			IBM_TDB)				
L256	21	((("SHUA") near3 ("Avi"))).INV.	(US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT)	ADJ	ON	ON	2023/01/10 07:54 AM
L257	19	L1 AND version	(US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT)	ADJ	ON	ON	2023/01/10 07:54 AM
L258	3	((("Orca") near3 ("Security"))).AS,AANM.	(USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM
L259	207643	(H04L63/1433 OR H04L63/1441 OR H04L63/14$2).cpc.	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L260	2490	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L261	205	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat) SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT;	ADJ	ON	ON	2023/01/10 07:54 AM

			IBM_TDB)				
L262	50	("10539275" "20170314745" "9109162" "9646093" "20050089749" "20150130346" "20150269204" "4591689" "5186249" "7392163" "20010046558" "20050106574" "20110003808" "5774249" "5999117" "6749768" "6749894" "8098282" "8868379" "10444489" "20020137625" "20040001966" "20040069969" "20050106712" "20060212502" "20080069735" "20090015670" "20100069660" "20110151019" "20120147382" "20130261976" "20140322337" "20160215211" "20170003497" "3554073" "3554085" "3554088" "3554091" "3554093" "3554095" "3554098" "3554099" "3554120" "3554135" "3554137" "3554147" "3554153" "3554155" "3554165" "3554167").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/10 07:54 AM
L263	5	L7 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/10 07:54 AM
L264	4	L7 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/10 07:54 AM
L265	50	("4770410" "5637144" "6145625" "4246184" "4269744" "4286481" "4301570" "4302939" "4436326" "4448396" "4551623" "4555258" "4796926" "4806203" "4836893" "4921707" "5188692" "5202667" "5250146" "5406914" "5494546" "6036446" "4243070" "4257482" "4269611" "4272640" "4273374" "4281805"	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/10 07:54 AM

		"4288558" "4299530" "4310334" "4320333" "4324078" "4324921" "4330553" "4334908" "4336606" "4338499" "4338737" "4347679" "4352957" "4355357" "4359761" "4360187" "4362072" "4365112" "4366353" "4367229" "4375510" "4381127").pn.					
L266	3	L10 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/10 07:54 AM
L267	0	L10 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2023/01/10 07:54 AM
L268	669	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L269	405	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version WITH (software OR app OR application OR software OR program OR code)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L270	384	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version WITH (software OR app OR application OR software OR program)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV,	ADJ	ON	ON	2023/01/10 07:54 AM

			MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L271	272	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version near2 (software OR app OR application OR software OR program)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L272	447	L4 AND (query$3 OR request$3) WITH cloud WITH source	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L273	65	L4 AND (query$3 OR request$3) WITH cloud WITH source WITH access$5 WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L274	1385	L4 AND (examin$5 OR	(US-PGPUB; USPAT;	ADJ	ON	ON	2023/01/10

		analyz$3 OR analysis) WITH source WITH access$5 WITH network	USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				07:54 AM
L275	480	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L276	96	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L277	61	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 WITH network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL,	ADJ	ON	ON	2023/01/10 07:54 AM

			PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L278	897	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L279	208	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log near3 network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L280	1225	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L281	201	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY,	ADJ	ON	ON	2023/01/10 07:54 AM

		fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner)	CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L282	79	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L283	6	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4 WITH (add$3 OR creat$3)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L284	42	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH,	ADJ	ON	ON	2023/01/10 07:54 AM

		(add$3 OR creat$3)	TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L285	1959	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L286	780	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L287	114	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L288	2	("20190281115" OR "20200137097").pn.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM
L289	447	L40 AND cloud WITH (api OR application	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM

		programming interface) WITH network					
L290	117	L40 AND cloud WITH (api OR application programming interface) WITH network WITH access$5	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM
L291	2211	L40 AND port WITH access$5 WITH (program OR software OR app OR application)	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM
L292	225	L40 AND port WITH access$5 WITH (program OR software OR app OR application) WITH (vulnerable OR risk$3 OR threat$4)	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM
L293	95	L40 AND port WITH access$5 WITH (program OR software OR app OR application) near3 (vulnerable OR risk$3 OR threat$4)	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM
L294	90	L40 AND (identify$3 OR identification OR determin$3 OR determination) WITH (vulnerab$5 OR threat OR risk) WITH port WITH network near3 access$7	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/10 07:54 AM
L295	1551	L40 AND (vulnerab$5 OR threat OR risk) WITH port WITH access$4	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L296	410	L40 AND (vulnerab$5 OR threat OR risk) WITH port near2 access$4	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO,	ADJ	ON	ON	2023/01/10 07:54 AM

			MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L297	164	L40 AND (vulnerab$5 OR threat OR risk) WITH port near2 access$4 AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L298	134	L40 AND (vulnerab$5 OR threat OR risk) WITH port near2 access$4 WITH (app OR application OR program OR software OR code)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L299	610	L40 AND (vulnerab$5 OR threat OR risk) WITH port WITH access$4 WITH (app OR application OR program OR software OR code)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 07:54 AM
L300	141	L40 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP,	ADJ	ON	ON	2023/01/10 07:54 AM

		near2 (app OR application OR program OR software OR code) WITH port WITH access$4	AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L301	3869	L40 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 10:56 AM
L302	1237	L40 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH network WITH (identify$3 OR identification OR determin$3 OR determination)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 10:57 AM
L303	132	L40 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH network WITH (identify$3 OR identification OR determin$3 OR determination) WITH known	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE,	ADJ	ON	ON	2023/01/10 11:00 AM

			SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L304	49	L40 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH (identify$3 OR identification OR determin$3 OR determination) WITH known SAME (cloud OR log OR record)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 02:02 PM
L305	75	L40 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH known WITH (cloud OR log OR record)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/10 02:57 PM
L306	22	("10152487" OR "10868825" OR "11036856" OR "20130262801" OR "20140366118" OR "20160330237" OR "20180205755" OR "20180367515" OR "20190250857" OR "20190327272" OR "20200097662" OR "20200220887" OR "20200244692" OR "20200396254" OR "20210342297" OR "20220100869" OR "20220182294" OR "20220191250" OR "20220210195" OR "7013395" OR	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 09:21 AM

		"7143143" OR "9794289").pn.					
L307	3	("20170012998" OR "20180288087" OR "8621073").pn.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 09:24 AM
L308	2	("20200059481" OR "8006285").pn.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 09:25 AM
L309	596426	(H04L63/1433 OR G06F9/45558 OR G06F21/54 OR G06F21/554 OR G06F21/568 OR G06F21/577 OR G06F21/78 OR H04L9/0825 OR H04L9/0894 OR H04L9/14 OR H04L63/10 OR H04L63/1441 OR H04L63/145 OR H04L67/1008 OR H04L67/101 OR G06F2009/4557 OR G06F2009/45595 OR G06F2221/034 OR G06F21/57 OR H04L63/14$2).cpc.	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/13 02:06 PM
L310	4542	L309 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/13 02:07 PM
L311	146	L309 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH network WITH (identify$3 OR identification OR determin$3 OR determination) WITH known	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN,	ADJ	ON	ON	2023/01/13 02:07 PM

			WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L312	51	L309 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH (identify$3 OR identification OR determin$3 OR determination) WITH known SAME (cloud OR log OR record)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/13 02:07 PM
L313	78	L309 AND port WITH (vulnerabilit$3 OR threat OR risk) WITH known WITH (cloud OR log OR record)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2023/01/13 02:07 PM

PE2E SEARCH - Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
N1	50293	(H04L63/1433 OR H04L63/1441 OR H04L63/14$2).cpc.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:21 AM
N2	2783	N1 AND cloud.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:21 AM
N3	274	N1 AND cloud.clm. AND port.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:22 AM
N4	45	N1 AND cloud.clm. AND port.clm. AND log.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:22 AM
N5	5	N1 AND cloud.clm. AND port.clm. AND log.clm. AND version.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:22 AM
N6	3	N1 AND cloud.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11

		AND port.clm. AND log.clm. AND version.clm. AND vulnerab$5.clm.					10:22 AM
N7	1	N1 AND cloud.clm. AND port.clm. AND log.clm. AND version.clm. AND vulnerab$5.clm. AND external.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:23 AM
N8	5	N1 AND cloud.clm. AND port.clm. AND log.clm. AND external.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:23 AM
N9	2	N1 AND cloud.clm. AND port.clm. AND log.clm. AND external.clm. AND software.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/11 10:23 AM
N10	146301	(H04L63/1433 OR G06F9/45558 OR G06F21/54 OR G06F21/554 OR G06F21/568 OR G06F21/577 OR G06F21/78 OR H04L9/0825 OR H04L9/0894 OR H04L9/14 OR H04L63/10 OR H04L63/1441 OR H04L63/145 OR H04L67/1008 OR H04L67/101 OR G06F2009/4557 OR G06F2009/45595 OR G06F2221/034 OR G06F21/57 OR H04L63/14$2).cpc.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:09 PM
N11	654	N10 AND cloud.clm. AND port.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:10 PM
N12	58	N10 AND cloud.clm. AND port.clm. AND log.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:10 PM
N13	8	N10 AND cloud.clm. AND port.clm. AND log.clm. AND version.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:10 PM
N14	3	N10 AND cloud.clm. AND port.clm. AND log.clm. AND version.clm. AND vulnerab$5.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:10 PM
N15	1	N10 AND cloud.clm. AND port.clm. AND	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:10 PM

		log.clm. AND version.clm. AND vulnerab$5.clm. AND external.clm.					
N16	8	N10 AND cloud.clm. AND port.clm. AND log.clm. AND external.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:10 PM
N17	2	N10 AND cloud.clm. AND port.clm. AND log.clm. AND external.clm. AND software.clm.	(US-PGPUB; USPAT)	ADJ	ON	ON	2023/01/13 02:10 PM

PE2E SEARCH - Search History (Similarity Search)

No Similarity Results available

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Use as many sheets as necessary)				Complete if Known	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	1	Attorney Docket Number	**15635.0005-00000**

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number / Number-Kind Code (if known)	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-8,006,285-B1	08-23-2011	Perlman	

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number / Number-Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2020/0059481-A1	02-20-2020	Sekar et al.	

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Foreign Patent Document — Country Code Number Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	Translation
		WO 2021/072877 A1	04-22-2021	PING AN TECH SHENZHEN CO LTD		YES
		CN 110708315 A	01-17-2020	DBAPP SECURITY INF TECH CO LTD		YES

NON PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Ex parte Quayle Action No. 17/657,981 dated October 31, 2022, in the United States Patent and Trademark Office. (15635.0008-00000)	N/A

Examiner Signature	/SARAH SU/	Date Considered	01/11/2023

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Use as many sheets as necessary)				Complete if Known	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	1	Attorney Docket Number	**15635.0005-00000**

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number Number-Kind Code (if known)	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number Number-Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Foreign Patent Document Country Code Number Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	Translation
		CN 114584359 A	06-03-2022	YUN TABLE TECH BEIJING CO LTD		YES
		CN 109995722 A	07-09-2019	GUANGZHOU MINGLEAD GENE TECH CO LTD		YES

NON PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Notice of Allowance, Application No. 17/657,981 dated December 7, 2022, in the United States Patent and Trademark Office. (15635.0008-00000)	N/A
		Notice of Allowance, Application No. 17/716583 dated December 8, 2022, in the United States Patent and Trademark Office. (15635.0011-00000)	N/A

Examiner Signature	/SARAH SU/	Date Considered	01/11/2023

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Use as many sheets as necessary)				Complete if Known	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	2	Attorney Docket Number	**15635.0005-00000**

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number Number-Kind Code (if known)	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-11,036,856-B2	06-15-2021	Graun et al.	
		US-9,794,289-B1	10-17-2017	Banerjee et al.	
		US-7,013,395-B1	03-14-2006	Swiler et al.	
		US-10,868,825-B1	12-15-2020	Dominessy et al.	
		US-10,152,487-B1	12-11-2018	Patwardhan et al.	
		US-7,143,143-B1	11-28-2006	Thompson	

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number Number-Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2020/0244692-A1	07-30-2020	SHUA	
		US-2019/0327272-A1	10-24-2019	Narayanaswamy	
		US-2020/0097662-A1	03-26-2020	Hufsmith et al.	
		US-2013/0262801-A1	10-03-2013	Sancheti et al.	
		US-20220210195-A1	06-30-2022	Parekh et al.	
		US-2019/0250857-A1	08-15-2019	Reddy et al.	
		US-2022/0191250-A1	06-16-2022	Laliberte	
		US-2014/0366118-A1	12-11-2014	Yin	
		US-2020/0220887-A1	07-09-2020	A. K.	
		US-2018/0367515-A1	12-20-2018	Anandam et al.	
		US-2022/0182294-A1	06-09-2022	Li et al.	
		US-2022/0100869-A1	03-31-2022	Berger et al.	
		US-2020/0396254-A1	12-17-2020	Crabtree et al.	
		US-2016/0330237-A1	11-10-2016	Edlabadkar	
		US-2018/0205755-A1	07-19-2018	Kavi et al.	
		US-2021/0342297-A1	11-04-2021	Gupta et al.	

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

Examiner Signature	/SARAH SU/	Date Considered	01/11/2023

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Use as many sheets as necessary)				Complete if Known	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	2	of	2	Attorney Docket Number	**15635.0005-00000**

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Foreign Patent Document — Country Code Number Kind Code (*if known*)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	Translation
		WO 2020/106973 A1	05-28-2020	ARAALI NETWORKS INC.		N/A
		WO 2019/051507 A1	03-14-2019	CARBON BLACK INC.		N/A
		WO 2017/153983 A1	09-14-2017	B. G. NEGEV TECHNOLOGIES AND APPLICATIONS LTD.		N/A

NON PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Non-Final Office Action Application No. 17/716,491 dated August 3, 2022, in the United States Patent and Trademark Office.	N/A
		Non-Final Office Action Application No. 17/657,986 dated August 17, 2022, in the United States Patent and Trademark Office.	N/A
		Non-Final Office Action Application No. 17/658,156 dated July 18, 2022, in the United States Patent and Trademark Office.	N/A
		Non-Final Office Action Application No. 17/716,583 dated August 18, 2022, in the United States Patent and Trademark Office.	N/A

Examiner Signature	/SARAH SU/	Date Considered	01/11/2023

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				***Complete if Known***	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi Shua
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	2	*Attorney Docket Number*	15635.0005-00000

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-11,120,124 B2	09-14-2021	Fusenig et al.	
		US-10,402,560 B2	09-03-2019	Gilbert	
		US-10,079,842 B1	09-18-2018	Brandwine et al.	
		US-9,229,758 B2	01-05-2016	Ammons et al.	
		US-9,734,325 B1	08-15-2017	Neumann et al.	
		US-11,216,563 B1	01-04-2022	Veselov et al.	
		US-9,756,070 B1	09-05-2017	Crowell et al.	
		US-9,858,105 B1	01-02-2018	Upadhyay et al.	
		US-10,782,952 B1	09-22-2020	Doring et al.	
		US-9,069,983 B1	06-30-2015	Nijjar	
		US-9,268,689 B1	02-23-2016	Chen et al.	
		US-10,534,915 B2	01-14-2020	Cherny et al.	
		US-8,479,294 B1	07-02-2013	Li et al.	
		US-10,956,484 B1	03-23-2021	Kulkarni et al.	

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2020/0042707 A1	02-06-2020	Kucherov et al.	
		US-2012/0323853 A1	12-20-2012	Fries et al.	
		US-2014/0096135 A1	04-03-2014	Kundu et al.	
		US-2009/0007100 A1	01-01-2009	Field et al.	
		US-2008/0263658 A1	10-23-2008	Michael et al.	
		US-2010/0017512 A1	01-21-2010	Ciano et al.	
		US-2021/0019404 A1	01-21-2021	Strogov et al.	
		US-2016/0063251 A1	03-03-2016	Spernow et al.	
		US-2015/0020203 A1	01-15-2015	Xie et al.	
		US-2014/0259168 A1	09-11-2014	McNamee et al.	
		US-2005/0055558 A1	03-10-2005	Carmona	
		US-2014/0007239 A1	01-02-2014	Sharpe et al.	
		US-2018/0198610 A1	07-12-2018	Mullins et al.	
		US-2022/0294817 A1	09-15-2022	Parekh et al.	

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				***Complete if Known***	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi Shua
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	2	of	2	*Attorney Docket Number*	15635.0005-00000

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

NON PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Rani et al.; "An Efficient Approach to Forensic Investigation in Cloud using VM Snapshots", 2015 International Conference on Pervasive Computing (ICPC), 5 pages, (2015).	N/A
		Wei et al., "Managing Security of Virtual Machine Images in a Cloud Environment", CCSW'09, pp. 91-96, November 13, 2009	N/A
		Almulla et al., "Digital Forensic of a Cloud Based Snapshot", The Sixth International Conference on Innovative Computing Technology (INTECH 2016), pp. 724-729, (2016)	N/A
		Pandey et al., "An Approach for Virtual Machine Image Security", Computer Science and Engineering MNNIT, Allahabad, 2014 International Conference on Signal Propagation and Computer Technology (ICSPCT), pp. 616-623, (2014)	N/A
		Rajasekaran et al., "Scalable Cloud Security via Asynchronous Virtual Machine Introspection", 8th USENIX Workshop on Hot Topics in Cloud Computing, Cover page and pages 1-6, (2016)	N/A
		Kaur et al., "Secure VM Backup and Vulnerability Removal in Infrastructure Clouds", 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1217-1226, (2014)	N/A
		Fernandez et al., "Two patterns for cloud computing: Secure Virtual Machine Image Repository and Cloud Policy Management Point", PLoP '13: Proceedings of the 20th Conference on Pattern Languages of Programs October 2013 Article No.: 15, Association for Computing Machinery (ACM), Cover sheet and pages 1-11, (2013)	N/A
		Fernandez et al. "Building a security reference architecture for cloud systems", Springer, Requirements Eng, vol. 21, pp. 225-249, (2016)	N/A
		Ammons et al., "Virtual machine images as structured data: the Mirage image library", IBM Research, Cover sheet 2 pages and pages 1-6, (2011)	N/A
		"IBM Point of View Security and Cloud Computing", IBM SmartCloud Enterprise, Cloud Computing White Paper, 13 sheets of cover pages and pages 1-20, (2009)	N/A
		Cui et al., "A Less Resource-Consumed Security Architecture on Cloud Platform", Wuhan University Journal of Natural Sciences, vol. 21, no. 5, pp. 407-414, (2016)	N/A
		Bugiel et al., "AmazonIA: When Elasticity Snaps Back", CCS' 11, ACM, pp. 389-400, (2011)	N/A
		Non-Final Office Action Application No. 17/658,149 dated December 30, 2022, in the United States Patent and Trademark Office. (15635.0007-00000)	N/A
		Non-Final Office Action Application No. 17/658,156 dated January 3, 2023, in the United States Patent and Trademark Office. (15635.0010-00000)	N/A
		Final Office Action Application No. 17/657,986 dated November 23, 2022, in the United States Patent and Trademark Office. (15635.0009-00000)	N/A

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**01/24/2023 12:10:03 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61482216	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 17

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
15635.0005-00000-IDS after RCE.pdf	2	Transmittal Letter	29 KB
15635.0005-00000-sb08.pdf	2	Information Disclosure Statement (IDS) Form (SB08)	34 KB
Warning: This is not a USPTO supplied IDS fillable form. Data in the form cannot be automatically loaded to other USPTO systems.			
NPL-Rani et al..pdf	5	Non Patent Literature	3087 KB
NPL-Rajasekaran et al..pdf	7	Non Patent Literature	6077 KB
NPL-Wei et al..pdf	6	Non Patent Literature	6285 KB

NPL-Almulla et al..pdf	6	Non Patent Literature	5700 KB
NPL-WEB-ARCHIVE - IBM - SmartCloud.pdf	33	Non Patent Literature	12233 KB
NPL-Ammons et al..pdf	8	Non Patent Literature	7102 KB
NPL-Cui et al..pdf	8	Non Patent Literature	5180 KB
NPL-Pandey et al..pdf	8	Non Patent Literature	13829 KB
NPL-Final Rejection-11-23-2022-15635.0009-00000.pdf	14	Non Patent Literature	432 KB
NPL-Bugiel et al..pdf	12	Non Patent Literature	12358 KB
NPL-Non-Final Rejection-1-3-2023-15635.0010-00000.pdf	22	Non Patent Literature	562 KB
NPL-Non-Final Rejection-12-30-2022-15635.0007-00000.pdf	19	Non Patent Literature	549 KB
NPL-Fernandez et al. Two Patterns for Cloud Computing.pdf	12	Non Patent Literature	5569 KB
NPL-Kaur et al..pdf	10	Non Patent Literature	10148 KB
NPL-Fernandez - Building a Security Reference Architecture.pdf	25	Non Patent Literature	22917 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
15635.0005-00000-IDS after	AD3D29EEB2BE50BA739B90E52069B0269C716F127BF8A40E4

RCE.pdf	BDF76C52E5FBB8EE829D36F47B7D2663BBBFFF65BE2859B2A6AB1839CB3C792ADFC320E1FF64452
15635.0005-00000-sb08.pdf	90CFC997FA3A3A9B89B0C510B07B701D2190F5480455AA97CDF9A3BD02DC9024CB8BBF3E8F37A3F9901065138F194A2A09E0DBA153C6E07AEFE90AEBEF2ACA7B
NPL-Rani et al..pdf	F165B7EA73ED18B1BF9C17E9ACD2F2FA33FE2E0F7EEBDE244246DA0559D60F27AA66E681D26858F34B085DE83FB65F53C53D2994B7AD04B9EDE12E80E4A75A87
NPL-Rajasekaran et al..pdf	97856A5770786D970FB46D606BDA5374C9386B25BE690C85B8CD37E81C63A80650E055EA5D77CBBA7C485CAA82BCD7C855FA4A63036E5297D58D27A7164BF4D8
NPL-Wei et al..pdf	1FFA3361F189451192C4A43B664793936DC9B8BF1ABA89CD8A9039FD0DE9DCE5534D6450F26A618E39F0F9BC027DDDC7B5F581AC48312049AE93AA1C73EBF0B0
NPL-Almulla et al..pdf	6F090A77C7C05DA6CF6503DFF5D07BB2EFE06C5DD5268AC3945C984F2C695DB6CB92F718AECB17400CEB6C57CDA1EA503CB2AD762FD4D5FE11566BE4872593CD
NPL-WEB-ARCHIVE - IBM - SmartCloud.pdf	7FA0D1BBE9E11351BB64A9BA65809AD0A1A3A5E426EE581A715E8775EE5E4D1C3FCC9084A4A5D2709C3EE0DD09F0DE1465DE039F7BEEE196A12A3BF12E3737B6
NPL-Ammons et al..pdf	747103FEF5246626D8D179C457238BAF76C36D62AD2A0514D6190FF22BA536D2572405A42326282628490A389826B53DE623DCAB8019E232707604B8BB822415
NPL-Cui et al..pdf	2B17CE2E6324A4871FCBF55D89B45E3274BCBA16DA1932A78ECF1247867416EF29C70C8CF0457EF8F04AA13792B7BB1AA59EB9D55B859B352856C0D1ED6CE4B4
NPL-Pandey et al..pdf	49EB3A155A60396F74B3540E1A474FFDC5D8005C2CFD2B5FA1D8C8807B748C817F906CAB6E3A5604D8827D194161CC180A51B6A26ED730FAB1EF007A463D013F
NPL-Final Rejection-11-23-	4D18C45D52FB20D4C8CAB9E02B20B4739E6D1EADC260DB86

2022-15635.0009-00000.pdf	E0EB1D350337A75800EEC72B94EABF72D0953F299693BF711B5599999EA865B12661A22750467A4E
NPL-Bugiel et al..pdf	AA12649F8EFE06E31721B40EC057B2A2F579D3DA3DDDBFE0C4562D7FD00B2E28223A9DDA1E50ACFE4156D524C2B4BE4ED7E4D470A0C8F4C335043E038E3A649D
NPL-Non-Final Rejection-1-3-2023-15635.0010-00000.pdf	3265609DA106CF454EA95F608E5F9025C08DA3681825AE5ED635A908BE661EFA457301E21BB78F6118A47CBB28D7FF111495EA2C7DB85B458FB7B2E53F8D232E
NPL-Non-Final Rejection-12-30-2022-15635.0007-00000.pdf	9A6CD73D75D08A5CA6F6BCF63BED25C83A43542CB8E47394FBBAD91B78B6097304357D6065BC9398DCEE4BF931393783E263EAED33E94ACE9D153FF6C13C3A08
NPL-Fernandez et al. Two Patterns for Cloud Computing.pdf	15BACFCB6E27481DD20EA4F6D2151DA3575C9EC290F1BFBB0BF64BA331DD5879754276BEFCC62C3DB4E5860674DA9CFBB077B9067D240262330AD97738370467
NPL-Kaur et al..pdf	D3C013C12B359601247549A613F32188D7B3C06B18252DA9A86640769DB19748DB40DB133193E99444284A6BBF7C6A37DF9BAE0A85E84303F642FD164FA83A4B
NPL-Fernandez - Building a Security Reference Architecture.pdf	1E2F12564C89D0182932AB3D9816B8047B93F7A20CDBD69478F6C7B9583C9E9AE551AB8F113358B80A27010A13D2C7B5707CD1E3793A776D5FE19DA4962A3BC3

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the

International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PATENT
Attorney Docket No. 15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.: 17/657,972	)	Examiner: SU, SARAH
	)	
Filed: April 5, 2022	)	Confirmation No.: 2547
	)	
For: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT UNDER 37 C.F.R. § 1.97(b)

Pursuant to 37 C.F.R. §§ 1.56 and 1.97(b), Applicant brings to the Examiner's attention the documents listed on attached Form PTO/SB/08. This Information Disclosure Statement is being filed before the mailing date of a first Office Action after the filing of a Request for Continued Examination in the above-referenced application.

Copies of the listed non-patent literature documents are attached.

Applicant respectfully requests that the Examiner consider the listed documents on the attached Form PTO/SB/08 and indicate that they were considered by making appropriate notations on the attached form.

This submission does not represent that a search has been made or that no better art exists and does not constitute an admission that each or all of the listed documents are material or constitute "prior art." If the Examiner applies any of the

documents as prior art against any claim in the application and Applicant determines that the cited documents do not constitute "prior art" under United States law, Applicant reserves the right to present to the U.S. Patent and Trademark Office the relevant facts and law regarding the appropriate status of such documents.

Applicant further reserves the right to take appropriate action to establish the patentability of the disclosed invention over the listed documents, should one or more of the documents be applied against the claims of the present application.

If there is any fee due in connection with the filing of this Statement, please charge the fee to deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: January 24, 2023

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				***Complete if Known***	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	1	*Attorney Docket Number*	**15635.0005-00000**

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Foreign Patent Document Country Code Number Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	Translation
		CN 114584359 A	06-03-2022	YUN TABLE TECH BEIJING CO LTD		YES
		CN 109995722 A	07-09-2019	GUANGZHOU MINGLEAD GENE TECH CO LTD		YES

NON PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Notice of Allowance, Application No. 17/657,981 dated December 7, 2022, in the United States Patent and Trademark Office. (15635.0008-00000)	N/A
		Notice of Allowance, Application No. 17/716583 dated December 8, 2022, in the United States Patent and Trademark Office. (15635.0011-00000)	N/A

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**12/22/2022 02:01:01 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61342779	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 6

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
IDS after RCE.pdf	3	Transmittal Letter	33 KB
sb08.pdf	1	Information Disclosure Statement (IDS) Form (SB08)	26 KB
Notice of Allowance- 12-7-2022-15635.0008-00000.pdf	55	Non Patent Literature	2232 KB
Notice of Allowance-12-8-2022-15635.0011-00000.pdf	13	Non Patent Literature	511 KB
FOR-CN114584359A.pdf	31	Foreign Reference	8416 KB
CN109995722A.pdf	15	Foreign Reference	3602 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
IDS after RCE.pdf	5F59AAE160824B56CF4455770AB2DF0ECCE1D5486275F81AF317B750D4088C89FE6BE3928F0B91983B6FBCD35D3E452FD63D35E6303B85F6F4987D1076880F3A
sb08.pdf	E5A5DF183FA5E261581D86EC3AD9765477170FE99773A0600D8661F8BDA0B63DD9498A934CC0F571EFAE6621990CD8A51AA5D8C75DA309BF278A5E8783F2C7F4
Notice of Allowance- 12-7-2022-15635.0008-00000.pdf	43C8D3BF2E6D804C53120BB7045A8E8D4AA8BC798714DFE2A004A11DE30B48D0E13A7CCEF633862F4F4D4EF626269A55E61D6BDBC4D89BE1EF4AA2D0AEA9BA72
Notice of Allowance-12-8-2022-15635.0011-00000.pdf	CC7213A7E65B267DBEBFD349AAE4F20680DFD4F78253EF0285BD2E2A18C2F06641E27FF9960061BF247FC370E2FEE341B40ED3AF98B6C73346F29678AAE5F425
FOR-CN114584359A.pdf	1F22B3D92A8E76AF02DF1BB0029CEA7F41C81D8656975BF3321B98B4F0A2201E5F9F0654177C2FACD77A4F77A165E21E92CD2AF06763CC55C2DEC6980437C366
CN109995722A.pdf	111D2017910BF09C15AFC0D2202CFB3B99540C4EF8DA9461F77D95ABDBAFA990222C6040BA79CF88CEE153C318C25777005385A06942718E63B68102ACC7AD54

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Espacenet

CN114584359A Safe trapping method and device and computer equipment

Applicants: YUN TABLE TECH BEIJING CO LTD

Inventors: FANG YONGCHENG,CHO JUNG-HO,LIU MAOLIN,GONG LIANGHUA

Classifications:
IPC ***H04L49/00; H04L9/40;***

CPC **H04L49/70 (CN); H04L63/1466 (CN); H04L63/1491 (CN);**

Priorities: CN202210178181A 2022-02-24

Application: CN202210178181A·2022-02-24

Publication: CN114584359A·2022-06-03

Published as: CN114584359A

Safe trapping method and device and computer equipment

Abstract

The present application is applicable to the technical field of network security, and provides a secure trapping method, comprising: generating a plurality of categories of first simulation devices and first virtual services, the first simulation devices and the first virtual services being generated through a simulator according to device parameters of physical devices and application scenarios of the physical devices; generating shadow equipment of the entity equipment, wherein each shadow equipment corresponds to a plurality of virtual communication addresses; and when the virtual communication address receives a traffic data packet from the attack device, if the shadow device corresponding to the virtual communication address is high interaction, forwarding the traffic data packet to the corresponding first simulation device and/or the first virtual service through the shadow device. According to the method, the simulation device is generated through the simulator corresponding to the entity device, and then deployment of the industrial internet deception trapping system is achieved. And a plurality of containers do not need to be generated, so that a large amount of server resources can be saved, and the cost of deploying the industrial internet deception trapping system is reduced.

safety trapping method, device and computer device

DOCUMENT ID	DATE PUBLISHED
CN-114584359-A	2022-06-03

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
FANG, Yong-cheng	N/A	N/A	N/A	N/A
ZHAO, Chong-hao	N/A	N/A	N/A	N/A
LIU, MAO-LIN	N/A	N/A	N/A	N/A
GONG, Liang-hua	N/A	N/A	N/A	N/A

DATE FILED
2022-02-24

CPC CURRENT

TYPE	CPC	DATE
CPCI	H04L49/70	2013-01-01
CPCI	H04L63/1491	2013-01-01
CPCI	H04L63/1466	2013-01-01

Abstract

The invention is suitable for the technical field of network safety, providing a safety for trapping, comprising: generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator; generating a shadow device of the entity device, each shadow device corresponding to a plurality of virtual communication address; when the virtual communication address receives the flow data packet from the attack device, if the shadow device corresponding to the virtual communication address is high interaction, the flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service through the shadow device. In the method, the simulation device is generated by the simulator corresponding to the entity device, so as to realize the deployment of the industrial internet spoofing trapping system. Because there is no need to generate a plurality of containers, so it can save a lot of server resource, so as to reduce the cost of deploying industrial internet spoofing trapping system.

Description

INVENTION-TITLE
Technical Field
The invention belongs to the technical field of network safety, especially relates to a safety and device for trapping and trapping.
Background

With the upgrading of industrial transformation, the industrial Internet is widely used. However, the industrial Internet has the problem of network information safety application, such information leakage, network attack, there is a safety of using the loophole and so on. At present, the active defence can be carried out by the industrial internet spoofing and trapping system, and the threat intelligence is obtained.
However, the existing industrial internet spoofing trapping system needs to occupy a large amount of server resource deployment, deployment cost is high.
Contents of the Invention
The embodiment of the invention claims safety trapping method and device, which can solve the problem that the industrial internet spoofing trapping system needs to occupy a large amount of server resource when deployed, and the deployment cost is high.
In a first aspect, the embodiment of the invention claims a safety for capturing, comprising: generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device; generating a shadow device of at least one entity device, the shadow device is a mapping of the entity device, each shadow device corresponding to a plurality of virtual communication address; when the virtual communication address receives the flow data packet from the attack device, if the shadow device corresponding to the virtual communication address is high interaction, the flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service through the shadow device.
In one possible implementation of the first aspect, generating shadow device of at least one entity device, comprising: in the idle IP address space, batch generating virtual IP address of a plurality of different physical addresses; generating the corresponding relation between the shadow device and the first simulation device and the first simulation service according to the mapping relationship between the shadow device and the entity device; according to the corresponding relation, associating a plurality of virtual IP address associated with each first simulation device and the first simulation service.
In one possible implementation of the first aspect, the traffic data packet comprises an application layer message, the method further comprises: when the virtual communication address receives the attack application layer message from the attack device, the shadow device corresponding to the virtual communication address forwarding the attack application layer message to the corresponding first simulation device and the first virtual service; when the first simulation device and the first virtual service receive the first message sent by the attack device, responding to the corresponding second message; correspondingly storing the contents of the first message and the second message in the data table one by one; when the first simulation device and the first virtual service receives the target application layer message of the target attack device, matching the content of the target message with the content of the first message in the data table, if the matching is successful, then sending the second message corresponding to the successfully matched first message to the attack device.
In one possible implementation of the first aspect, the method further comprises: generating the attack image of the target attack device according to the history attack flow data log; collecting the attack behaviour of the attack image of the target attack device, generating the alarm behaviour log corresponding to the target attack device.
In one possible implementation of the first aspect, the method further comprises: according to history attack flow data log generating attack image of the target attack device, comprising: obtaining the historical flow data record generated by each attacking device; obtaining key field from the attack flow data record, wherein the key field comprises attack device communication address, attack device fingerprint and the user login of the specific software if at least one of the key field in the flow data sent by any attack device and the target attack device is the same, then determining the attack device and the target attack device are the same attack device.
In one possible implementation of the first aspect, the method further comprises: sending the alarm behaviour log corresponding to the attack device to the target server, so that the target server sends the alarm behaviour log to the client.
In one possible implementation of the first aspect, the method further comprises: orderly counting the attack behaviour total number in the attack flow data in each preset time period and the total number of the alarm attack behaviour; generating an alarm attack behaviour tendency chart according to the total number of

attack behaviour and the total number of alarm attack behaviour in each preset time period, and displaying the alarm attack behaviour tendency chart in the monitoring large screen.
In a second aspect, the embodiment of the invention claims a device safety trapping, comprising: service simulation module, used for generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device; a device generating module, a shadow device generating at least one entity device, the shadow device is a mapping of the entity device, each shadow device corresponding to a plurality of virtual communication address; a flow forwarding module, used for when the virtual communication address receives the flow data packet from the attack device, if the virtual communication address corresponding to the shadow device is high interaction, the flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service through the shadow device.
the third aspect, the embodiment of the invention claims a terminal device, the terminal device comprises a memory and a processor, the memory stores a computer program, the processor executes the computer program to realize the following steps: Including: generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device; generating a shadow device of at least one entity device, the shadow device is a mapping of the entity device, each shadow device corresponding to a plurality of virtual communication address; when the virtual communication address receives the flow data packet from the attack device, if the shadow device corresponding to the virtual communication address is high interaction, the flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service through the shadow device.
the fourth aspect, the embodiment of the invention claims a computer readable storage medium computer, readable storage medium is stored with a computer program, the computer program is executed by a processor to implement the following steps: Including: generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device; generating a shadow device of at least one entity device, the shadow device is a mapping of the entity device, each shadow device corresponding to a plurality of virtual communication address; when the virtual communication address receives the flow data packet from the attack device, if the virtual communication address corresponding to the flow data packet is forwarded to the corresponding first simulation device.
the fifth aspect, the embodiment of the invention claims a computer program product, when the computer program product is run on the terminal device, the terminal device executes any one safety the first aspect of the trapping method.
It can be understood that the second aspect to the fifth aspect of the beneficial effect can be described in the first aspect of the related description, which will not be repeated here.
The embodiment of the invention compared with the existing technology has the beneficial effects as follows: The safety, device, computer device, storage medium computer program product, in the server, generating the first simulation device and the first virtual service by the simulator corresponding to the entity device, and configuring the virtual communication address for each first simulation device, when the virtual communication address receives the flow data packet from the attack device, and forwarding it to the corresponding first simulation device and/or the first virtual service. In the method, through the simulator corresponding to the entity device to generate the simulation device, so as to realize the deployment of industrial internet spoofing trapping system. Because there is no need to generate a plurality of containers, so it can save a lot of server resource, so as to reduce the cost of deploying industrial internet spoofing trapping system.
Description of pictures
In order to more clearly explain the technical solution of the embodiment of the present application, the embodiments or the prior art described in the description of the accompanying drawings is briefly described, it is obvious, the following description of the accompanying drawings is only some embodiments of the invention, for the common technician in the field, under the premise of not paying creative labour, it can

obtain other drawings according to the drawings.
FIG. 1 is an embodiment of the invention provides safety environment diagram of the trapping method;
FIG. 2 is a flow diagram of the device generating step provided by one embodiment of the present application;
FIG. 3 is a flow diagram of the safety for trapping according to one embodiment of the invention;
FIG. 4 is another embodiment of the invention claims safety capture method of the flow diagram;
FIG. 5 is a schematic diagram of an application scenario provided by an embodiment of the present application;
FIG. 6 is a structure schematic safety of the trapping device provided by the embodiment of the invention;
FIG. 7 is a structure schematic diagram of the terminal device provided by the embodiment of the invention.
Specific implementation examples
In order to make the purpose of the present application, technical solution and advantages are more clearly understood, the following combine the accompanying drawings and examples, for further detailed explanation of this application. It should be understood that the specific embodiment described herein is only for explaining the application, and not for defining the application.
FIG. 1 is an embodiment of the invention claims safety environment diagram of the trapping method.
safety of the invention claims a trapping method, which can be applied to the application environment shown in FIG. 1. wherein, the attack device terminal 102 communicates with the server 104 through the network. The data storage system may store the data to be processed by the server 104. the data storage system can be integrated on the server 104, also can be placed on the cloud or other network server. server 104 generates a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device; generating a shadow device of at least one entity device, the shadow device is a mapping of the entity device, each shadow device corresponding to a plurality of virtual communication address; when the virtual communication address received from the attack device 102 sent by the flow data packet, if the virtual communication address corresponding to the shadow device is high interaction, then the flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service through the shadow device. wherein the attack device terminal can be but not limited to mobile phone, the tablet computer can also be internet of things device and portable wearable device, the internet of things device can be a intelligent, a intelligent TV, a intelligent air conditioner, intelligent vehicle-mounted device and so on. The portable wearable device can be intelligent watch, a smart wristband, a head-mounted device and so on. The server 104 may be implemented using an independent server or a server cluster consisting of a plurality of servers.
FIG. 2 is a flow diagram of the device generating step provided by one embodiment of the present application.
In one embodiment, as shown in FIG. 2, there is provided a safety for capturing, using the method applied to FIG. 1 in the server 104 as an example for explanation, comprising the following steps:
S202, generating a plurality of categories of the first simulation device and the first virtual service, wherein the first simulation device and the first virtual service is the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device.
Specifically, the server can realize the simulation of the entity industrial control device by entity industrial control device issued by the simulator corresponding to the entity industrial control device, and then the corresponding firmware is loaded into the industrial control device simulation model, so as to realize the simulation of the entity industrial control device to obtain the first simulation device. For example, when the server needs to simulate the industrial control device is Siemens river graph industrial control device, the Siemens manufacturer issued by the Siemens simulation software simulated with the firmware, so as to obtain the first simulation device corresponding to the Siemens river graph industrial control device.
the first virtual service can be defined according to the protocol, through the protocol simulation of the code level, so as to realize the network service simulation. For example, when the simulation ssh (Secure Shell safety shell protocol), the service end receives the request sent by the client end, handshake for three times, establishing tcp (Transmission ControlProtocol, transmission control protocol). then the two parties send the version number and algorithm supported by itself, negotiating to determine the version and algorithm. then

the client generates a pair of keys according to the data sent by the service end, and sends the public key to the server. when the client receives the identity of the first verification server, then storing it in the local folder, then using the temporary public key sent by the server to generate the shared key, and then encrypting the signature, comparing the result with the calculated hash value, when the data is verification problem, so as to finish the simulation of ssh protocol.

when the simulation simulation of the entity industrial control device and simulation simulation of the network protocol, according to the production process of each industry, office network, the scene of the production network and the simulation parameter of each entity device are used for configuring each simulation device and the virtual service combination into different types of industrial internet spoofing trapping system. It should be noted that the industrial internet spoofing trapping system also can be called honey net system, and the honey net system is composed of honey field and honey net, wherein the honey field is composed of a plurality of simulation device and a plurality of virtual service, honey net is covered on the network of the honey field.

S204, generating at least one shadow device of the entity device, the shadow device is a mapping of the entity device, each shadow device corresponding to a plurality of virtual communication address.

wherein the shadow device is a virtual image on the existing operating system, it has the same function system is completely the same. after entering the shadow system, all operations are virtual, will not generate any influence to the real system. When the first simulation device is operated on the existing operation system, the shadow device is the mapping of the first simulation device on the virtual operation system. Because the flow rate is introduced into the honey field, actually the attacker attacks the virtual device in the honey field, service, but for the attacker, the attack is the shadow device through the communication address virtual virtual technology, service, device.

In other embodiments, a shadow device may correspond to one entity device, also can correspond to a plurality of entity device. when a shadow device corresponding to a plurality of entity device, shadow device obtains a plurality of virtual communication address, and the plurality of virtual communication address are distributed to each entity device.

S206, when the virtual communication address receives the flow data packet from the attack device, if the virtual communication address corresponding to the shadow device is high interaction, then the flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service through the shadow device.

Specifically, when the virtual communication address set by the shadow device in the virtual operation system receives the data flow packet from the attack device, if the virtual communication address corresponding to the shadow device is high interaction, the shadow device introduces the received flow data packet into the first simulation device or the first simulation service corresponding to the virtual communication address by means of the net address conversion. wherein the shadow device can be configured to high interaction, also can be configured to low interaction. The shadow device with low interaction only can imitate a small amount of internet protocol and network service, and the shadow device with high interaction can simulate the industrial control device, also can simulate most internet protocol and network service.

It should be noted that when the shadow device is low interaction, it only needs to transmit the data flow sent by the attack device to the first simulation device or the first simulation service through specific port on the shadow device.

The safety for capturing, generating a first simulation device and a first virtual service by the simulator corresponding to the entity device in the server, and configuring a virtual address for each first simulation device, when the virtual communication address receives the flow data packet from the attack device, and forwarding it to the corresponding first simulation device and/or the first virtual service, the method generates the simulation device by the simulator corresponding to the entity device, it can save a lot of server resource, so as to reduce the cost of deploying the honey net.

In one embodiment, the server generates a shadow device of at least one entity device, comprising: in the idle IP address space, batch generating virtual IP address of a plurality of different physical addresses; generating the corresponding relation between the shadow device and the first simulation device and the first simulation service according to the mapping relationship between the shadow device and the entity device; according to the corresponding relation, associating a plurality of virtual IP address associated with

each first simulation device and the first simulation service.
Specifically, the virtual communication address of the idle virtual communication address in the virtual operation system selects virtual communication address of different physical addresses, because the shadow device is a mapping of the entity device, the mapping relation of the shadow device and the entity device can be one-to-one mapping relation, also can be multiple-to-one mapping relationship. the first simulation device is generated by the simulator corresponding to the entity device, so the first simulation device and the entity device are one-to-one corresponding relation, according to the corresponding relation between the first simulation device and the entity device and the mapping relation of the shadow device and the entity device to obtain the corresponding relation between the shadow device and the first simulation device, then the plurality of virtual addresses according to the respective first simulation device associated with each other.
In this embodiment, the virtual communication address is configured for the first simulation device and the first simulation service, so that the attacker sends attack flow packet to the virtual communication address, and the attack flow packet is forwarded to the first simulation device and the first simulation service through the shadow device, so as to collect the attack information passive discovery attack behaviour, resisting external network attack.
FIG. 3 is a flow diagram of the safety for trapping according to one embodiment of the invention.
In one embodiment, the traffic data packet comprises an application layer message, referring to FIG. 3, the trapping safety further comprises:
S302, when the virtual communication address receives the first message from the attack device, the shadow device corresponding to the virtual communication address forwarding the first message to the corresponding first simulation device and the first virtual service.
wherein the first message refers to the application layer message in the flow data packet sent by the attack device to the virtual communication address. Specifically, the virtual communication address in the virtual operation system receives the first message from the attack device, the shadow device can through the network address conversion technology to the public address of the first message into local area network address, transmitting the first message to the first simulation device or the first simulation service according to the local area network address.
S304, when the first simulation device and the first virtual service receives the first message sent by the attack device response corresponding to the second message.
Because the first virtual service comprises a simulation network protocol, the first simulation device and/or the first virtual service receives the first message sent by the attack device, according to the simulation network protocol response to the first message, the response message is sent to the attack device.
S306, the content of the first message and the second message are stored in the data table one by one.
Specifically, for part using the unencrypted protocol of the industrial control device, capturing the flow of the client and the service end, extracting the interactive application layer message, to " key name: The form of the key value is in the data table in the database.
Illustratively, the application layer message sent by the attack device is AAAA, the application layer message replied by the service end is BBBB, the database is stored as AAAA: BBBB; after using the flow learning result, if the server detects the application layer message sent by the attack device is AAAA, the service end returns the application layer message BBBB corresponding to the AAAA.
S308, when the first simulation device and the first virtual service receives the target application layer message of the target attack device, the content of the target message is matched with the content of the first message in the data table, if the matching is successful, then the second message corresponding to the successfully matched first message is sent to the attack device.
Illustratively, when the first simulation device and the first virtual service when using flow learning, function the server detects the application layer message sent by the attack device is AAAA, searching the same content with the AAAA message from the data table stored in the server, if the content is the first message of AAAA in the data table, then obtaining the second message corresponding to the first message, and forwarding the second message to the attack device through the shadow device corresponding to the virtual communication address.
In this embodiment, according to the flow data packet between the attack device and the server, extracting the application layer message in the interactive data packet, when detecting the same application layer

message of the first message stored in the database, the server can directly return the second message corresponding to the first message in the database, The invention improves the response efficiency of the flow data packet sent by the attack device.
FIG. 4 is another embodiment of the invention claims safety capture method of the flow diagram.
In one embodiment, the trapping safety further comprises:
S402, according to the history attack flow data log generating attack image of the target attack device. wherein the attack portrait refers to the intrusion attack device for identity authentication.
Specifically, the server obtains the historical flow data record generated by each attack device; then obtaining the key field from the history attack flow data record, wherein the key field comprises attack device address, attack device fingerprint and the user name of the specific software of the attack device login Illustratively, specific software Ai-cui, QQ music, China's professional IT community, and so on. if at least one of the key field in the flow data sent by any attack device and the target attack device is the same, then determining the attack device and the target attack device are the same attack device.
S404, collecting the attack behaviour of the attack image of the target attack device, generating the alarm behaviour log corresponding to the target attack device.
Specifically, the server analyzes the attack behaviour belonging to the same device from the historical flow data packet, analyzing all the attack behaviour, and extracting the alarm behaviour, combining the alarm behaviour of the execution time, the executor and generating the alarm behaviour log corresponding to the attack device.
In this embodiment, the server to attack the communication address of the device, device fingerprint, social account number three latitude associated with the external attacker image, and recording the change condition of the communication address of the attack device, solving the manual modification of the communication address, dynamic host configuration protocol lease expiration of communication address change condition caused by subsequent attack device is difficult to trace.
In one embodiment, the method further comprises: sending the alarm behaviour log corresponding to the attack device to the target server, so that the target server sends the alarm behaviour log to the client. wherein the target server is used for storing and analyzing various alarm behaviour log generated in the interaction process of the server and the attack device, interactive flow log and other data.
honey pot generating server sends the alarm action log generated by interaction with the attack device to the target server, the target server sends the alarm action log to the client, so as to alarm the manager to timely manually intervene the attack device.
In one embodiment, the method further comprises: orderly counting the attack behaviour total number in the attack flow data in each preset time period and the total number of the alarm attack behaviour; generating an alarm attack behaviour tendency chart according to the total number of attack behaviour and the total number of alarm attack behaviour in each preset time period, and displaying the alarm attack behaviour tendency chart in the monitoring large screen.
Specifically, the server the attack behaviour total number in each preset time period and alarm attack behaviour total number, and counting the ratio of alarm attack behaviour number and attack behaviour number of each preset time period, and the alarm attack behaviour number and number ratio of each preset time period to generate trend graph, and displaying the trend graph in the monitoring large screen; In addition, the large screen can also display early warning level, total number of assets (total number of internal attackers), asset high risk number (internal attacker high risk), shadow device number, attack network topology, high risk behaviour rolling playing, attack shadow device type statistics, attack behaviour type statistics, high risk alarm trend/total alarm trend
In this embodiment, the high risk behaviour rolling play, attack shadow device type statistics, attack behaviour type statistics, high risk alarm trend/total alarm trend and real-time display on the monitoring large screen, can make the manager timely monitors the attack device high risk behaviour and alarm behaviour.
FIG. 5 is an application scene schematic diagram provided by one embodiment of the present application.
In one embodiment, as shown in FIG. 5, the invention claims an industrial internet fraud trapping system applied to industrial internet platform.
The system comprises a data display layer, a data storage layer, a data processing layer and a data collecting layer. wherein the data collecting layer comprises an analogue service module and a shadow device module, the simulation service comprises industrial control service, traditional IT service, network

device and each industrial control device; The shadow device comprises a virtual communication address response module and a port response module. the data processing layer comprises the original flow log, original attack behaviour log, original attacker information data merging, data analysis, collecting external threat intelligence and multiple reverse tracing method. the data storage layer is used for the alarm behaviour log, alarm event log, request flow log, interactive flow log and other data to store. the data display layer comprises a real-time monitoring module and a platform display module.

in the trapping safety, firstly simulating the corresponding first simulation device in the data collecting layer through the industrial control device simulator, then simulating the protocol of the code level to obtain the first virtual service, the server through the trapping management module to the first simulation device and the first virtual service according to field deployment condition to generate honey field . For example: if the safety is deployed under the working condition, the server generates the host computer, printer and switch and other analog network device, then generating various simulation network protocol needed by the office network through the server.

the server sends the shadow device management strategy to the shadow device module, the shadow device module in the virtual system idle IP address space, batch generating a plurality of virtual IP address of different physical address; generating the corresponding relation between the shadow device and the first simulation device and the first simulation service according to the mapping relationship between the shadow device and the entity device; then according to the corresponding relation, associating a plurality of virtual IP address associated with each first simulation device and the first simulation service, so that the first simulation device and the first simulation service are covered by the network. forming a low interaction service simulation for the shadow device without high interaction according to the policy response specific port and the service open state, introducing the attack flow into the simulation service module by means of network address conversion for the shadow device configured with high interaction, forming a high interaction service simulation. when the attack device attacks the target network in the process of touching the virtual shadow device, the shadow device the attack flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service.

at the same time, the data collecting and analyzing module will collect and record the attack flow data packet of the recording attack device interaction with the shadow device, and analyzing and analyzing the flow, generating the original flow log and sending to the data processing module. when the attack device starts the network scan and network attack, the data analysis module of the data collecting and analyzing module will detect the corresponding attack, such as scanning, password blasting, vulnerability utilization, industrial control attack, and other network attack behaviour, and generating the original alarm behaviour log to the data processing module. when the attacker finishes the attack and obtains the simulation service authority, the data collecting and analyzing module will collect the host action of the attacker to the simulation service, comprising a sample transmitted to the simulation service, generating the original alarm behaviour log to send to the data processing module.

after the data processing module receives the original flow log, generating the attack image of the target attack device according to the history attack flow data log. Specifically, the data processing module takes the physical communication address as root base, the external network uses communication address, device fingerprint, social account number three latitude extracting the attacker information and inquiring by domain name, domain name reverse analysis, domain name location, external threat intelligence, port scanning and other reverse traceability path to the attacker to trace the source, completing the attacker information then the data processing module divides the attack flow into interaction flow and request flow according to the type of the shadow device, then supplementing shadow device according to type of attack flow, source communication address geographic location and session address and so information, and storing all the information into the database. the other data processing module is further used for collecting the attack behaviour of the attack image of the target attack device, generating the alarm behaviour log corresponding to the target attack device. then taking the attacker and the attacked shadow device as the condition, merging the alarm behaviour log, extracting the label, attack times and so on information generate alarm event log. the data processing module is used for analyzing the sample extracted from the original alarm behaviour log in the sandbox, and calling the threat information to inquire the related information and finally the sample data.

at last, collecting the industrial internet spoofing trapping system in the data display layer, the analyzed data

is visually displayed. the real-time monitoring module mainly displays the real-time situation of the industrial internet spoofing trapping system deployment area, comprising a pre-warning level, the total number of attacks in the first month, the number of the high-risk behavior in the first month, the total number of the assets (total number of the attackers), the high risk number of the assets (high risk number of the inner network attackers), the number of the shadow devices, and the attack network topology, high risk behaviour rolling playing, attack shadow device type statistics, attack behaviour type statistics, high risk alarm trend/total alarm trend statistics data. and the data display layer also can send the alarm behaviour log timing to the target server, so as to prompt the manager to pay attention to the high-risk behaviour of the attack device.

The system is a safety trapping product suitable for industrial internet. different from the traditional honey net system is only used for collecting the attack information passive discovery attack behaviour, resisting external network attack, industrial internet fraud trapping system has active defense capability, can by simulating a plurality of industrial control device to construct virtual shadow device, building virtual honey net, trapping the attack behaviour of the industrial control network, and realizing the detection of each kind of attack behaviour, recognition and early warning ability. At the same time, through reverse engineering and large data analysis technology, for the attack of the industrial control network, quickly locating the suspicious behaviour, recognition network attack, timely implementing the safety protection, omni-directional threat early warning.

It should be understood that, although each of the steps involved in the above embodiments is displayed in turn according to the indication of the arrows, these steps are not necessarily performed sequentially in the order indicated by the arrows. Unless otherwise specified herein, the execution of these steps is not strictly limited in order, and these steps can be performed in other orders. Furthermore, at least a portion of the steps involved in the above embodiments may include a plurality of steps or stages, these steps or stages are not necessarily performed at the same time, but can be executed at different times, the execution order of these steps or stages is not necessarily performed in turn, and it can be executed alternately or alternately with other steps or at least a part of the step or stage in other steps.

FIG. 6 is a structure schematic safety of the trapping device provided by the embodiment of the invention. In one embodiment, as shown in FIG. 6, providing a safety device 600, comprising: The service simulation module 602, the device generation module 604 and the traffic forwarding module 606, wherein:

service simulation module 602, for generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device.

The device generating module 604 generates at least one shadow device of the entity device, the shadow device is a mapping of the entity device, and each shadow device corresponds to a plurality of virtual communication addresses.

flow forwarding module 606, for when the virtual communication address received from the attack device of the flow data packet, if the virtual communication address corresponding to the shadow device is high interaction, then the flow data packet is forwarded to the corresponding first simulation device and/or the first virtual service through the shadow device.

In one embodiment, the device generating module 604 comprises: an address generating sub-module, for in the idle IP address space, batch generating a plurality of virtual IP address of different physical address; corresponding relation sub-module, for according to the mapping relationship between the shadow device and the entity device to generate the corresponding relation between the shadow device and the first simulation device and the first simulation service; associated sub-module, for according to the corresponding relation, the plurality of virtual IP address associated with each first simulation device and the first simulation service.

In one embodiment, the device further comprises: a forwarding sub-module, for when the virtual communication address receives the attack application layer message from the attack device, the shadow device corresponding to the virtual communication address of the attack application layer message is forwarded to the corresponding first simulation device and the first virtual service; a second message generating sub-module, for when the first simulation device and the first virtual service receiving the first message sent by the attack device response corresponding to the second message; a storage sub-module,

for correspondingly storing the content of the first message and the second message in the data table one by one; a flow learning sub-module, for when the first simulation device and the first virtual service receives the target application layer message of the target attack device, the content of the target message is matched with the content of the first message in the data table, if the matching is successful, then sending the second message corresponding to the successfully matched first message to the attack device.
In one embodiment, the device further comprises: attack picture sub-module, used for according to the history attack flow data log generating attack picture of the target attack device; a behaviour sun sub-module, used for collecting the attack behaviour of the attack image of the target attack device, generating the alarm behaviour log corresponding to the target attack device.
In one embodiment, the attack image sub-module is further used for obtaining the historical flow data record generated by each attack device; obtaining key field from the attack flow data record, wherein the key field comprises attack device communication address, attack device fingerprint and the user login of the specific software if at least one of the key field in the flow data sent by any attack device and the target attack device is the same, then determining the attack device and the target attack device are the same attack device.
In one embodiment, the device is further used for sending the alarm behaviour log corresponding to the attack device to the target server, so that the target server sends the alarm behaviour log to the client.
In one embodiment, the device is further used for orderly counting the attack behaviour total number in the attack flow data in each preset time period and the total number of alarm attack behaviour; generating an alarm attack behaviour tendency chart according to the total number of attack behaviour and the total number of alarm attack behaviour in each preset time period, and displaying the alarm attack behaviour tendency chart in the monitoring large screen.
Each module in the robot teaching device can be all or partially implemented by software, hardware and combinations thereof. the each module can be embedded in the hardware form or independent from the processor in the computer device, also can be stored in the memory of the computer device in software form, so as to facilitate the processor to perform the operation corresponding to each module.
FIG. 7 is a structure schematic diagram of the terminal device provided by the embodiment of the invention. In one embodiment, there is provided a computer device, the computer device can be a server, the internal structure diagram can be shown in FIG. 7. The computer device comprises a processor connected by a system bus, a memory and a network interface. wherein the processor of the computer device is used for providing calculation and control capability. The memory of the computer device includes a non-volatile storage medium and an internal memory. The non-volatile storage medium is stored with an operating system, a computer program and a database. The internal memory provides an environment for operating systems and computer programs in a non-volatile storage medium The database of the computer device is used for storing path track data. The network interface of the computer device is used for communicating with the external terminal through the network connection The computer program is executed by a processor to implement a robot teaching method.
It will be understood by those skilled in the art, the structure shown in FIG. 7, is only a block diagram of the partial structure associated with the present application scheme, and does not form the application scheme applied to the computer device of the definition, the specific computer device may include more or less than shown in the figure of the component, or a combination of certain components, or a different component arrangement.
In one embodiment, there is provided a computer device, comprising a memory and a processor, the memory is stored with a computer program, the processor executes the computer program to realize the safety one of the embodiments provided in the trapping method.
In one embodiment, there is provided a computer readable storage medium which is stored with a computer program, a computer program when executed by a processor to realize the safety provided in any one embodiment of the trapping method.
In one embodiment, there is provided a computer program product, comprising a computer program, when the computer program is executed by a processor to implement the method provided in any one safety of the capture method.
It should be noted that, the present application relates to the user (information but not limited to user equipment information user personal information and so on) and data (including but not limited to data for analysis, stored data, data display), are authorized by the user or fully authorized by each information of the

data.
Those skilled in the art can understand all or part of the flow in the above embodiment method, which can be finished by computer program instruction related hardware to finish, the computer program can be stored in a non-volatile computer storage medium, the computer program when executing, The flow of the embodiments of the above methods may be included. wherein any reference of the memory, database or other medium used in each embodiment provided by the present application, may include at least one of non-volatile and volatile memory. The non-volatile memory may include a Read Only Memory (Read-OnlyMemory, ROM), a tape, a floppy disk, a flash memory, an optical memory, a high density embedded non-volatile memory, a resistive random access memory (ReRAM), Magnetoresistive Random Access Memory (MRAM), Ferroelectric Random Access Memory (FRAM), phase change memory (Phase Change Memory, PCM), graphene memory. The volatile memory may include a Random Access Memory (Random Access Memory, RAM) or an external cache. As explanation and not limitation, RAM can be a plurality of forms, such as static random access memory (Static Random Access Memory, SRAM) or dynamic random access memory (Dynamic RandomAccess Memory, DRAM) and so on. The database involved in each embodiment of the present application may include at least one of a type database and a non-type database. The non-type database may include a blockchain-based distributed database, and the like, not limited to this. The processor involved in the embodiments provided by the present application may be a general purpose processor, a central processor, a graphics processor, a digital signal processor, a programmable logic device, a data processing logic based on quantum computation, and so on, not limited to this.
each technical feature of the above embodiments can be any combination, to make the description is simple, not all possible combinations of all technical features in the above embodiments are described, however, as long as the combination of these technical characteristics does not exist contradiction, should be considered as the scope of the specification.
The above embodiments only express several embodiments of the present application, the description is more specific and detailed, but not to be understood as the limit of the patent application range of the invention. It should be noted that, for the common technician in the field, without departing from the conception of the invention, it also can make several deformation and improvement, these belong to the protection range of the invention. Therefore, the protection scope of the present application should be based on the appended claims.

Claims

1. A safety for trapping and trapping, wherein the safety comprises the following steps: generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service is according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device; generating at least one shadow device of the entity device, the shadow device is a mapping of the entity device, each of the shadow device corresponding to a plurality of virtual communication address; when the virtual communication address receives the flow data packet from the attack device, if the shadow device corresponding to the virtual communication address is high interaction, then forwarding the flow data packet to the corresponding first simulation device and/or the first virtual service through the shadow device.
2. The safety according to claim 1, wherein the step of generating at least one shadow device of the entity device comprises: generating at least one shadow device of the entity device according to the image information of the object device, in the idle IP address space, batch generating virtual IP address of a plurality of different physical addresses; generating the corresponding relation between the shadow device and the first simulation device and the first simulation service according to the mapping relationship between the shadow device and the entity device; according to the corresponding relation, associating a plurality of the virtual IP address associated with each of the first simulation device and the first simulation service.
3. The safety for trapping according to claim 1, wherein the flow data packet comprises an application layer message, the method further comprises the following steps: when the virtual communication address receives the attack application layer message from the attack device, the shadow device corresponding to

the virtual communication address transfers the attack application layer message to the corresponding first simulation device and the first virtual service; when the first analogue device and the first virtual service receive the first message sent by the attack device, responding to the corresponding second message; correspondingly storing the contents of the first message and the second message in the data table one by one; when the first simulation device and the first virtual service receives the target application layer message of the target attack device, matching the content of the target message with the content of the first message in the data table, if the matching is successful, then sending the second message corresponding to the first message matched successfully to the attack device.

4. The safety for trapping according to claim 1, wherein the method further comprises: generating the attack image of the target attack device according to the history attack flow data log; collecting the attack behaviour of the attack image of the target attack device, generating the alarm behaviour log corresponding to the target attack device.

5. The safety according to claim 4, wherein the step of generating the attack image of the target attack device according to the history attack flow data log comprises: obtaining the historical flow data record generated by each said attacking device; obtaining key field from the attack flow data record, wherein the key field comprises attack device communication address, attack device fingerprint and user login of the specific software if at least one of the key field in the flow data sent by any attack device and the target attack device is the same, then determining that the attack device and the target attack device are the same attack device.

6. The safety for trapping according to claim 1, wherein the method further comprises: sending the alarm behaviour log corresponding to the attack device to the target server, so that the target server sends the alarm behaviour log to the client.

7. The safety for trapping according to claim 1, wherein the method further comprises: orderly counting the attack behaviour total number in the attack flow data in each preset time period and the total number of the alarm attack behaviour; generating an alarm attack behaviour tendency chart according to the total number of attack behaviour and the total number of alarm attack behaviour in each preset time period, and displaying the alarm attack behaviour trend graph in the monitoring large screen.

8. An safety device, wherein the device comprises: service simulation module, used for generating a plurality of categories of the first simulation device and the first virtual service, the first simulation device and the first virtual service according to the device parameter of the entity device and the application scene of the entity device, generated by the simulator corresponding to the entity device; device generating module, generating at least one shadow device of the entity device, the shadow device is a mapping of the entity device, each of the shadow device corresponding to a plurality of virtual communication address; a flow forwarding module, for when the virtual communication address receives the flow data packet from the attack device, if the shadow device corresponding to the virtual communication address is high interaction, then forwarding the flow data packet to the corresponding first simulation device and/or the first virtual service through the shadow device.

9. A terminal device, comprising a memory, a processor and a computer program stored in the memory and capable of running on the processor, wherein the processor executes the method according to any one of claims 1 to 1 to 7 when executing the computer program.

10. A computer readable storage medium wherein the computer readable storage medium stored with a computer program, wherein the computer program is executed by a processor to implement the method according to any one of claims 1 to 7.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 114584359 A
(43)申请公布日 2022.06.03

(21)申请号 202210178181.1

(22)申请日 2022.02.24

(71)申请人 烽台科技(北京)有限公司
地址 100000 北京市海淀区北坞村路甲25号静心园I座一层101室

(72)发明人 方永成 赵重浩 刘茂林 龚亮华

(74)专利代理机构 深圳中一联合知识产权代理有限公司 44414
专利代理师 张延薇

(51)Int.Cl.
H04L 9/40(2022.01)
H04L 49/00(2022.01)

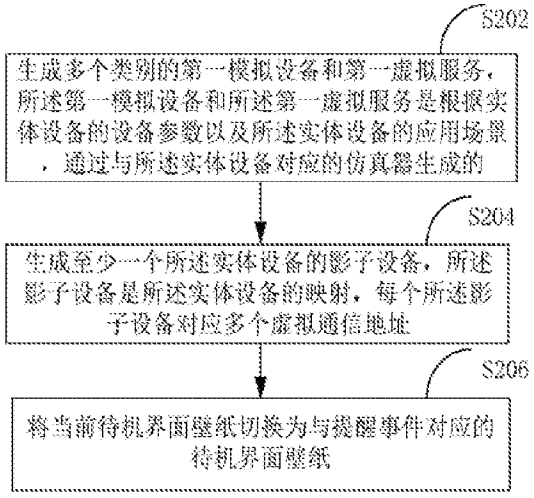
权利要求书2页 说明书10页 附图5页

(54)发明名称

安全诱捕方法、装置和计算机设备

(57)摘要

本申请适用于网络安全技术领域，提供了一种安全诱捕方法，包括：生成多个类别的第一模拟设备和第一虚拟服务，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的；生成实体设备的影子设备，每个影子设备对应多个虚拟通信地址；当所述虚拟通信地址接收到来自攻击设备的流量数据包时，若虚拟通信地址对应的影子设备为高交互，则通过影子设备将流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。该方法中，通过与实体设备对应的仿真器来生成模拟设备，进而实现工业互联网欺骗诱捕系统的部署。由于无需生成多个容器，所以可以节省大量的服务器资源，从而减少部署工业互联网欺骗诱捕系统的成本。



CN 114584359 A

1.一种安全诱捕方法,其特征在于,所述安全诱捕方法,包括:

生成多个类别的第一模拟设备和第一虚拟服务,所述第一模拟设备和所述第一虚拟服务是根据实体设备的设备参数以及所述实体设备的应用场景,通过与所述实体设备对应的仿真器生成的;

生成至少一个所述实体设备的影子设备,所述影子设备是所述实体设备的映射,每个所述影子设备对应多个虚拟通信地址;

当所述虚拟通信地址接收到来自攻击设备的流量数据包时,若所述虚拟通信地址对应的影子设备为高交互,则通过所述影子设备将所述流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

2.如权利要求1所述的安全诱捕方法,其特征在于,所述生成至少一个所述实体设备的影子设备,包括:

在空闲的IP地址空间中,批量生成多个不同物理地址的虚拟IP地址;

根据影子设备与所述实体设备之间的映射关系生成影子设备与所述第一模拟设备以及第一模拟服务之间的对应关系;

根据所述对应关系,将多个所述虚拟IP地址关联上各个所述第一模拟设备以及第一模拟服务。

3.如权利要求1所述的安全诱捕方法,其特征在于,所述流量数据包中包含应用层报文,所述方法还包括:

当所述虚拟通信地址接收到来自攻击设备的攻击应用层报文,与所述虚拟通信地址相对应的影子设备将所述攻击应用层报文转发至对应的第一模拟设备以及第一虚拟服务;

当所述第一模拟设备以及第一虚拟服务接收到所述攻击设备发送的第一报文后响应相对应的第二报文;

将所述第一报文与所述第二报文的内容一一对应地存储在数据表中;

当所述第一模拟设备以及第一虚拟服务接收到目标攻击设备的目标应用层报文时,将所述目标报文的内容与所述数据表中的第一报文的内容相匹配,若匹配成功,则将匹配成功的第一报文相对应的第二报文发送至所述攻击设备。

4.如权利要求1所述的安全诱捕方法,其特征在于,所述方法还包括:

根据所述历史攻击流量数据日志生成目标攻击设备的攻击画像;

收集所述目标攻击设备的攻击画像的攻击行为,生成与所述目标攻击设备对应的告警行为日志。

5.如权利要求4所述的安全诱捕方法,其特征在于,所述根据所述历史攻击流量数据日志生成目标攻击设备的攻击画像,包括:

获取各个所述攻击设备产生的历史流量数据记录;

从所述攻击流量数据记录中获取关键字段,其中所述关键字段包括攻击设备地通信地址、攻击设备指纹以及攻击设备上登录的特定软件的用户名;

若任一攻击设备与目标攻击设备发送的流量数据中的关键字段中至少一项相同,则认定所述攻击设备与所述目标攻击设备为同一攻击设备。

6.如权利要求1所述的安全诱捕方法,其特征在于,所述方法还包括:

将与所述攻击设备对应的告警行为日志发送目标服务器,以使目标服务器将所述告警

行为日志发送至客户端。

7.如权利要求1所述的安全诱捕方法，其特征在于，所述方法还包括：

依次统计每一预设时间段内的攻击流量数据中的攻击行为总数以及告警攻击行为总数；

根据每一预设时间段内的攻击行为总数以及告警攻击行为总数生成告警攻击行为趋势图，并将所述告警攻击行为趋势图显示在监视大屏中。

8.一种安全诱捕装置，其特征在于，所述装置包括：

服务模拟模块，用于生成多个类别的第一模拟设备和第一虚拟服务，所述第一模拟设备和所述第一虚拟服务是根据实体设备的设备参数以及所述实体设备的应用场景，通过与所述实体设备对应的仿真器生成的；

设备生成模块，生成至少一个所述实体设备的影子设备，所述影子设备是所述实体设备的映射，每个所述影子设备对应多个虚拟通信地址；

流量转发模块，用于当所述虚拟通信地址接收到来自攻击设备的流量数据包时，若所述虚拟通信地址对应的影子设备为高交互，则通过所述影子设备将所述流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

9.一种终端设备，包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序，其特征在于，所述处理器执行所述计算机程序时实现如权利要求1至7任一项所述的方法。

10.一种计算机可读存储介质，所述计算机可读存储介质存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现如权利要求1至7任一项所述的方法。

安全诱捕方法、装置和计算机设备

技术领域

[0001] 本申请属于网络安全技术领域，尤其涉及一种安全诱捕方法及装置。

背景技术

[0002] 随着工业转型升级，工业互联网被广泛应用。但是，工业互联网在应用时存在网络信息安全方面的问题，如信息泄露、遭受网络攻击、存在可被利用的安全漏洞等问题。目前可以通过工业互联网欺骗诱捕系统进行主动防御，获取威胁情报。

[0003] 但是，现有的工业互联网欺骗诱捕系统在部署时需要占用大量的服务器资源，部署成本较高。

发明内容

[0004] 本申请实施例提供了安全诱捕方法及装置，可以解决工业互联网欺骗诱捕系统在部署时需要占用大量的服务器资源，部署成本较高的问题。

[0005] 第一方面，本申请实施例提供了一种安全诱捕方法，包括：生成多个类别的第一模拟设备和第一虚拟服务，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的；生成至少一个实体设备的影子设备，影子设备是实体设备的映射，每个影子设备对应多个虚拟通信地址；当虚拟通信地址接收到来自攻击设备的流量数据包时，若虚拟通信地址对应的影子设备为高交互，则通过影子设备将流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

[0006] 在第一方面的一种可能的实现方式中，生成至少一个实体设备的影子设备，包括：在空闲的IP地址空间中，批量生成多个不同物理地址的虚拟IP地址；根据影子设备与实体设备之间的映射关系生成影子设备与第一模拟设备以及第一模拟服务之间的对应关系；根据对应关系，将多个虚拟IP地址关联上各个第一模拟设备以及第一模拟服务。

[0007] 在第一方面的一种可能的实现方式中，流量数据包中包含应用层报文，方法还包括：当虚拟通信地址接收到来自攻击设备的攻击应用层报文，与虚拟通信地址相对应的影子设备将攻击应用层报文转发至对应的第一模拟设备以及第一虚拟服务；当第一模拟设备以及第一虚拟服务接收到攻击设备发送的第一报文后响应相对应的第二报文；将第一报文与第二报文的内容一一对应地存储在数据表中；当第一模拟设备以及第一虚拟服务接收到目标攻击设备的目标应用层报文时，将目标报文的内容与数据表中的第一报文的内容相匹配，若匹配成功，则将匹配成功的第一报文相对应的第二报文发送至攻击设备。

[0008] 在第一方面的一种可能的实现方式中，该方法还包括：根据历史攻击流量数据日志生成目标攻击设备的攻击画像；收集目标攻击设备的攻击画像的攻击行为，生成与目标攻击设备对应的告警行为日志。

[0009] 在第一方面的一种可能的实现方式中，该方法还包括：根据历史攻击流量数据日志生成目标攻击设备的攻击画像，包括：获取各个攻击设备产生的历史流量数据记录；从攻击流量数据记录中获取关键字段，其中关键字段包括攻击设备地通信地址、攻击设备指纹

0417

以及攻击设备上登录的特定软件的用户名；若任一攻击设备与目标攻击设备发送的流量数据中的关键字段中至少一项相同，则认定攻击设备与目标攻击设备为同一攻击设备。

[0010] 在第一方面的一种可能的实现方式中，该方法还包括：将与攻击设备对应的告警行为日志发送目标服务器，以使目标服务器将告警行为日志发送至客户端。

[0011] 在第一方面的一种可能的实现方式中，该方法还包括：依次统计每一预设时间段内的攻击流量数据中的攻击行为总数以及告警攻击行为总数；根据每一预设时间段内的攻击行为总数以及告警攻击行为总数生成告警攻击行为趋势图，并将告警攻击行为趋势图显示在监视大屏中。

[0012] 第二方面，本申请实施例提供了一种安全诱捕装置，包括：服务模拟模块，用于生成多个类别的第一模拟设备和第一虚拟服务，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的；设备生成模块，生成至少一个实体设备的影子设备，影子设备是实体设备的映射，每个影子设备对应多个虚拟通信地址；流量转发模块，用于当虚拟通信地址接收到来自攻击设备的流量数据包时，若虚拟通信地址对应的影子设备为高交互，则通过影子设备将流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

[0013] 第三方面，本申请实施例提供了一种终端设备，终端设备包括存储器和处理器，存储器存储有计算机程序，处理器执行计算机程序时实现以下步骤：包括：生成多个类别的第一模拟设备和第一虚拟服务，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的；生成至少一个实体设备的影子设备，影子设备是实体设备的映射，每个影子设备对应多个虚拟通信地址；当虚拟通信地址接收到来自攻击设备的流量数据包时，若虚拟通信地址对应的影子设备为高交互，则通过影子设备将流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

[0014] 第四方面，本申请实施例提供了一种计算机可读存储介质，计算机可读存储介质，其上存储有计算机程序，计算机程序被处理器执行时实现以下步骤：包括：生成多个类别的第一模拟设备和第一虚拟服务，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的；生成至少一个实体设备的影子设备，影子设备是实体设备的映射，每个影子设备对应多个虚拟通信地址；当虚拟通信地址接收到来自攻击设备的流量数据包时，若虚拟通信地址对应的流量数据包转发至相对应的第一模拟设备。

[0015] 第五方面，本申请实施例提供了一种计算机程序产品，当计算机程序产品在终端设备上运行时，使得终端设备执行上述第一方面中任一项安全诱捕方法。

[0016] 可以理解的是，上述第二方面至第五方面的有益效果可以参见上述第一方面中的相关描述，在此不再赘述。

[0017] 本申请实施例与现有技术相比存在的有益效果是：上述安全诱捕方法、装置、计算机设备、存储介质和计算机程序产品，在服务器中通过与实体设备对应的仿真器生成第一模拟设备以及第一虚拟服务，并为各个第一模拟设备配置虚拟通信地址，当该虚拟通信地址接收到来自攻击设备的流量数据包时，并将其转发至发生对应的第一模拟设备和/或第一虚拟服务。在本方法中，通过与实体设备对应的仿真器来生成模拟设备，进而实现工业互联网欺骗诱捕系统的部署。由于无需生成多个容器，所以可以节省大量的服务器资源，从而

减少部署工业互联网欺骗诱捕系统的成本。

附图说明

[0018] 为了更清楚地说明本申请实施例中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本申请的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

[0019] 图1是本申请一实施例提供的安全诱捕方法的应用环境图；

[0020] 图2是本申请一实施例提供的设备生成步骤的流程示意图；

[0021] 图3是本申请一实施例提供的安全诱捕方法的流程示意图；

[0022] 图4是本申请另一实施例提供的安全诱捕方法的流程示意图；

[0023] 图5是本申请一实施例提供的应用场景示意图；

[0024] 图6是本申请实施例提供的安全诱捕装置的结构示意图；

[0025] 图7是本申请实施例提供的终端设备的结构示意图。

具体实施方式

[0026] 为了使本申请的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本申请进行进一步详细说明。应当理解，此处描述的具体实施例仅仅用以解释本申请，并不用于限定本申请。

[0027] 图1是本申请一实施例提供的安全诱捕方法的应用环境图。

[0028] 本申请实施例提供的安全诱捕方法，可以应用于如图1所示的应用环境中。其中，攻击设备终端102通过网络与服务器104进行通信。数据存储系统可以存储服务器104需要处理的数据。数据存储系统可以集成在服务器104上，也可以放在云上或其他网络服务器上。服务器104生成多个类别的第一模拟设备和第一虚拟服务，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的；生成至少一个实体设备的影子设备，影子设备是实体设备的映射，每个影子设备对应多个虚拟通信地址；当虚拟通信地址接收到来自攻击设备102发送的流量数据包时，若虚拟通信地址对应的影子设备为高交互，则通过影子设备将流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。其中，攻击设备终端可以但不限于手机、平板电脑还可以是物联网设备和便携式可穿戴设备，物联网设备可为智能音箱、智能电视、智能空调、智能车载设备等。便携式可穿戴设备可为智能手表、智能手环、头戴设备等。服务器104可以用独立的服务器或者是多个服务器组成的服务器集群来实现。

[0029] 图2是本申请一实施例提供的设备生成步骤的流程示意图。

[0030] 在一个实施例中，如图2所示，提供了一种安全诱捕方法，以该方法应用于图1中的服务器104为例进行说明，包括以下步骤：

[0031] S202，生成多个类别的第一模拟设备和第一虚拟服务，其中，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的。

[0032] 具体地，服务器可以通过实体工控设备厂家发布的与实体工控设备相对应的仿真

器来实现对实体工控设备的仿真，再将对应的固件加载进入工控设备仿真模型中，从而实现对实体工控设备的仿真得到第一模拟设备。例如，当服务器需要模拟的工控设备为西门子河图工控设备时，将西门子厂家发布的西门子仿真软件与固件一起进行仿真，从而得到与西门子河图工控设备相对应的第一模拟设备。

[0033] 第一虚拟服务则可以是根据协议的定义，通过代码层级的协议仿真，进而实现对网络服务仿真得到的。例如，当仿真ssh(Secure Shell，安全外壳协议)时，服务端接收到来自客户端发送过来的请求后，先进行三次握手，建立tcp(Transmission Control Protocol，传输控制协议)连接。然后双方发送自己所支持的版本号以及算法，协商确定版本及算法。接着客户端根据服务端发送过来的数据生成一对密钥，将公钥发送给服务器。当客户端接收到后会通过先验证服务器的身份，然后将其保存到本地文件夹中，再使用服务器发送过来的临时公钥生成共享密钥，再加密签名，将其结果与自己所计算的哈希值进行比较，验证数据时没有问题的，从而完成对ssh协议的仿真。

[0034] 当完成对实体工控设备的仿真模拟以及网络协议的仿真模拟后，根据各个行业的生产工艺、办公网、生产网等的场景以及各个实体设备的仿真参数来将各个模拟设备以及虚拟服务组合配置成各种不同类型的工业互联网欺骗诱捕系统。需要说明的是，工业互联网欺骗诱捕系统也可以叫蜜网系统，而蜜网系统又是由蜜场与蜜网组成，其中蜜场由多个模拟设备以及多种虚拟服务组成，蜜网是指覆盖在蜜场上的网络。

[0035] S204，生成至少一个实体设备的影子设备，影子设备是实体设备的映射，每个影子设备对应多个虚拟通信地址。

[0036] 其中，影子设备是可以现有操作系统上的虚拟影像，它具有真实系统完全一样的功能。进入影子系统后，所有操作都是虚拟的，不会对真正的系统产生任何影响。当在现有的操作系统上运行出第一模拟设备时，影子设备是在虚拟操作系统上第一模拟设备的映射。由于流量引入蜜场，实际上攻击者攻击的是蜜场里面的虚拟设备、服务，但是对于攻击者而言，他攻击的是影子设备通过ip虚拟技术虚拟出的通信地址、服务、设备。

[0037] 在其他实施例中，一个影子设备可以对应一个实体设备，也可以对应多个实体设备。当一个影子设备对应多个实体设备时，影子设备获取多个虚拟通信地址，并将多个虚拟通信地址依次分配给各个实体设备。

[0038] S206，当虚拟通信地址接收到来自攻击设备的流量数据包时，若虚拟通信地址对应的影子设备为高交互，则通过影子设备将流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

[0039] 具体地，当虚拟操作系统中影子设备设置的虚拟通信地址接收到来自攻击设备的数据流量包时，若此时虚拟通信地址对应的影子设备为高交互时，影子设备将接收到的流量数据包通过net地址转换的方式把攻击流量数据包引入与该虚拟通信地址对应的第一模拟设备或第一模拟服务。其中，影子设备可以配置成高交互的，也可以配置成低交互的。低交互的影子设备只能模仿少量的互联网协议和网络服务，而高交互的影子设备既可以模仿工控设备、也可以模拟大多数的互联网协议以及网络服务。

[0040] 需要说明的是，当该影子设备为低交互时，则只需通过影子设备上特定的端口将攻击设备发来的数据流量转发至第一模拟设备或者第一模拟服务。

[0041] 上述安全诱捕方法，在服务器中通过与实体设备对应的仿真器生成第一模拟设备

以及第一虚拟服务，并为各个第一模拟设备配置虚拟地址，当该虚拟通信地址接收到来自攻击设备的流量数据包时，并将其转发至发生对应的第一模拟设备和/或第一虚拟服务，该方法中通过与实体设备对应的仿真器来生成模拟设备，可以节省大量的服务器资源，从而减少部署蜜网的成本。

[0042] 在一个实施例中，服务器生成至少一个实体设备的影子设备，包括：在空闲的IP地址空间中，批量生成多个不同物理地址的虚拟IP地址；根据影子设备与实体设备之间的映射关系生成影子设备与第一模拟设备以及第一模拟服务之间的对应关系；根据对应关系，将多个虚拟IP地址关联上各个第一模拟设备以及第一模拟服务。

[0043] 具体地，从虚拟操作系统中的空闲虚拟通信地址批量选择出不同物理地址的虚拟通信地址，由于影子设备是实体设备的映射，影子设备与实体设备的映射关系可以是一对一的映射关系，也可以是多对一的映射关系。第一模拟设备是由与实体设备对应的仿真器生成的，因此第一模拟设备与实体设备之间是一对一的对应关系，根据第一模拟设备与实体设备之间的对应关系以及影子设备与实体设备的映射关系得出影子设备与第一模拟设备之间的对应关系，然后将多个虚拟地址根据该对应关联上各个第一模拟设备。

[0044] 本实施例中，通过为第一模拟设备以及第一模拟服务配置虚拟通信地址，从而引诱攻击者向虚拟通信地址发送攻击流量包，并通过影子设备将攻击流量包转发至第一模拟设备以及第一模拟服务，从而可以收集攻击信息、被动发现攻击行为、抵御外部网络攻击。

[0045] 图3是本申请一实施例提供的安全诱捕方法的流程示意图。

[0046] 在一实施例中，流量数据包中包含应用层报文，参考图3，该安全诱捕方法还包括：

[0047] S302，当虚拟通信地址接收到来自攻击设备的第一报文，与虚拟通信地址相对应的影子设备将第一报文转发至对应的第一模拟设备以及第一虚拟服务。

[0048] 其中，第一报文是指攻击设备发送给虚拟通信地址的流量数据包中的应用层报文。具体地，当虚拟操作系统中的虚拟通信地址接收到来自攻击设备的第一报文，影子设备可以通过网络地址转换技术将第一报文的公网地址转换成局域网地址，然后根据该局域网地址将第一报文转发至第一模拟设备或者第一模拟服务。

[0049] S304，当第一模拟设备以及第一虚拟服务接收到攻击设备发送的第一报文后响应相对应的第二报文。

[0050] 由于第一虚拟服务中包含有仿真网络协议，第一模拟设备和/或第一虚拟服务接收到攻击设备发送的第一报文后，会根据仿真网络协议响应第一报文，将响应得到报文发送给攻击设备。

[0051] S306，将第一报文与第二报文的内容一一对应地存储在数据表中。

[0052] 具体地，针对部分使用未加密协议的工控设备，抓取客户端和服务端的流量，提取出交互的应用层报文，以“键名:键值”的形式存在数据库中数据表中。

[0053] 示例性地，攻击设备发送的应用层报文为AAAA，服务端回复的应用层报文为BBBB，数据库中保存为AAAA:BBBB；当使用流量学习的结果后，如果服务器检测到攻击设备发送的应用层报文为AAAA，服务端就返回AAAA对应的应用层报文BBBB。

[0054] S308，当第一模拟设备以及第一虚拟服务接收到目标攻击设备的目标应用层报文时，将目标报文的内容与数据表中的第一报文的内容相匹配，若匹配成功，则将匹配成功的第一报文相对应的第二报文发送至攻击设备。

[0055] 示例性地，当第一模拟设备以及第一虚拟服务当使用流量学习功能后，如果服务器检测到攻击设备发送的应用层报文为AAAA，从服务器中的存储的数据表中查找与AAAA相同内容的报文，若在数据表中存在内容为AAAA的第一报文，则获取与该第一报文相对应的第二报文，并将第二报文通过与该虚拟通信地址相对应的影子设备转发至攻击设备。

[0056] 本实施例中，根据攻击设备与服务器之间的流量数据包，提取出交互数据包中的应用层报文，当检测到与数据库中存储的第一报文相同的应用层报文后，服务器可以直接返回数据库中与该第一报文对应的第二报文，提高了对对攻击设备发送的流量数据包的响应效率。

[0057] 图4是本申请另一实施例提供的安全诱捕方法的流程示意图。

[0058] 在一个实施例中，该安全诱捕方法还包括：

[0059] S402，根据历史攻击流量数据日志生成目标攻击设备的攻击画像。

[0060] 其中，攻击画像是指对入侵攻击设备进行身份认证。

[0061] 具体地，首先服务器获取各个攻击设备产生的历史流量数据记录；然后从历史攻击流量数据记录中获取关键字段，其中关键字段包括攻击设备地通信地址、攻击设备指纹以及攻击设备上登录的特定软件的用户名等。示例性的，特定软件包括爱奇艺、QQ音乐、中国专业IT社区等。若任一攻击设备与目标攻击设备发送的流量数据中的关键字段中至少一项相同，则认定该攻击设备与目标攻击设备为同一攻击设备。

[0062] S404，收集目标攻击设备的攻击画像的攻击行为，生成与目标攻击设备对应的告警行为日志。

[0063] 具体地，服务器从历史流量数据包中解析出属于同一设备的攻击行为，对所有攻击行为进行解析，并从中提取出告警行为，结合告警行为的实行时间、执行者等生成与该攻击设备对应的告警行为日志。

[0064] 本实施例中，服务器以攻击设备的通信地址、设备指纹、社交账号三个纬度对外网攻击者进行关联画像，并记录下攻击设备的通信地址的变化情况，解决针对手动修改通信地址、动态主机配置协议租约到期等通信地址变化的情况导致后续攻击设备追溯困难的问题。

[0065] 在一个实施例中，该方法还包括：将与攻击设备对应的告警行为日志发送目标服务器，以使目标服务器将告警行为日志发送至客户端。

[0066] 其中，目标服务器是用于存储以及解析服务器与攻击设备交互过程中产生的各种告警行为日志、交互流量日志等数据。

[0067] 蜜罐生成服务器将与攻击设备交互产生的告警行为日志发送个目标服务器，目标服务器将该告警行为日志发送给至客户端，以警示管理者能够及时对攻击设备进行人为干预。

[0068] 在一个实施例中，该方法还包括：依次统计每一预设时间段内的攻击流量数据中的攻击行为总数以及告警攻击行为总数；根据每一预设时间段内的攻击行为总数以及告警攻击行为总数生成告警攻击行为趋势图，并将告警攻击行为趋势图显示在监视大屏中。

[0069] 具体地，服务器将每一预设时间段内的攻击行为总数以及告警攻击行为总数，并统计出每一预设时间段的告警攻击行为数量与攻击行为数量之比，并将每一预设时间段的告警攻击行为数量以及数量比生成趋势图，并将该趋势图显示在监视大屏中；另外，大屏上

还可以展示预警等级，资产总数(内网攻击者总数)，资产高危数(内网攻击者高危数)，影子设备数，攻击网络拓扑、高危行为滚动播放、被攻击影子设备类型统计、攻击行为类型统计、高危告警趋势/总体告警趋势等统计数据。

[0070] 本实施例中，将高危行为滚动播放、被攻击影子设备类型统计、攻击行为类型统计、高危告警趋势/总体告警趋势等实时显示在监控大屏上，可以使管理者及时地监察到攻击设备地高危行为和告警行为。

[0071] 图5是本申请一实施例提供的应用场景示意图。

[0072] 在一实施例中，如图5所示，提出了一种应用于工业互联网平台的工业互联网欺骗诱捕系统。

[0073] 该系统中包括数据展示层、数据存储层、数据处理层和数据采集层。其中，数据采集层包括模拟服务模块和影子设备模块，模拟服务中包含工控服务、传统IT服务、网络设备以及各工控设备；影子设备中包括虚拟通信地址响应模块以及端口响应模块。数据处理层包括对原始流量日志、原始攻击行为日志、原始攻击者信息日志进行数据归并、数据分析、收集外网威胁情报等多项反向溯源方法。数据存储层用于对告警行为日志、告警事件日志、请求流量日志、交互流量日志等数据进行存储。数据展示层包括实时监控模块和平台展示模块。

[0074] 在该安全诱捕方法中首先在数据采集层中通过工控设备仿真器仿真出对应的第一模拟设备，然后对代码层级的协议进行仿真得到第一虚拟服务，服务器通过诱捕管理模块将第一模拟设备和第一虚拟服务根据现场部署情况生成蜜场。例如：若该安全诱捕方法是部署在办公工况下，则通过服务器生成主机、打印机以及交换机等模拟网络设备，再通过服务器生成办公网所需的各种仿真网络协议。

[0075] 服务器再将影子设备管理策略下发到影子设备模块，影子设备模块在虚拟系统中空闲的IP地址空间中，批量生成多个不同物理地址的虚拟IP地址；根据影子设备与实体设备之间的映射关系生成影子设备与第一模拟设备以及第一模拟服务之间的对应关系；然后根据该对应关系，将多个虚拟IP地址关联上各个第一模拟设备以及第一模拟服务，使得第一模拟设备以及第一模拟服务被网络所覆盖。对于没有配置高交互的影子设备根据策略响应特定的端口与服务开放状态形成低交互服务模拟，对于配置高交互的影子设备通过网络地址转换的方式把攻击流量引入模拟服务模块，形成高交互服务模拟。当攻击设备攻击目标网络的过程中触碰到虚拟的影子设备时，该影子设备将攻击流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

[0076] 同时，数据采集分析模块会全程采集记录攻击设备与影子设备交互的攻击流量数据包，并对流量进行解析和分析，生成原始流量日志发送给数据处理模块。当攻击设备发动网络扫描和网络攻击时，数据采集分析模块的数据分析模块会通过检测到对应的攻击，比如扫描、密码爆破、漏洞利用、工控攻击、等网络攻击行为，并生成原始告警行为日志发送给数据处理模块。当攻击者在攻击完成并取得模拟服务权限后，数据采集分析模块会采集攻击者对模拟服务做的主机行为，包括传输到模拟服务中样本，生成原始告警行为日志发送给数据处理模块。

[0077] 数据处理模块接收到原始流量日志后，会根据历史攻击流量数据日志生成目标攻击设备的攻击画像。具体地，数据处理模块以物理通信地址为根基，外网以通信地址、设备

指纹、社交账号三个纬度提取出攻击者信息，并且通过域名查询、域名反向解析、域名定位、外网威胁情报、端口扫描等多种反向溯源途径对攻击者进行溯源追踪，补全攻击者信息。然后数据处理模块根据影子设备的类型将攻击流量分为交互流量和请求流量，然后根据攻击流量的类型补全影子设备、源通信地址地理位置以及会话地址等信息，并将所有信息存入数据库。另外数据处理模块还用于收集目标攻击设备的攻击画像的攻击行为，生成与目标攻击设备对应的告警行为日志。再以攻击者和被攻击影子设备为条件，对告警行为日志进行归并，提取出标签，攻击次数等信息生成告警事件日志。数据处理模块对原始告警行为日志中提取出的样本放入沙箱进行分析，并调用威胁情报查询相关信息，最后样本数据。

[0078] 最后在数据展示层中对工业互联网欺骗诱捕系统收集、分析的数据进行可视化展示。实时监控模块主要展示工业互联网欺骗诱捕系统部署区域的实时态势，包括预警等级、近1月攻击行为总数，近1月高危行为数，资产总数(内网攻击者总数)，资产高危数(内网攻击者高危数)，影子设备数，攻击网络拓扑、高危行为滚动播放、被攻击影子设备类型统计、攻击行为类型统计、高危告警趋势/总体告警趋势等统计数据。并且数据展示层还可以将告警行为日志定时发送给目标服务器，以提示管理人员及时关注攻击设备的高危行为。

[0079] 本系统是一款适用于工业互联网的安全诱捕产品。不同于传统的蜜网系统仅用于收集攻击信息、被动发现攻击行为、抵御外部网络攻击，工业互联网欺骗诱捕系统具备主动防御能力，可以通过模拟多种工控设备来构建虚拟影子设备、搭建虚拟蜜网，诱捕对工控网络的攻击行为，并实现对各种攻击行为的检测、识别和预警能力。同时，通过逆向工程和大数据分析技术解构，针对工控网络的攻击，做到迅速定位可疑行为，精确识别网络攻击，及时实施安全防护，全方位的完成威胁预警。

[0080] 应该理解的是，虽然如上的各实施例所涉及的流程图中的各个步骤按照箭头的指示依次显示，但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明，这些步骤的执行并没有严格的顺序限制，这些步骤可以以其它的顺序执行。而且，如上的各实施例所涉及的流程图中的至少一部分步骤可以包括多个步骤或者多个阶段，这些步骤或者阶段并不必然是在同一时刻执行完成，而是可以在不同的时刻执行，这些步骤或者阶段的执行顺序也不必然是依次进行，而是可以与其它步骤或者其它步骤中的步骤或者阶段的至少一部分轮流或者交替地执行。

[0081] 图6是本申请实施例提供的安全诱捕装置的结构示意图。

[0082] 在一个实施例中，如图6所示，提供了一种安全诱捕装置600，包括：服务模拟模块602、设备生成模块604和流量转发模块606，其中：

[0083] 服务模拟模块602，用于生成多个类别的第一模拟设备和第一虚拟服务，第一模拟设备和第一虚拟服务是根据实体设备的设备参数以及实体设备的应用场景，通过与实体设备对应的仿真器生成的。

[0084] 设备生成模块604，生成至少一个实体设备的影子设备，影子设备是实体设备的映射，每个影子设备对应多个虚拟通信地址。

[0085] 流量转发模块606，用于当虚拟通信地址接收到来自攻击设备的流量数据包时，若虚拟通信地址对应的影子设备为高交互，则通过影子设备将流量数据包转发至对应的第一模拟设备和/或第一虚拟服务。

[0086] 在一个实施例中，设备生成模块604包括：地址生成子模块，用于在空闲的IP地址

0424

空间中，批量生成多个不同物理地址的虚拟IP地址；对应关系子模块，用于根据影子设备与实体设备之间的映射关系生成影子设备与第一模拟设备以及第一模拟服务之间的对应关系；关联子模块，用于根据对应关系，将多个虚拟IP地址关联上各个第一模拟设备以及第一模拟服务。

[0087] 在一个实施例中，该装置还包括：转发子模块，用于当虚拟通信地址接收到来自攻击设备的攻击应用层报文，与虚拟通信地址相对应的影子设备将攻击应用层报文转发至对应的第一模拟设备以及第一虚拟服务；第二报文生成子模块，用于当第一模拟设备以及第一虚拟服务接收到攻击设备发送的第一报文后响应相对应的第二报文；存储子模块，用于将第一报文与第二报文的内容一一对应地存储在数据表中；流量学习子模块，用于当第一模拟设备以及第一虚拟服务接收到目标攻击设备的目标应用层报文时，将目标报文的内容与数据表中的第一报文的内容相匹配，若匹配成功，则将匹配成功的第一报文相对应的第二报文发送至攻击设备。

[0088] 在一个实施例中，该装置还包括：攻击画像子模块，用于根据历史攻击流量数据日志生成目标攻击设备的攻击画像；行为日志子模块，用于收集目标攻击设备的攻击画像的攻击行为，生成与目标攻击设备对应的告警行为日志。

[0089] 在一个实施例中，攻击画像子模块还用于获取各个攻击设备产生的历史流量数据记录；从攻击流量数据记录中获取关键字段，其中关键字段包括攻击设备地通信地址、攻击设备指纹以及攻击设备上登录的特定软件的用户名；若任一攻击设备与目标攻击设备发送的流量数据中的关键字段中至少一项相同，则认定攻击设备与目标攻击设备为同一攻击设备。

[0090] 在一个实施例中，装置还用于将与攻击设备对应的告警行为日志发送目标服务器，以使目标服务器将告警行为日志发送至客户端。

[0091] 在一个实施例中，装置还用于依次统计每一预设时间段内的攻击流量数据中的攻击行为总数以及告警攻击行为总数；根据每一预设时间段内的攻击行为总数以及告警攻击行为总数生成告警攻击行为趋势图，并将告警攻击行为趋势图显示在监视大屏中。

[0092] 上述机器人示教装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备中的处理器中，也可以以软件形式存储于计算机设备中的存储器中，以便于处理器调用执行以上各个模块对应的操作。

[0093] 图7是本申请实施例提供的终端设备的结构示意图。

[0094] 在一个实施例中，提供了一种计算机设备，该计算机设备可以是服务器，其内部结构图可以如图7所示。该计算机设备包括通过系统总线连接的处理器、存储器和网络接口。其中，该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质和内存储器。该非易失性存储介质存储有操作系统、计算机程序和数据库。该内存储器为非易失性存储介质中的操作系统和计算机程序的运行提供环境。该计算机设备的数据库用于存储路径轨迹数据。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机程序被处理器执行时以实现一种机器人示教方法。

[0095] 本领域技术人员可以理解，图7中示出的结构，仅仅是与本申请方案相关的部分结构的框图，并不构成对本申请方案所应用于其上的计算机设备的限定，具体的计算机设备可以包括比图中所示更多或更少的部件，或者组合某些部件，或者具有不同的部件布置。

[0096] 在一个实施例中，提供了一种计算机设备，包括存储器和处理器，存储器中存储有计算机程序，该处理器执行计算机程序时实现上述任一实施例中提供的安全诱捕方法。

[0097] 在一个实施例中，提供了一种计算机可读存储介质，其上存储有计算机程序，计算机程序被处理器执行时实现上述任一实施例中提供的安全诱捕方法。

[0098] 在一个实施例中，提供了一种计算机程序产品，包括计算机程序，该计算机程序被处理器执行时实现上述任一实施例中提供的安全诱捕方法。

[0099] 需要说明的是，本申请所涉及的用户信息(包括但不限于用户设备信息、用户个人信息等)和数据(包括但不限于用于分析的数据、存储的数据、展示的数据等)，均为经用户授权或者经过各方充分授权的信息和数据。

[0100] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机程序来指令相关的硬件来完成，的计算机程序可存储于一非易失性计算机可读取存储介质中，该计算机程序在执行时，可包括如上述各方法的实施例的流程。其中，本申请所提供的各实施例中所使用的对存储器、数据库或其它介质的任何引用，均可包括非易失性和易失性存储器中的至少一种。非易失性存储器可包括只读存储器(Read-Only Memory，ROM)、磁带、软盘、闪存、光存储器、高密度嵌入式非易失性存储器、阻变存储器(ReRAM)、磁变存储器(Magnetoresistive Random Access Memory，MRAM)、铁电存储器(Ferroelectric Random Access Memory，FRAM)、相变存储器(Phase Change Memory，PCM)、石墨烯存储器等。易失性存储器可包括随机存取存储器(Random Access Memory，RAM)或外部高速缓冲存储器等。作为说明而非局限，RAM可以是多种形式，比如静态随机存取存储器(Static Random Access Memory，SRAM)或动态随机存取存储器(Dynamic Random Access Memory，DRAM)等。本申请所提供的各实施例中所涉及的数据库可包括关系型数据库和非关系型数据库中至少一种。非关系型数据库可包括基于区块链的分布式数据库等，不限于此。本申请所提供的各实施例中所涉及的处理器可为通用处理器、中央处理器、图形处理器、数字信号处理器、可编程逻辑器、基于量子计算的数据处理逻辑器等，不限于此。

[0101] 以上实施例的各技术特征可以进行任意的组合，为使描述简洁，未对上述实施例中的各个技术特征所有可能的组合都进行描述，然而，只要这些技术特征的组合不存在矛盾，都应当认为是本说明书记载的范围。

[0102] 以上实施例仅表达了本申请的几种实施方式，其描述较为具体和详细，但并不能因此而理解为对本申请专利范围的限制。应当指出的是，对于本领域的普通技术人员来说，在不脱离本申请构思的前提下，还可以做出若干变形和改进，这些都属于本申请的保护范围。因此，本申请的保护范围应以所附权利要求为准。

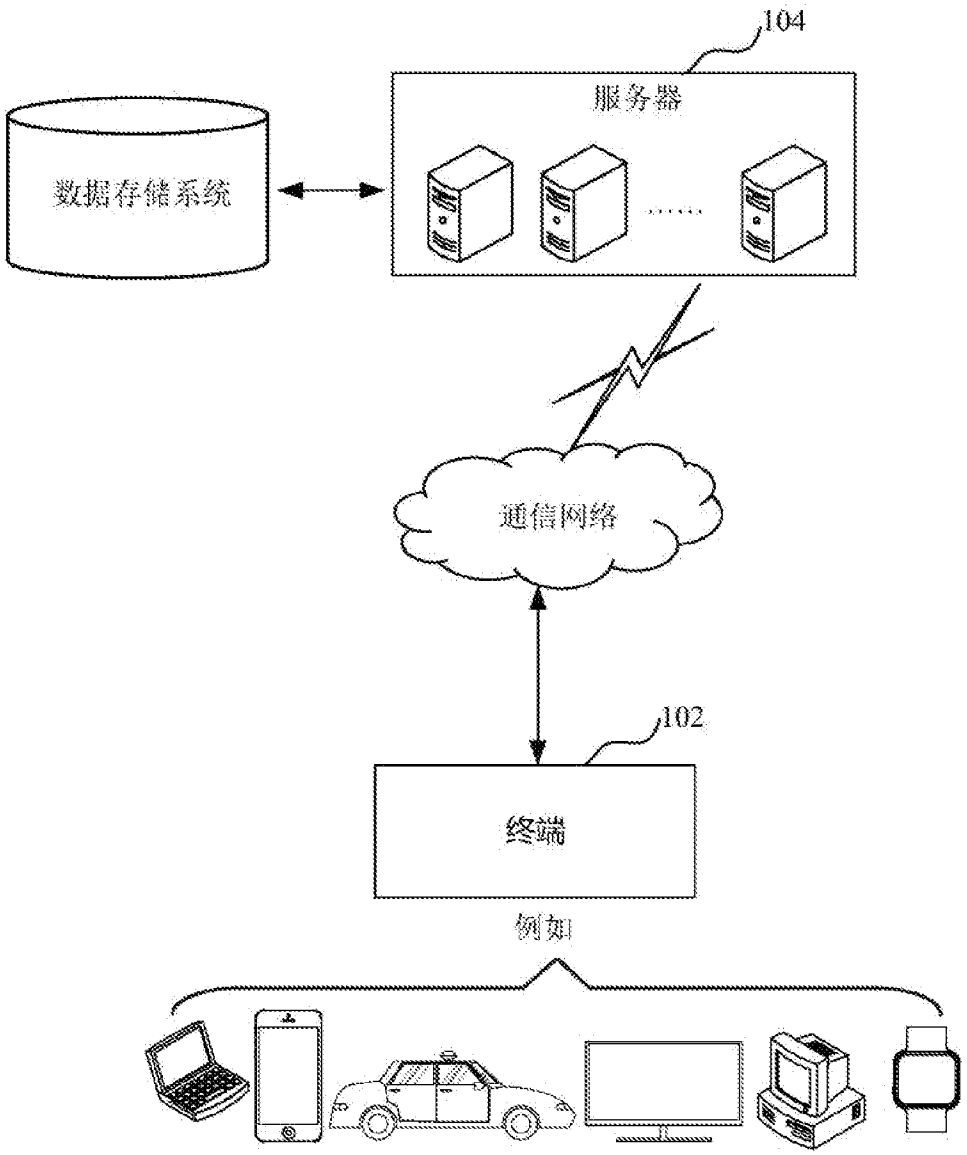


图1

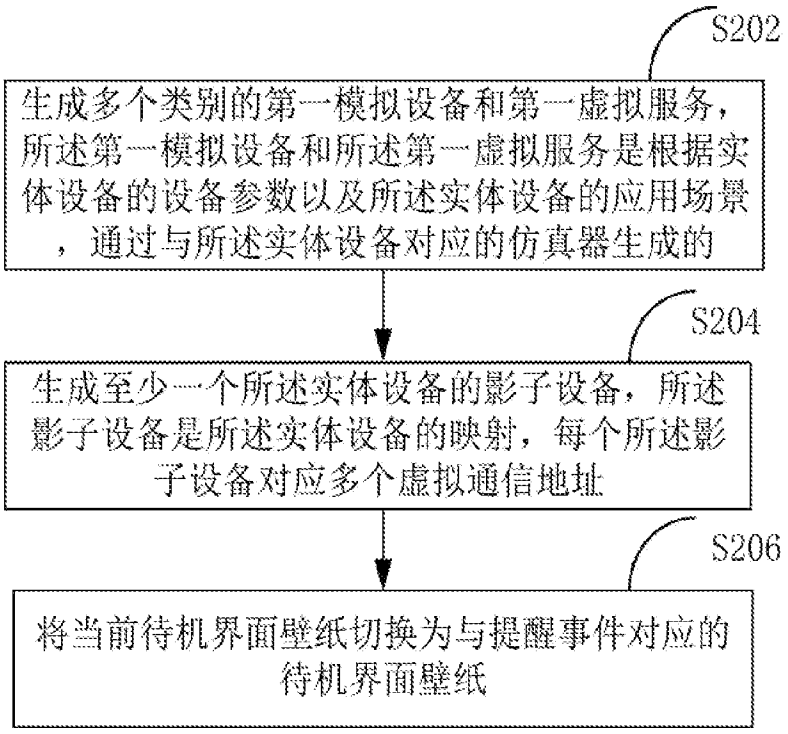

S202
生成多个类别的第一模拟设备和第一虚拟服务，所述第一模拟设备和所述第一虚拟服务是根据实体设备的设备参数以及所述实体设备的应用场景，通过与所述实体设备对应的仿真器生成的
S204
生成至少一个所述实体设备的影子设备，所述影子设备是所述实体设备的映射，每个所述影子设备对应多个虚拟通信地址
S206
将当前待机界面壁纸切换为与提醒事件对应的待机界面壁纸

图2

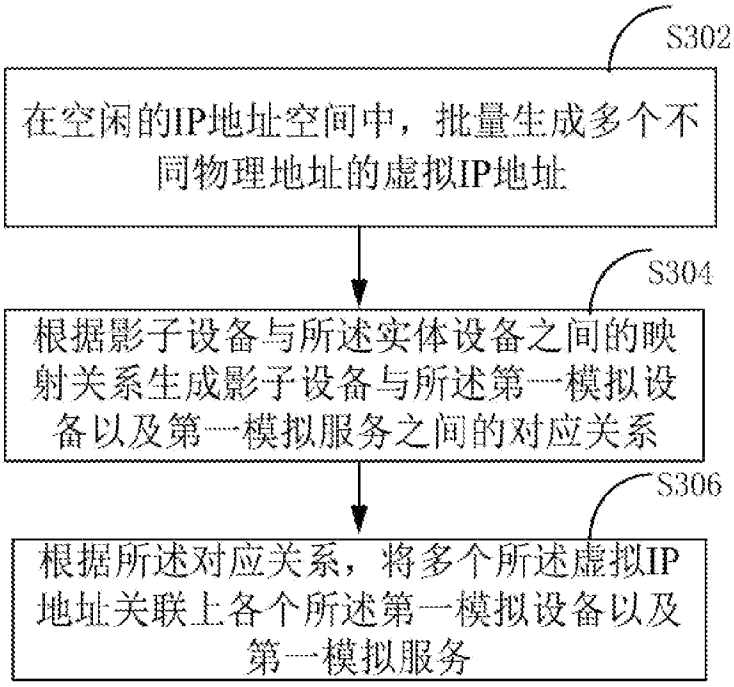

S302
在空闲的IP地址空间中，批量生成多个不同物理地址的虚拟IP地址
S304
根据影子设备与所述实体设备之间的映射关系生成影子设备与所述第一模拟设备以及第一模拟服务之间的对应关系
S306
根据所述对应关系，将多个所述虚拟IP地址关联上各个所述第一模拟设备以及第一模拟服务

图3

0428

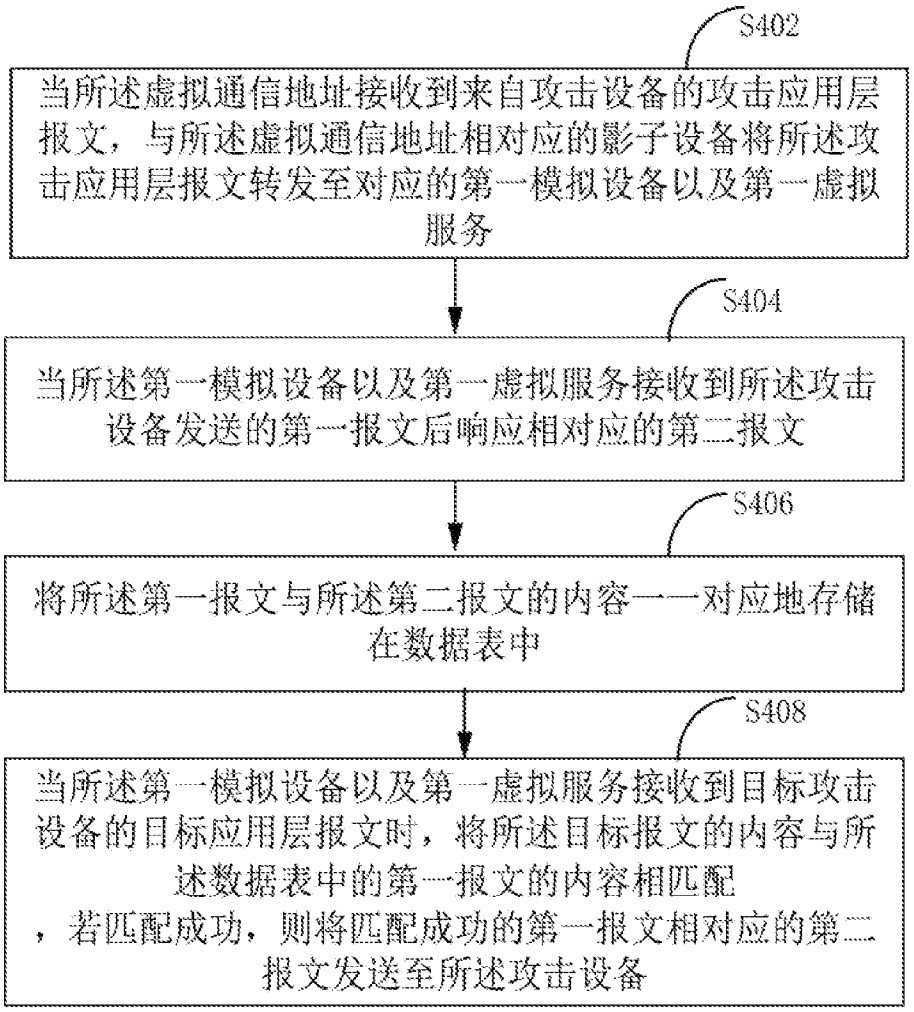

S402
当所述虚拟通信地址接收到来自攻击设备的攻击应用层报文，与所述虚拟通信地址相对应的影子设备将所述攻击应用层报文转发至对应的第一模拟设备以及第一虚拟服务
S404
当所述第一模拟设备以及第一虚拟服务接收到所述攻击设备发送的第一报文后响应相对应的第二报文
S406
将所述第一报文与所述第二报文的内容一一对应地存储在数据表中
S408
当所述第一模拟设备以及第一虚拟服务接收到目标攻击设备的目标应用层报文时，将所述目标报文的内容与所述数据表中的第一报文的内容相匹配
，若匹配成功，则将匹配成功的第一报文相对应的第二报文发送至所述攻击设备

图4

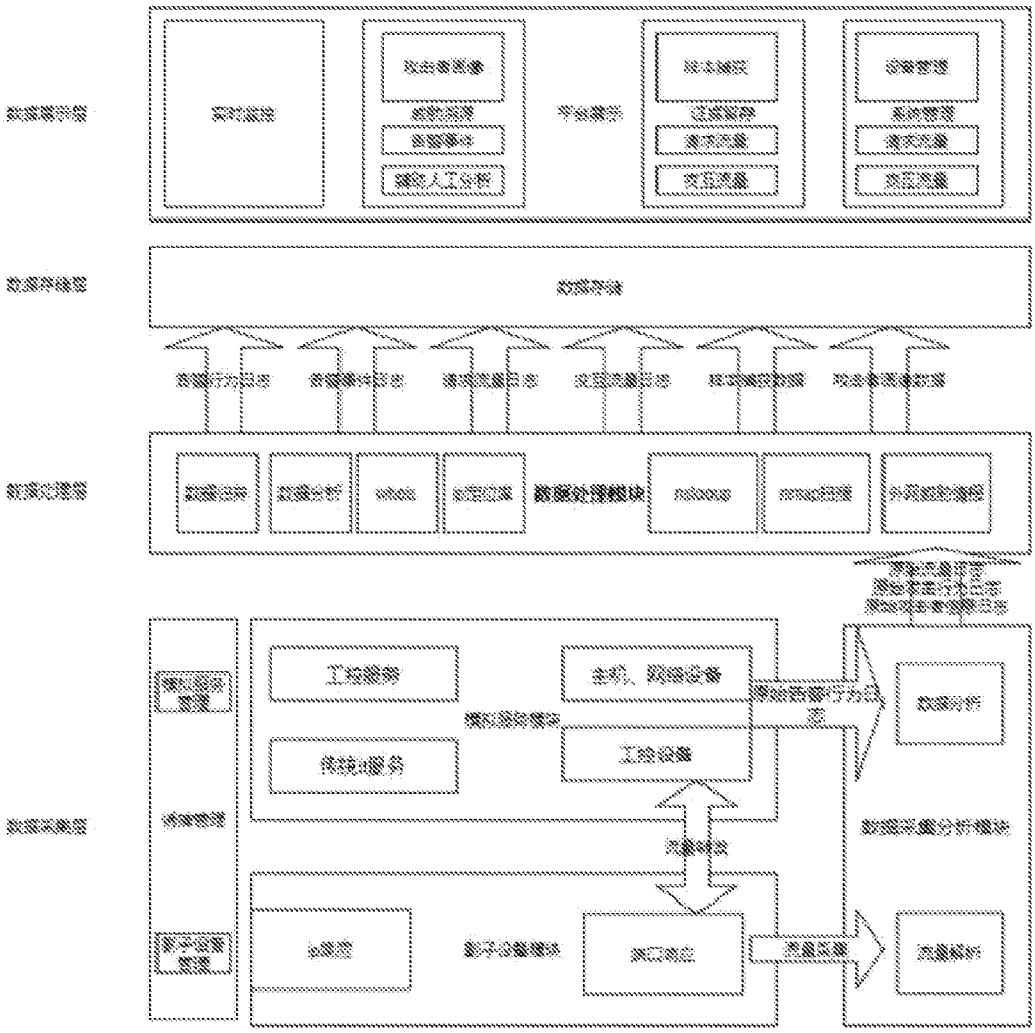


图5

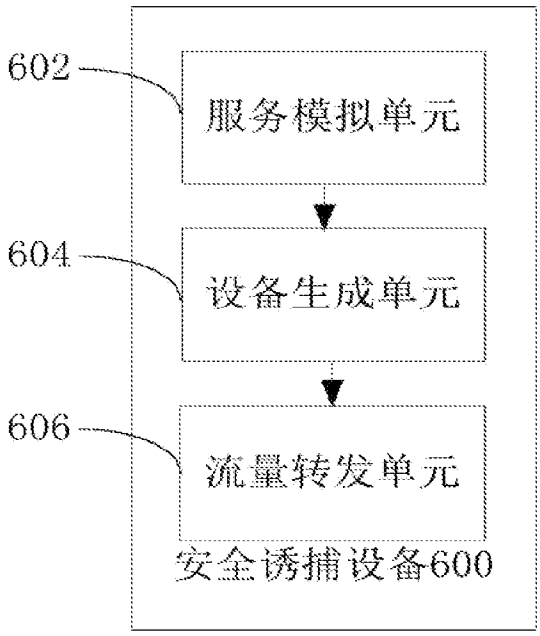

602
服务模拟单元
604
设备生成单元
606
流量转发单元
安全诱捕设备600

图6

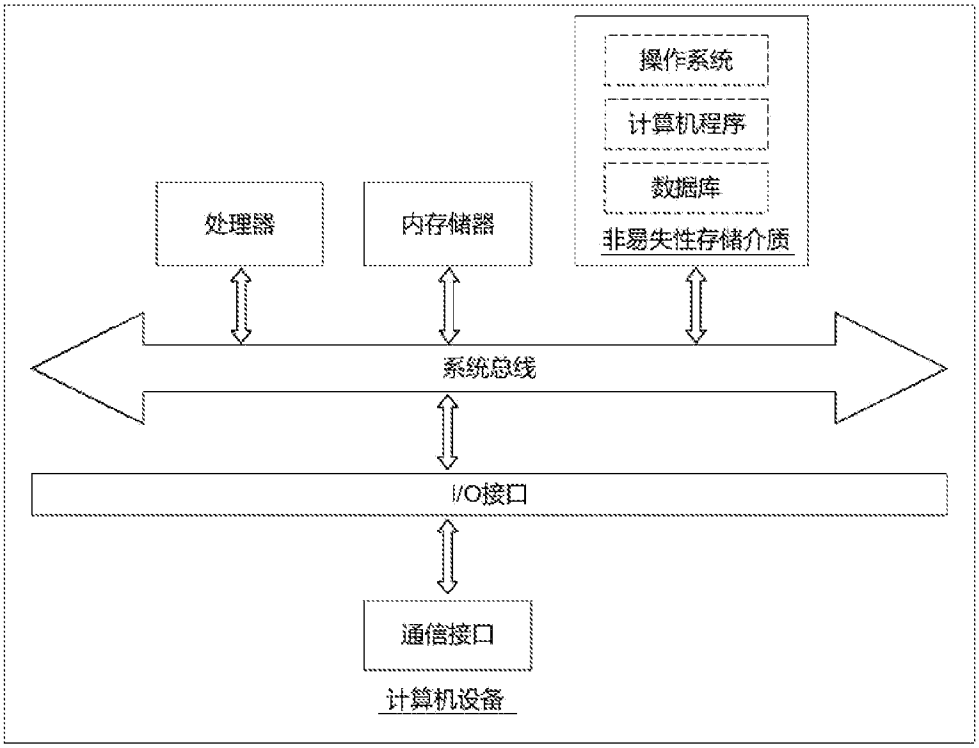

操作系统
计算机程序
数据库
非易失性存储介质
处理器
内存储器
系统总线
I/O接口
通信接口
计算机设备

图7

Espacenet

CN109995722A Mass detection data analysis system for APT protection

Applicants: GUANGZHOU MINGLEAD GENE TECH CO LTD

Inventors: XU JIFENG,QI JIANMING,ZHOU JUNSONG,CHEN DUNJIN

Classifications:
IPC ***H04L29/06;***

CPC **H04L63/1416 (CN); H04L63/1466 (CN); H04L63/20 (CN);**

Priorities: CN201711489965A 2017-12-30

Application: CN201711489965A·2017-12-30

Publication: CN109995722A·2019-07-09

Published as: CN109995722A

Mass detection data analysis system for APT protection

Abstract

The invention discloses a mass detection data analysis system for APT protection. The system comprises a data acquisition layer and a data processing layer, wherein the data acquisition layer consistsof a plurality of mass heterogeneous data sources with adjacent or non-adjacent physical positions, and is responsible for providing multi-angle security data and scenes for the data processing layer; and the data processing layer is responsible for subscribing the data of the data acquisition layer according to an association analysis rule so as to carry out analysis processing. According to thescheme of the invention, the big data technology is utilized to carry out deep correlation analysis on the monitoring detection data, so that not only can pre-warning be realized, but also blocking in an event can be realized, and post audit traceability can be realized.

mass detection data analysis system for APT protection

DOCUMENT ID	DATE PUBLISHED
CN-109995722-A	2019-07-09

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
XU, Ji-feng	N/A	N/A	N/A	N/A
QI, Jian-ming	N/A	N/A	N/A	N/A
ZHOU, Jun-song	N/A	N/A	N/A	N/A
CHEN, Dun-jin	N/A	N/A	N/A	N/A

DATE FILED
2017-12-30

CPC CURRENT

TYPE	CPC	DATE
CPCI	H04L63/1466	2013-01-01
CPCI	H04L63/20	2013-01-01
CPCI	H04L63/1416	2013-01-01

Abstract

The invention claims a APT protection of mass detection data analysis system, the system comprising: a data collecting layer and a data processing layer, several mass heterogeneous data sources, wherein the data collecting layer by physical location adjacent or non-adjacent, responsible for providing security data to the data processing layer and scene multi-angle; said data processing layer is responsible for data correlation analysis rule to subscribe the data collecting layer so as to perform analysis and processing. The solution of the invention using large data technology for monitoring and detecting data for depth correlation analysis, not only can realize beforehand alarm, also can realize the blocking in the event, also can realize traceability of post-audit.

Description

INVENTION-TITLE
technical field
The invention belongs to the big data analysis technology field, relating to a detection data analysis system for mass facing APT protection.
BACKGROUND
APT (advanced persistent threat) advanced persistent threat attack, which is different from the traditional network man, APT emphasizes penetrant and get information to steal the enterprises and the government core information asset as the main purpose.
the characteristics of the APT on A and P2 properties: A represents Advanced, mainly expressed on

hypersonic attack means, character information of hacker attack behavior is difficult to extract, attack single point strong hidden performance, diversified attack and attack space is not determined. P represents a Persistent, mainly showing the attack duration and shielding time length, it makes the network threat from random attack of the type become objects, organized, with a population of type attack, which makes the traditional in real time detection, defense mode real-time blocking body is hard to exert the effect. Currently, the APT attack event has been explosive growth. Although already caused by the height the attention of government and enterprise, but the existing protection scheme has certain limitations, cannot cover the global, to possibly caused by missing, and many product can only detect APT attacks, but can not provide real-time protection function.

summary of the invention

The purpose of the invention is to provide a mass detection data analysis system facing APT protection, such a population for APT type attack to the traditional real time monitoring, real-time blocking huge impact is caused by main body of the detecting technology, the monitoring detection data for depth correlation analysis using large data technology, not only can the comprehensive analysis target system whether there is risk being attacked so as to realize beforehand alarm, it can find the attack is carried out, more accurate understanding of the intent and backward tracing. realize the comprehensive determination of the threat so as to timely take related policy to prevent attack to realize blocking event, also for big data analysis the security audit information, reproduction history state and evolution process of the data according to the tracing path to realize post-audit tracing.

In order to solve the above technical problem, the present invention adopts the following technical solution: a detection data analysis system for mass facing APT protection, the system comprising: a data collecting layer and a data processing layer, wherein several mass heterogeneous data sources of the data collecting layer by physical location adjacent or not adjacent, is responsible for the data processing layer provides security data and scene multi-angle; said data processing layer is responsible for data association analyzing rules subscribing to the data collecting layer; so as to carry out analysis processing.

Further data source, the data acquisition layer comprises a flow mirroring, firewall, the UDS and IPS.

Further, the data processing layer by the safety detecting module, safety audit module and large data correlation analysis module.

Further, the safety detection module comprises detection and high detection function.

Further, the correlation analysis module of larger data containing transverse association analysis and longitudinal association analysis function.

Further, the security audit module comprises analysis of vulnerabilities and attack source function.

Compared with the existing technology, the invention has the following beneficial effects:

The solution of the invention for the APT attack to traditional in this population in real time monitoring, real-time blocking huge impact is caused by main body of the detecting technology, using large data technology for monitoring and detecting data for depth correlation analysis so as to realize beforehand alarm and comprehensive determination of the threat, to take the relevant policy block the attack, realize event in blocking and post-audit tracing.

Description

Picture 1 the whole frame image data analysis of mass detection system facing APT protection.

Specific implementation methods

With reference to the accompanying drawings and specific embodiment of this invention will be described in a more detailed and complete. It should be understood that the specific embodiments described here are only used to explain the present invention, but not to limit the scope of the invention.

Referring to FIG. 1, a facing APT protection of mass detection data analysis system of the present invention, the system comprises a data collecting layer and a data processing layer, wherein said data processing layer is responsible for data association analyzing rules subscribing to the data collecting layer; so as to analyze and process, multiple mass heterogeneous data source data source of the data collecting layer comprises physical location flow mirroring, firewall, the UDS, IPS and a series of adjacent or not adjacent, is responsible for providing security data to the data processing layer and multi-angle scene.

a data processing layer with a safety detecting module, safety audit module and large data correlation analysis module, wherein the security detection module further comprises a low level detecting module and position detecting module, a data correlation analysis module is divided into transverse relation analyzing

module and longitudinal associated analysis module, a safety audit module is divided into the vulnerability analysis module and attack source modules, each module is implemented by a variety of techniques.
1) low detection module
Because APT generally uses Oday vulnerability by unknown Trojan obtained authority, remote control, and the detecting device of the traditional based on feature matching always to firstly capture the malicious code sample, then extracting the characteristic and the attack based on feature identification, the presence of hysteresis. low detection is emergency subsystem response platform facing the malfunction detection early warning terminal. system and the traditional safety protection software has intrinsic technical difference. safety enhancing module using the malfunction detection theory model, which not only can effectively monitor the known threat, and can find the unknown threat.
malfunction detection analysis and warning subsystem comprises an application malfunction detection module, system malfunction detection module, a communication malfunction detection module, a transmission malfunction detection module, abnormal data discovery module and other abnormal discovery module. the inner main logic is associated abnormal discovery, decision judging and early warning. associated malfunction detection is the hazard level abnormal set according to the strategy base information, i.e., risk, the potential threat for the 2 time estimation, the policy library and to update the warning through the I/O interface with high monitoring communication, for the 3 inference, and can through I/O interface uploads the abnormal data set, searching for the 4 threat more accurate inference of threat according to user attribute database information. the result set of the data of the user attribute database can be collected through the self-defined by the user or by associating each module of user operation habit for slowly adjusting to generate. constructing the target user attribute database is associated user social attribute, to sense the customizability of threat penetrant. low detection not only can support high detection, also can through the output interface of the warning I/O, the warning information is sent to other safety protection system, such as antivirus and firewall software, honey.
2) high detection module
position detection is facing the core response platform malfunction detection early warning emergency source and pathway. the advantage of high detection is that it can capture discovery abnormality of a backbone network, find a lot of resource attack implementation, detecting and finding that a large source a malicious application. associating low-level detection data, can make high detection has large "view" capability, can effectively analyze and detect unknown threat, tracing the attacker identity, responding to the APT attack.
position detection technology comprises processing fast identification and classification technology for network service based on streaming mode, for cognitive recognition and feature extraction technology to flow based on information theory, network monitoring technology based on content analysis based on detecting technology of behaviour characteristic; to the network data stream and network behaviour as the centre, a multi-angle detection analyzing and calculating abnormal discovery technology, for detecting and analyzing abnormal network technology obtained, uses multi-level network anomaly associated with data fusion technology, multi-layered network abnormal association rule library building technology and so on. position detection by statistical characteristic of the protocol layer of the IP layer, transport layer, application layer, establishing the normal user network access model, detecting the abnormal behavior possible, including abnormal network behaviour of the abnormal flow transmission, DNS of the speed change behavior and other Trojan attack. different behaviour characteristics of user research service network and a gateway on the network, respectively establishing abnormal model to service the network behaviour feature and boundary network behaviour characteristic, discovering unknown threat. the core of the high-position detection is abnormal decision engine, data low detecting malfunction detection decision to provide support for high through I/O interface. In addition, in low detection transmits the abnormal data packet in the malfunction detection module also can incoming high detection system for depth analysis via an I/O interface corresponding to the network traffic. the result of the abnormality decision engine is composed of the corresponding I/O interface output, for low detection reference. result set of malfunction detection also can provide technical support is other safety protective platform through the I/O interface.
3) security audit module
security audit module collecting a system log, resource usage, user behavior and network traffic log and data record, and the vulnerability analysis sub-module and attack tracing sub-module initially analyzes the

collected data, besides using the traditional blacklist and white list analysis method, has to be combined big data correlation analysis module to perform the comprehensive determination, from a lot of log record and data record of mixed in digging the potential threatening problem.

4) large data correlation analysis module

monitoring data of the low level detection and high detection, security audit information for association analysis and data mining to find correlation between the recording, discovery of new features. for the sample data digging out of the manual classification or clustering method uses machine learning for classification result building the feature base training set, thus improving the sorting accuracy of feature database.

log record using large collected data technology to the security audit module, detecting module and monitoring data of the high low level detecting module for big data correlation analysis to comprehensively analyze the target system whether there is likely to be attack to realize beforehand alarm. the attack is being performed for big data analysis, can more accurately understand the intent and backward tracing, realize the comprehensive determination of the threat so as to timely take related policy to prevent attack to realize blocking in the event. and the security audit information of the whole attack process for big data analysis, reproduction history state and evolution process of the data according to the tracing path, realizing traceability of post-audit.

detecting data generated by all kinds of APT attack detection tool, key, context and information flow, log information of the external to form the mass data with multiple different structures. data of the multi-source multi-structural, long time, low density property, defense method to the big data analysis brings the protective challenge. is the key problem of how to efficiently process data analysis by data technology.

1) data management technology

in today's large data age, complete difficult development of relational data management system (RDBMS) encountered a single data type, insufficient expandability and so on. along with the increase of the data amount, data analysis of basic policy is pushed to the computing data, rather than data calculation. In recent years, the non-relational data management technology rise suddenly, the plurality of types of data for effective management, processing and analyzing, obtaining good system performance through parallel processing technology, and satisfies the processing requirement of the constantly increasing data amount to its high expansibility.

the data engine relational database as the core, the data from various sources by the ETL tool leads the relational database system. client end tool realized by SQL language routine report generation. for complex analysis, the SQL expression ability to expose its limitations. If the data extracted from the database, the leading front end analysis tool (SAS.SPSS) for subsequent analysis, will result in a large amount of data movement. meanwhile, the front end analysis data processing ability of the tool the limited memory size, data analysis efficiency will deteriorate. In addition, because data tend to types, large scale relational database organization and processing ability cannot deal with high-dimensional large scale when the challenge of null data. Therefore, in the data analysis process is necessarily parallel calculation, the selection of the high performance data management.

relative to the RDBMS, MapReduce technology is represented by the non-relational data management technology support a higher fault tolerance from the storage model and the calculation model, stronger expansibility, analysis task is big data analysis provides a good operation platform is ensured, at the same time, cannot be used for more easily by using SQL to express the MapReduce computing function expression. As a aiming at solving the large-scale non-structured data fast and batch processing of parallel technique frame, MapReduce to obtain the cyrtotrachelus buqueti development. low performance problem caused by too simple to overcome the scheduling algorithm, for multi-core CPU, GPU, heterogeneous system and the cloud platform of different environment such as scheduling policy can be optimized. through the logic model of the defined data processing flow, converting the logical model into a physical model using model conversion algorithm and code generation arithmetic to realize the multi-data source for processing MapReduce data support. flow and localized by an intermediate result cache, improves the MapReduce data stream processing ability, and claims a method for high speed sensing data stream real-time MapReduce method.

To detect APT attack behaviour of each attack stage from the multi-dimensional data, the data must be deeply analyzed, rather than merely generating a simple report. This complex analysis must rely on complicated analysis model, it is difficult to perform expression using SQL, belonging to the depth analysis.

Therefore, the challenge facing large data depth analysis, in MapReduce technology is represented by the non-relational data management technology has obvious advantages.

2) the depth analysis technology

Techtarget defined the depth analysis is to use complex data processing technology to be integrated obtaining knowledge from the structured, non-structured and semi-structured multi-source data. it starts from the requirement of the user, the way to effectively, the user can accept a large amount of complicated distributed data for analysis, extraction and summarization, so as to provide the corresponding guidance to the user decision.

Because the depth analysis usually needs to finish the PB level even level EB data for accurately locating and complex query, and normally used for environment real-time and near real-time response, a typical data analysis operation (e.g., aggregation, rotation, slicing and summary) analysis technique cannot meet the requirement, requirement people is more complex, for example, time series analysis, path analysis, What-if analysis, social network analysis and complex statistical analysis model and did not attempt due to hardware/software limitation. following the time sequence analysis and pattern analysis is described: analyzing the time sequence). time sequence analyzing method of mathematical statistics according to the time order of data sequence is processed, and thus predicting the future development trend. it not only admit the continuity of business development but also considers the randomness of object development. Because the time sequence analysis comprises predicting future data according to the history data, so the commonly used in market potential, hydrology forecast prediction, weather forecast, national economic macro-control and enterprise operation and management and so on.

(social network analysis and graph analysis. mode of social network analysis focuses on relationships and relationship, on the concept, the manner and method differs from the traditional statistical analysis and data processing method. In the figure, 1 independent of entity 1 node represents a social network contact 1, in edge represents the entity between the 2 points. using the social network analysis, can obtain some useful knowledge, such as found in the network key entity and so on (each group key entity in the network connected together). Such information can be used for sentiment analysis, product sales and potential threat prediction and so on.

APT attack data based on complex and multi-source characteristics, introduction depth analysis can effectively extract knowledge from the history data, and to predict the attack action has not occurred so as to nip in the bud.

3) large data mining technology

safety detection based on large data mining is a class of important safety detecting technique. the traditional data mining algorithm based on serial calculation, processing data size limited, once the migration to large data environment, the executing efficiency will greatly decrease even cannot run and cannot meet the present requirement of large data.

Cloud computing is a big data analysis and mining technology can support of the computing platform, mining and analyzing large data point is the mass data, it can not only finish operation using a single computer, but must adopts the distributed computing architecture. Therefore, relying on distributed processing, distributed database of cloud computing, cloud storage and virtualization technology is an effective means for improving the large data mining efficiency, dispersing the computing task while running on a different computer, the various application systems can be adjusted according to needs to acquire storage resource, computing resource and other service resource. MapReduce, Hadoop, Spark and other high performance, high expandability of the programming model of parallel computing, distributed processing large data frame and continuously emerge and development of related key technology, makes the data storage and distributed computing of large data. based on distributed computing, large data mining technology can be better to provide new theoretical and technical support based on security detection problem of the large data. current security detection technology based on large data mining main research various data mining parallelization strategy based on distributed computing algorithm, new large data mining method and realizing abnormity analysis and detection of large scale on MapReduce, Spark frame and so on.

The above description is only the preferred embodiments of the present invention, which are not used to restrict the present invention to those skilled in the art, the present invention may have various modifications and changes. Any modifications, within the spirit and principle of the invention the any modification, equivalent replacing, improved and so on, are contained in the protection range of the invention.

Claims

1. for mass of APT protection detection data analysis system, wherein the system comprises a data collecting layer and a data processing layer, several mass heterogeneous data sources, wherein the data collecting layer by physical location adjacent or non-adjacent, responsible for providing security data to the data processing layer and scene multi-angle; said data processing layer is responsible for data correlation analysis rule to subscribe the data collecting layer so as to perform analysis and processing.
2. The facing APT protection according to claim 1, the mass detection data analysis system, wherein the data source of the data collecting layer comprises a flow mirroring, firewall, the UDS and IPS.
3. The APT protection-oriented data analysis of mass detection system according to claim 1, wherein, the data processing layer formed by the safety detecting module, safety audit module and large data correlation analysis module.
4. The APT protection-oriented data analysis of mass detection system according to claim 3, wherein, said safety detection module comprises detection and high detection function.
5. The APT-oriented protection according to claim 3 mass detection data analysis system, wherein the large data correlation analysis module comprising transverse association analysis and longitudinal association analysis function.
6. The APT protection-oriented data analysis of mass detection system according to claim 3, wherein, said safe audit module comprises vulnerability analysis and attack source function.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 109995722 A
(43)申请公布日 2019.07.09

(21)申请号 201711489965.1

(22)申请日 2017.12.30

(71)申请人 广州明领基因科技有限公司
地址 510610 广东省广州市天河区林和东路281号天伦控股大厦17层自编05单元

(72)发明人 徐继峰 祁建明 周峻松 陈墩金

(51)Int.Cl.
H04L 29/06(2006.01)

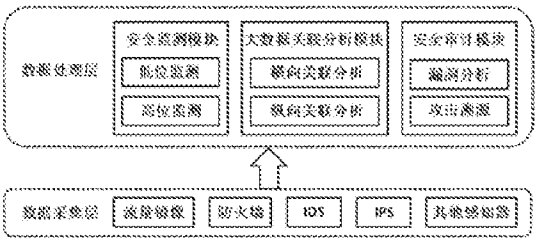
权利要求书1页 说明书5页 附图1页

(54)发明名称

面向APT防护的海量检测数据分析系统

(57)摘要

本发明公开了一种面向APT防护的海量检测数据分析系统，该系统包括：数据采集层以及数据处理层；其中，所述数据采集层由物理位置相邻或不相邻的若干海量异构数据源组成，负责为所述数据处理层提供多角度的安全数据及情景；所述数据处理层负责根据关联分析规则订阅所述数据采集层的数据，从而进行分析处理。本发明方案利用大数据技术对监控检测数据进行深度关联分析，不仅能够实现事前预警，也可实现事中阻断，还可实现事后审计溯源。



CN 109995722 A

1.面向APT防护的海量检测数据分析系统，其特征在于，所述系统包括：数据采集层以及数据处理层；其中，所述数据采集层由物理位置相邻或不相邻的若干海量异构数据源组成，负责为所述数据处理层提供多角度的安全数据及情景；所述数据处理层负责根据关联分析规则订阅所述数据采集层的数据，从而进行分析处理。

2.根据权利要求1所述的面向APT防护的海量检测数据分析系统，其特征在于，所述数据采集层的数据来源包括流量镜像、防火墙、UDS及IPS等。

3.根据权利要求1所述的面向APT防护的海量检测数据分析系统，其特征在于，所述数据处理层由安全检测模块、安全审计模块以及大数据关联分析模块组成。

4.根据权利要求3所述的面向APT防护的海量检测数据分析系统，其特征在于，所述安全检测模块包含低位检测及高位检测功能。

5.根据权利要求3所述的面向APT防护的海量检测数据分析系统，其特征在于，所述大数据关联分析模块含有横向关联分析及纵向关联分析功能。

6.根据权利要求3所述的面向APT防护的海量检测数据分析系统，其特征在于，所述安全审计模块包含漏洞分析及攻击溯源功能。

面向APT防护的海量检测数据分析系统

技术领域

[0001] 本发明属于大数据分析技术领域，涉及一种面向APT防护的海量检测数据分析系统。

背景技术

[0002] APT(advanced persistent threat)即高级持续性威胁攻击，其不同于传统的网络人侵，APT侧重渗透和获取信息，以窃取企业和政府核心信息资产为主要目的。

[0003] APT的特点表现在A和P2个特性上：A代表Advanced，主要表现在攻击手段高超，攻击行为特征难以提取，攻击单点隐蔽性强，攻击渠道多样化以及攻击空间不确定。P代表Persistent，主要表现在攻击持续和隐蔽时间长，它使网络威胁从散兵游勇式的随机攻击变成有目的、有组织、有预谋的群体式攻击，使传统的以实时检测、实时阻断为主体的防御方式难以再发挥作用。

[0004] 目前，APT攻击事件已趋于爆发式增长。虽说已引起了政府和企业的高度重视，但现有防御方案大都具有一定的局限性，无法覆盖全局，从而可能导致漏报，而且很多产品只能检测APT攻击，却无法提供实时防护功能。

发明内容

[0005] 本发明目的在于提供一种面向APT防护的海量检测数据分析系统，针对APT这种群体式攻击给传统的以实时监测、实时阻断为主体的检测技术带来的巨大冲击，利用大数据技术对监控检测数据进行深度关联分析，不仅能够综合分析目标系统是否存在被攻击的风险，进而实现事前预警，也可发现正在进行的攻击，更加准确地理解意图和反向追踪，实现对威胁的综合研判，从而及时采取相关的策略阻止攻击，实现事中阻断，还可同时对安全审计信息进行大数据分析，根据追踪路径重现数据的历史状态和演变过程，实现事后审计溯源。

[0006] 为解决上述技术问题，本发明采用如下的技术方案：一种面向APT防护的海量检测数据分析系统，该系统包括：数据采集层以及数据处理层；其中，所述数据采集层由物理位置相邻或不相邻的若干海量异构数据源组成，负责为所述数据处理层提供多角度的安全数据及情景；所述数据处理层负责根据关联分析规则订阅所述数据采集层的数据，从而进行分析处理。

[0007] 进一步地，所述数据采集层的数据来源包括流量镜像、防火墙、UDS及IPS等。

[0008] 进一步地，所述数据处理层由安全检测模块、安全审计模块以及大数据关联分析模块组成。

[0009] 进一步地，所述安全检测模块包含低位检测及高位检测功能。

[0010] 进一步地，所述大数据关联分析模块含有横向关联分析及纵向关联分析功能。

[0011] 进一步地，所述安全审计模块包含漏洞分析及攻击溯源功能。

[0012] 本发明与现有技术相比具有以下的有益效果：

[0013] 本发明方案针对APT这种群体式攻击给传统的以实时监测、实时阻断为主体的检测技术带来的巨大冲击，利用大数据技术对监控检测数据进行深度关联分析，进而实现事前预警及对威胁的综合研判，及时采取相关的策略阻止攻击，实现事中阻断以及事后审计溯源。

附图说明

[0014] 图1是面向APT防护的海量检测数据分析系统的整体框架图。

具体实施方式

[0015] 下面结合附图及具体实施例对本发明进行更加详细与完整的说明。可以理解的是，此处所描述的具体实施例仅用于解释本发明，而非对本发明的限定。

[0016] 参照图1，本发明的一种面向APT防护的海量检测数据分析系统，该系统包括：数据采集层以及数据处理层；其中，所述数据处理层负责根据关联分析规则订阅所述数据采集层的数据，从而进行分析处理；所述数据采集层的数据来源包括流量镜像、防火墙、UDS、IPS等一系列物理位置相邻或不相邻的若干海量异构数据源组成，负责为所述数据处理层提供多角度的安全数据及情景。

[0017] 数据处理层有安全检测模块、安全审计模块及大数据关联分析模块，其中安全检测模块又包括低位检测模块和高位检测模块，大数据关联分析模块分为横向关联分析模块和纵向关联分析模块，安全审计模块分为漏洞分析模块及攻击溯源模块，每个模块由多种技术实现。

[0018] 1)低位检测模块

[0019] 由于APT普遍采用0day漏洞获取权限，通过未知木马进行远程控制，而传统基于特征匹配的检测设备总是要先捕获恶意代码样本，才能提取特征并基于特征进行攻击识别，这就存在先天的滞后性。低位检测是面向终端的异常发现预警应急响应平台的子系统。系统与传统的安全防护软件有本质技术区别。安全增强模块采用异常发现理论模型，不仅能够对已知威胁进行有效监测，同时能够对未知威胁及时发现。

[0020] 异常发现分析与预警子系统包含应用异常发现模块、系统异常发现模块、通信异常发现模块、传输异常发现模块、数据异常发现模块和其他异常发现模块。内部主要逻辑是关联异常发现、决策判决和预警。关联异常发现是根据策略库信息判断异常集的危害等级，即风险，对潜在威胁进行2次推测；其策略库可通过I/O接口与高位监测通信并进行更新、预警，根据用户属性库信息对威胁进行第3次推断，并可通过I/O接口将异常数据集上传，以寻求更为精确的第4次威胁推断。用户属性库的数据可通过用户自主定义或者通过关联各模块收集的用户操作习惯的结果集进行缓慢调整而生成。构建用户属性库的目的是关联用户社会属性，以感知定制性的威胁渗透。低位检测不仅可以支持高位检测，还可以通过预警I/O的输出接口，将预警信息发送给其他安全防护系统，如杀毒软件、蜜网和防火墙等。

[0021] 2)高位检测模块

[0022] 高位检测是面对源头和途径的异常发现预警应急响应平台的核心。高位检测的优势在于可以捕获发现大量骨干网络的异常，挖掘发现大量攻击实施者的资源，检测发现大量源头恶意应用。关联低位检测数据，可使高位检测具备“大视野”的能力，可以有效地分析

0442

发现未知威胁，追溯攻击者身份，应对APT攻击。

[0023] 高位检测技术包括基于流模式对网络服务进行快速识别和分类技术；基于信息论对流量进行认知识别和特征提取技术；基于内容分析的网络监测技术；基于行为特征的检测技术；以网络数据流和网络行为为中心，多角度检测分析计算异常发现技术；针对检测分析获取的网络异常技术；采用多层次网络异常关联与数据融合技术；多层次网络异常关联规则库构建技术等。

[0024] 高位检测通过统计IP层、传输层、应用层等各协议层的特征，建立正常用户网络访问模型，检测可能的异常行为，包括流量传输异常、DNS的速变攻击行为和其他木马的异常网络行为。研究业务网络和出入口网络上用户的不同行为特征，分别对业务网络行为特征和边界网络行为特征建立异常模型，发现未知威胁。高位检测的核心是异常决策引擎，低位检测数据通过I/O接口为高位异常发现决策提供支持。此外低位检测中传输异常发现模块中的异常数据包也可通过与网络信息流对应的I/O接口传入高位检测体系进行深度分析。异常决策引擎的结果可由对应的I/O接口输出，供低位检测参考。异常发现的结果集亦可通过I/O接口为其他安全防护平台提供技术支持。

[0025] 3)安全审计模块

[0026] 安全审计模块收集包括系统日志、资源使用、用户行为以及网络流量等日志和数据记录，并利用漏洞分析子模块和攻击溯源子模块初步分析所收集的数据，除使用传统的黑名单、白名单等分析方法外，更要结合大数据关联分析模块进行综合研判，以从大量的、混杂的日志记录和数据记录中挖掘出潜在威胁问题。

[0027] 4)大数据关联分析模块

[0028] 对低位检测与高位检测的监测数据、安全审计信息进行关联分析和数据挖掘，找出记录之间的相关性，发现新的特征。对于挖掘出来的样本数据进行人工的分类或聚类，以机器学习的方法用分类结果构建特征库训练集，不断提高特征库的分类准确率。

[0029] 利用大数据技术对安全审计模块收集的日志记录、高位检测模块以及低位检测模块的监控数据进行大数据关联分析，以综合分析目标系统是否存在被攻击的可能，实现事前预警。对正在进行的攻击进行大数据分析，可以更准确地理解意图和反向追踪，实现对威胁的综合研判，从而及时采取相关的策略阻止攻击，实现事中阻断。同时对攻击全过程的安全审计信息进行大数据分析，根据追踪路径重现数据的历史状态和演变过程，实现事后审计溯源。

[0030] 各类APT攻击检测工具产生的检测数据、关键流量、日志、上下文以及外部的情报信息等形成了具有多种不同结构的海量数据。数据的多来源性、多结构性、长时序性、低密度性，给大数据分析的防御方法带来了防护挑战。如何有效地利用大数据技术处理分析数据是关键问题。

[0031] 1)数据管理技术

[0032] 在当今大数据时代，曾发展完善的关系数据管理系统(RDBMS)遇到了数据类型单一、扩展性不足等困难。随着数据量的增长，数据分析的基本策略是把计算推向数据，而不是把数据推向计算。近年来，非关系数据管理技术异军突起，对多种类型的数据进行有效的管理、处理和分析；通过并行处理技术获得良好的系统性能；并以其高度的扩展性满足不断增长的数据量的处理要求。

[0033] 关系数据库作为核心的数据引擎，各种来源的数据通过ETL工具导入关系数据库系统。客户端工具通过SQL语言实现例行性的报表生成。针对复杂的分析，SQL的表达能力就暴露出其局限性。如果把数据从数据库中提取出来，导入前端分析工具(SAS.SPSS)进行后续分析，将导致大量的数据移动。同时前端分析工具的数据处理能力将受限于内存大小，数据分析效率将大打折扣。另外，由于当今数据趋向类型多样、规模巨大，关系数据库的组织和处理能力已无法应对大规模的高维时空数据的挑战。因此，在数据分析过程中引入并行计算，将是实现高性能数据管理的必然选择。

[0034] 相对于RDBMS，以MapReduce技术为代表的非关系数据管理技术从存储模型和计算模型上支持更高的容错性、更强的扩展性，为大数据分析提供了很好的运行平台保障，同时，难以用SQL进行表达的分析任务更容易用MapReduce计算函数表达。作为一种旨在解决大规模非结构化数据快速批量处理的并行技术框架，MapReduce获得了长足的发展。为了克服调度算法过于简单导致的性能低下问题，面向多核CPU、GPU、异构系统以及云平台等不同环境的调度策略得以优化。通过定义数据处理流程的逻辑模型，使用模型转换算法和代码生成算法将逻辑模型转化为物理模型，实现了MapReduce对多数据源的数据处理的支持。通过中间结果缓存、流水化和本地化改进了MapReduce的数据流处理能力，进而提出一种针对高速传感数据流的实时MapReduce方法。

[0035] 为了从多维度的数据中检测APT攻击的各个攻击阶段的行为，必须对数据进行深入的分析，而不是仅仅生成简单的报表。这种复杂的分析必须依赖于复杂的分析模型，很难用SQL来进行表达，属于深度分析。因此，面对大数据深度分析的挑战，以MapReduce技术为代表的非关系数据管理技术具有明显的优势。

[0036] 2)深度分析技术

[0037] Techtarget对深度分析的定义是利用复杂数据处理技术从包含结构化、非结构化及半结构化的多源数据集中获取知识。其从用户的需求出发，以有效、用户可接受的方式对大量复杂分布式数据进行分析、提取与汇总，从而为用户决策提供相应的指导。

[0038] 由于深度分析通常需要完成PB级甚至EB级数据的精确定位及复杂查询，并常用于需要实时和近实时响应的环境，典型的数据分析操作(例如：聚集、旋转、切片和汇总)已无法满足其要求，需要引入更复杂的分析技术，例如：时间序列分析、路径分析、What-if分析、社会网络分析以及由于硬件/软件限制而未曾尝试过的复杂统计分析模型。以下对时间序列分析和图分析进行简单介绍：

[0039] ①时间序列分析。时间序列分析利用数理统计的方法对按时间顺序的数据序列加以处理，并以此预测未来事物发展趋势。其既承认事物发展的延续性又考虑到事物发展的随机性。由于时间序列分析具备根据历史数据预测未来数据的特点，故其常应用于市场潜量预测、气象预报、水文预报、国民经济宏观控制以及企业经营管理等领域。

[0040] ②社会网络分析和大规模图分析。社会网络分析侧重于关系和关系的模式，在概念上，采用的方式和方法有别于传统的统计分析和数据处理方法。在图中，1个节点代表社会网络中1个独立的实体，2点之间的1条边代表实体之间的联系。利用社会网络分析，可以获取一些有用的知识，比如发现网络中的关键实体等(关键实体把网络中的各个小组连接在一起)。这些信息可以用于舆情分析、产品销售以及潜在威胁预测等领域。

[0041] 基于APT攻击数据的复杂和多源性特征，引入深度分析可以有效地从历史数据中

提取知识，并以此对未发生的攻击行为进行预测，防患于未然。

[0042] 3)大数据挖掘技术

[0043] 基于大数据挖掘的安全检测是一类重要的安全检测技术。传统的数据挖掘算法基于串行计算，处理数据规模有限，一旦迁移到大数据环境，其执行效率会大幅下降甚至无法运行，无法满足当今大数据时代的需求。

[0044] 云计算是大数据分析与挖掘技术可依托的计算平台，大数据的重点在于对海量数据的挖掘与分析，无法仅使用单台计算机完成运算，而必须采用分布式计算架构。因此，依托云计算的分布式处理、分布式数据库、云存储和虚拟化技术是提高大数据挖掘效率的有效手段，将计算任务分散在不同的计算机上同时运行，使各种应用系统可以根据需要获取存储资源、计算资源和其他服务资源。MapReduce，Hadoop，Spark等高性能、高扩展性的并行计算编程模型、分布式大数据处理框架以及相关关键技术的不断涌现和发展，使得大数据的数据存储和分布式计算成为现实。以分布式计算为基础，大数据挖掘技术可以更好地为基于大数据的安全检测问题提供新的理论与技术支持。目前基于大数据挖掘的安全检测技术主要研究各种数据挖掘算法的并行化策略，基于分布式计算的新型大数据挖掘方法以及在MapReduce，Spark框架上实现大规模的异常分析和检测等。

[0045] 以上所述仅为本发明的优选实施例，并不用于限制本发明，对于本领域技术人员而言，本发明可以有各种改动和变化。凡在本发明的精神和原理之内所作的任何修改、等同替换、改进等，均应包含在本发明的保护范围之内。

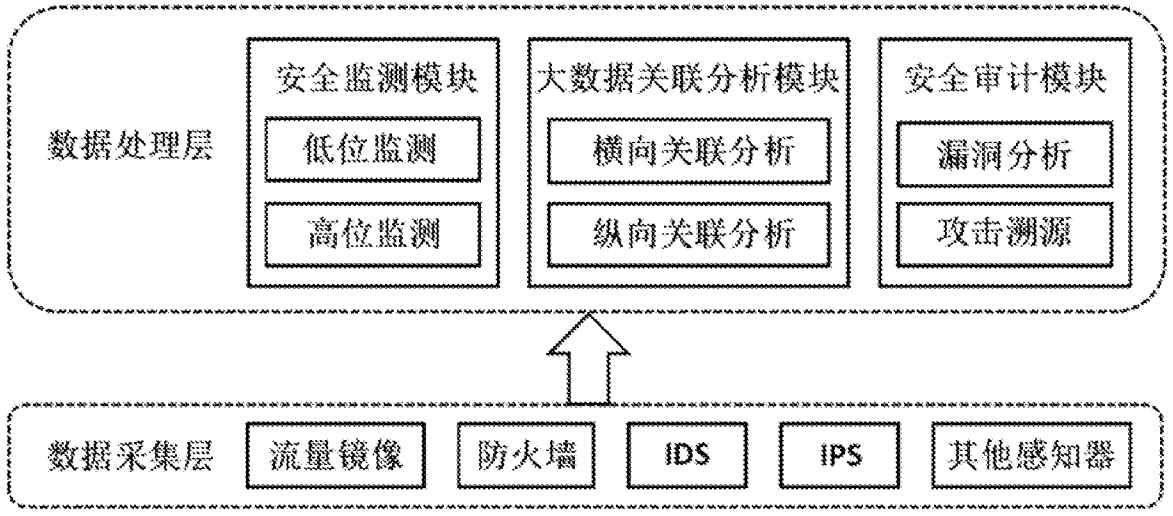

数据处理层
安全监测模块
低位监测
高位监测
大数据关联分析模块
横向关联分析
纵向关联分析
安全审计模块
漏洞分析
攻击溯源
数据采集层
流量镜像
防火墙
IDS
IPS
其他感知器

图1

PATENT
Attorney Docket No. 15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.: 17/657,972	)	Examiner: SU, SARAH
	)	
Filed: April 5, 2022	)	Confirmation No.: 2547
	)	
For: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT UNDER 37 C.F.R. § 1.97(b)

Pursuant to 37 C.F.R. §§ 1.56 and 1.97(b), Applicant brings to the Examiner's attention the documents listed on attached Form PTO/SB/08. This Information Disclosure Statement is being filed before the mailing date of a first Office Action after the filing of a Request for Continued Examination in the above-referenced application.

Statement under 37 C.F.R. § 1.704(d)

The following constitutes the statement specified by § 1.704(d) required to avoid a negative patent term adjustment. No item of information contained in the IDS was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the IDS was known to any individual designated in § 1.56(c) more than thirty days prior to the filing of this

Information Disclosure Statement. In accordance with 37 C.F.R. § 1.704(d), no negative patent term adjustment will result from this Information Disclosure Statement.

Copies of the listed foreign and non-patent literature documents are attached.

The following documents are listed on the accompanying PTO/SB/08 and are in a non-English language:

1. CN 114584359 A - An English-language translation of this document is also enclosed.
2. CN 109995722 A - An English-language translation of this document is also enclosed.

Applicant respectfully requests that the Examiner consider the listed documents on the attached Form PTO/SB/08 and indicate that they were considered by making appropriate notations on the attached form.

This submission does not represent that a search has been made or that no better art exists and does not constitute an admission that each or all of the listed documents are material or constitute "prior art." If the Examiner applies any of the documents as prior art against any claim in the application and Applicant determines that the cited documents do not constitute "prior art" under United States law, Applicant reserves the right to present to the U.S. Patent and Trademark Office the relevant facts and law regarding the appropriate status of such documents.

Applicant further reserves the right to take appropriate action to establish the patentability of the disclosed invention over the listed documents, should one or more of the documents be applied against the claims of the present application.

If there is any fee due in connection with the filing of this Statement, please charge the fee to deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: December 22, 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

PTO/SB/30EFS (01-22)
Approved for use through 05/31/2024. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

REQUEST FOR CONTINUED EXAMINATION(RCE)TRANSMITTAL
(Submitted Only via EFS-Web)

Application Number	17657972	Filing Date	2022-04-05	Docket Number (if applicable)	15635.0005-00000	Art Unit	2431
First Named Inventor	Avi SHUA			Examiner Name	SU, SARAH		

This is a Request for Continued Examination (RCE) under 37 CFR 1.114 of the above-identified application.
Request for Continued Examination (RCE) practice under 37 CFR 1.114 does not apply to any utility or plant application filed prior to June 8, 1995, or to any design application. The Instruction Sheet for this form is located at WWW.USPTO.GOV

SUBMISSION REQUIRED UNDER 37 CFR 1.114

Note: If the RCE is proper, any previously filed unentered amendments and amendments enclosed with the RCE will be entered in the order in which they were filed unless applicant instructs otherwise. If applicant does not wish to have any previously filed unentered amendment(s) entered, applicant must request non-entry of such amendment(s).

☐ Previously submitted. If a final Office action is outstanding, any amendments filed after the final Office action may be considered as a submission even if this box is not checked.

- ☐ Consider the arguments in the Appeal Brief or Reply Brief previously filed on _____
- ☐ Other _____

☒ Enclosed

- ☒ Amendment/Reply
- ☐ Information Disclosure Statement (IDS)
- ☐ Affidavit(s)/ Declaration(s)
- ☒ Other Do not enter the Amendment after Final filed 13 October 2022

MISCELLANEOUS

☐ Suspension of action on the above-identified application is requested under 37 CFR 1.103(c) for a period of months _____ (Period of suspension shall not exceed 3 months; Fee under 37 CFR 1.17(i) required)

☐ Other _____

FEES

The RCE fee under 37 CFR 1.17(e) is required by 37 CFR 1.114 when the RCE is filed.
☒ The Director is hereby authorized to charge any underpayment of fees, or credit any overpayments, to Deposit Account No 060916

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED

☒ Patent Practitioner Signature

Applicant Signature

PTO/SB/30EFS (01-22)
Approved for use through 05/31/2024. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Signature of Registered U.S. Patent Practitioner			
Signature	/Christopher C. Johns/	Date (YYYY-MM-DD)	2022-12-14
Name	Christopher C. Johns	Registration Number	68,664

This collection of information is required by 37 CFR 1.114. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**12/14/2022 04:09:37 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61300389	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 5

DOCUMENT		PAGES	DESCRIPTION	SIZE (KB)
Response to Advisory Action 15635.0005-00000.pdf		10	-	46 KB
Response to Advisory Action 15635.0005-00000-A.NE.pdf	(1-1)	1	Response After Final Action	13 KB
Response to Advisory Action 15635.0005-00000-CLM.pdf	(2-7)	6	Claims	29 KB
Response to Advisory Action 15635.0005-00000-REM.pdf	(8-10)	3	Applicant Arguments/Remarks Made in an Amendment	24 KB

15635.0005 EOT.pdf	2	Extension of Time	167 KB
15635.0005 RCE.pdf	3	Request for Continued Examination (RCE)	1628 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
Response to Advisory Action 15635.0005-00000.pdf	00ED2EA7FC5B4377DC9D5D09679E55300E54A90C7213C42B8F89DAC9EB914B71579ADA26B8A77D7367B93FDFE107ADF3D87F1C5A45907B759994CA83AFEF160C
Response to Advisory Action 15635.0005-00000-A.NE.pdf	8EECD2E194FB6C3AE923C00C7C1884B22F35CF2A95DE2A2D3AA752D646ACCB9545B07EA9A59D9A510C306C4BCB16309801928313A96FD75D1A48F166CE9121B7
Response to Advisory Action 15635.0005-00000-CLM.pdf	1AAEB90D68FE0E6FB5E1FF3B779B660CB622C54F44DE0F5C1B90988D1C29914D48A9CF8809BD05F062E26CCDBD71A1CD4F08E2F4925FA2F588A0130D62E7939E
Response to Advisory Action 15635.0005-00000-REM.pdf	CCD93BEF8FB32A651E37D4C33C9C2389E24AC1F328EAF57F3BB44B8650EF80FC83D2E0B82CF35DE7E495553D4085572C7F02315C8C0626AE1F540753C51C35AD
15635.0005 EOT.pdf	A9B571273E89B89C51D2A5499E16A6664F5A9B137FFCBAD24096A54DE3706F9480CDA67750E930E8DFDA564EF785BF83CE5DDD7AEDCFA2B0F70B5064BD5FD0D5
15635.0005 RCE.pdf	631A2AE03136A867AE655319B9B3ACAE8926BD794FDD2A814C96579648ED403A572D905FD01BF0B29E741041A1784962004 3005F341DB08E74359F22F42AFF29

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d)

and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/AIA/22 (10-20)
Approved for use through 11/30/2020. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PETITION FOR EXTENSION OF TIME UNDER 37 CFR 1.136(a)

Docket Number (Optional): 15635.0005-00000

Application Number: 17/657,972

Filed: April 5, 2022

For: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Art Unit: 2431

Examiner: SU, SARAH

This is a request under the provisions of 37 CFR 1.136(a) to extend the period for filing a reply in the above-identified application.

The requested extension and fee are as follows (check time period desired and enter the appropriate fee below):

		Fee	Small Entity Fee	Micro Entity Fee	
☑	One month (37 CFR 1.17(a)(1))	$220	$110	$55	$ 220
☐	Two months (37 CFR 1.17(a)(2))	$640	$320	$160	$
☐	Three months (37 CFR 1.17(a)(3))	$1,480	$740	$370	$
☐	Four months (37 CFR 1.17(a)(4))	$2,320	$1,160	$580	$
☐	Five months (37 CFR 1.17(a)(5))	$3,160	$1,580	$790	$

☐ Applicant asserts small entity status. See 37 CFR 1.27.

☐ Applicant certifies micro entity status. See 37 CFR 1.29.
Form PTO/SB/15A or B or equivalent must either be enclosed or have been submitted previously.

☐ A check in the amount of the fee is enclosed.

☐ Payment by credit card. Form PTO-2038 is attached.

☐ The Director has already been authorized to charge fees in this application to a Deposit Account.

☑ The Director is hereby authorized to charge any fees which may be required, or credit any overpayment, to Deposit Account Number 06-0916.

☑ Payment made via EFS-Web.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

I am the

☐ applicant.

☐ attorney or agent of record. Registration number _____.

☑ attorney or agent acting under 37 CFR 1.34. Registration number 68,664.

/Christopher C. Johns/ — Signature

14 December 2022 — Date

Christopher C. Johns — Typed or printed name

(202) 408-4155 — Telephone Number

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4 for signature requirements and certifications. Submit multiple forms if more than one signature is required, see below*.

☑ * Total of 1 forms are submitted.

This collection of information is required by 37 CFR 1.136(a). The information is required to obtain or retain a benefit by the public, which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 6 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Mail Stop PCT, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	12/14/2022 04:09:37 PM ET	15635.0005-00000

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61300389	AUTHORIZED BY	Christopher Johns
CUSTOMER #	22852	FILING DATE	04/05/2022
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Avi SHUA

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
CARD / 1007	E2022BDG11533540	Michelle Daniels

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
1251	EXTENSION FOR RESPONSE WITHIN FIRST MONTH	220.00	1	220.00
1801	REQUEST FOR CONTINUED EXAMINATION (RCE) - 1ST REQUEST (SEE 37 CFR 1.114)	1360.00	1	1360.00
			TOTAL AMOUNT:	$1,580.00

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PATENT
Attorney Docket No.15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.:17/657,972	)	Examiner: Sarah Su
	)	
Filed: April 5, 2022	)	
	)	
For: **Systems and Methods for Managing Cyber Vulnerabilities**	)	Confirmation No.: 2547
	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

AMENDMENT WITH REQUEST FOR CONTINUED EXAMINATION

In reply to the Final Office Action mailed on September 1, 2022, and the Advisory Action mailed on October 31, 2022, Applicant responds to the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims in this paper.

Remarks follow the amendment sections of this paper.

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1-20. (canceled)

21. (Currently Amended) A cyber security system for a cloud environment, the system comprising:

at least one processor configured to:

using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment;

identify an installed software application in the accessed block storage volume;

analyze the identified installed software application to determine an associated software version;

access a data structure of known software vulnerabilities for a plurality of versions of software applications;

perform a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and

~~perform at least one of:~~

~~query the cloud provider API to determine network accessibility information related to the workload;~~

~~identify at least one port on which the vulnerable application is accessible; or~~

use network accessibility information and at least one port to identify one or more <u>of the known and identified</u> vulnerabilities susceptible to attack from

outside the workload, wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.

22. (Previously Presented) The system of claim 21, wherein the at least one processor is further configured to implement a remedial action in response to the identified one or more vulnerabilities.

23. (Previously Presented) The system of claim 22, wherein the remedial action includes transmitting an alert to a device associated with an administrator.

24. (Currently Amended) The system of claim 21, wherein the at least one processor is further configured to:

~~querying~~ query the cloud provider API to determine network accessibility information related to the workload further comprises examining data sources associated with the workload; and

~~determining~~ determine the network accessibility information based on the examined data sources.

25. (Cancelled)

26. (Previously Presented) The system of claim 21, wherein to identify the installed software application, the at least one processor is configured to:

extract data from at least one of operating system packages, libraries, or program language libraries; and

identify the installed software application based on the extracted data.

27. (Previously Presented) The system of claim 21, wherein the at least one processor is further configured to identify a version of the installed software application.

28. (Previously Presented) The system of claim 21, wherein the identified installed software application includes one or more scripts.

29. (Previously Presented) The system of claim 21, wherein the data structure includes aggregated vulnerability data.

30. (Previously Presented) The system of claim 29, wherein the aggregated vulnerability data includes data from one or more third-party vendors.

31. (Previously Presented) The system of claim 29, wherein the aggregated vulnerability data includes data collected by a scanner.

32. (Previously Presented) The system of claim 29, wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug.

33. (Currently Amended) A method, comprising:

using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment;

identifying an installed software application in the accessed block storage volume;

analyzing the identified installed software application to determine an associated software version;

accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;

performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and

~~perform at least one of the following:~~

~~query the cloud provider API to determine network accessibility information related to the workload;~~

~~identify at least one port on which the vulnerable application is accessible; or~~

use network accessibility information and at least one port to identify one or more <u>of the known and identified</u> vulnerabilities susceptible to attack from outside the workload, <u>wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log</u>.

34. (Previously Presented) The method of claim 33, further comprising implementing a remedial action in response to the identified one or more vulnerabilities.

35. (Previously Presented) The method of claim 34, wherein the remedial action includes transmitting an alert to a device associated with an administrator.

36. (Currently Amended) The method of claim 33, wherein the method further comprises querying the cloud provider API to determine the network accessibility information related to the workload, by: ~~further comprises:~~

examining data sources associated with the workload; and

determining the network accessibility information based on the examined data sources.

37. (Canceled)

38. (Previously Presented) The method of claim 33, wherein identify the installed software application comprises:

extracting data from at least one of OS packages, libraries, or program language libraries; and

identifying the installed software application based on the extracted data.

39. (Previously Presented) The method of claim 38, wherein the at least one processor is further configured to identify a version of the installed software application.

40. (Currently Amended) A non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising:

using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment;

identifying an installed software application in the accessed block storage volume;

analyzing the identified installed software application to determine an associated software version;

accessing a data structure of known software vulnerabilities for a plurality of versions of software applications; and

~~perform at least one of the following:~~

~~query the cloud provider API to determine network accessibility information related to the workload;~~

~~identify at least one port on which the vulnerable application is accessible; or~~

use network accessibility information and at least one port to identify one or more of the known and identified vulnerabilities susceptible to attack from outside the workload, wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.

41-159 (Cancelled)

REMARKS

The present response is intended to be fully responsive to all points of rejection raised by the instant Advisory Action mailed October 31, 2022, and is believed to place the application in condition for allowance.

In the Advisory Action, the Examiner referenced the abstract, lines 2-14 of Stopel et al. (US2019/0116199 A1) as allegedly teaching elements of the pending claims.

In response, Applicant proposes to amend claims 21, 33, and 40 No new matter has been added.

Applicants respectfully request that the Examiner consider this Response After Final based on the After Final Consideration Pilot (AFCP) 2.0. A Certification and Request for Consideration under the After Final Consideration Pilot Program 2.0 (Form PTO/SB/434) is being filed concurrently herewith.

35 U.S.C. § 102 REJECTIONS

Claims 21 (and dependent claims 22, 26, 27, 29, 30, and 32) and claims 33 (and dependent claims 34 and 38-39) and 40 are rejected under 35 U.S.C. § 102 as anticipated by US 20200389483 to Dye. The Examiner further raises concern that specific elements of claims 21, 33, and 40 are allegedly disclosed by US 20190116199A1 to Stopel et al. and that additional prior art searching is required. The Applicant respectfully traverses this rejection in view of the remarks that follow.

Claim 21, as amended, recites, *inter alia*:

> *use network accessibility information and at least one port to identify one or more of the known and identified vulnerabilities susceptible to attack from outside the workload, wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.*

The Examiner contends that Stopel et al. discloses claim element "identify at least one port on which the vulnerable application is accessible" because the abstract discloses this element. However, this is incorrect.

Stopel et al. discloses a system that, upon identification of a misconfigured application, can detect at least one port through which an application is accessible. *See, e.g.*, *Stopel* at Abstract, ¶¶ [0003], [0016]-[0018], [0027]-[0028]. However, this teaching in Stopel fails to teach or suggest "network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log," at least because Stopel teaches a host device using an at least one port to attempt to access an application to determine whether a vulnerability exists based on an application's misconfiguration, ¶ [0016], rather than identifying "one or more of the known and identified vulnerabilities" from a "data structure of known software vulnerabilities." *Stopel* at ¶ [0016].

For at least the foregoing reasons, independent claims 21, 33, and 40, and dependent claims 22, 26, 27, 29, 30, 32, 34, and 38-39, as amended, is allowable over this rejection and the Applicant respectfully requests reconsideration based on the above.

35 U.S.C. § 103 REJECTIONS

Claims 23-25, 28, 31, and 35-37 are rejected under 35 U.S.C. § 103 as being unpatentable over various combinations of Dye, Jones, Joffe, Liu. As discussed above, Dye fails to teach elements of the claims. The Examiner does not assert that any of Jones, Joffe, Liu, or Stopel make up for these deficiencies, nor do they. Therefore, the rejection of claims 23-25, 28, 31, and 35-37 should be withdrawn.

CONCLUSION

Applicant respectfully points out that the final action by the Examiner presented new arguments as to the application of the references against the pending claims.

Finally, Applicant submits that the entry of the amendment would place the application in better form for appeal, should the Examiner dispute the patentability of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: December 14, 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				***Complete if Known***	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	1	*Attorney Docket Number*	**15635.0005-00000**

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-8,006,285-B1	08-23-2011	Perlman	

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2020/0059481-A1	02-20-2020	Sekar et al.	

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Foreign Patent Document Country Code Number Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	Translation
		WO 2021/072877 A1	04-22-2021	PING AN TECH SHENZHEN CO LTD		YES
		CN 110708315 A	01-17-2020	DBAPP SECURITY INF TECH CO LTD		YES

NON PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Ex parte Quayle Action No. 17/657,981 dated October 31, 2022, in the United States Patent and Trademark Office. (15635.0008-00000)	N/A

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

PATENT
Attorney Docket No. 15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.: 17/657,972	)	Examiner: SU, SARAH
	)	
Filed: April 5, 2022	)	Confirmation No.: 2547
	)	
For: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT UNDER 37 C.F.R. § 1.97(d)

Pursuant to 37 C.F.R. §§ 1.56 and 1.97(d), Applicant brings to the attention of the Examiner the documents on the attached listing. This Information Disclosure Statement is being filed after a Final Office Action and is accompanied by a fee of $260.00 as specified under § 1.17(p) and a statement as specified under § 1.97(e).

Statement under 37 C.F.R. § 1.97(e)

The following constitutes the statement specified by 37 C.F.R. § 1.97(e)(2). No item of information contained in the IDS was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the IDS was known to any individual designated in 37 CFR 1.56(c) more

than three months prior to the filing of this Information Disclosure Statement. See 37 CFR 1.97(e)(2).

Copies of the listed foreign and non-patent literature documents are attached.

The following documents are listed on the accompanying PTO/SB/08 and are in a non-English language:

1. WO 2021/072877 A1 - An English-language translation of this document is also enclosed.
2. CN 110708315 A - An English-language translation of this document is also enclosed.

Applicant respectfully requests that the Examiner consider the listed documents on the attached Form PTO/SB/08 and indicate that they were considered by making appropriate notations on the attached form.

This submission does not represent that a search has been made or that no better art exists and does not constitute an admission that each or all of the listed documents are material or constitute "prior art." If the Examiner applies any of the documents as prior art against any claim in the application and Applicant determines that the cited documents do not constitute "prior art" under United States law, Applicant reserves the right to present to the U.S. Patent and Trademark Office the relevant facts and law regarding the appropriate status of such documents.

Applicant further reserves the right to take appropriate action to establish the patentability of the disclosed invention over the listed documents, should one or more of the documents be applied against the claims of the present application.

If there is any fee due in connection with the filing of this Statement, please charge the fee to deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: November 9, 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

Espacenet

CN110708315A Asset vulnerability identification method, device and system

Applicants: DBAPP SECURITY INF TECH CO LTD

Inventors: WANG YINGXIN,FAN YUAN

Classifications:
IPC ***H04L29/06;***

CPC **H04L63/1433 (CN);**

Priorities: CN201910956125A 2019-10-09

Application: CN201910956125A·2019-10-09

Publication: CN110708315A·2020-01-17

Published as: CN110708315A

Asset vulnerability identification method, device and system

Abstract

The invention provides an asset vulnerability identification method, device and system, and relates to the technical field of network security, and the method comprises the steps: obtaining the assetinformation of to-be-detected assets; sending the asset information to a cloud data center; the cloud data center performing vulnerability identification detection on the to-be-detected assets based on the asset information and generating a vulnerability identification result; wherein the cloud data center stores target threat intelligence data, and the target threat intelligence data comprises atleast one of threat intelligence data uploaded by a security service provider, threat intelligence data uploaded by a security intelligence organization and threat intelligence data uploaded by an enterprise to which assets to be detected belong; and obtaining the vulnerability identification result, and generating early warning information based on the vulnerability identification result, thereby solving the technical problem that an existing asset vulnerability identification method is relatively single in asset vulnerability detection.

identification method, device and system of asset vulnerability

DOCUMENT ID
CN-110708315-A

DATE PUBLISHED
2020-01-17

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
WANG, YING-XIN	N/A	N/A	N/A	N/A
FAN, Yuan	N/A	N/A	N/A	N/A

DATE FILED
2019-10-09

Abstract

The invention claims a method of asset vulnerability identification method, device and system, referring to the technical field of network security, comprising: obtaining the asset information of the detected asset, the asset information is sent to the cloud data center. to make the cloud data centre based on asset information asset to be detected for detecting vulnerability identification, and generating a vulnerability identification result, wherein the cloud data center stores target threat intelligence data, target threat intelligence data comprises at least one of the following: the security service provider upload of threat intelligence data. threat intelligence data uploaded by the security information organization and the threat information data uploaded by the enterprise detection asset belongs to, obtaining the vulnerability identification result, and generating the pre-warning information based on vulnerability identification result, solves the technical problem of detecting the asset vulnerability of existing asset vulnerability identification method is single.

Description

INVENTION-TITLE
technical field
This invention claims network security technology field, especially refers to a kind of asset vulnerability identification method, device and system.
BACKGROUND
under the current network safety technology condition, the occurrence of a network attack vulnerability basic network threat using information assets exposed by malicious action is carried out. in the network space, network threat is objectively present, to reduce the risk of the network attack, vulnerability reinforcing the information asset is a very effective security technical measures. but before the asset vulnerability reinforcing, needs to know the vulnerability information asset with which, therefore, information asset vulnerability detection is abnormal important for network security protection.
However, the existing vulnerability detection method has the technical problems is single vulnerability detection to the asset.
Aiming at said problem, yet provides effective solution.
summary of the invention
Therefore, it is an object of the present invention is to provide a kind of asset vulnerability identification method, device and system, so as to solve the technical problem of the asset vulnerability detecting existing

of asset vulnerability identification method is single.
In a first aspect, the invention provides a method for identifying asset vulnerability, comprising: obtaining the asset information of the detected asset, said asset information to the cloud data center. to make the cloud data centre for detecting vulnerability identification based on said asset information for the asset to be detected, and generating vulnerability identification result, wherein the cloud data center stores a target threat intelligence data, the target threat intelligence data comprises at least one of the following: the security service provider upload of threat intelligence data. threat intelligence data uploaded by the security information organization and the threat information data uploaded by the enterprise detection asset belongs to, obtaining the vulnerability identification result and generating the pre-warning information based on the vulnerability identification result.
Further, obtaining the asset information of asset detection, comprising: sending the instruction to the local asset information base so that the local asset information database based on the obtaining instruction collecting attribute information and fingerprint information of the asset to be detected, wherein officer name of the asset to be detected belongs to said fingerprint information include, but are not limited to, the following information: the classification information of the to-be-detected asset factory manufacturer name, the to-be-detected asset, the asset to be detected operating system information; obtaining the attribute information and the fingerprint information, the attribute information and the fingerprint information is determined as the asset information, wherein the classification information comprises: first level classification information and second level classification information, first level classification comprises the following types: server, a network device, a security device, a secondary classification comprises the following types: a firewall, a switch, a router and a database.
Further, if the two-stage classification of the asset to be detected is a database, then the fingerprint information further comprises the type and version of the asset to be detected.
Further, the asset information is sent to the cloud data center to make the cloud data centre based on said asset information for the asset to be detected for detecting vulnerability identification, comprising: the cloud service port of said asset information provided by the cloud data centre sending to the cloud data center, so that the cloud data centre based on preset leak detection program and the asset information for the asset to be detected for detecting vulnerability identification.
Further, the cloud data centre for providing infrastructure as a service and platform as a service, the cloud data centre stores the target threat intelligence data based on Hadoop framework.
In a second aspect, the invention provides a kind of asset vulnerability identification device, comprising: obtaining unit, sending unit and a warning unit, wherein the obtaining unit is used for obtaining the asset information of the detected asset; the sending unit is used for sending the asset information to the cloud data center, so that the cloud data centre performing vulnerability recognition detection to the to-be-detected asset based on the asset information, and generating a vulnerability identification result, wherein the cloud data center stores a target threat intelligence data. the target threat intelligence data comprises at least one of the following: the security service provider upload of threat intelligence data, threat information data uploaded by the security information organization and the threat information data uploaded by the enterprise detection asset belongs to; the warning unit for obtaining the vulnerability identification result, and generating the pre-warning information based on the vulnerability identification result.
Further, the obtaining unit is further used for sending the instruction to the local asset information base so that the local asset information database based on the obtaining instruction collecting attribute information and fingerprint information of the asset to be detected, wherein officer name of the asset to be detected belongs to said fingerprint information include, but are not limited to, the following information: the classification information of the to-be-detected asset factory manufacturer name, the to-be-detected asset, the asset to be detected of operating system information, obtaining the attribute information and the fingerprint information; the attribution information and the fingerprint information is determined as the asset information, wherein the classification information comprises: first level classification information and second level classification information, first level classification comprises the following types: server, a network device, a security device, a secondary classification comprises the following types: a firewall, a switch, a router and a database.
Further, if the two-stage classification of the asset to be detected is a database, then the fingerprint information further comprises the type and version of the asset to be detected.

Further, the sending unit is further used for: the cloud service port of said asset information provided by the cloud data center is sent to the cloud data center, so that the cloud data centre based on preset leak detection program and the asset information for the asset to be detected for detecting vulnerability identification.

In a third aspect, the invention claims a kind of asset vulnerability identification system, comprising: a production information collecting device, a cloud data center and leak warning platform, wherein, the local asset information base for collecting the asset information of asset to be detected; the cloud data centre is used for performing vulnerability recognition detection on the to-be-detected asset based on the asset information, and generating the vulnerability identification result, the vulnerability warning platform for generating the pre-warning information based on the vulnerability identification result.

In the embodiment of the invention, firstly, obtaining the to-be-detected asset information of the asset, the asset information is sent to the cloud data center, to make the cloud data centre based on asset information asset to be detected for detecting vulnerability identification, and generating vulnerability identification result, wherein cloud data center stores target threat intelligence data, target threat intelligence data comprises at least one of the following: the security service provider upload of threat intelligence data, security threat information data uploaded by the information organization and the threat information data uploaded by the enterprise detection asset belongs, vulnerability identification result for characterizing the asset to be detected whether there is vulnerability, obtaining vulnerability identification result, and generating the pre-warning information based on vulnerability identification result.

In the invention, the asset information of the asset to be detected to the obtained sent to the cloud data center to make the cloud data centre according to the target the stored threat intelligence data, assets to be detected of the comprehensive vulnerability detection. Because the cloud data centre according to the collected vulnerability multi-party provides the threat intelligence data asset to be detected for leak detection, the asset to be detected for overall vulnerability detection, so as to solve the technical problem of detecting of the asset vulnerability in the existing technology is single. so as to realize the asset to be detected to the technical effect of the comprehensive vulnerability detection.

Other features and advantages of the present invention will be described in the following specification, and partly become apparent from the specification, or by implementing the present invention understood. structure for purpose of this invention and the other advantages particularly pointed out in the specification, claims and drawings realized and obtained.

To make the above object, features and advantages of the present invention can be best understood, lower Went preferred embodiments, and the accompanying drawings, to the following detailed description.

Description

In order to clearly illustrate the technical solution of the invention specifically implementation manner or in the existing technology, will now be simply described drawings required to be used in the specific embodiment or existing technology description, obviously, the drawings in the following description are some embodiments of the present invention, for this field common technician, under the premise of without paying creative labor, it also can be other drawings according to these drawings to obtain.

FIG. 1 an embodiment of the present invention provides a method for identifying asset vulnerability of FIG.

FIG. 2 an embodiment of the present invention to provide an asset vulnerability identification device of the present invention;

FIG. 3 an embodiment of the present invention to provide an asset vulnerability identification system of the present invention;

FIG. 4 is a schematic diagram of a server according to an embodiment of the invention.

Specific implementation methods

In order to make the purpose of the invention, technology scheme advantages and more clearly, Following combine the accompanying drawings to the technical solution of the invention will be described in a clearly and fully, obviously, the embodiment description of the invention embodiment part, but not all of the embodiments. Based on the embodiments of the invention, all other embodiments obtained by ordinary technician in present technical field without any creative work is within the protective scope of the present invention.

about a vulnerability detection technology. the current vulnerability detection technology of the information asset by actively detecting technology mainly relies on special vulnerability scanning device to achieve, in

this way, the vulnerability scanning device using which carries the "vulnerability rule library" asset to actively scan the various kinds of information, such as when to scan for a certain asset, the rule R if the scanning result hits the "vulnerability rule library", it is determined that the asset has vulnerability R vulnerability corresponding to the rule. result in such a case, the vulnerability scanning depends on the quantity and quality of "vulnerability rule library" carried by the vulnerability scanning device. if a certain a certain vulnerability information exposed out of the asset vulnerability in the "rule library", then the leak can be detected by the vulnerability scanning device. if a certain vulnerability not leak the "rule library", then the vulnerability scanning device cannot be detected, the vulnerability exists even if true this vulnerability.
to "vulnerability rule base" of the update. For the discovery of vulnerability model, vulnerability scanning service provider will generally periodically or aperiodically the vulnerability are packaged into software upgrading packet, vulnerability scanning device by the upgrade of the online or offline to finish upgrading of vulnerability rule library ", of vulnerability scanning device side after finishing the upgrading can have the newly discovered vulnerability detection capability.
identification method of the existing asset vulnerability has the following disadvantages:
1 based on the vulnerability scanning technology of active scanning needs to send data to the scanned target by a network packet, due to the complexity of the real environment, which may cause operation of the target host is not stable, causing downtime of the information asset, stopping service, data loss and so on.
2, the hysteresis of the vulnerability detection. For the Oday vulnerability and vulnerability, the vulnerability scanning device are not disclosed because of hysteresis vulnerability rule library " upgrade, may cause some vulnerability cannot be detected, so there is the risk the malicious threat using the vulnerability to attack.
3, single detection capability. Generally, the leak detection device of a manufacturer has its inherent technical and limit of vulnerability rule library, cannot exert a plurality of security vendor and vulnerability detection capability of various security organization, cannot realize the polymerization of vulnerability detection capability.
4, the efficiency of vulnerability detection is low. The traditional vulnerability scanning equipment technology implementation reasons, a plurality of asset for vulnerability detection time, low efficiency, typically complete vulnerability detection of one asset requires several minutes.
Aiming at the above disadvantages, the invention adopts the following method to overcome said disadvantages:
First Embodiment
according to an embodiment of the present invention, there is provided a method for identifying asset vulnerability of embodiment, it should be noted that the illustration of drawings in the step flow of execution in computer system can be such as a group of computer-executable instructions, and although the logical order is shown in the flow chart, but in some cases, steps may be executed in different order shown or described.
FIG. 1 according to an embodiment of the present invention a method for identifying asset vulnerability, as shown in FIG. 1, the method comprises the following steps:
step S102, obtaining the asset information of the detected asset;
step S104, the asset information is sent to the cloud data center to make the cloud data centre performing vulnerability recognition detection to the to-be-detected asset based on the asset information, and generating a vulnerability identification result, wherein the cloud data center stores a target threat intelligence data, the target threat intelligence data comprises at least one of the following: the security service provider upload of threat intelligence data, threat information data uploaded by the security information organization and the threat information data uploaded by the enterprise detection asset belongs to;
step S106, obtaining the vulnerability identification result, and generating the pre-warning information based on the vulnerability identification result.
In the invention, the asset information of the asset to be detected to the obtained sent to the cloud data center to make the cloud data centre according to the target the stored threat intelligence data, assets to be detected of the comprehensive vulnerability detection. Because the cloud data centre according to the collected vulnerability multi-party provides the threat intelligence data asset to be detected for leak detection, the asset to be detected for overall vulnerability detection, so as to solve the technical problem of detecting of the asset vulnerability in the existing technology is single. so as to realize the asset to be detected to the

technical effect of the comprehensive vulnerability detection.
It should be noted that, the capability data by establishing threat intelligence data uploaded by the security service provider, a security threat intelligence data uploaded by the information organization and the cloud data centre threat intelligence data uploaded by the enterprise detection asset belongs, may be incorporated consisting of and abutting each party security threat intelligence data, form a threat intelligence synergy and multi-security information of the synergy of security threat information sharing mechanism construction.
In addition, it should be noted that, the vulnerability warning information can be sent to the worker through short message, mail and so on, so that the worker can timely processing the asset vulnerability to be detected, so as to reduce the risk to be detected asset faces.
In the embodiment of the invention, the step S102 further comprises the following steps:
step S11, sending the instruction to the local asset information base so that the local asset information library collects attribute information and fingerprint information of the asset to be detected based on the acquired instruction, name of responsible person, the asset to be detected belongs, the fingerprint information include, but are not limited to, the following information: the classification information of the to-be-detected asset factory manufacturer name, the to-be-detected asset, the asset to be detected operating system information;
step S11, obtaining the attribute information and the fingerprint information, the attribute information and the fingerprint information is determined as the asset information.
In the embodiment of the invention, after sending the instruction to the local asset information base, local asset information library according to the obtaining instruction collecting the attribute information and fingerprint information of the detected assets.
Then, the obtained local asset information base collected after detecting the attribute information and fingerprint information of the asset as the asset information of the detected asset.
It should be noted that the attribute information include, but are not limited to the following information: name of the asset to be detected, after detecting the asset belongs to responsible person name and other information.
said fingerprint information include, but are not limited to, the following information: operating system information detection asset to be factory manufacturer name, the classification information of the detected asset, asset to be detected, wherein the classification information comprises: first level classification information and second level classification information, first level classification comprises the following types: server, a network device, a security device, a secondary classification comprises the following types: a firewall, a switch, a router and a database.
In addition, it should be noted that operating system to be detected comprises: type of information assets to the operating system, the version of the operating system and other information.
if the asset is the database to be detected, then the version and type of fingerprint information of the detected asset comprises a cash asset.
in the asset information of the asset to be detected comprises fingerprint information, so that each of the fingerprint information to determine one implementation asset to be detected, so as to accurately by asset fingerprint matching technology assets to be detected for the passive vulnerability detection accurately, reduce the vulnerability detection when running effect of assets to be detected.
In addition, local asset information database in the to-be-detected asset information of asset collection can be performed through automatic or manual input, it automatically collects the asset information mainly by asset detection device finishes, manual input, the local asset information library can provide visualization of the input interface for the worker to the asset information manually recording local asset information in the database.
In the embodiment of the invention, the step S104 further comprises the following steps:
step S21, the cloud service port of said asset information provided by the cloud data centre sends to the cloud data center, so that the cloud data centre inlet vulnerability identification detection based on preset leak detection program and the asset information for the asset to be detected.
In the embodiment of the invention, the threat intelligence database provided by the cloud data centre IaaS (infrastructure as a service) and relatedness (platform as a service) as a base layer, using Hadoop framework of building based on the large data processing architecture, the threat of data source information in the database may be from various security service providers, security intelligence organization, enterprise

itself. data in the PaaS (platform as a service) constructing leak detection cloud application program (i.e., the predetermined leak detection program), the program can call the threat intelligence database in order for assets to be detected for detecting vulnerability identification. at the same time, the program provided by via the Internet cloud service interface to realize the service of vulnerability detection interface.
Because the cloud data centre can by real-time update of threat information cloud data as basic data, by means of cloud interface provides leak detection service, it can real time provide fast and in time of the detection result is 0day vulnerability detection and position detection of the asset, solve the vulnerability detection service due to the hysteresis caused by the risk vulnerability rule library, and cloud data center to cloud interface provided by way of can batch the asset vulnerability detection and improves the detection efficiency of the asset vulnerability.
Second Embodiment
The invention further claims an asset vulnerability identification device, the device identification method for performing asset vulnerability of embodiment of the above content of the present invention, specifically claims the following device is identifying asset vulnerability provided by the embodiment of the invention.
As shown in FIG. 2, the asset identification device comprises an obtaining unit 10, a transmitting unit 20 and a warning unit 30.
the obtaining unit 10 for obtaining the asset information of asset to be detected;
the sending unit 20 for sending the asset information to the cloud data center, so that the cloud data centre performing vulnerability recognition detection to the to-be-detected asset based on the asset information, and generating a vulnerability identification result, wherein the cloud data center stores a target threat intelligence data, the target threat intelligence data comprises at least one of the following: the security service provider upload of threat intelligence data, threat information data uploaded by the security information organization and the threat information data uploaded by the enterprise detection asset belongs to;
the warning unit 30 for obtaining the vulnerability identification result, and generating the pre-warning information based on the vulnerability identification result.
In the invention, the asset information of the asset to be detected to the obtained sent to the cloud data center to make the cloud data centre according to the target the stored threat intelligence data, assets to be detected of the comprehensive vulnerability detection. Because the cloud data centre according to the collected vulnerability multi-party provides the threat intelligence data asset to be detected for leak detection, the asset to be detected for overall vulnerability detection, so as to solve the technical problem of detecting of the asset vulnerability in the existing technology is single. so as to realize the asset to be detected to the technical effect of the comprehensive vulnerability detection.
Preferably, the obtaining unit is further used for sending the instruction to the local asset information base so that the local asset information database based on the obtaining instruction collecting attribute information and fingerprint information of the asset to be detected, wherein officer name of the asset to be detected belongs to said fingerprint information include, but are not limited to, the following information: the classification information of the to-be-detected asset factory manufacturer name, the to-be-detected asset, the asset to be detected of operating system information, obtaining the attribute information and the fingerprint information; the attribution information and the fingerprint information is determined as the asset information, wherein the classification information comprises: first level classification information and second level classification information, first level classification comprises the following types: server, a network device, a security device, a secondary classification comprises the following types: a firewall, a switch, a router and a database.
Preferably, if two-stage classification of the asset to be detected is a database, then the fingerprint information further comprises the type and version of the asset to be detected.
Preferably, the sending unit is further used for: the cloud service port of said asset information provided by the cloud data center is sent to the cloud data center, so that the cloud data centre based on preset leak detection program and the asset information for the asset to be detected for detecting vulnerability identification.
Third Embodiment
The invention further claims an asset vulnerability identification system, the device identification system for performing asset vulnerability of embodiment of the above content of the present invention, the following are

specifically described identification system of asset vulnerabilities according to an embodiment of the invention.
As shown in FIG. 3, the asset identification system comprising: a local asset information library 100, the cloud data center 200 and leak warning platform 300.
the local asset information database 100 for collecting and storing the to-be-detected asset information of asset;
the cloud data center 200 for performing vulnerability recognition detection based on said asset information for the asset to be detected, and generating a vulnerability identification result;
asset information the vulnerability warning platform 300 for obtaining the local asset information stored in the asset to be detected, and sending the asset information to the cloud data centre, and generating the pre-warning information based on the vulnerability identification result.
In the invention, the asset information of the asset to be detected to the obtained sent to the cloud data center to make the cloud data centre according to the target the stored threat intelligence data, assets to be detected of the comprehensive vulnerability detection. Because the cloud data centre according to the collected vulnerability multi-party provides the threat intelligence data asset to be detected for leak detection, the asset to be detected for overall vulnerability detection, so as to solve the technical problem of detecting of the asset vulnerability in the existing technology is single. so as to realize the asset to be detected to the technical effect of the comprehensive vulnerability detection.
Referring to FIG. 4, an embodiment of the invention further claims a server 400, including a processor 50, a memory 51, a bus 52, and a communication interface 53, the processor 50, communication interface 53, and memory 51 connected via bus 52, a processor 50 for executing module to execute, such as a computer program stored in the memory 51.
wherein, the memory 51 may include high speed random access memory (RAM, Random Access Memory), and may also include memory (non volatile memory) of non-stable -, such as at least one disk memory. via the at least one communication interface 53 (which may be wired or wireless) to realize the system network and the communication connection between the at least one other network element, it can use the internet, wide area network, local area network, metropolitan area network, and the like.
Bus 52 can be an ISA bus, PCI bus or an EISA bus. the bus can be divided into address bus, data bus, control bus, and the like. For ease of representation, FIG. 4 is only represented by a double arrow, but does not indicate that only one bus or one type of bus.
wherein, a memory 51 for storing a program, the processor 50 after receiving the execution instruction, executes the program, the method performed by the device of any embodiment of the invention embodiment claims a flow process definition of can be used for processor 50, or implemented by the processor 50.
Processor 50 may be an integrated circuit chip with a processing capability of the signal. finishing instruction of the integrated logic circuit in the realizing process, each step of the above method can by processor 50 in hardware or software form. the processor 50 may be a general processor, comprising a central processing unit (Central Processing Unit (CPU), network processor (Network Processor, abbreviated NP), but also can be digital signal processor (Digital SignalProcessing (DSP), an application specific integrated circuit (Application Specific Integrated Circuit, abbreviated as ASIC), programmable gate array (Field-Programmable Gate Array (FPGA) or other programmable logic device, discrete gate or transistor logic device, discrete hardware components. it can realize or perform each method disclosed in an embodiment of the invention, and the steps and logic diagram. a general purpose processor may be a microprocessor or the processor can be any conventional processor and so on. The embodiment of the invention claims a step of combining method may be directly embodied as hardware decoding processor execution is completed, or for combination in the decoding processor hardware and software module execution is completed. software module may be located in random access memory, flash memory, read only memory, programmable read only memory or an electrically erasable programmable memory, registers, and so the mature storage medium. the storage medium is located in memory 51, steps the processor 50 reads information in memory 51, combines its hardware to complete the method.
In addition, in the description of the embodiment of the invention, unless otherwise explicitly specified and defined, the terms "mounted", "connected", "connection" should be understood broadly, for example, may be a fixed connection, also can be detachably connected or integrally connected, can be mechanically connected, also can be electric connected can be directly connected, or may be indirectly connected by an

intermediate medium, can be connected in the two elements. For those of ordinary skill in the art, can understand the specific meaning of the term in the present invention the specific condition.
In the description of the invention, it should be noted that the term "central", "upper", "lower", "left", "right", "vertical", "horizontal", "inner", orientation or positional relationship of "outside") indication is based on orientation or positional relationship shown in the drawings is only to facilitate description of the present invention and to simplify the description, but not indicate or imply the apparatus or element must have a specific orientation in a specific orientation structure and operation, so it can not be understood as limiting the present invention. In addition, the terms "first", "second", "third" used for description purposes only, but are not intended to indicate or imply relative importance.
In the several embodiments provided in the present application, it should be understood that the system, apparatus and method disclosed, can be realized by other ways. apparatus embodiments described above are schematic only, for example, the division of the unit, is only a logical function division, the actual implementation can have additional dividing mode of, for example, multiple units or components can be combined or integrated into another system, or some features may be ignored or not performed. the other point, the displayed or discussed mutual coupling between or direct coupling or communication connection may be via some communication interface, indirect coupling or communication connection of the apparatus or unit can be electrically, mechanically or other form.
as the separation member unit can be or may not be physically separated, as unit display module may be or may not be physical units, that is, may be located in one position, or may be distributed to multiple network units. can be adjusted according to the actual need to select part or all of unit wherein to realize the purpose of the embodiment.
In addition, each function unit in each embodiment of the present invention can be integrated in a processing unit, or may be separate physical presence of each unit, also can be two or more than two unit are integrated in one unit.
at last, it should be noted that the above embodiment is only the specific embodiments of the invention, used to illustrate the technical solution of the invention, and not to its limit, the protection scope of the present invention is not limited to this. Although reference to the foregoing embodiment of the invention proceed the detailed explanation, one of ordinary skill in the art should understand that the technical solution in the technical scope any familiar with the technical field of technician in this invention claims, it still can be the preceding embodiment according to modifying or can easily think of changes. or wherein the part technical feature for equivalent replacement, and these modifications, changes or substitutions do not make the essence of the corresponding technical solutions departing from the spirit and scope of the technical solution of the embodiment of the invention, should be covered in the protection scope of the present invention. Therefore, the protection range of the invention should be compatible with the protection scope.

Claims

1. An asset vulnerability identification method, wherein, applied to vulnerability warning platform, comprising: obtaining the asset information of the detected asset, said asset information to the cloud data center. to make the cloud data centre for detecting vulnerability identification based on said asset information for the asset to be detected, and generating vulnerability identification result, wherein the cloud data center stores a target threat intelligence data, the target threat intelligence data comprises at least one of the following: the security service provider upload of threat intelligence data. threat intelligence data uploaded by the security information organization and the threat information data uploaded by the enterprise detection asset belongs to, obtaining the vulnerability identification result and generating the pre-warning information based on the vulnerability identification result.
2. The method according to claim 1, wherein obtaining the asset information of asset detection, comprising: sending the instruction to the local asset information base, to make the local asset information database based on the obtaining instruction collecting attribute information and fingerprint information of the asset to be detected, wherein the attribute information includes, but is not limited to the following information: name of the asset to be detected, the asset to be detected of the responsible person name belongs to said fingerprint information include, but are not limited to, the following information: factory name of the

manufacturer of the asset to be detected, classification information of the asset to be detected, operation system information of the asset to be detected, obtaining the attribute information and the fingerprint information; the attribution information and the fingerprint information is determined as the asset information, wherein the classification information comprises: first level classification information and second level classification information, first level classification comprises the following types: server, network equipment, safety equipment, the secondary classification comprises the following types: firewall, switch, router, database.

3. The method according to claim 2, wherein, if the two-stage classification of the asset to be detected is a database, then the fingerprint information further comprises the type and version of the asset to be detected.

4. The method according to claim 1, wherein the asset information to the cloud data center to make the cloud data centre based on said asset information for the asset to be detected for detecting vulnerability identification, comprising: the cloud service port of said asset information provided by the cloud data centre sending to the cloud data center, so that the cloud data centre inlet vulnerability identification detection based on preset leak detection program and the asset information for the asset to be detected.

5. The method according to claim 1, wherein the cloud data centre for providing infrastructure as a service and platform as a service, the cloud data centre stores the target threat intelligence data based on Hadoop framework.

6. An asset vulnerability identification device, wherein the device comprises an obtaining unit, sending unit and a warning unit, wherein the obtaining unit is used for obtaining the asset information of the detected asset; the sending unit is used for sending the asset information to the cloud data center, so that the cloud data centre performing vulnerability recognition detection to the to-be-detected asset based on the asset information, and generating a vulnerability identification result, wherein the cloud data center stores a target threat intelligence data. the target threat intelligence data threat data comprises at least one of the following: the security service provider upload of threat intelligence data, threat information data uploaded by the security information organization and the threat information data uploaded by the enterprise detection asset belongs to; the warning unit for obtaining the vulnerability identification result, and generating the pre-warning information based on the vulnerability identification result.

7. The device according to claim 6, wherein the obtaining unit is further used for sending the instruction to the local asset information base. to make the local asset information database collecting attribute information and fingerprint information of the asset to be detected based on the obtaining instruction, name of responsible person, the asset to be detected belongs, said fingerprint information include, but are not limited to, the following information: the classification information of the to-be-detected asset factory manufacturer name, the to-be-detected asset, the asset to be detected operating system information; obtaining the attribute information and the fingerprint information, the attribute information and the fingerprint information is determined as the asset information, wherein the classification information comprises: first level classification information and second level classification information, first level classification comprises the following types: server, a network device, a security device, a secondary classification comprises the following types: a firewall, a switch, a router and a database.

8. The device according to claim 7, wherein, if the two-stage classification of the asset to be detected is a database, then the fingerprint information further comprises the type and version of the asset to be detected.

9. The device according to claim 6, wherein said sending unit is further used for: the cloud service port of said asset information provided by the cloud data center is sent to the cloud data center, so that the cloud data centre based on preset leak detection program and the asset information for the asset to be detected for detecting vulnerability identification.

10. An asset vulnerability identification system, wherein the system comprises a local asset information database, a cloud data center and leak warning platform, wherein, the local asset information database for collecting and storing the to-be-detected asset information of asset; the cloud data centre for performing vulnerability recognition detection on the to-be-detected asset based on the asset information, and generating a vulnerability identification result, the vulnerability warning platform for asset information for obtaining the local asset information stored in the asset to be detected, and sending said asset information to the cloud data centre, and generating the pre-warning information based on the vulnerability identification result.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 110708315 A
(43)申请公布日 2020.01.17

(21)申请号 201910956125.4

(22)申请日 2019.10.09

(71)申请人 杭州安恒信息技术股份有限公司
地址 310000 浙江省杭州市滨江区西兴街道联慧街188号

(72)发明人 王英新 范渊

(74)专利代理机构 北京超凡宏宇专利代理事务所(特殊普通合伙) 11463
代理人 徐彦圣

(51)Int.Cl.
H04L 29/06(2006.01)

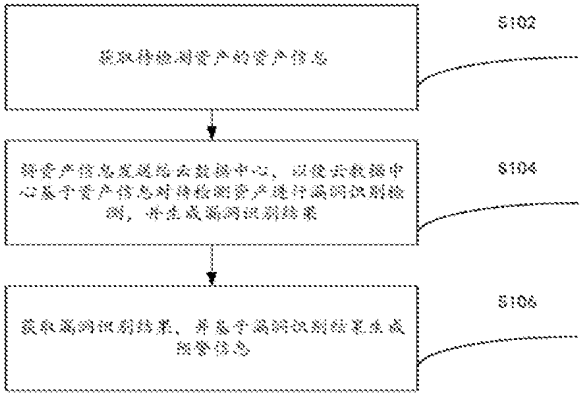
权利要求书2页 说明书8页 附图2页

(54)发明名称

资产漏洞的识别方法、装置和系统

(57)摘要

本发明提供了一种资产漏洞的识别方法、装置和系统，涉及网络安全的技术领域，包括：获取待检测资产的资产信息；将资产信息发送给云数据中心，以使云数据中心基于资产信息对待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，云数据中心存储有目标威胁情报数据，目标威胁情报数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据；获取漏洞识别结果，并基于漏洞识别结果生成预警信息，解决了现有的资产漏洞识别方法对资产漏洞的检测较为单一的技术问题。



CN 110708315 A

1.一种资产漏洞的识别方法，其特征在于，应用于漏洞预警平台，包括：

获取待检测资产的资产信息；

将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，所述云数据中心存储有目标威胁情报数据，所述目标威胁情报数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据；

获取所述漏洞识别结果，并基于所述漏洞识别结果生成预警信息。

2.根据权利要求1所述的方法，其特征在于，获取待检测资产的资产信息，包括：

向本地资产信息库发送获取指令，以使所述本地资产信息库基于所述获取指令采集所述待检测资产的属性信息和指纹信息，其中，所述属性信息包括但不限于以下信息：所述待检测资产的名称，所述待检测资产的所属责任人名称，所述指纹信息包括但不限于以下信息：所述待检测资产的出厂厂商名称，所述待检测资产的分类信息，所述待检测资产的操作系统信息；

获取所述属性信息和所述指纹信息，将所述属性信息和所述指纹信息确定为所述资产信息；

其中，所述分类信息包括：一级分类信息和二级分类信息，一级分类包括以下种类：服务器，网络设备，安全设备，二级分类包括以下种类：防火墙，交换机，路由器，数据库。

3.根据权利要求2所述的方法，其特征在于，若所述待检测资产的二级分类为数据库，则所述指纹信息还包括所述待检测资产的类型和版本。

4.根据权利要求1所述的方法，其特征在于，将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，包括：

将所述资产信息通过所述云数据中心提供的云服务端口发送给所述云数据中心，以使所述云数据中心基于预设漏洞检测程序和所述资产信息对所述待检测资产进漏洞识别检测。

5.根据权利要求1所述的方法，其特征在于，

所述云数据中心用于提供基础设施即服务和平台即服务；

所述云数据中心基于Hadoop框架存储所述目标威胁情报数据。

6.一种资产漏洞的识别装置，其特征在于，所述装置包括：获取单元，发送单元和预警单元，其中，

所述获取单元用于获取待检测资产的资产信息；

所述发送单元用于将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，所述云数据中心存储有目标威胁情报数据，所述目标威胁情报数据威胁数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据；

所述预警单元用于获取所述漏洞识别结果，并基于所述漏洞识别结果生成预警信息。

7.根据权利要求6所述的装置，其特征在于，所述获取单元还用于：

向本地资产信息库发送获取指令，以使所述本地资产信息库基于所述获取指令采集所述待检测资产的属性信息和指纹信息，其中，所述待检测资产的所属责任人名称，所述指纹

信息包括但不限于以下信息：所述待检测资产的出厂厂商名称，所述待检测资产的分类信息，所述待检测资产的操作系统信息；

获取所述属性信息和所述指纹信息，将所述属性信息和所述指纹信息确定为所述资产信息；

其中，所述分类信息包括：一级分类信息和二级分类信息，一级分类包括以下种类：服务器，网络设备，安全设备，二级分类包括以下种类：防火墙，交换机，路由器，数据库。

8.根据权利要求7所述的装置，其特征在于，若所述待检测资产的二级分类为数据库，则所述指纹信息还包括所述待检测资产的类型和版本。

9.根据权利要求6所述的装置，其特征在于，所述发送单元还用于：

将所述资产信息通过所述云数据中心提供的云服务端口发送给所述云数据中心，以使所述云数据中心基于预设漏洞检测程序和所述资产信息对所述待检测资产进行漏洞识别检测。

10.一种资产漏洞的识别系统，其特征在于，所述系统包括：本地资产信息库，云数据中心和漏洞预警平台，其中，

所述本地资产信息库用于采集并存储待检测资产的资产信息；

所述云数据中心用于基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果；

所述漏洞预警平台用于获取所述本地资产信息库中存储的待检测资产的资产信息，并将所述资产信息发送给所述云数据中心，以及基于所述漏洞识别结果生成预警信息。

资产漏洞的识别方法、装置和系统

技术领域

[0001] 本发明涉及网络安全技术领域，尤其是涉及一种资产漏洞的识别方法、装置和系统。

背景技术

[0002] 在当前网络安全技术条件下，网络攻击的发生基本是网络威胁利用信息资产所暴露的漏洞而实施的恶意行为。在网络空间中，网络威胁是客观存在的，要降低网络攻击的风险，对信息资产进行漏洞加固是非常有效的安全技术措施。但在进行资产漏洞加固之前，需要知道信息资产所面临的漏洞有哪些，因此，信息资产漏洞检测对于网络安全防护来说是异常重要的。

[0003] 但是，现有的漏洞检测方法存在对资产漏洞检测较为单一的技术问题。

[0004] 针对上述问题，还未提出有效的解决方案。

发明内容

[0005] 有鉴于此，本发明的目的在于提供一种资产漏洞的识别方法、装置和系统，以解决了现有的资产漏洞识别方法对资产漏洞的检测较为单一的技术问题。

[0006] 第一方面，本发明实施例提供了一种资产漏洞的识别方法，包括：获取待检测资产的资产信息；将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，所述云数据中心存储有目标威胁情报数据，所述目标威胁情报数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据；获取所述漏洞识别结果，并基于所述漏洞识别结果生成预警信息。

[0007] 进一步地，获取待检测资产的资产信息，包括：向本地资产信息库发送获取指令，以使所述本地资产信息库基于所述获取指令采集所述待检测资产的属性信息和指纹信息，其中，所述待检测资产的所属责任人名称，所述指纹信息包括但不限于以下信息：所述待检测资产的出厂厂商名称，所述待检测资产的分类信息，所述待检测资产的操作系统信息；获取所述属性信息和所述指纹信息，将所述属性信息和所述指纹信息确定为所述资产信息；其中，所述分类信息包括：一级分类信息和二级分类信息，一级分类包括以下种类：服务器，网络设备，安全设备，二级分类包括以下种类：防火墙，交换机，路由器，数据库。

[0008] 进一步地，若所述待检测资产的二级分类为数据库，则所述指纹信息还包括所述待检测资产的类型和版本。

[0009] 进一步地，将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，包括：将所述资产信息通过所述云数据中心提供的云服务端口发送给所述云数据中心，以使所述云数据中心基于预设漏洞检测程序和所述资产信息对所述待检测资产进行漏洞识别检测。

[0010] 进一步地，所述云数据中心用于提供基础设施即服务和平台即服务；所述云数据

中心基于Hadoop框架存储所述目标威胁情报数据。

[0011] 第二方面，本发明实施例提供了一种资产漏洞的识别装置，包括：获取单元，发送单元和预警单元，其中，所述获取单元用于获取待检测资产的资产信息；所述发送单元用于将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，所述云数据中心存储有目标威胁情报数据，所述目标威胁情报数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据；所述预警单元用于获取所述漏洞识别结果，并基于所述漏洞识别结果生成预警信息。

[0012] 进一步地，所述获取单元还用于：向本地资产信息库发送获取指令，以使所述本地资产信息库基于所述获取指令采集所述待检测资产的属性信息和指纹信息，其中，所述待检测资产的所属责任人名称，所述指纹信息包括但不限于以下信息：所述待检测资产的出厂厂商名称，所述待检测资产的分类信息，所述待检测资产的操作系统信息；获取所述属性信息和所述指纹信息，将所述属性信息和所述指纹信息确定为所述资产信息；其中，所述分类信息包括：一级分类信息和二级分类信息，一级分类包括以下种类：服务器，网络设备，安全设备，二级分类包括以下种类：防火墙，交换机，路由器，数据库。

[0013] 进一步地，若所述待检测资产的二级分类为数据库，则所述指纹信息还包括所述待检测资产的类型和版本。

[0014] 进一步地，所述发送单元还用于：将所述资产信息通过所述云数据中心提供的云服务端口发送给所述云数据中心，以使所述云数据中心基于预设漏洞检测程序和所述资产信息对所述待检测资产进行漏洞识别检测。

[0015] 第三方面，本发明实施例提供了一种资产漏洞的识别系统，包括：产信息采集设备，云数据中心和漏洞预警平台，其中，所述本地资产信息库用于采集待检测资产的资产信息；所述云数据中心用于基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果；所述漏洞预警平台用于基于所述漏洞识别结果生成预警信息。

[0016] 在本发明实施例中，首先，获取待检测资产的资产信息；将资产信息发送给云数据中心，以使云数据中心基于资产信息对待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，云数据中心存储有目标威胁情报数据，目标威胁情报数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据，漏洞识别结果用于表征待检测资产是否存在漏洞；获取漏洞识别结果，并基于漏洞识别结果生成预警信息。

[0017] 本申请中，通过将获取到的待检测资产的资产信息发送给云数据中心，以使云数据中心能够根据其存储的目标威胁情报数据，对待检测资产进行全面的漏洞检测，由于云数据中心能够根据汇集了多方提供的漏洞威胁情报数据对待检测资产进行漏洞检测，达到了对待检测资产进行全面漏洞检测的目的，进而解决了现有技术中资产漏洞的检测较为单一的技术问题，从而实现了对待检测资产进行全面漏洞检测的技术效果。

[0018] 本发明的其他特征和优点将在随后的说明书中阐述，并且，部分地从说明书中变得显而易见，或者通过实施本发明而了解。本发明的目的和其他优点在说明书、权利要求书以及附图中所特别指出的结构来实现和获得。

[0019] 为使本发明的上述目的、特征和优点能更明显易懂，下文特举较佳实施例，并配合

0489

所附附图，作详细说明如下。

附图说明

[0020] 为了更清楚地说明本发明具体实施方式或现有技术中的技术方案，下面将对具体实施方式或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图是本发明的一些实施方式，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

[0021] 图1为本发明实施例提供的一种资产漏洞的识别方法的流程图；

[0022] 图2为本发明实施例提供的一种资产漏洞的识别装置的示意图；

[0023] 图3为本发明实施例提供的一种资产漏洞的识别系统的示意图；

[0024] 图4为本发明实施例提供的一种服务器的示意图。

具体实施方式

[0025] 为使本发明实施例的目的、技术方案和优点更加清楚，下面将结合附图对本发明的技术方案进行清楚、完整地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0026] 关于漏洞检测技术。当前针对信息资产的漏洞检测技术主要依靠专门的漏洞扫描设备通过主动探测技术来实现，在这种方式下，漏洞扫描设备利用其携带的“漏洞规则库”对各类信息资产进行主动扫描，如在对某一个资产进行扫描时，若其扫描结果命中了“漏洞规则库”中的规则R，则判定该资产具有R漏洞规则所对应的漏洞。在这种情况下，漏洞扫描的结果取决于漏洞扫描设备所携带的“漏洞规则库”的数量和质量。若某个信息资产所暴露出的某个漏洞在“漏洞规则库”中，则此漏洞可被漏洞扫描设备检测出来。若某个漏洞不在“漏洞规则库”中，则漏洞扫描设备无法检测出该漏洞，即便这个漏洞真实的存在。

[0027] 关于“漏洞规则库”的更新。对于新发现的漏洞，漏洞扫描设备提供商一般会定期或不定期的将这些漏洞打包成软件升级包，漏洞扫描设备通过在线或离线的升级完成“漏洞规则库”的升级，升级完成之后的漏洞扫描设备方可具备对新发现漏洞的检测能力。

[0028] 现有资产漏洞的识别方法存在以下的缺点：

[0029] 1、基于主动扫描的漏洞扫描技术需要通过网络向被扫描目标发送数据包，由于现实环境的复杂性，这可能造成目标主机的运行不稳定，造成信息资产的宕机、停止服务、数据丢失等。

[0030] 2、漏洞检测的滞后性。对于0day漏洞及未公开的漏洞，漏洞扫描设备由于“漏洞规则库”升级的滞后性，可能导致某些漏洞无法及时检测出来，进而存在恶意威胁利用该漏洞进行攻击的风险。

[0031] 3、检测能力单一。一般来说，某一厂商的漏洞检测设备都有其固有的技术及漏洞规则库的限制，无法发挥多个安全厂商及各类安全组织的漏洞检测能力，无法实现漏洞检测能力的聚合。

[0032] 4、漏洞检测的效率较低。由于传统的漏洞扫描设备技术实现方式原因，对多台资产进行漏洞检测时，效率较低，通常完成一台资产的漏洞检测需要数分钟。

[0033] 针对上述缺点，本申请采用以下方法克服上述缺点：

[0034] 实施例一：

[0035] 根据本发明实施例，提供了一种资产漏洞的识别方法实施例，需要说明的是，在附图的流程图示出的步骤可以在诸如一组计算机可执行指令的计算机系统中执行，并且，虽然在流程图中示出了逻辑顺序，但是在某些情况下，可以以不同于此处的顺序执行所示出或描述的步骤。

[0036] 图1是根据本发明实施例的一种资产漏洞的识别方法，如图1所示，该方法包括如下步骤：

[0037] 步骤S102，获取待检测资产的资产信息；

[0038] 步骤S104，将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，所述云数据中心存储有目标威胁情报数据，所述目标威胁情报数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据；

[0039] 步骤S106，获取所述漏洞识别结果，并基于所述漏洞识别结果生成预警信息。

[0040] 本申请中，通过将获取到的待检测资产的资产信息发送给云数据中心，以使云数据中心能够根据其存储的目标威胁情报数据，对待检测资产进行全面的漏洞检测，由于云数据中心能够根据汇集了多方提供的漏洞威胁情报数据对待检测资产进行漏洞检测，达到了对待检测资产进行全面漏洞检测的目的，进而解决了现有技术中资产漏洞的检测较为单一的技术问题，从而实现了对待检测资产进行全面漏洞检测的技术效果。

[0041] 需要说明的是，通过建立由安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上传的威胁情报数据组成的云数据中心，可以整合和对接各方安全威胁情报数据，形成威胁情报的数据协同和多方安全情报的能力协同，有利于安全威胁情报的共享机制建设。

[0042] 另外，还需要说明的是，上述的漏洞预警信息可以通过短信，邮件等方式发送给工作人员，以使工作人员能够及时对待检测资产的漏洞进行处理，从而降低待检测资产面临的风险。

[0043] 在本发明实施例中，步骤S102还包括如下步骤：

[0044] 步骤S11，向本地资产信息库发送获取指令，以使所述本地资产信息库基于所述获取指令采集所述待检测资产的属性信息和指纹信息，其中，所述待检测资产的所属责任人名称，所述指纹信息包括但不限于以下信息：所述待检测资产的出厂厂商名称，所述待检测资产的分类信息，所述待检测资产的操作系统信息；

[0045] 步骤S11，获取所述属性信息和所述指纹信息，将所述属性信息和所述指纹信息确定为所述资产信息。

[0046] 在本发明实施例中，在向本地资产信息库发送获取指令之后，本地资产信息库能够根据获取指令采集待检测资产的属性信息和指纹信息。

[0047] 然后，将获取本地资产信息库采集到的待检测资产的属性信息和指纹信息作为待检测资产的资产信息。

[0048] 需要说明的是，上述的属性信息包括但不限于以下信息：待检测资产的名称，待检

测资产的所属责任人名称等信息。

[0049] 上述的指纹信息包括但不限于以下信息：待检测资产的出厂厂商名称，待检测资产的分类信息，待检测资产的操作系统信息，其中，分类信息包括：一级分类信息和二级分类信息，一级分类包括以下种类：服务器，网络设备，安全设备，二级分类包括以下种类：防火墙，交换机，路由器，数据库。

[0050] 另外，还需要说明的是，待检测资产给的操作系统信息包括：操作系统的类型，操作系统的版本等信息。

[0051] 如果，待检测资产为数据库，那么待检测资产的指纹信息还包括代金侧资产的版本和类型。

[0052] 由于待检测资产的资产信息中包含指纹信息，因此，每个指纹信息能够确定一个待检测资产，进而可以通过资产指纹匹配技术准确的实现对待检测资产进行准确的被动的漏洞检测，降低漏洞检测时对被待检测资产的运行影响。

[0053] 另外，本地资产信息库在对待检测资产的资产信息进行采集可以通过自动方式或手动录入进行，资产信息的自动采集主要借助资产探测类设备完成，手动录入时，本地资产信息库可提供可视化的录入界面，以供工作人员将资产信息手动录入本地资产信息库中。

[0054] 在本发明实施例中，步骤S104还包括如下步骤：

[0055] 步骤S21，将所述资产信息通过所述云数据中心提供的云服务端口发送给所述云数据中心，以使所述云数据中心基于预设漏洞检测程序和所述资产信息对所述待检测资产进漏洞识别检测。

[0056] 在本发明实施例中，由于云数据中心提供的IaaS(基础设施即服务)及PaaS(平台即服务)作为基础层，利用开源的Hadoop架构搭建基于大数据处理架构的威胁情报数据库，威胁情报数据库中的数据来源可以是来自各个安全服务提供商、安全情报组织、企业自身。在PaaS(平台即服务)之上构建漏洞检测云应用程序(即，预设漏洞检测程序)，本程序可调用威胁情报数据库中的数据，以便进行对待检测资产进行漏洞识别检测。同时，云应程序用通过互联网提供云服务接口，实现漏洞检测接口的服务化。

[0057] 由于云数据中心能够通过实时更新的威胁情报云数据作为基础数据，以云端接口的方式提供漏洞检测服务，可实时的为资产的0day漏洞检测和位置漏洞检测提供快速及时的检测结果，解决由于漏洞规则库滞后性所带来的风险，同时云数据中心以云端接口方式提供的漏洞检测服务能够批量对资产进行漏洞检测，提高了资产漏洞的检测效率。

[0058] 实施例二：

[0059] 本发明还提供了一种资产漏洞的识别装置，该装置用于执行本发明实施例上述内容所提供的资产漏洞的识别方法，以下是本发明实施例提供的资产漏洞的识别装置的具体介绍。

[0060] 如图2所示，上述的资产识别装置包括：获取单元10，发送单元20和预警单元30。

[0061] 所述获取单元10用于获取待检测资产的资产信息；

[0062] 所述发送单元20用于将所述资产信息发送给云数据中心，以使所述云数据中心基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果，其中，所述云数据中心存储有目标威胁情报数据，所述目标威胁情报数据包括以下至少之一：安全服务提供商上传的威胁情报数据，安全情报组织上传的威胁情报数据和待检测资产所属企业上

0492

传的威胁情报数据；

[0063] 所述预警单元30用于获取所述漏洞识别结果，并基于所述漏洞识别结果生成预警信息。

[0064] 本申请中，通过将获取到的待检测资产的资产信息发送给云数据中心，以使云数据中心能够根据其存储的目标威胁情报数据，对待检测资产进行全面的漏洞检测，由于云数据中心能够根据汇集了多方提供的漏洞威胁情报数据对待检测资产进行漏洞检测，达到了对待检测资产进行全面漏洞检测的目的，进而解决了现有技术中资产漏洞的检测较为单一的技术问题，从而实现了对待检测资产进行全面漏洞检测的技术效果。

[0065] 优选地，所述获取单元还用于：向本地资产信息库发送获取指令，以使所述本地资产信息库基于所述获取指令采集所述待检测资产的属性信息和指纹信息，其中，所述待检测资产的所属责任人名称，所述指纹信息包括但不限于以下信息：所述待检测资产的出厂厂商名称，所述待检测资产的分类信息，所述待检测资产的操作系统信息；获取所述属性信息和所述指纹信息，将所述属性信息和所述指纹信息确定为所述资产信息；其中，所述分类信息包括：一级分类信息和二级分类信息，一级分类包括以下种类：服务器，网络设备，安全设备，二级分类包括以下种类：防火墙，交换机，路由器，数据库。

[0066] 优选地，若所述待检测资产的二级分类为数据库，则所述指纹信息还包括所述待检测资产的类型和版本。

[0067] 优选地，所述发送单元还用于：将所述资产信息通过所述云数据中心提供的云服务端口发送给所述云数据中心，以使所述云数据中心基于预设漏洞检测程序和所述资产信息对所述待检测资产进行漏洞识别检测。

[0068] 实施例三：

[0069] 本发明还提供了一种资产漏洞的识别系统，该装置用于执行本发明实施例上述内容所提供的资产漏洞的识别系统，以下是本发明实施例提供的资产漏洞的识别系统的具体介绍。

[0070] 如图3所示，上述的资产识别系统包括：本地资产信息库100，云数据中心200和漏洞预警平台300。

[0071] 所述本地资产信息库100用于采集并存储待检测资产的资产信息；

[0072] 所述云数据中心200用于基于所述资产信息对所述待检测资产进行漏洞识别检测，并生成漏洞识别结果；

[0073] 所述漏洞预警平台300用于获取所述本地资产信息库中存储的待检测资产的资产信息，并将所述资产信息发送给所述云数据中心，以及基于所述漏洞识别结果生成预警信息。

[0074] 本申请中，通过将获取到的待检测资产的资产信息发送给云数据中心，以使云数据中心能够根据其存储的目标威胁情报数据，对待检测资产进行全面的漏洞检测，由于云数据中心能够根据汇集了多方提供的漏洞威胁情报数据对待检测资产进行漏洞检测，达到了对待检测资产进行全面漏洞检测的目的，进而解决了现有技术中资产漏洞的检测较为单一的技术问题，从而实现了对待检测资产进行全面漏洞检测的技术效果。

[0075] 参见图4，本发明实施例还提供一种服务器400，包括：处理器50，存储器51，总线52和通信接口53，所述处理器50、通信接口53和存储器51通过总线52连接；处理器50用于执行

存储器51中存储的可执行模块，例如计算机程序。

[0076] 其中，存储器51可能包含高速随机存取存储器(RAM，Random Access Memory)，也可能还包括非不稳定的存储器(non-volatile memory)，例如至少一个磁盘存储器。通过至少一个通信接口53(可以是有线或者无线)实现该系统网元与至少一个其他网元之间的通信连接，可以使用互联网，广域网，本地网，城域网等。

[0077] 总线52可以是ISA总线、PCI总线或EISA总线等。所述总线可以分为地址总线、数据总线、控制总线等。为便于表示，图4中仅用一个双向箭头表示，但并不表示仅有一根总线或一种类型的总线。

[0078] 其中，存储器51用于存储程序，所述处理器50在接收到执行指令后，执行所述程序，前述本发明实施例任一实施例揭示的流过程定义的装置所执行的方法可以应用于处理器50中，或者由处理器50实现。

[0079] 处理器50可能是一种集成电路芯片，具有信号的处理能力。在实现过程中，上述方法的各步骤可以通过处理器50中的硬件的集成逻辑电路或者软件形式的指令完成。上述的处理器50可以是通用处理器，包括中央处理器(Central Processing Unit，简称CPU)、网络处理器(Network Processor，简称NP)等；还可以是数字信号处理器(Digital Signal Processing，简称DSP)、专用集成电路(Application Specific Integrated Circuit，简称ASIC)、现成可编程门阵列(Field-Programmable Gate Array，简称FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件。可以实现或者执行本发明实施例中的公开的各方法、步骤及逻辑框图。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。结合本发明实施例所公开的方法的步骤可以直接体现为硬件译码处理器执行完成，或者用译码处理器中的硬件及软件模块组合执行完成。软件模块可以位于随机存储器，闪存、只读存储器，可编程只读存储器或者电可擦写可编程存储器、寄存器等本领域成熟的存储介质中。该存储介质位于存储器51，处理器50读取存储器51中的信息，结合其硬件完成上述方法的步骤。

[0080] 另外，在本发明实施例的描述中，除非另有明确的规定和限定，术语“安装”、“相连”、“连接”应做广义理解，例如，可以是固定连接，也可以是可拆卸连接，或一体地连接；可以是机械连接，也可以是电连接；可以是直接相连，也可以通过中间媒介间接相连，可以是两个元件内部的连通。对于本领域的普通技术人员而言，可以具体情况理解上述术语在本发明中的具体含义。

[0081] 在本发明的描述中，需要说明的是，术语“中心”、“上”、“下”、“左”、“右”、“竖直”、“水平”、“内”、“外”等指示的方位或位置关系为基于附图所示的方位或位置关系，仅是为了便于描述本发明和简化描述，而不是指示或暗示所指的装置或元件必须具有特定的方位、以特定的方位构造和操作，因此不能理解为对本发明的限制。此外，术语“第一”、“第二”、“第三”仅用于描述目的，而不能理解为指示或暗示相对重要性。

[0082] 在本申请所提供的几个实施例中，应该理解到，所揭露的系统、装置和方法，可以通过其它的方式实现。以上所描述的装置实施例仅仅是示意性的，例如，所述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，又例如，多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些通信接口，装置或单元的间接

0494

耦合或通信连接，可以是电性，机械或其它的形式。

[0083] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0084] 另外，在本发明各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。

[0085] 最后应说明的是：以上所述实施例，仅为本发明的具体实施方式，用以说明本发明的技术方案，而非对其限制，本发明的保护范围并不局限于此，尽管参照前述实施例对本发明进行了详细的说明，本领域的普通技术人员应当理解：任何熟悉本技术领域的技术人员在本发明揭露的技术范围内，其依然可以对前述实施例所记载的技术方案进行修改或可轻易想到变化，或者对其中部分技术特征进行等同替换；而这些修改、变化或者替换，并不使相应技术方案的本质脱离本发明实施例技术方案的精神和范围，都应涵盖在本发明的保护范围之内。因此，本发明的保护范围应所述以权利要求的保护范围为准。

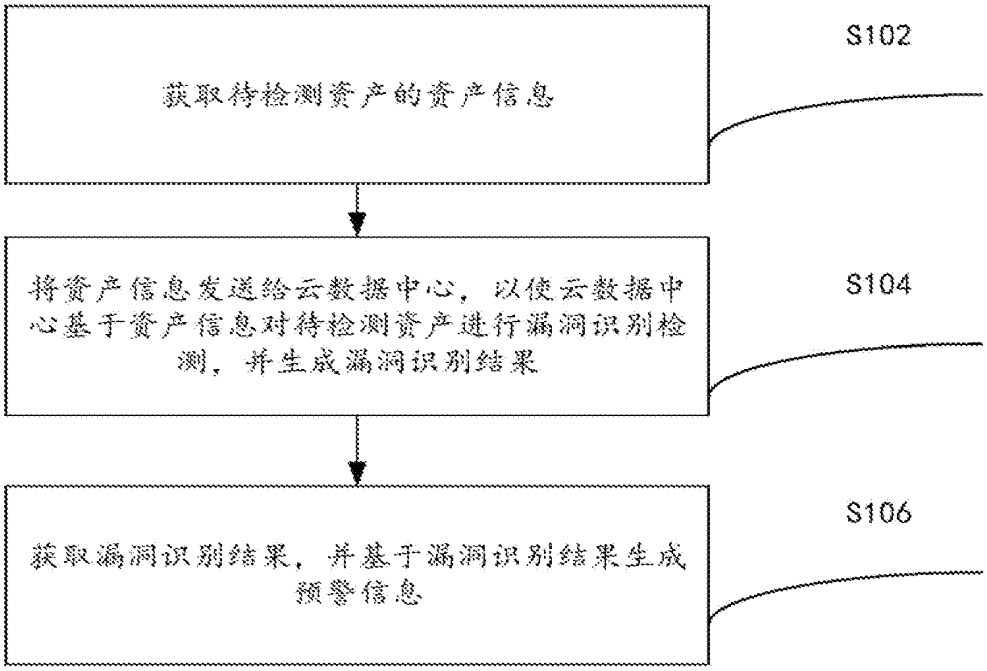

获取待检测资产的资产信息
S102
将资产信息发送给云数据中心，以使云数据中心基于资产信息对待检测资产进行漏洞识别检测，并生成漏洞识别结果
S104
获取漏洞识别结果，并基于漏洞识别结果生成预警信息
S106

图1

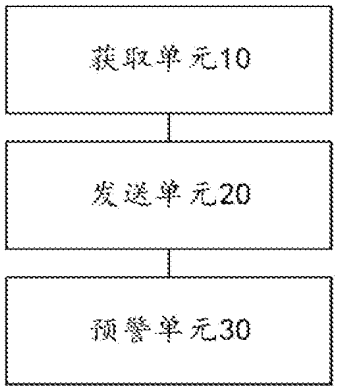

获取单元10
发送单元20
预警单元30

图2

0496

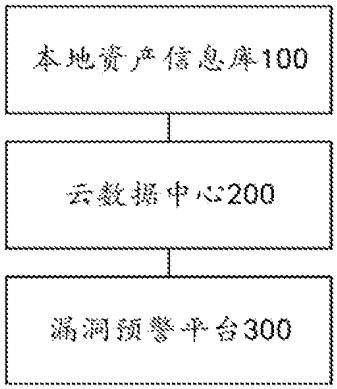

本地资产信息库100
云数据中心200
漏洞预警平台300

图3

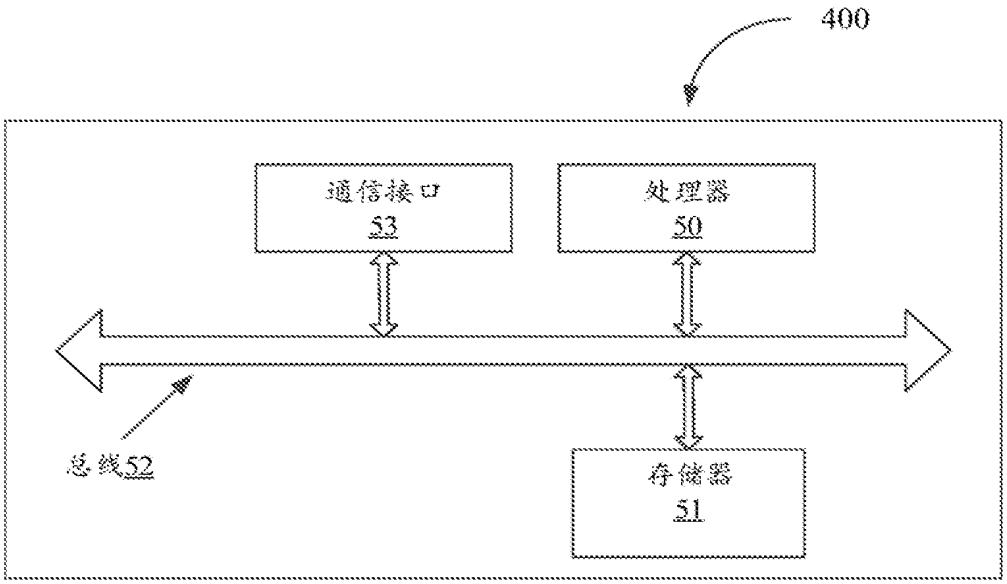

400
通信接口
53
处理器
50
总线52
存储器
51

图4

SECURE STARTING METHOD AND APPARATUS FOR CLOUD HOST, AND COMPUTER DEVICE AND STORAGE MEDIUM

DOCUMENT ID
WO-2021072877-A1

DATE PUBLISHED
2021-04-22

INVENTOR INFORMATION

NAME	CITY	STATE	ZIP CODE	COUNTRY
SHEN, YONG	N/A	N/A	518000	CN

DATE FILED
2019-11-14

Abstract

Disclosed are a secure starting method and apparatus for a cloud host, and a computer device and a storage medium. The method comprises: when a starting instruction, triggered by a user, for a specified cloud host is received, acquiring, according to a preset rule, asset information corresponding to the specified cloud host; determining whether there are vulnerabilities, corresponding to the asset information of the specified cloud host, stored in a preset vulnerability database; if so, acquiring the last shutdown time of the specified cloud host and release times respectively corresponding to the vulnerabilities; according to the last shutdown time and the release times, determining whether there is a specified vulnerability, of which the release time is within a specified time period, in all the vulnerabilities; and if so, starting the specified cloud host in a pre-created secure isolation area. According to the preset application, the specified cloud host can be effectively prevented from being invaded or attacked by vulnerabilities during use, and the security of the specified cloud host after starting is guaranteed.

Description

INVENTION-TITLE
This application claims the priority of a Chinese patent application filed with the Chinese Patent Office on October 15, 2019, the application number is 2019109790501, and the invention title is "the secure boot method, device, computer equipment and storage medium of the cloud host", and its entire content Incorporated in this application by reference.
TECHNICAL-FIELD
Technical field
This application relates to the technical field of cloud computing, and in particular to a secure startup method, device, computer equipment, and storage medium of a cloud host.
BACKGROUND-ART
Background technique
Cloud computing is currently an important field and direction in the development of IT technology, and it is gradually infiltrating all areas of society. Cloud host is an important part of cloud computing in infrastructure applications. It is located at the bottom of the cloud computing industry chain and its products are derived from cloud computing platforms. The platform integrates the three core elements of Internet applications: computing, storage, and network, and provides users with public Internet infrastructure services. Cloud

hosting is a kind of similar to VPS (Virtual Private Server, virtual private server) the virtualization technology of the host, VPS uses virtual software (such as VZ or VM) to virtualize multiple parts similar to independent hosts on a host, which can realize single-machine multi-user, and each part can be used. As a separate operating system, the management method is the same as that of the host. The cloud host virtualizes multiple parts similar to independent hosts on a group of host machine clusters. Each host machine in the cluster has an image of the cloud host machine, which greatly improves the security and stability of the virtual host machine. In addition, in order to facilitate users to access cloud computing anytime and anywhere, cloud hosts can usually be accessed directly from the public network (Internet), and even managed.

In the existing various cloud computing systems, each cloud host will face the security risk of the protection gap. Specifically, when the cloud host is in the shutdown state, since the cloud host will repair all currently known security vulnerabilities when it is shut down, it can be considered that the cloud host in the shutdown state is also in a safe state. However, when the cloud host is restarted and exposed to the public network, the external environment at this time may change compared to before the shutdown. If there are new security vulnerabilities related to the cloud host, the cloud host may be attacked by attackers on the public network at any time, and the security of the cloud host is greatly threatened.

BEST-MODE

The best mode of the present invention

It should be understood that the specific embodiments described here are only used to explain the present application, and are not used to limit the present application.

Referring to Fig. 1, a secure boot method of a cloud host according to an embodiment of the present application includes:

S1: Acquire asset information corresponding to the designated cloud host according to preset rules when receiving a startup instruction for a designated cloud host triggered by a user, where the asset information includes at least operating system information;

S2: Determine whether there are vulnerabilities corresponding to the asset information of the designated cloud host stored in the preset vulnerability database, where the number of the vulnerabilities is one or more;

S3: If yes, obtain the last shutdown time of the designated cloud host and the release time corresponding to each of the vulnerabilities;

S4: According to the last shutdown time and each of the release times, determine whether there are specified vulnerabilities whose release time is within a specified period of time among all the vulnerabilities;

S5: If yes, start the designated cloud host, and when the designated cloud host is started, place the designated cloud host in a preset security isolation area to suspend the network access function of the designated cloud host.

As described in the above steps S1 to S5, the execution subject of the embodiment of the present invention is a secure boot device for a cloud host, specifically a host cluster used to virtual out the cloud host, through which the host cluster can realize the For the secure startup of the host, the above-mentioned designated cloud host is any cloud host among all the cloud hosts virtualized by the host cluster. Specifically, it is first judged whether a user-triggered start instruction for the designated cloud host is received; when the start instruction is received, the asset information corresponding to the designated cloud host is obtained according to preset rules, wherein the asset information includes at least the operating system Information, for example, the operating system is windowsserver2012, and the RDP service version 11.0 is used to provide a remote desktop. In addition, the asset information may also include cloud host service information and key application information. In addition, the foregoing preset rules may include: obtaining asset information corresponding to the specified cloud host by querying the cloud platform corresponding to the specified cloud host, or by invoking the specified scanning tool. Then, according to the above asset information, it is determined whether there are vulnerabilities corresponding to the asset information of the above designated cloud host stored in the preset vulnerability database, wherein the number of the above vulnerabilities is one or more, and the above vulnerability database stores passing The crawler tool crawls the vulnerability information related to various operating systems corresponding to the cloud host and the vulnerability information related to the cloud host service from the target website in real time. If the vulnerability database stores the vulnerability corresponding to the asset information of the specified cloud host, the last shutdown time of the specified cloud host and the release time corresponding to each of the

aforementioned vulnerabilities are obtained. Among them, for example, the form of any vulnerability information crawled can be CVE20190004-rdp vulnerability, released on July 02, 2019. The affected system is all versions of the rdp service before version 12.0, and the release time of the corresponding vulnerability is 2019 /07/02. Then, based on the last shutdown time and each of the above release times, it is determined whether there are specified vulnerabilities in all the vulnerabilities whose release time is within the specified time period, where the above specified time period is included between the above shutdown time and the above current time period. If there are specified vulnerabilities whose release time is within the specified time period, the above-mentioned designated cloud host will be started in the pre-created security quarantine area, and the network access function of the designated cloud host will be disabled in the above-mentioned security quarantine area. In this embodiment, when a new vulnerability related to the designated cloud host occurs during the last shutdown period, the designated cloud host will be intelligently placed in the pre-created security isolation zone for startup, thereby starting on the designated cloud host Then temporarily disable the network access function of the designated cloud host, effectively avoiding the intrusion or attack of the designated cloud host by vulnerabilities during use, and ensuring the security of the designated cloud host after it is started. Furthermore, if there are no specified vulnerabilities whose release time is within the specified period of time among all vulnerabilities, indicating that no related vulnerabilities appeared during the last shutdown of the specified cloud host, the startup of the specified cloud host will be completed directly , And there is no need to place the designated cloud host in a security quarantine area. In order to effectively guarantee the user experience of the designated cloud host.

Further, in an embodiment of the present application, the above step S1 includes:

S100: Determine whether an asset management tool is installed in the designated cloud host, where the asset management tool is used to periodically synchronize asset information corresponding to the designated cloud host to the designated cloud platform;

S101: If an asset management tool is installed, search for the asset information from the designated cloud platform;

S102: If an asset management tool is not installed, call a designated scanning tool to perform scanning processing on the designated cloud host to obtain the asset information.

As described in the above steps S100 to S102, the asset information may specifically include the operating system information corresponding to the specified cloud host, as well as the cloud host service information and key application information in the specified cloud host. The cloud platform corresponding to the specified cloud host can be queried. Or by calling the specified scanning tool to obtain the asset information corresponding to the specified cloud host. Specifically, first determine whether an asset management tool is installed in the specified cloud host. The asset management tool has a synchronization function for asset information of the specified cloud host. When the asset management tool is installed in the specified cloud host, the specified cloud host The host will periodically synchronize the internal asset information to the corresponding designated cloud platform through the above asset management tool, so that the designated cloud platform will store the asset information of the designated cloud host. If it is determined that an asset management tool is installed in the designated cloud host, the above asset information can be found from the designated cloud platform corresponding to the above designated cloud host. If there is no asset management tool installed in the designated cloud host, the designated scanning tool can be called to scan the cloud platform to obtain the above asset information. The designated scanning tool can be nmap, and the nmap tool can be used to perform the scanning process on the cloud platform. Specify the cloud host to send the operating system detection command \$nmap-O[target IP], so that the designated cloud host will query its own asset information after receiving the system detection command, and then the asset information returned by the designated cloud host can be obtained. After obtaining the asset information of the designated cloud host, it can then intelligently identify whether there are vulnerabilities corresponding to the designated cloud host in the vulnerability database based on the asset information.

Further, in an embodiment of the present application, before the foregoing step S2, the method includes:

S200: Use crawler tools to grab the first vulnerability information related to various operating systems corresponding to the cloud host and the second vulnerability information related to the cloud host service from the target website in real time;

S201: Create a vulnerability database;

S202: Store the first vulnerability information and the second vulnerability information in the vulnerability

database.
As described in the above steps S200 to S202, before the judgment process of judging whether the vulnerability corresponding to the specified cloud host is stored in the preset vulnerability database according to the asset information, the process of generating the vulnerability database is also included. Specifically, first, a crawler tool is used to grab the first vulnerability information related to various operating systems corresponding to the cloud host and the second vulnerability information related to the cloud host service from the target website in real time. Wherein, the above-mentioned first vulnerability information may include vulnerability information corresponding to the first vulnerability, and first patch information corresponding to the first vulnerability. Similarly, the aforementioned second vulnerability information may include vulnerability information corresponding to the second vulnerability, and second patch information corresponding to the second vulnerability. In addition, crawler tools refer to web crawlers (also known as web spiders or web robots), which are programs or scripts that automatically crawl information on the World Wide Web in accordance with certain rules. The crawler tool includes but is not limited to the Python crawler tool. In addition, the aforementioned target website is an authoritative website at home and abroad for publishing vulnerabilities related to cloud hosting. Specifically, the host cluster uses a crawler tool to execute a crawler file to crawl vulnerability information that meets the data crawling conditions set by the crawler file. The crawler file includes, but is not limited to, two data crawling conditions of target URL and search keywords. The target URL is the URL of the target website corresponding to the first vulnerability information that needs to be crawled and the second vulnerability information in the crawler file, and the URL (short for Uniform Resource Locator, that is, uniform resource locator) is correct A concise representation of the location and access method of resources obtained from the Internet is the address of a standard resource on the Internet. Search keywords refer to keywords in the crawler file that are used to limit the common characteristics of the first vulnerability information and the second vulnerability information to be crawled by the crawler file. After obtaining the above-mentioned first vulnerability information and the second vulnerability information, create a vulnerability database, and then store the first vulnerability information and the second vulnerability information in the above-mentioned vulnerability database to form a subsequent use to determine whether there is a possible threat Specify the vulnerability database of the security vulnerabilities of the cloud host.
Further, in an embodiment of the present application, before the above step S4, the method includes:
S400: Obtain the current time;
S401: Determine the time period included between the last shutdown time and the current time as the designated time period.
As described in the above steps S400 to S401, before the step of judging whether there are specified vulnerabilities whose release time is within a specified period of time among all the foregoing vulnerabilities, the step of determining the specified period of time is also included. Specifically, the current time is first obtained, The above-mentioned current time has the same accuracy as the above-mentioned last shutdown time, but the accuracy is not specifically limited. For example, the above-mentioned accuracy can be accurate to year, month and day, for example, the current time can be 2019/07/11, or even It can be accurate to the year, month, day, and so on, so I won't go into too much detail here. After the current time and the aforementioned last shutdown time are obtained, the time period included between the last shutdown time and the aforementioned current time is determined as the aforementioned designated time period. For example, if the current time is 2019/07/11 and the last shutdown time is 2019/07/01, it can be determined that the above specified time period includes: 2019/07/01-2019/07/11. This embodiment determines the specified time period based on the current time and the last shutdown time of the specified cloud host, which is beneficial to subsequently accurately identifying whether there are specified vulnerabilities in the vulnerability database that threaten the security of the specified cloud host based on the specified time period. , And then intelligently select the opening method for the designated cloud host based on the recognition result.
Further, in an embodiment of the present application, after the above step S5, the method includes:
S500: Determine whether the specified patch corresponding to the specified vulnerability is stored in the vulnerability database;
S501: If yes, download the designated patch, and store the designated patch in a designated storage area in the designated cloud host through a designated transmission method;

S502: Send an installation instruction for installing the specified patch to the specified cloud host, so that the specified cloud host installs the specified patch according to the installation instruction, and returns a corresponding patch installation result, where the patch Installation results include successful or failed installation;
S503: Determine whether the specified patch is successfully installed according to the patch installation result returned by the specified cloud host;
S504: If yes, transfer the designated cloud host out of the security isolation area to restore the normal network access function of the designated cloud host.
As mentioned in the above steps S500 to S504, after the designated cloud host is placed in the security quarantine area, the designated cloud host will be removed from the security only after the designated cloud host has successfully installed the designated patch corresponding to the above-mentioned specified vulnerability. Remove from the quarantine area to ensure the safety of the designated cloud host. Specifically, it is first determined whether the vulnerability database stores the designated patch corresponding to the designated vulnerability. If the above specified patch is stored in the vulnerability database, the specified patch will be downloaded, and the specified patch will be stored in the specified storage area in the specified cloud host through the specified transmission method. Among them, the above-mentioned designated transmission method is the local area network transmission method or the Bluetooth transmission method. After the host cluster downloads the above-mentioned designated patch, it transmits the downloaded designated patch to the designated storage area in the designated cloud host by using the designated transmission method internally. , Since the transmission process does not need to use network resources, the designated cloud host will not be attacked by attackers on the public network when receiving the designated patch, which effectively guarantees the security of the designated cloud host in the process of storing the designated patch In addition, the installation process of the designated cloud host to install the designated patch does not require a network, which further increases the security of the designated cloud host. In addition, the above-mentioned designated storage area is not specifically limited, and can be set according to actual needs. For example, it can be a newly created area dedicated to storing the designated patch, or an area with a larger storage space in the designated cloud host. ,and many more. After finishing storing the specified patch in the above specified storage area, an installation instruction to install the specified patch will be sent to the specified cloud host, so that the specified cloud host can download and install the specified patch according to the above installation instruction, and return to the corresponding patch installation As a result, the patch installation result includes installation success or installation failure. Then, according to the patch installation result returned by the designated cloud host, it is further judged whether the above-mentioned designated patch is successfully installed. If the specified patch is successfully installed in the specified cloud host, the specified cloud host will be transferred out of the aforementioned security quarantine area to restore the normal network access function of the specified cloud host, so that the specified cloud host can perform normal operations after the network access function is restored. Work to ensure the user's sense of experience when using the designated cloud host. Further, if the installation of the specified patch fails, the specified cloud host will return the patch installation result that the installation failed. At this time, the specified cloud host will not be transferred out of the security quarantine area, but the reason for the failure of the specified patch installation will be found first; Then, according to the failure reason of the failure of the specified patch installation, control the specified cloud host to re-download and install the above specified patch, and after confirming that the specified patch is successfully installed in the specified cloud host, transfer the specified cloud host out of the above security The quarantine area enables the normal network access function of the designated cloud host to be restored only when the designated patch is successfully installed, effectively ensuring the security of the designated cloud host.
In an embodiment of the present application, after the above step S504, the method includes:
S5040: Acquire first usage data of a first host machine corresponding to the designated cloud host according to a first preset period, where the first usage data includes at least a CPU usage rate, a memory usage rate, and a network interface bandwidth usage rate
S5041: Analyze and process the first usage data, and determine whether the first usage data is greater than a preset standard threshold;
S5042: If yes, find out the second host machine whose second usage data is less than the standard

threshold from all the host machines;
S5043: Migrate the designated cloud host from the first host to the second host.
As described in the above steps S5040 to S5043, after the designated cloud host is transferred out of the security quarantine area, in the normal working process of the designated cloud host, if the first host corresponding to the designated cloud host is in a high load state , The host cluster also has the function of intelligent migration for the specified cloud host. Specifically, after the startup of the specified cloud host is completed, the first usage data of the first host corresponding to the specified cloud host is acquired according to the first preset period, where the first preset period is not specifically limited, for example Can be set to one hour. In addition, the aforementioned usage data includes CPU usage, memory usage, and network interface bandwidth usage. Then analyze and process the first usage data, and determine whether the above-mentioned first usage data is greater than a preset standard threshold. For different types of first usage data, the same number of standard thresholds corresponding to the type are set. . In addition, the specific value of the standard threshold is not specifically limited, and can be set by the host cluster, or can also be set by the user according to requirements. For example, the standard threshold corresponding to CPU usage can be set to 80%, the standard threshold corresponding to memory usage can be set to 75%, and the standard threshold corresponding to network interface bandwidth usage can be set to 85%. If the above-mentioned first usage data is greater than the preset standard threshold value, it indicates that the first host machine corresponding to the designated cloud host is currently in a high-load state, and the first host machine has fewer computable resources. At this time, the second host with the second usage data less than the above standard threshold will be found from all the hosts in the host cluster, and the designated cloud host will be migrated from the above first host to the above second host to achieve Intelligently migrate the designated cloud host from the first host with excessive load to the second host with lower load, so that the computing resources of the designated cloud host are optimized, thereby effectively improving the operating efficiency and response speed of the designated cloud host. It also guarantees the safety of the first host. Further, if the number of second host machines is more than one, the designated host machine with the lowest data value corresponding to the second usage data is selected from all the second host machines, and then the designated cloud host machine is selected from the above-mentioned first host machine. The host machine is migrated to the above specified host machine. By filtering out the designated host with the lowest data value corresponding to the second usage data from all the second hosts, not only the operating efficiency and response speed of the designated cloud host can be improved, but also the load balance for all the hosts can be achieved.
Further, in an embodiment of the present application, after the above step S504, the method includes:
S5044: Acquire the available resource space in the cloud hard disk corresponding to the designated cloud host according to the second preset period;
S5045: Analyze and process the available resource space, and determine whether the available resource space is less than a preset resource space threshold;
S5046: If yes, filter out the junk data in the cloud hard disk;
S5047: Send a garbage removal instruction to the designated cloud host to control the designated cloud host to clear the junk data according to the garbage removal instruction.
As described in the above steps S5044 to S5047, after the designated cloud host is transferred out of the above-mentioned security quarantine area, the available resource space in the designated cloud host in a normal working state will be monitored and processed to ensure that the designated cloud host can have Sufficient available resource space. Specifically, after the specified cloud host is started, the available resource space in the cloud hard disk corresponding to the specified cloud host is acquired according to a second preset period, where the second preset period is not specifically limited, for example, it can be set to One hour. Then it is judged whether the above-mentioned available resource space is less than the preset resource space threshold. The specific value of the above-mentioned resource space threshold is not specifically limited. It can be set by the host cluster, or can also be set by the user according to the needs. For example, the resource space The threshold can be set to 5g. If the available resource space is less than the preset resource space threshold, it indicates that the current available resource space of the designated cloud host is small, and the cloud hard disk corresponding to the designated cloud host is under high pressure. At this time, the junk data in the cloud hard disk will be filtered out. The junk data includes the remaining designated cloud host configuration files, useless temporary files, useless files that have been

deleted/uninstalled, and the hard disk and/or volume have not been formatted. Successfully leftover files, etc. After the junk data is filtered out, the garbage removal instruction is sent to the designated cloud host to control the designated cloud host to clean up the junk data according to the garbage removal instruction. Specifically, the management tool or the designated cloud host can be used to clean up the junk data. Application software to realize the automatic removal of the above-mentioned junk data. This embodiment automatically and intelligently removes junk data in the cloud hard disk corresponding to the specified cloud host when the current available resources of the specified cloud host are low, effectively reducing the pressure and burden of the specified cloud host, and releasing the specified cloud host. The redundant and useless resources on the cloud host can be used to process the junk data in the cloud hard disk corresponding to the designated cloud host in an orderly and timely manner, avoid unnecessary waste of resources, improve the utilization efficiency of the internal resources of the designated cloud host, and finally improve the designated cloud The operating efficiency and response speed of the host.

2, an embodiment of the present application also provides a secure boot device for a cloud host, including:

The first obtaining module 1 is configured to obtain asset information corresponding to the specified cloud host according to a preset rule when a user-triggered start instruction for a specified cloud host is received, wherein the asset information includes at least operating system information;

The first judgment module 2 is used for judging whether there are vulnerabilities corresponding to the asset information of the designated cloud host stored in the preset vulnerability database, wherein the number of the vulnerabilities is one or more;

The second obtaining module 3 is configured to, if yes, obtain the last shutdown time of the designated cloud host and the release time corresponding to each of the vulnerabilities;

The second judgment module 4 is used for judging whether there are specified vulnerabilities whose release time is within a specified period of time among all the vulnerabilities according to the last shutdown time and each of the release times;

The isolation module 5 is configured to, if yes, start the designated cloud host in a pre-created security isolation area, wherein the network access function of the specified cloud host will be disabled in the security isolation area.

In this embodiment, the realization process of the functions and functions of the first acquisition module, the first judgment module, the second acquisition module, the second judgment module, and the isolation module in the secure boot device of the cloud host is detailed in the above cloud host The implementation process corresponding to steps S1 to S5 in the safe boot method will not be repeated here.

Further, in an embodiment of the present application, the above-mentioned first acquisition module includes:

The first determining unit is configured to determine whether an asset management tool is installed in the designated cloud host, wherein the asset management tool is used to periodically synchronize asset information corresponding to the designated cloud host to the designated cloud platform;

The first searching unit is configured to search for the asset information from the designated cloud platform if an asset management tool is installed;

The first calling unit is configured to, if the asset management tool is not installed, call a designated scanning tool to perform scanning processing on the designated cloud host, so as to obtain the asset information.

In this embodiment, the implementation process of the functions and roles of the first determining unit, the first searching unit, and the first calling unit in the secure boot device of the cloud host is detailed in the corresponding steps S100 to S100 to the secure boot method of the cloud host. The implementation process of S102 will not be repeated here.

Further, in an embodiment of the present application, the secure boot device of the cloud host includes:

The crawling module is used to grab the first vulnerability information related to various operating systems corresponding to the cloud host and the second vulnerability information related to the cloud host service from the target website in real time through a crawler tool;

Create module to create a vulnerability database;

The first storage module is used to store the first vulnerability information and the second vulnerability information in the vulnerability database.

In this embodiment, the implementation process of the functions and roles of the grab module, creation module, and first storage module in the secure boot device of the cloud host is detailed in the implementation of corresponding steps S200 to S202 in the secure boot method of the cloud host. The

process will not be repeated here.
Further, in an embodiment of the present application, the secure boot device of the cloud host includes:
The third obtaining module is used to obtain the current time;
The first determining module is configured to determine the time period included between the last shutdown time and the current time as the designated time period.
In this embodiment, the implementation process of the functions and roles of the third acquiring module and the first determining module in the secure startup device of the cloud host is detailed in the implementation process corresponding to steps S400 to S401 in the secure startup method of the cloud host. I won't repeat them here.
Further, in an embodiment of the present application, the secure boot device of the cloud host includes:
The second determining module is used to determine whether the specified patch corresponding to the specified vulnerability is stored in the vulnerability database;
The second storage module is configured to download the specified patch if it is, and store the specified patch in a specified storage area in the specified cloud host through a specified transmission method;
The first sending module is configured to send an installation instruction for installing the specified patch to the specified cloud host, so that the specified cloud host installs the specified patch according to the installation instruction, and returns the corresponding patch installation result, Wherein, the patch installation result includes installation success or installation failure;
The third judgment module is configured to judge whether the specified patch is successfully installed according to the patch installation result returned by the specified cloud host;
The transfer module is used for transferring the designated cloud host out of the security isolation area, so as to restore the normal network access function of the designated cloud host.
In this embodiment, the realization process of the functions and roles of the second determination module, the second storage module, the first sending module, the third judgment module, and the transfer module in the secure boot device of the cloud host is detailed in the above cloud host The implementation process corresponding to steps S500 to S504 in the secure boot method will not be repeated here.
Further, in an embodiment of the present application, the secure boot device of the cloud host includes:
The fourth acquiring module is configured to acquire the first usage data of the first host machine corresponding to the designated cloud host according to the first preset period, where the first usage data includes at least a CPU usage rate, a memory usage rate, and Network interface bandwidth usage rate;
The fourth judgment module is configured to analyze and process the first usage data, and judge whether the first usage data is greater than a preset standard threshold;
The searching module is configured to, if yes, find out the second host machine whose second usage data is less than the standard threshold value from all the host machines;
The migration module is used to migrate the designated cloud host from the first host to the second host.
In this embodiment, the implementation process of the functions and roles of the fourth acquisition module, the fourth judgment module, the search module, and the migration module in the secure boot device of the cloud host is detailed in the corresponding step S5040 in the secure boot method of the cloud host. The implementation process to S5043 will not be repeated here.
Further, in an embodiment of the present application, the secure boot device of the cloud host includes:
A fifth acquiring module, configured to acquire the available resource space in the cloud hard disk corresponding to the designated cloud host according to a second preset period;
The fifth judgment module is configured to analyze and process the available resource space, and judge whether the available resource space is less than a preset resource space threshold;
The screening module is used to screen out junk data in the cloud hard disk if so;
The second sending module is configured to send a garbage removal instruction to the designated cloud host to control the designated cloud host to clear the junk data according to the garbage removal instruction.
In this embodiment, the implementation process of the functions and roles of the fifth acquiring module, the fifth determining module, the screening module, and the second sending module in the secure boot device of the cloud host is detailed in the corresponding method in the secure boot method of the cloud host. The implementation process of steps S5044 to S5047 will not be repeated here.
Referring to FIG. 3, an embodiment of the present application also provides a computer device. The computer device may be a server, and its internal structure may be as shown in FIG. 3. The computer

equipment includes a processor, a memory, a network interface, and a database connected through a system bus. Among them, the processor designed for the computer equipment is used to provide calculation and control capabilities. The memory of the computer device includes a volatile or non-volatile storage medium and internal memory. The volatile or non-volatile storage medium stores an operating system, computer readable instructions, and a database. The internal memory provides an environment for the operation of the operating system and computer-readable instructions in the volatile or non-volatile storage medium. The database of the computer equipment is used to store data such as asset information and vulnerabilities. The network interface of the computer device is used to communicate with an external terminal through a network connection. When the computer-readable instruction is executed by the processor, the method for secure startup of the cloud host shown in any of the above exemplary embodiments is realized. Those skilled in the art can understand that the structure shown in FIG. 3 is only a block diagram of a part of the structure related to the solution of the present application, and does not constitute a limitation on the devices and computer equipment to which the solution of the present application is applied.

An embodiment of the present application also provides a computer-readable storage medium. The computer-readable storage medium may be volatile or non-volatile, and computer-readable instructions are stored thereon. When executed by one or more processors, the steps in the foregoing embodiment of the secure boot method for a cloud host are realized when executed by one or more processors.

In summary, the secure startup method, device, computer equipment, and storage medium of the cloud host provided in the embodiments of the present application, upon receiving a user-triggered startup instruction for a specified cloud host, obtain the Asset information corresponding to the designated cloud host, where the asset information includes at least operating system information; determine whether a predetermined vulnerability database stores vulnerabilities corresponding to the asset information of the designated cloud host, wherein the number of vulnerabilities Is one or more; if so, obtain the last shutdown time of the designated cloud host and the release time corresponding to each of the vulnerabilities; according to the last shutdown time and each release time, it is determined that all the Whether there are specified vulnerabilities in the vulnerabilities whose release time is within a specified period of time; if so, start the designated cloud host in the pre-created security isolation zone, where the network access function of the specified cloud host is in the security isolation zone Will be disabled. This application will intelligently place the designated cloud host in the pre-created security isolation zone for startup when a new vulnerability related to it occurs during the last shutdown of the designated cloud host, so that after the designated cloud host is started Temporarily disable the network access function of the designated cloud host, effectively avoiding the intrusion or attack of the designated cloud host by vulnerabilities during use, and ensuring the security of the designated cloud host after it is started.

A person of ordinary skill in the art can understand that all or part of the processes in the methods of the foregoing embodiments can be implemented by computer-readable instructions to instruct relevant hardware. The computer-readable instructions can be stored in a non-volatile or In a volatile computer-readable storage medium, when the computer-readable instructions are executed, they may include the processes of the above-mentioned method embodiments. Wherein, any reference to memory, storage, database or other media provided in this application and used in the embodiments may include non-volatile and/or volatile memory. Non-volatile memory may include read-only memory (ROM), programmable ROM (PROM), electrically programmable ROM (EPROM), electrically erasable programmable ROM (EEPROM), or flash memory. Volatile memory may include random access memory (RAM) or external cache memory. As an illustration and not a limitation, RAM is available in many forms, such as static RAM (SRAM), dynamic RAM (DRAM), synchronous DRAM (SDRAM), dual-rate SDRAM (SSRSDRAM), enhanced SDRAM (ESDRAM), synchronous Link (Synchlink) DRAM (SLDRAM), memory bus (Rambus) direct RAM (RDRAM), direct memory bus dynamic RAM (DRDRAM), and memory bus dynamic RAM (RDRAM), etc.

The above are only the preferred embodiments of this application, and do not limit the scope of this application. Any equivalent structure or equivalent process transformation made using the content of the specification and drawings of this application, or directly or indirectly applied to other related The technical field is equally included in the scope of patent protection of this application.

Claims

A method for securely starting a cloud host, which is characterized in that it includes:
When receiving a user-triggered start instruction for a designated cloud host, acquiring asset information corresponding to the designated cloud host according to a preset rule, where the asset information includes at least operating system information;
Judging whether there are vulnerabilities corresponding to the asset information of the designated cloud host stored in the preset vulnerability database, wherein the number of the vulnerabilities is one or more;
If yes, obtain the last shutdown time of the designated cloud host and the release time corresponding to each of the vulnerabilities;
According to the last shutdown time and each of the release times, determine whether there is a specified vulnerability whose release time is within a specified period of time among all the vulnerabilities;
If yes, start the designated cloud host in a pre-created security isolation zone, wherein the network access function of the designated cloud host will be disabled in the security isolation zone.
The secure startup method of a cloud host according to claim 1, wherein the step of obtaining asset information corresponding to the designated cloud host according to a preset rule comprises:
Determining whether an asset management tool is installed in the designated cloud host, wherein the asset management tool is used to periodically synchronize asset information corresponding to the designated cloud host to the designated cloud platform;
If an asset management tool is installed, search for the asset information from the designated cloud platform;
If the asset management tool is not installed, call the designated scanning tool to scan the designated cloud host to obtain the asset information.
The method for secure startup of a cloud host according to claim 1, wherein before the step of determining whether a vulnerability corresponding to the asset information of the designated cloud host is stored in the preset vulnerability database, the method further include:
Through the crawler tool, the first vulnerability information related to various operating systems corresponding to the cloud host and the second vulnerability information related to the cloud host service are captured from the target website in real time;
Create a vulnerability database;
The first vulnerability information and the second vulnerability information are stored in the vulnerability database.
The method for secure startup of a cloud host according to claim 1, wherein said determining whether there is a release time in all said vulnerabilities is within a specified time period according to said last shutdown time and each said release time Before the step of specifying vulnerabilities, the method also includes:
Get the current time;
The time period included between the last shutdown time and the current time is determined as the designated time period.
The method for safely starting a cloud host according to claim 1, wherein after the step of starting the designated cloud host in a pre-created security isolation zone, the method further comprises:
Determining whether the specified patch corresponding to the specified vulnerability is stored in the vulnerability database;
If yes, download the designated patch, and store the designated patch in a designated storage area in the designated cloud host through a designated transmission method;
Send an installation instruction for installing the specified patch to the specified cloud host, so that the specified cloud host installs the specified patch according to the installation instruction, and returns a corresponding patch installation result, wherein the patch installation result Including successful or failed installation;
Judging whether the specified patch is successfully installed according to the patch installation result returned by the specified cloud host;
If yes, transfer the designated cloud host out of the security isolation area to restore the normal network access function of the designated cloud host.
The secure startup method of a cloud host according to claim 5, wherein after the step of transferring the designated cloud host out of the security isolation area to restore the normal network access function of the

designated cloud host, The method also includes:
Acquiring first usage data of a first host machine corresponding to the designated cloud host according to a first preset period, where the first usage data includes at least a CPU usage rate, a memory usage rate, and a network interface bandwidth usage rate;
Analyzing and processing the first usage data, and determining whether the first usage data is greater than a preset standard threshold;
If yes, find out the second host machine whose second usage data is less than the standard threshold from all the host machines;
Migrating the designated cloud host from the first host to the second host.
The secure startup method of a cloud host according to claim 5, wherein after the step of transferring the designated cloud host out of the security isolation area to restore the normal network access function of the designated cloud host, The method also includes:
Acquiring, according to a second preset period, the available resource space in the cloud hard disk corresponding to the designated cloud host;
Analyzing and processing the available resource space, and determining whether the available resource space is less than a preset resource space threshold;
If yes, filter out the junk data in the cloud hard disk;
Sending a garbage removal instruction to the designated cloud host to control the designated cloud host to clear the garbage data according to the garbage removal instruction.
A secure boot device for a cloud host, which is characterized in that it includes:
The first obtaining module is configured to obtain asset information corresponding to the specified cloud host according to preset rules when a user-triggered start instruction for the specified cloud host is received, wherein the asset information includes at least operating system information;
The first judgment module is used to judge whether there are vulnerabilities corresponding to the asset information of the designated cloud host stored in the preset vulnerability database, wherein the number of the vulnerabilities is one or more;
The second obtaining module is configured to, if yes, obtain the last shutdown time of the designated cloud host and the release time corresponding to each of the vulnerabilities;
The second judgment module is used for judging whether there are specified vulnerabilities whose release time is within a specified period of time among all the vulnerabilities according to the last shutdown time and each of the release times;
The isolation module is configured to, if yes, start the designated cloud host in a pre-created security isolation area, wherein the network access function of the specified cloud host will be disabled in the security isolation area.
The secure boot device for a cloud host according to claim 8, wherein the first obtaining module comprises:
The first determining unit is configured to determine whether an asset management tool is installed in the designated cloud host, wherein the asset management tool is used to periodically synchronize asset information corresponding to the designated cloud host to the designated cloud platform;
The first searching unit is configured to search for the asset information from the designated cloud platform if an asset management tool is installed;
The first calling unit is configured to, if the asset management tool is not installed, call a designated scanning tool to perform scanning processing on the designated cloud host, so as to obtain the asset information.
The secure boot device of a cloud host according to claim 8, wherein the device comprises:
The crawling module is used to grab the first vulnerability information related to various operating systems corresponding to the cloud host and the second vulnerability information related to the cloud host service from the target website in real time through a crawler tool;
Create module to create a vulnerability database;
The first storage module is used to store the first vulnerability information and the second vulnerability information in the vulnerability database.
The secure boot device of a cloud host according to claim 8, wherein the device comprises:
The third obtaining module is used to obtain the current time;
The first determining module is configured to determine the time period included between the last shutdown time and the current time as the designated time period.

The secure boot device of a cloud host according to claim 8, wherein the device comprises:
The second determining module is used to determine whether the specified patch corresponding to the specified vulnerability is stored in the vulnerability database;
The second storage module is configured to download the specified patch if it is, and store the specified patch in a specified storage area in the specified cloud host through a specified transmission method;
The first sending module is configured to send an installation instruction for installing the specified patch to the specified cloud host, so that the specified cloud host installs the specified patch according to the installation instruction, and returns the corresponding patch installation result, Wherein, the patch installation result includes installation success or installation failure;
The third judgment module is configured to judge whether the specified patch is successfully installed according to the patch installation result returned by the specified cloud host;
The transfer module is used for transferring the designated cloud host out of the security isolation area, so as to restore the normal network access function of the designated cloud host.
The secure boot device of a cloud host according to claim 12, wherein the device comprises:
The fourth acquiring module is configured to acquire the first usage data of the first host machine corresponding to the designated cloud host according to the first preset period, where the first usage data includes at least a CPU usage rate, a memory usage rate, and Network interface bandwidth usage rate;
The fourth judgment module is configured to analyze and process the first usage data, and judge whether the first usage data is greater than a preset standard threshold;
The searching module is configured to, if yes, find out the second host machine whose second usage data is less than the standard threshold value from all the host machines;
The migration module is used to migrate the designated cloud host from the first host to the second host.
The secure boot device of a cloud host according to claim 12, wherein the device comprises:
A fifth acquiring module, configured to acquire the available resource space in the cloud hard disk corresponding to the designated cloud host according to a second preset period;
The fifth judgment module is configured to analyze and process the available resource space, and judge whether the available resource space is less than a preset resource space threshold;
The screening module is used to screen out junk data in the cloud hard disk if so;
The second sending module is configured to send a garbage removal instruction to the designated cloud host to control the designated cloud host to clear the junk data according to the garbage removal instruction.
A computer device includes a memory and a processor, the memory stores computer-readable instructions, and is characterized in that, when the processor executes the computer-readable instructions, a method for safely starting a cloud host is realized, and the cloud host is Start method, including:
When receiving a user-triggered start instruction for a designated cloud host, acquiring asset information corresponding to the designated cloud host according to a preset rule, where the asset information includes at least operating system information;
Judging whether there are vulnerabilities corresponding to the asset information of the designated cloud host stored in the preset vulnerability database, wherein the number of the vulnerabilities is one or more;
If yes, obtain the last shutdown time of the designated cloud host and the release time corresponding to each of the vulnerabilities;
According to the last shutdown time and each of the release times, determine whether there is a specified vulnerability whose release time is within a specified period of time among all the vulnerabilities;
If yes, start the designated cloud host in a pre-created security isolation zone, wherein the network access function of the designated cloud host will be disabled in the security isolation zone.
The computer device according to claim 15, wherein the step of obtaining asset information corresponding to the designated cloud host according to a preset rule comprises:
Determining whether an asset management tool is installed in the designated cloud host, wherein the asset management tool is used to periodically synchronize asset information corresponding to the designated cloud host to the designated cloud platform;
If an asset management tool is installed, search for the asset information from the designated cloud platform;
If the asset management tool is not installed, call the designated scanning tool to scan the designated cloud host to obtain the asset information.
The computer device according to claim 15, wherein before the step of determining whether a vulnerability

corresponding to the asset information of the designated cloud host is stored in the preset vulnerability database, the method further comprises:
Through the crawler tool, the first vulnerability information related to various operating systems corresponding to the cloud host and the second vulnerability information related to the cloud host service are captured from the target website in real time;
Create a vulnerability database;
The first vulnerability information and the second vulnerability information are stored in the vulnerability database.
A computer-readable storage medium having computer-readable instructions stored thereon, wherein the computer-readable instructions are executed by a processor to realize a secure startup method of a cloud host, and the secure startup method of the cloud host includes:
When receiving a user-triggered start instruction for a designated cloud host, acquiring asset information corresponding to the designated cloud host according to a preset rule, where the asset information includes at least operating system information;
Judging whether there are vulnerabilities corresponding to the asset information of the designated cloud host stored in the preset vulnerability database, wherein the number of the vulnerabilities is one or more;
If yes, obtain the last shutdown time of the designated cloud host and the release time corresponding to each of the vulnerabilities;
According to the last shutdown time and each of the release times, determine whether there is a specified vulnerability whose release time is within a specified period of time among all the vulnerabilities;
If yes, start the designated cloud host in a pre-created security isolation zone, wherein the network access function of the designated cloud host will be disabled in the security isolation zone.
The computer-readable storage medium according to claim 18, wherein the step of obtaining asset information corresponding to the designated cloud host according to a preset rule comprises:
Determining whether an asset management tool is installed in the designated cloud host, wherein the asset management tool is used to periodically synchronize asset information corresponding to the designated cloud host to the designated cloud platform;
If an asset management tool is installed, search for the asset information from the designated cloud platform;
If the asset management tool is not installed, call the designated scanning tool to scan the designated cloud host to obtain the asset information.
The computer-readable storage medium according to claim 18, wherein before the step of determining whether a vulnerability corresponding to the asset information of the designated cloud host is stored in the preset vulnerability database, the method further comprises :
Through the crawler tool, the first vulnerability information related to various operating systems corresponding to the cloud host and the second vulnerability information related to the cloud host service are captured from the target website in real time;
Create a vulnerability database;
The first vulnerability information and the second vulnerability information are stored in the vulnerability database.

Copyright © 2022 Clarivate Analytics. All rights reserved. Republication or redistribution of Clarivate Analytics content, including by framing or similar means, is prohibited without the prior written consent of Clarivate Analytics. Clarivate and its logo are trademarks of the Clarivate Analytics group.

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2021 年 4 月 22 日 (22.04.2021)

WIPO | PCT

(10) 国际公布号
WO 2021/072877 A1

(51) 国际专利分类号:
G06F 9/4401 (2018.01) *G06F 21/53* (2013.01)
G06F 9/455 (2006.01) *G06F 21/57* (2013.01)
G06F 8/61 (2018.01)

(21) 国际申请号: PCT/CN2019/118430

(22) 国际申请日: 2019 年 11 月 14 日 (14.11.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910979050.1 2019年10月15日 (15.10.2019) CN

(71) 申请人: 平安科技（深圳）有限公司 (PING AN TECHNOLOGY(SHENZHEN)CO.,LTD.) [CN/CN]; 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518000 (CN)。

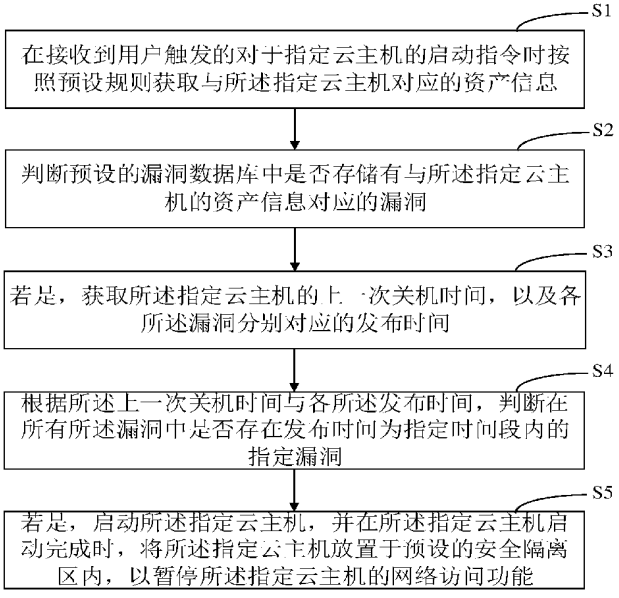
(72) 发明人: 沈勇 (SHEN, Yong); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518000 (CN)。

(74) 代理人: 深圳市明日今典知识产权代理事务所（普通合伙） (SHENZHEN MINGRIJINDIAN INTELLECTUAL PROPERTY AGENCY FIRM(GENERAL)); 中国广东省深圳市南山区南头街道安乐社区关口二路15号智恒产业园30栋4层406室, Guangdong 518000 (CN)。

(81) 指定国(除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB,

(54) **Title:** SECURE STARTING METHOD AND APPARATUS FOR CLOUD HOST, AND COMPUTER DEVICE AND STORAGE MEDIUM

(54) **发明名称:** 云主机的安全启动方法、装置、计算机设备和存储介质



S1 When a starting instruction, triggered by a user, for a specified cloud host is received, acquire, according to a preset rule, asset information corresponding to the specified cloud host
S2 Determine whether there are vulnerabilities, corresponding to the asset information of the specified cloud host, stored in a preset vulnerability database
S3 If so, acquire the last shutdown time of the specified cloud host and release times respectively corresponding to the vulnerabilities
S4 According to the last shutdown time and the release times, determine whether there is a specified vulnerability, of which the release time is within a specified time period, in all vulnerabilities
S5 If so, start the specified cloud host, and place the specified cloud host in a preset secure isolation area when the starting of the specified cloud host is completed, so as to suspend a network access function of the specified cloud host

图 1

(57) **Abstract:** Disclosed are a secure starting method and apparatus for a cloud host, and a computer device and a storage medium. The method comprises: when a starting instruction, triggered by a user, for a specified cloud host is received, acquiring, according to a preset rule, asset information corresponding to the specified cloud host; determining whether there are vulnerabilities, corresponding to the asset information of the specified cloud host, stored in a preset vulnerability database; if so, acquiring the last shutdown time of the specified cloud host and release times respectively corresponding to the vulnerabilities; according to the last shutdown time and the

WO 2021/072877 A1

GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布：

— 包括国际检索报告(条约第21条(3))。

release times, determining whether there is a specified vulnerability, of which the release time is within a specified time period, in all the vulnerabilities; and if so, starting the specified cloud host in a pre-created secure isolation area. According to the preset application, the specified cloud host can be effectively prevented from being invaded or attacked by vulnerabilities during use, and the security of the specified cloud host after starting is guaranteed.

(57) 摘要：本申请揭示了一种云主机的安全启动方法、装置、计算机设备和存储介质，其中方法包括：在接收到用户触发的对于指定云主机的启动指令时，按照预设规则获取与指定云主机对应的资产信息；判断预设的漏洞数据库中是否存储有与指定云主机的资产信息对应的漏洞；若是，获取指定云主机的上一次关机时间，以及各漏洞分别对应的发布时间；根据上一次关机时间与各发布时间，判断在所有漏洞中是否存在发布时间为指定时间段内的指定漏洞；若是，在预创建的安全隔离区内启动所述指定云主机。本申请可以有效的避免出现指定云主机在使用过程中受到漏洞的入侵或攻击的情况，保障了指定云主机启动后的安全性。

云主机的安全启动方法、装置、计算机设备和存储介质

[0001] 本申请要求于2019年10月15日提交中国专利局、申请号为2019109790501，发明名称为“云主机的安全启动方法、装置、计算机设备和存储介质”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

[0002] 本申请涉及云计算技术领域，具体涉及一种云主机的安全启动方法、装置、计算机设备和存储介质。

背景技术

[0003] 云计算当前是IT技术发展的一个重要领域和方向，并正在逐步渗透到社会的各领域。云主机是云计算在基础设施应用上的重要组成部分，位于云计算产业链金字塔底层，产品源自云计算平台。该平台整合了互联网应用三大核心要素：计算、存储、网络，面向用户提供公用化的互联网基础设施服务。云主机是一种类似VPS(Virtual Private Server，虚拟专用服务器)主机的虚拟化技术，VPS是采用虚拟软件(如VZ或VM)在一台主机上虚拟出多个类似独立主机的部分，能够实现单机多用户，每个部分都可以做单独的操作系统，管理方法同主机相同。而云主机是在一组宿主机集群上虚拟出多个类似独立主机的部分，集群中每个宿主机上都有云主机的一个镜像，从而大大提高了虚拟主机的安全稳定性。另外，为了便于用户随时随地访问云计算，云主机通常能从公网（Internet）直接访问，甚至是管理。

[0004] 在现有的各类云计算系统中，各个云主机均会面临着防护间隙的安全风险。具体地，在云主机处于关机状态时，由于云主机在关机时会对当前所有已知的安全漏洞进行修复处理，因而可认为处于关机状态的云主机也处于一个安全的状态。但是，当云主机被重新启动并暴露在公网时，此时的外界环境相比于关机前可能会发生改变。如果当前出现了与云主机相关的新的安全漏洞，则云主机随时可能会遭受到公网上的攻击者的攻击，云主机的安全性受到极大的威胁。

发明概述

技术问题

[0005] 本申请的主要目的为提供一种云主机的安全启动方法、装置、计算机设备和存储介质，旨在解决在现有的各类云计算系统中，当云主机从关机状态被重新启动并暴露在公网时，随时可能会遭受到公网上的攻击者的攻击，云主机的安全性容易受到极大的威胁的技术问题。

问题的解决方案

技术解决方案

[0006] 为了实现上述申请目的，本申请提出一种云主机的安全启动方法，所述方法包括步骤：

[0007] 在接收到用户触发的对于指定云主机的启动指令时按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

[0008] 若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

[0009] 根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

[0010] 若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。

[0011] 本申请还提供一种云主机的安全启动装置，包括：

[0012] 第一获取模块，用于在接收到用户触发的对于指定云主机的启动指令时按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；

[0013] 第一判断模块，用于判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

[0014] 第二获取模块，用于若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

[0015] 第二判断模块，用于根据所述上一次关机时间与各所述发布时间，判断在所有

所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

[0016] 隔离模块，用于若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。

[0017] 本申请还提供一种计算机设备，包括存储器和处理器，所述存储器中存储有计算机可读指令，所述处理器执行所述计算机可读指令时实现上述方法的步骤。

[0018] 本申请还提供一种计算机可读存储介质，其上存储有计算机可读指令，所述计算机可读指令被处理器执行时实现上述方法的步骤。

发明的有益效果

有益效果

[0019] 本申请中提供的云主机的安全启动方法、装置、计算机设备和存储介质，在接收到用户触发的对于指定云主机的启动指令时，按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。本申请当指定云主机在上一次关机期间时出现了新的与其相关的漏洞的情况下，会智能的将指定云主机放置在预创建的安全隔离区内进行启动，从而在指定云主机启动后暂时禁用指定云主机的网络访问功能，有效的避免出现指定云主机在使用过程中受到漏洞的入侵或攻击的情况，保障了指定云主机启动后的安全性。

对附图的简要说明

附图说明

[0020] 图1是本申请一实施例的云主机的安全启动方法的流程示意图；

[0021] 图2是本申请一实施例的云主机的安全启动装置的结构示意图；

[0022] 图3是本申请一实施例的计算机设备的结构示意图。

实施该发明的最佳实施例

本发明的最佳实施方式

[0023] 应当理解，此处所描述的具体实施例仅仅用于解释本申请，并不用于限定本申请。

[0024] 参照图1，本申请一实施例的云主机的安全启动方法，包括：

[0025] S1：在接收到用户触发的对于指定云主机的启动指令时按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；

[0026] S2：判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

[0027] S3：若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

[0028] S4：根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

[0029] S5：若是，启动所述指定云主机，并在所述指定云主机启动完成时，将所述指定云主机放置于预设的安全隔离区内，以暂停所述指定云主机的网络访问功能。

[0030] 如上述步骤S1至S5所述，本发明实施例的执行主体为一种云主机的安全启动装置，具体可为用于虚拟出云主机的宿主机集群，通过该宿主机集群可实现对于指定云主机的安全启动，上述指定云主机为宿主机集群虚拟出的所有云主机中任意一台云主机。具体地，首先判断是否接收到用户触发的对于指定云主机的启动指令；当接收到该启动指令时，按照预设规则获取与指定云主机对应的资产信息，其中，上述资产信息至少包括操作系统信息，例如可为操作系统为windowsserver2012，使用RDP服务11.0版本提供远程桌面，另外该资产信息还可包括云主机服务信息与关键应用信息。另外，上述预设规则可包括：通过查询与指定云主机对应的云平台的方式，或者通过调用指定扫描工具的方式来获取与指定云主机对应的资产信息。然后，根据上述资产信息，判断预设的漏洞数据库中是否存储有与上述指定云主机的资产信息对应的漏洞，其中，上述漏洞的数量为一个或多个，另外，上述漏洞数据库内存储有通过爬虫工具实时从目标

网站抓取与云主机对应的各类操作系统相关的漏洞信息，以及与云主机服务相关的漏洞信息。如果漏洞数据库中存储有与上述指定云主机的资产信息对应的漏洞，则获取上述指定云主机的上一次关机时间以及各上述漏洞分别对应的发布时间。其中，举例地，爬取的任一漏洞信息的形式可为CVE20190004-rdp漏洞，2019/07/02发布，受影响系统为rdp服务12.0版本之前的所有版本，则对应的漏洞的发布时间为2019/07/02。然后根据上述上一次关机时间与各上述发布时间，判断在所有漏洞中是否存在发布时间为指定时间段内的指定漏洞，其中，上述指定时间段为上述关机时间与上述当前时间之间所包含的时间段。如果存在发布时间为指定时间段内的指定漏洞，则在预创建的安全隔离区内启动上述指定云主机，其中，指定云主机的网络访问功能在上述安全隔离区内会被禁用。本实施例当指定云主机在上一次关机期间时出现了新的与其相关的漏洞的情况下，会智能的将指定云主机放置在预创建的安全隔离区内进行启动，从而在指定云主机启动后暂时禁用指定云主机的网络访问功能，有效的避免出现指定云主机在使用过程中受到漏洞的入侵或攻击的情况，保障了指定云主机启动后的安全性。进一步地，如果在所有漏洞中不存在发布时间为指定时间段内的指定漏洞，表明在指定云主机的上一次关机期间并未出现与其相关的漏洞，则直会接完成对于指定云主机的启动，以及不用将指定云主机放置于安全隔离区。以有效的保证用户对于指定云主机的使用体验。

[0031] 进一步地，本申请一实施例中，上述步骤S1，包括：

[0032] S100：确定所述指定云主机内是否安装有资产管理工具，其中，所述资产管理工具用于定期将所述指定云主机对应的资产信息同步至指定云平台；

[0033] S101：若安装有资产管理工具，从所述指定云平台中查找出所述资产信息；

[0034] S102：若未安装有资产管理工具，调用指定扫描工具对所述指定云主机进行扫描处理，以得到所述资产信息。

[0035] 如上述步骤S100至S102所述，上述资产信息具体可包括指定云主机对应的操作系统信息，以及指定云主机内的云主机服务信息与关键应用信息，可通过查询与指定云主机对应的云平台的方式，或者通过调用指定扫描工具的方式来获取与指定云主机对应的资产信息。具体地，首先确定上述指定云主机内是否安装

有资产管理工具，其中，上述资产管理工具具有对指定云主机的资产信息的同步功能，当指定云主机内安装有资产管理工具时，则指定云主机会定期通过上述资产管理工具来将内部的资产信息同步至对应的上述指定云平台，从而该指定云平台会存储有指定云主机的资产信息。如果确定出指定云主机内安装有资产管理工具，则可从与上述指定云主机对应的指定云平台中查找出上述资产信息。而如果指定云主机内未安装有资产管理工具，则可调用指定扫描工具对上述云平台进行扫描处理，以得到上述资产信息，其中，上述指定扫描工具可为nmap，可通过nmap工具来对该指定云主机发送操作系统探测命令$nmap-O[target IP]，使得指定云主机在接收到该作系统探测命令后会查询自身的资产信息，进而可以得到由指定云主机返回的资产信息。在得到了指定云主机的资产信息后，后续便可智能根据该资产信息来识别出漏洞数据库中是否存在与指定云主机对应的漏洞。

[0036] 进一步地，本申请一实施例中，上述步骤S2之前，包括：

[0037] S200：通过爬虫工具，实时从目标网站抓取出与云主机对应的各类操作系统相关的第一漏洞信息，以及与云主机服务相关的第二漏洞信息；

[0038] S201：创建一个漏洞数据库；

[0039] S202：将所述第一漏洞信息与所述第二漏洞信息存储于所述漏洞数据库内。

[0040] 如上述步骤S200至S202所述，在根据上述资产信息，进行判断预设的漏洞数据库中是否存储有与上述指定云主机对应的漏洞的判断过程之前，还包括生成该漏洞数据库的生成过程。具体地，首先通过爬虫工具，实时从目标网站抓取与云主机对应的各类操作系统相关的第一漏洞信息，以及与云主机服务相关的第二漏洞信息。其中，上述第一漏洞信息可包括与第一漏洞对应的漏洞信息，以及与第一漏洞对应的第一补丁信息。同理，上述第二漏洞信息可包括与第二漏洞对应的漏洞信息，以及与第二漏洞对应的第二补丁信息。另外，爬虫工具是指网络爬虫(又被称为网页蜘蛛或网络机器人)，是一种按照一定的规则，自动地抓取万维网信息的程序或者脚本。该爬虫工具包括但不限于Python爬虫工具。另外，上述目标网站为用于发布与云主机相关漏洞的国内外权威网站。具体地，宿主机集群采用爬虫工具执行爬虫文件，以爬取符合爬虫文件设置的数据爬取

条件的漏洞信息。其中，该爬虫文件包括但不限于目标URL和搜索关键词这两个数据爬取条件。该目标URL是该爬虫文件中用于限定所需爬取的第一漏洞信息与第二漏洞信息对应的目标网站的URL，该URL(Uniform Resource Locator的简称，即统一资源定位符)是对可以从互联网上得到的资源的位置和访问方法的一种简洁的表示，是互联网上标准资源的地址。搜索关键词是指该爬虫文件中用于限定该爬虫文件所需爬取的第一漏洞信息与第二漏洞信息的共同特性的关键词。在得到上述第一漏洞信息与第二漏洞信息后，再创建一个漏洞数据库，然后将该第一漏洞信息与第二漏洞信息存储于上述漏洞数据库内，以形成后续用于确定是否存在可能会威胁指定云主机安全的漏洞的漏洞数据库。

[0041] 进一步地，本申请一实施例中，上述步骤S4之前，包括：

[0042] S400：获取当前时间；

[0043] S401：将所述上一次关机时间与所述当前时间之间所包含的时间段，确定为所述指定时间段。

[0044] 如上述步骤S400至S401所述，上述判断所有上述漏洞中是否存在发布时间为指定时间段内的指定漏洞的步骤之前，还包括确定出该指定时间段的步骤，具体地，首先获取当前时间，其中，上述当前时间与上述上一次关机时间的精确度相同，但对该精确度不做具体限定，例如上述精确度可精确到年月日，例如当前时间可为2019/07/11，或者还可精确到年月日时等，在此不做过多叙述。在得到当前时间以及上述上一次关机时间后，再将该上一次关机时间与上述当前时间之间所包含的时间段确定为上述指定时间段。举例地，如果当前时间为2019/07/11，上一次关机时间为2019/07/01，则可确定出上述指定时间段包括：2019/07/01-2019/07/11。本实施例根据当前时间与指定云主机的上一次关机时间来确定出指定时间段，有利于后续根据该指定时间段来精确的识别出漏洞数据库中是否存有会威胁指定云主机安全的指定漏洞，进而根据识别结果来智能的选择对于指定云主机的开启方式。

[0045] 进一步地，本申请一实施例中，上述步骤S5之后，包括：

[0046] S500：确定所述漏洞数据库中是否存储有与所述指定漏洞对应的指定补丁；

[0047] S501：若是，下载所述指定补丁，并通过指定传输方式将所述指定补丁存储于

所述指定云主机内的指定存储区域中；

[0048] S502：向所述指定云主机发送安装所述指定补丁的安装指令，以便所述指定云主机根据所述安装指令对所述指定补丁进行安装，并返回对应的补丁安装结果，其中，所述补丁安装结果包括安装成功或安装失败；

[0049] S503：根据所述指定云主机返回的所述补丁安装结果，判断所述指定补丁是否安装成功；

[0050] S504：若是，将所述指定云主机转移出所述安全隔离区，以恢复所述指定云主机正常的网络访问功能。

[0051] 如上述步骤S500至S504所述，在将指定云主机放置于安全隔离区后，只有在该指定云主机成功安装成功了与上述指定漏洞对应的指定补丁后，才会将该指定云主机从安全隔离区内移除，以保证指定云主机的使用安全。具体地，首先判断上述漏洞数据库是否存储有与上述指定漏洞对应的指定补丁。如果漏洞数据库内存储有上述指定补丁，则会下载该指定补丁，并通过指定传输方式将该指定补丁存储于指定云主机内的指定存储区域中。其中，上述指定传输方式为局域网传输方式或蓝牙传输方式，宿主机集群在下载了上述指定补丁后，通过在内部利用该指定传输方式将下载好的指定补丁传输至指定云主机内的指定存储区域，由于传输过程并不需要使用到网络资源，使得指定云主机在接收到指定补丁时也不会遭受公网上的攻击者的攻击，有效的保证了指定云主机在存储指定补丁的过程中的安全性，而且指定云主机对指定补丁进行安装的安装过程也不需要网络，进一步增加了指定云主机的使用安全性。另外，对于上述指定存储区域不作具体限定，可以根据实际需求进行设置，例如可以是一个新创建的专门用于存储该指定补丁的区域，也可以是指定云主机中的存储空间较大的一个区域，等等。在完成了将指定补丁存储于上述指定存储区域中后，会向上述指定云主机发送安装指定补丁的安装指令，以便指定云主机根据上述安装指令下载并安装该指定补丁，并返回对应的补丁安装结果，其中，所述补丁安装结果包括安装成功或安装失败。然后根据指定云主机返回的补丁安装结果，进一步判断上述指定补丁是否安装成功。如果指定补丁在指定云主机内安装成功，则会将指定云主机转移出上述安全隔离区，以恢复指定云主机正常的网络访问

功能，从而使得指定云主机在恢复网络访问功能后可以进行正常的工作，保证用户在使用指定云主机时的体验感。进一步地，如果指定补丁安装失败，指定云主机会返回安装失败的补丁安装结果，则此时不会将指定云主机转移出上述安全隔离区，而会先查找出指定补丁安装失败的失败原因；然后根据该指定补丁安装失败的失败原因，控制该指定云主机对上述指定补丁进行重新下载与安装，并在确定出指定补丁在指定云主机内安装成功后，再将指定云主机转移出上述安全隔离区，使得只有在指定补丁安装成功的情况下，才会恢复指定云主机正常的网络访问功能，有效的保障了指定云主机的使用安全。

[0052] 本申请一实施例中，上述步骤S504之后，包括：

[0053] S5040：按照第一预设周期获取与所述指定云主机对应的第一宿主机的第一使用数据，其中，所述第一使用数据至少包括CPU使用率、内存使用率以及网络接口带宽使用率；

[0054] S5041：对所述第一使用数据进行分析处理，判断所述第一使用数据是否大于预设的标准阈值；

[0055] S5042：若是，从所有的宿主机中查找出第二使用数据小于所述标准阈值的第二宿主机；

[0056] S5043：将所述指定云主机从所述第一宿主机迁移至所述第二宿主机。

[0057] 如上述步骤S5040至S5043所述，在将指定云主机转移出上述安全隔离区后，在指定云主机的正常工作过程中，如果出现了指定云主机对应的第一宿主机处于高负荷状态的情况，宿主机集群还具有对于指定云主机的智能迁移的功能。具体地，在上述指定云主机启动完成后，按照第一预设周期获取与上述指定云主机对应的第一宿主机的第一使用数据，其中，对上述第一预设周期不作具体限定，例如可设为一个小时。另外，上述使用数据包括CPU使用率、内存使用率以及网络接口带宽使用率等。然后对该第一使用数据进行分析处理，并判断上述第一使用数据是否大于预设的标准阈值，其中，对于不同类型的第一使用数据，设置有该类型一一对应的相同数量的标准阈值。另外，对标准阈值的具体数值不作具体限定，可由宿主机集群设定，或者也可由用户根据需求进行设置。举例地，与CPU使用率对应的标准阈值可设为80%，与内存使用率对应的标准阈

值可设为75%，与网络接口带宽使用率的标准阈值可设为85%。如果上述第一使用数据大于预设的标准阈值，则表明与指定云主机对应的第一宿主机当前处于高负荷状态，第一宿主机的可计算资源较少。此时会从宿主机集群内的所有宿主机中查找出第二使用数据小于上述标准阈值的第二宿主机，并将指定云主机从上述第一宿主机迁移至上述第二宿主机，以实现将指定云主机从负载过高的第一宿主机智能的迁移到负载较低的第二宿主机，使得指定云主机的计算资源得到优化，从而有效的提升指定云主机的运行效率和响应速度，也保障了第一宿主机的使用安全。进一步地，如果第二宿主机的数量为多个，则从所有的第二宿主机中筛选出与第二使用数据对应的数据值最低的指定宿主机，再将上述指定云主机从上述第一宿主机迁移至上述指定宿主机。通过从所有的二宿主机中筛选出第二使用数据对应的数据值最低的指定宿主机，不仅可以提升指定云主机的运行效率和响应速度，还有利于实现对于所有宿主机的负载均衡。

[0058] 进一步地，本申请一实施例中，上述步骤S504之后，包括：

[0059] S5044：按照第二预设周期获取与所述指定云主机对应的云硬盘内的可用资源空间；

[0060] S5045：对所述可用资源空间进行分析处理，判断所述可用资源空间是否小于预设的资源空间阈值；

[0061] S5046：若是，筛选出所述云硬盘中的垃圾数据；

[0062] S5047：向所述指定云主机发送垃圾清除指令，以控制所述指定云主机根据所述垃圾清除指令，对所述垃圾数据进行清除处理。

[0063] 如上述步骤S5044至S5047所述，在将指定云主机转移出上述安全隔离区后，后续还会对处于正常工作状态的指定云主机中的可用资源空间进行监控处理，以保证指定云主机可具备充足的可用资源空间。具体地，在上述指定云主机启动后，按照第二预设周期获取与上述指定云主机对应的云硬盘内的可用资源空间，其中，对上述第二预设周期不作具体限定，例如可设为一个小时。然后判断上述可用资源空间是否小于预设的资源空间阈值，其中，对上述资源空间阈值的具体数值不作具体限定，可由宿主机集群设定，或者也可由用户根据需求进行设置，举例地，资源空间阈值可设为5g。如果可用资源空间小于预设的资源

空间阈值，表明指定云主机当前的可用资源空间较少，与指定云主机对应的云硬盘的使用压力较大。此时会筛选出上述云硬盘中的垃圾数据，其中，上述垃圾数据包括残余的指定云主机配置文件、无用的临时文件、已删除/卸载后残留的无用文件、硬盘和/或卷格式化未成功残留的文件等。在筛选出该垃圾数据后，再向上述指定云主机发送垃圾清除指令，以控制上述指定云主机根据该垃圾清除指令，对上述垃圾数据进行清除处理，具体可通过指定云主机上的管理工具或应用软件来实现对于上述垃圾数据的自动清除。本实施例在指定云主机当前的可用资源较少时，会自动并智能的对与指定云主机对应的云硬盘内的垃圾数据进行清除，有效的减少了指定云主机的压力和负担，释放指定云主机上多余、无用的资源，从而实现有序并及时地处理指定云主机对应的云硬盘内的垃圾数据，避免不必要的资源浪费，提高指定云主机内部资源的利用效率，最终提升指定云主机的运行效率和响应速度。

[0064] 参照图2，本申请一实施例中还提供了一种云主机的安全启动装置，包括：

[0065] 第一获取模块1，用于在接收到用户触发的对于指定云主机的启动指令时按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；

[0066] 第一判断模块2，用于判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

[0067] 第二获取模块3，用于若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

[0068] 第二判断模块4，用于根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

[0069] 隔离模块5，用于若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。

[0070] 本实施例中，上述云主机的安全启动装置中的第一获取模块、第一判断模块、第二获取模块、第二判断模块与隔离模块的功能和作用的实现过程具体详见上述云主机的安全启动方法中对应步骤S1至S5的实现过程，在此不再赘述。

[0071] 进一步地，本申请一实施例中，上述第一获取模块，包括：

[0072] 第一确定单元，用于确定所述指定云主机内是否安装有资产管理工具，其中，所述资产管理工具用于定期将所述指定云主机对应的资产信息同步至指定云平台；

[0073] 第一查找单元，用于若安装有资产管理工具，从所述指定云平台中查找出所述资产信息；

[0074] 第一调用单元，用于若未安装有资产管理工具，调用指定扫描工具对所述指定云主机进行扫描处理，以得到所述资产信息。

[0075] 本实施例中，上述云主机的安全启动装置中的第一确定单元、第一查找单元与第一调用单元的功能和作用的实现过程具体详见上述云主机的安全启动方法中对应步骤S100至S102的实现过程，在此不再赘述。

[0076] 进一步地，本申请一实施例中，上述云主机的安全启动装置，包括：

[0077] 抓取模块，用于通过爬虫工具，实时从目标网站抓取出与云主机对应的各类操作系统相关的第一漏洞信息，以及与云主机服务相关的第二漏洞信息；

[0078] 创建模块，用于创建一个漏洞数据库；

[0079] 第一存储模块，用于将所述第一漏洞信息与所述第二漏洞信息存储于所述漏洞数据库内。

[0080] 本实施例中，上述云主机的安全启动装置中的抓取模块、创建模块与第一存储模块的功能和作用的实现过程具体详见上述云主机的安全启动方法中对应步骤S200至S202的实现过程，在此不再赘述。

[0081] 进一步地，本申请一实施例中，上述云主机的安全启动装置，包括：

[0082] 第三获取模块，用于获取当前时间；

[0083] 第一确定模块，用于将所述上一次关机时间与所述当前时间之间所包含的时间段，确定为所述指定时间段。

[0084] 本实施例中，上述云主机的安全启动装置中的第三获取模块与第一确定模块的功能和作用的实现过程具体详见上述云主机的安全启动方法中对应步骤S400至S401的实现过程，在此不再赘述。

[0085] 进一步地，本申请一实施例中，上述云主机的安全启动装置，包括：

[0086] 第二确定模块，用于确定所述漏洞数据库中是否存储有与所述指定漏洞对应的

指定补丁；

[0087] 第二存储模块，用于若是，下载所述指定补丁，并通过指定传输方式将所述指定补丁存储于所述指定云主机内的指定存储区域中；

[0088] 第一发送模块，用于向所述指定云主机发送安装所述指定补丁的安装指令，以便所述指定云主机根据所述安装指令对所述指定补丁进行安装，并返回对应的补丁安装结果，其中，所述补丁安装结果包括安装成功或安装失败；

[0089] 第三判断模块，用于根据所述指定云主机返回的所述补丁安装结果，判断所述指定补丁是否安装成功；

[0090] 转移模块，用于若是，将所述指定云主机转移出所述安全隔离区，以恢复所述指定云主机正常的网络访问功能。

[0091] 本实施例中，上述云主机的安全启动装置中的第二确定模块、第二存储模块、第一发送模块、第三判断模块与转移模块的功能和作用的实现过程具体详见上述云主机的安全启动方法中对应步骤S500至S504的实现过程，在此不再赘述。

[0092] 进一步地，本申请一实施例中，上述云主机的安全启动装置，包括：

[0093] 第四获取模块，用于按照第一预设周期获取与所述指定云主机对应的第一宿主机的第一使用数据，其中，所述第一使用数据至少包括CPU使用率、内存使用率以及网络接口带宽使用率；

[0094] 第四判断模块，用于对所述第一使用数据进行分析处理，判断所述第一使用数据是否大于预设的标准阈值；

[0095] 查找模块，用于若是，从所有的宿主机中查找出第二使用数据小于所述标准阈值的第二宿主机；

[0096] 迁移模块，用于将所述指定云主机从所述第一宿主机迁移至所述第二宿主机。

[0097] 本实施例中，上述云主机的安全启动装置中的第四获取模块、第四判断模块、查找模块与迁移模块的功能和作用的实现过程具体详见上述云主机的安全启动方法中对应步骤S5040至S5043的实现过程，在此不再赘述。

[0098] 进一步地，本申请一实施例中，上述云主机的安全启动装置，包括：

[0099] 第五获取模块，用于按照第二预设周期获取与所述指定云主机对应的云硬盘内的可用资源空间；

[0100] 第五判断模块，用于对所述可用资源空间进行分析处理，判断所述可用资源空间是否小于预设的资源空间阈值；

[0101] 筛选模块，用于若是，筛选出所述云硬盘中的垃圾数据；

[0102] 第二发送模块，用于向所述指定云主机发送垃圾清除指令，以控制所述指定云主机根据所述垃圾清除指令，对所述垃圾数据进行清除处理。

[0103] 本实施例中，上述云主机的安全启动装置中的第五获取模块、第五判断模块、筛选模块与第二发送模块的功能和作用的实现过程具体详见上述云主机的安全启动方法中对应步骤S5044至S5047的实现过程，在此不再赘述。

[0104] 参照图3，本申请实施例中还提供一种计算机设备，该计算机设备可以是服务器，其内部结构可以如图3所示。该计算机设备包括通过系统总线连接的处理器、存储器、网络接口和数据库。其中，该计算机设备设计的处理器用于提供计算和控制能力。该计算机设备的存储器包括易失性或非易失性存储介质、内存储器。该易失性或非易失性存储介质存储有操作系统、计算机可读指令和数据库。该内存储器为易失性或非易失性存储介质中的操作系统和计算机可读指令的运行提供环境。该计算机设备的数据库用于存储资产信息、漏洞等数据。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机可读指令被处理器执行时以实现上述任一个示例性实施例所示出的云主机的安全启动方法。本领域技术人员可以理解，图3中示出的结构，仅仅是与本申请方案相关的部分结构的框图，并不构成对本申请方案所应用于其上的装置、计算机设备的限定。

[0105] 本申请一实施例还提供一种计算机可读存储介质，该计算机可读存储介质可以是易失性的，也可以是非易失性的，其上存储有计算机可读指令，该计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器执行时实现上述云主机的安全启动方法实施例中的步骤。

[0106] 综上所述，本申请实施例中提供的云主机的安全启动方法、装置、计算机设备和存储介质，在接收到用户触发的对于指定云主机的启动指令时，按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息

对应的漏洞，其中，所述漏洞的数量为一个或多个；若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。本申请当指定云主机在上一次关机期间时出现了新的与其相关的漏洞的情况下，会智能的将指定云主机放置在预创建的安全隔离区内进行启动，从而在指定云主机启动后暂时禁用指定云主机的网络访问功能，有效的避免出现指定云主机在使用过程中受到漏洞的入侵或攻击的情况，保障了指定云主机启动后的安全性。

[0107] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机可读指令来指令相关的硬件来完成，所述的计算机可读指令可存储于一非易失性或易失性计算机可读取存储介质中，该计算机可读指令在执行时，可包括如上述各方法的实施例的流程。其中，本申请所提供的和实施例中所使用的对存储器、存储、数据库或其它介质的任何引用，均可包括非易失性和/或易失性存储器。非易失性存储器可以包括只读存储器（ROM）、可编程ROM（PROM）、电可编程ROM（EPROM）、电可擦除可编程ROM（EEPROM）或闪存。易失性存储器可包括随机存取存储器（RAM）或者外部高速缓冲存储器。作为说明而非局限，RAM通过多种形式可得，诸如静态RAM（SRAM）、动态RAM（DRAM）、同步DRAM（SDRAM）、双速据率SDRAM（SSRSDRAM）、增强型SDRAM（ESDRAM）、同步链路（Synchlink）DRAM（SLDRAM）、存储器总线（Rambus）直接RAM（RDRAM）、直接存储器总线动态RAM（DRDRAM）、以及存储器总线动态RAM（RDRAM）等。

[0108] 以上所述仅为本申请的优选实施例，并非因此限制本申请的专利范围，凡是利用本申请说明书及附图内容所作的等效结构或等效流程变换，或直接或间接运用在其他相关的技术领域，均同理包括在本申请的专利保护范围内。

权利要求书

[权利要求 1] 一种云主机的安全启动方法，其特征在于，包括：

在接收到用户触发的对于指定云主机的启动指令时，按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；

判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。

[权利要求 2] 根据权利要求1所述的云主机的安全启动方法，其特征在于，所述按照预设规则获取与所述指定云主机对应的资产信息的步骤，包括：

确定所述指定云主机内是否安装有资产管理工具，其中，所述资产管理工具用于定期将所述指定云主机对应的资产信息同步至指定云平台；

若安装有资产管理工具，从所述指定云平台中查找出所述资产信息；

若未安装有资产管理工具，调用指定扫描工具对所述指定云主机进行扫描处理，以得到所述资产信息。

[权利要求 3] 根据权利要求1所述的云主机的安全启动方法，其特征在于，所述判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞的步骤之前，所述方法还包括：

通过爬虫工具，实时从目标网站抓取出与云主机对应的各类操作系统相关的第一漏洞信息，以及与云主机服务相关的第二漏洞信息；

创建一个漏洞数据库；

将所述第一漏洞信息与所述第二漏洞信息存储于所述漏洞数据库内。

[权利要求 4] 根据权利要求1所述的云主机的安全启动方法，其特征在于，所述根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞的步骤之前，所述方法还包括：

获取当前时间；

将所述上一次关机时间与所述当前时间之间所包含的时间段，确定为所述指定时间段。

[权利要求 5] 根据权利要求1所述的云主机的安全启动方法，其特征在于，所述在预创建的安全隔离区内启动所述指定云主机的步骤之后，所述方法还包括：

确定所述漏洞数据库中是否存储有与所述指定漏洞对应的指定补丁；

若是，下载所述指定补丁，并通过指定传输方式将所述指定补丁存储于所述指定云主机内的指定存储区域中；

向所述指定云主机发送安装所述指定补丁的安装指令，以便所述指定云主机根据所述安装指令对所述指定补丁进行安装，并返回对应的补丁安装结果，其中，所述补丁安装结果包括安装成功或安装失败；

根据所述指定云主机返回的所述补丁安装结果，判断所述指定补丁是否安装成功；

若是，将所述指定云主机转移出所述安全隔离区，以恢复所述指定云主机正常的网络访问功能。

[权利要求 6] 根据权利要求5所述的云主机的安全启动方法，其特征在于，所述将所述指定云主机转移出所述安全隔离区，以恢复所述指定云主机正常的网络访问功能的步骤之后，所述方法还包括：

按照第一预设周期获取与所述指定云主机对应的第一宿主机的第一使用数据，其中，所述第一使用数据至少包括CPU使用率、内存使用率以及网络接口带宽使用率；

对所述第一使用数据进行分析处理，判断所述第一使用数据是否大于预设的标准阈值；

若是，从所有的宿主机中查找出第二使用数据小于所述标准阈值的第二宿主机；

将所述指定云主机从所述第一宿主机迁移至所述第二宿主机。

[权利要求 7] 根据权利要求5所述的云主机的安全启动方法，其特征在于，所述将所述指定云主机转移出所述安全隔离区，以恢复所述指定云主机正常的网络访问功能的步骤之后，所述方法还包括：

按照第二预设周期获取与所述指定云主机对应的云硬盘内的可用资源空间；

对所述可用资源空间进行分析处理，判断所述可用资源空间是否小于预设的资源空间阈值；

若是，筛选出所述云硬盘中的垃圾数据；

向所述指定云主机发送垃圾清除指令，以控制所述指定云主机根据所述垃圾清除指令，对所述垃圾数据进行清除处理。

[权利要求 8] 一种云主机的安全启动装置，其特征在于，包括：

第一获取模块，用于在接收到用户触发的对于指定云主机的启动指令时按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；

第一判断模块，用于判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

第二获取模块，用于若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

第二判断模块，用于根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

隔离模块，用于若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。

[权利要求 9] 根据权利要求8所述的云主机的安全启动装置，其特征在于，所述第

一获取模块，包括：

第一确定单元，用于确定所述指定云主机内是否安装有资产管理工具，其中，所述资产管理工具用于定期将所述指定云主机对应的资产信息同步至指定云平台；

第一查找单元，用于若安装有资产管理工具，从所述指定云平台中查找出所述资产信息；

第一调用单元，用于若未安装有资产管理工具，调用指定扫描工具对所述指定云主机进行扫描处理，以得到所述资产信息。

[权利要求 10] 根据权利要求8所述的云主机的安全启动装置，其特征在于，所述装置包括：

抓取模块，用于通过爬虫工具，实时从目标网站抓取出与云主机对应的各类操作系统相关的第一漏洞信息，以及与云主机服务相关的第二漏洞信息；

创建模块，用于创建一个漏洞数据库；

第一存储模块，用于将所述第一漏洞信息与所述第二漏洞信息存储于所述漏洞数据库内。

[权利要求 11] 根据权利要求8所述的云主机的安全启动装置，其特征在于，所述装置包括：

第三获取模块，用于获取当前时间；

第一确定模块，用于将所述上一次关机时间与所述当前时间之间所包含的时间段，确定为所述指定时间段。

[权利要求 12] 根据权利要求8所述的云主机的安全启动装置，其特征在于，所述装置包括：

第二确定模块，用于确定所述漏洞数据库中是否存储有与所述指定漏洞对应的指定补丁；

第二存储模块，用于若是，下载所述指定补丁，并通过指定传输方式将所述指定补丁存储于所述指定云主机内的指定存储区域中；

第一发送模块，用于向所述指定云主机发送安装所述指定补丁的安装

指令，以便所述指定云主机根据所述安装指令对所述指定补丁进行安装，并返回对应的补丁安装结果，其中，所述补丁安装结果包括安装成功或安装失败；

第三判断模块，用于根据所述指定云主机返回的所述补丁安装结果，判断所述指定补丁是否安装成功；

转移模块，用于若是，将所述指定云主机转移出所述安全隔离区，以恢复所述指定云主机正常的网络访问功能。

[权利要求 13] 根据权利要求12所述的云主机的安全启动装置，其特征在于，所述装置包括：

第四获取模块，用于按照第一预设周期获取与所述指定云主机对应的第一宿主机的第一使用数据，其中，所述第一使用数据至少包括CPU使用率、内存使用率以及网络接口带宽使用率；

第四判断模块，用于对所述第一使用数据进行分析处理，判断所述第一使用数据是否大于预设的标准阈值；

查找模块，用于若是，从所有的宿主机中查找出第二使用数据小于所述标准阈值的第二宿主机；

迁移模块，用于将所述指定云主机从所述第一宿主机迁移至所述第二宿主机。

[权利要求 14] 根据权利要求12所述的云主机的安全启动装置，其特征在于，所述装置包括：

第五获取模块，用于按照第二预设周期获取与所述指定云主机对应的云硬盘内的可用资源空间；

第五判断模块，用于对所述可用资源空间进行分析处理，判断所述可用资源空间是否小于预设的资源空间阈值；

筛选模块，用于若是，筛选出所述云硬盘中的垃圾数据；

第二发送模块，用于向所述指定云主机发送垃圾清除指令，以控制所述指定云主机根据所述垃圾清除指令，对所述垃圾数据进行清除处理 。

[权利要求 15] 一种计算机设备，包括存储器和处理器，所述存储器存储有计算机可读指令，其特征在于，所述处理器执行所述计算机可读指令时实现云主机的安全启动方法，该云主机的安全启动方法，包括：

在接收到用户触发的对于指定云主机的启动指令时，按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；

判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。

[权利要求 16] 根据权利要求15所述的计算机设备，其特征在于，所述按照预设规则获取与所述指定云主机对应的资产信息的步骤，包括：

确定所述指定云主机内是否安装有资产管理工具，其中，所述资产管理工具用于定期将所述指定云主机对应的资产信息同步至指定云平台；

若安装有资产管理工具，从所述指定云平台中查找出所述资产信息；

若未安装有资产管理工具，调用指定扫描工具对所述指定云主机进行扫描处理，以得到所述资产信息。

[权利要求 17] 根据权利要求15所述的计算机设备，其特征在于，所述判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞的步骤之前，所述方法还包括：

通过爬虫工具，实时从目标网站抓取出与云主机对应的各类操作系统相关的第一漏洞信息，以及与云主机服务相关的第二漏洞信息；

创建一个漏洞数据库；

将所述第一漏洞信息与所述第二漏洞信息存储于所述漏洞数据库内。

[权利要求 18] 一种计算机可读存储介质，其上存储有计算机可读指令，其特征在于，所述计算机可读指令被处理器执行时实现云主机的安全启动方法，该云主机的安全启动方法，包括：

在接收到用户触发的对于指定云主机的启动指令时，按照预设规则获取与所述指定云主机对应的资产信息，其中，所述资产信息至少包括操作系统信息；

判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞，其中，所述漏洞的数量为一个或多个；

若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间；

根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞；

若是，在预创建的安全隔离区内启动所述指定云主机，其中，所述指定云主机的网络访问功能在所述安全隔离区内会被禁用。

[权利要求 19] 根据权利要求18所述的计算机可读存储介质，其特征在于，所述按照预设规则获取与所述指定云主机对应的资产信息的步骤，包括：

确定所述指定云主机内是否安装有资产管理工具，其中，所述资产管理工具用于定期将所述指定云主机对应的资产信息同步至指定云平台；

若安装有资产管理工具，从所述指定云平台中查找出所述资产信息；

若未安装有资产管理工具，调用指定扫描工具对所述指定云主机进行扫描处理，以得到所述资产信息。

[权利要求 20] 根据权利要求18所述的计算机可读存储介质，其特征在于，所述判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞的步骤之前，所述方法还包括：

通过爬虫工具，实时从目标网站抓取出与云主机对应的各类操作系统相关的第一漏洞信息，以及与云主机服务相关的第二漏洞信息；

创建一个漏洞数据库；

将所述第一漏洞信息与所述第二漏洞信息存储于所述漏洞数据库内。

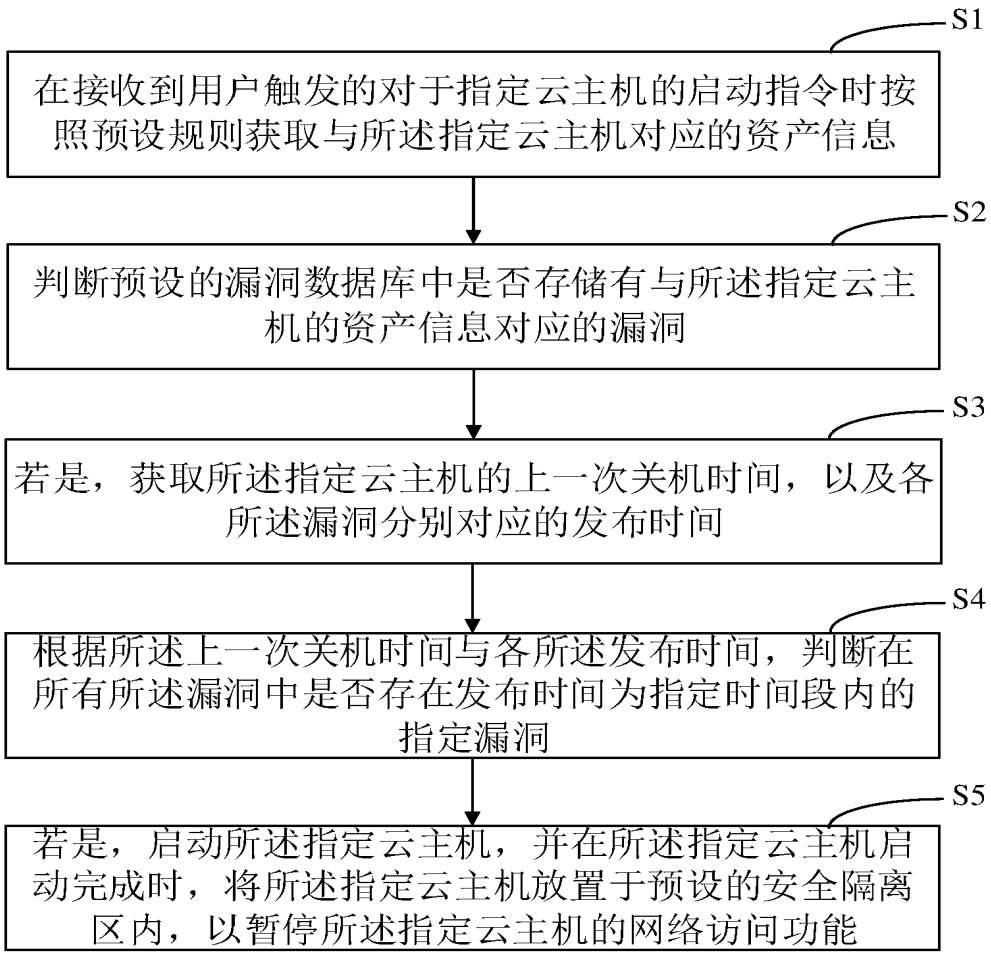

S1
在接收到用户触发的对于指定云主机的启动指令时按照预设规则获取与所述指定云主机对应的资产信息
S2
判断预设的漏洞数据库中是否存储有与所述指定云主机的资产信息对应的漏洞
S3
若是，获取所述指定云主机的上一次关机时间，以及各所述漏洞分别对应的发布时间
S4
根据所述上一次关机时间与各所述发布时间，判断在所有所述漏洞中是否存在发布时间为指定时间段内的指定漏洞
S5
若是，启动所述指定云主机，并在所述指定云主机启动完成时，将所述指定云主机放置于预设的安全隔离区内，以暂停所述指定云主机的网络访问功能

图 1

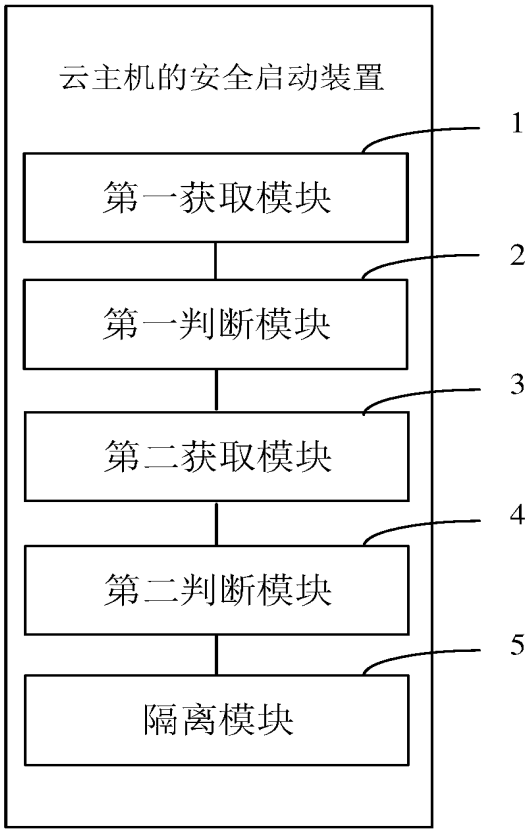

云主机的安全启动装置
1
第一获取模块
2
第一判断模块
3
第二获取模块
4
第二判断模块
5
隔离模块

图 2

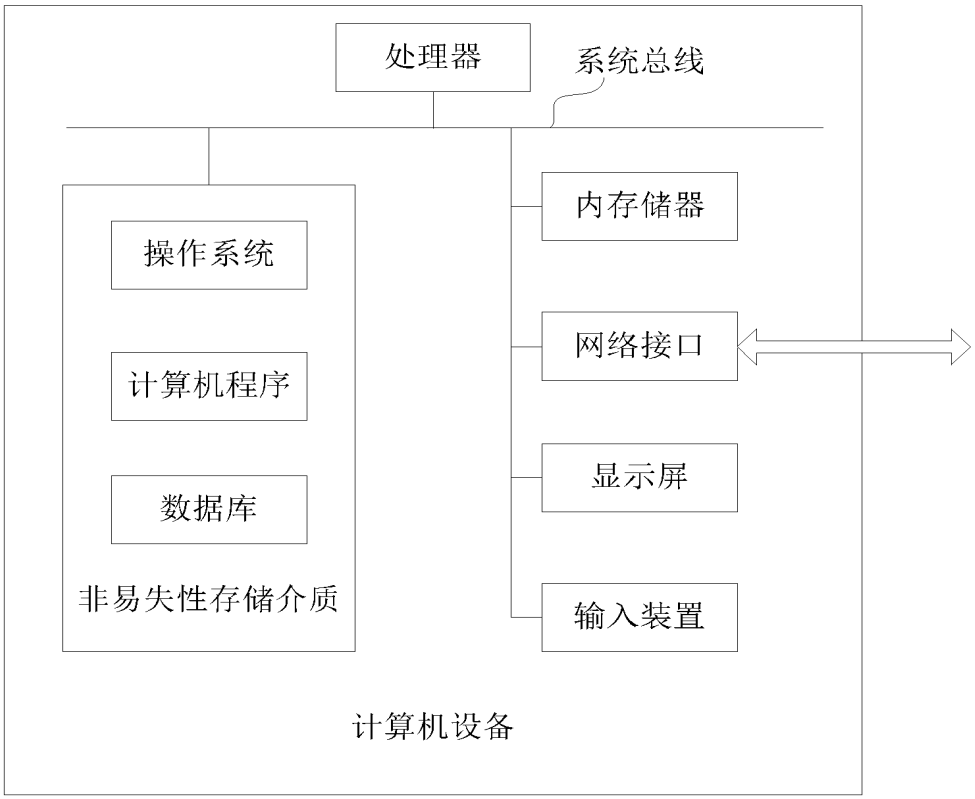

处理器
系统总线
内存储器
操作系统
网络接口
计算机程序
显示屏
数据库
非易失性存储介质
输入装置
计算机设备

图 3

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2019/118430

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/4401(2018.01)i; G06F 9/455(2006.01)i; G06F 8/61(2018.01)i; G06F 21/53(2013.01)i; G06F 21/57(2013.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS; CNTXT; CNKI; SIPOABS; DWPI; USTXT; WOTXT; EPTXT: 云, 主机, 启动, 漏洞, 虚拟机, 操作系统, 关机, 发布, 时间, 隔离区, 安全环境, 沙箱, 沙盒, cloud, startup, bug, VM, VZ, VPS, OS, shut, down, release, time, security, environment, sandbox, insulate, separate

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 105204902 A (HUAWEI TECHNOLOGIES CO., LTD.) 30 December 2015 (2015-12-30) description, paragraphs [0043]-[0129]	1-20
A	CN 110059007 A (BEIJING QIANXIN TECHNOLOGY CO., LTD.) 26 July 2019 (2019-07-26) entire document	1-20
A	US 2018053001 A1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 22 February 2018 (2018-02-22) entire document	1-20

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
27 May 2020	**08 July 2020**
Name and mailing address of the ISA/CN **China National Intellectual Property Administration (ISA/CN)** **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China** Facsimile No. **(86-10)62019451**	Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2019/118430

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	105204902	A	30 December 2015	CN	105204902	B	07 December 2018
CN	110059007	A	26 July 2019	None			
US	2018053001	A1	22 February 2018	US	10460113	B2	29 October 2019

国际检索报告

国际申请号
PCT/CN2019/118430

A. 主题的分类

G06F 9/4401(2018.01)i; G06F 9/455(2006.01)i; G06F 8/61(2018.01)i; G06F 21/53(2013.01)i; G06F 21/57(2013.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用）)

CNABS;CNTXT;CNKI;SIPOABS;DWPI;USTXT;WOTXT;EPTXT：云，主机，启动，漏洞，虚拟机，操作系统，关机，发布，时间，隔离区，安全环境，沙箱，沙盒，cloud，startup，bug，VM，VZ，VPS，OS，shut，down，release，time，security，environment，sandbox，insulate，separate

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
X	CN 105204902 A (华为技术有限公司) 2015年 12月 30日 (2015-12-30) 说明书第[0043]-[0129]段	1-20
A	CN 110059007 A (北京奇安信科技有限公司) 2019年 7月 26日 (2019-07-26) 全文	1-20
A	US 2018053001 A1 (IBM) 2018年 2月 22日 (2018-02-22) 全文	1-20

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：
“A” 认为不特别相关的表示了现有技术一般状态的文件
“E” 在国际申请日的当天或之后公布的在先申请或专利
“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）
“O” 涉及口头公开、使用、展览或其他方式公开的文件
“P” 公布日先于国际申请日但迟于所要求的优先权日的文件
“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件
“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性
“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性
“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2020年 5月 27日	2020年 7月 8日
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 潘秋羽 电话号码 (86-512) 88995784

PCT/ISA/210 表(第2页) (2015年1月)

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2019/118430

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	105204902	A	2015年 12月 30日	CN	105204902	B	2018年 12月 7日
CN	110059007	A	2019年 7月 26日		无		
US	2018053001	A1	2018年 2月 22日	US	10460113	B2	2019年 10月 29日

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	11/09/2022 03:20:36 PM ET	15635.0005-00000

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61161465	AUTHORIZED BY	Christopher Johns
CUSTOMER #	22852	FILING DATE	04/05/2022
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Avi SHUA

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
CARD / 1007	E2022A9F21123656	Michelle Daniels

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
1806	SUBMISSION OF AN INFORMATION DISCLOSURE STATEMENT	260.00	1	260.00
			TOTAL AMOUNT:	$260.00

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**11/09/2022 03:20:36 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61161465	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 5

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
IDS after FINAL Office Action.pdf	3	Transmittal Letter	19 KB
sb08.pdf	1	Information Disclosure Statement (IDS) Form (SB08)	26 KB
WO2021072877A1_FOR.pdf	44	Foreign Reference	3036 KB
CN110708315A_FOR.pdf	24	Foreign Reference	5243 KB
Ex Parte Quayle Action-10-31-2022-15635.0008-00000_NPL.pdf	82	Non Patent Literature	2749 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
IDS after FINAL Office Action.pdf	88678D463DEDFC808B61EEDD4A3CF42CA3381903D1D6CC27EA24AF8BC31BD382F46FF1856FBADEC5DCA8211F2C9E08F691BEA7F62CBE87D4E81C08FD7F67EBEF
sb08.pdf	665EBB33831D7BFB86ADD5C6C7F8D656B1F319B00DF5FAA9037F65A89040D22F29F71118BC768E29EF64DF67ADDB391AB397ECC507D6F52338A2E17C86948940
WO2021072877A1_FOR.pdf	03C361B23C37D6FC9ACFE9AD6DD5FE35F823E1F44C0A0897525D56FE71ABFFE958B1DDADD03ACBABE5F8B8C90865EF79073444F4A7C62519B0609C388CF032C6
CN110708315A_FOR.pdf	50F9A5DE497ECB1FF12DD73F6FD6B6094F14FEEEFF6B9F85A569A4CAEBAD0D0C622D4A8FF2F8018C7071988FFCF14DF82217D6E5E8033A9D9607528ECC184D37
Ex Parte Quayle Action-10-31-2022-15635.0008-00000_NPL.pdf	2059F34E3274F9EC2E4A046AA41C435B24544EF6349CFB03A8160A7C3745937CA1BF0138A8E5E8C47BA4826AA2D07B30564BBFDE4C52686D23C20AFC7542C8F2

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/657,972	04/05/2022	Avi SHUA	15635.0005-00000	2547

22852 7590 10/31/2022

FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

EXAMINER	
SU, SARAH	

ART UNIT	PAPER NUMBER
2431	

NOTIFICATION DATE	DELIVERY MODE
10/31/2022	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

regional-desk@finnegan.com

Advisory Action Before the Filing of an Appeal Brief	**Application No.** 17/657,972	**Applicant(s)** SHUA, Avi	
	Examiner SARAH SU	**Art Unit** 2431	**AIA (FITF) Status** Yes

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

THE REPLY FILED 13 October 2022 FAILS TO PLACE THIS APPLICATION IN CONDITION FOR ALLOWANCE.

NO NOTICE OF APPEAL FILED

1. ☑ The reply was filed after a final rejection. No Notice of Appeal has been filed. To avoid abandonment of this application, applicant must timely file one of the following replies: (1) an amendment, affidavit, or other evidence, which places the application in condition for allowance; (2) a Notice of Appeal (with appeal fee) in compliance with 37 CFR 41.31; or (3) a Request for Continued Examination (RCE) in compliance with 37 CFR 1.114 if this is a utility or plant application. Note that RCEs are not permitted in design applications. The reply must be filed within one of the following time periods:
   a) ☐ The period for reply expires ____ months from the mailing date of the final rejection.
   b) ☑ The period for reply expires on: (1) the mailing date of this Advisory Action; or (2) the date set forth in the final rejection, whichever is later. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of the final rejection.
   c) ☐ A prior Advisory Action was mailed more than 3 months after the mailing date of the final rejection in response to a first after-final reply filed within 2 months of the mailing date of the final rejection.The current period for reply expires ____ months from the mailing date of *the prior Advisory Action* or SIX MONTHS from the mailing date of the final rejection, whichever is earlier.
   Examiner Note: If box 1 is checked, check either box (a), (b) or (c). ONLY CHECK BOX (b) WHEN THIS ADVISORY ACTION IS THE FIRST RESPONSE TO APPLICANTS FIRST AFTER-FINAL REPLY WHICH WAS FILED WITHIN TWO MONTHS OF THE FINAL REJECTION. ONLY CHECK BOX (c) IN THE LIMITED SITUATION SET FORTH UNDER BOX (c). See MPEP 706.07(f).

Extensions of time may be obtained under 37 CFR 1.136(a). The date on which the petition under 37 CFR 1.136(a) and the appropriate extension fee have been filed is the date for purposes of determining the period of extension and the corresponding amount of the fee. The appropriate extension fee under 37 CFR 1.17(a) is calculated from: (1) the expiration date of the shortened statutory period for reply originally set in the final Office action; or (2) as set forth in (b) or (c) above, if checked. Any reply received by the Office later than three months after the mailing date of the final rejection, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

NOTICE OF APPEAL

2. ☐ The Notice of Appeal was filed on _____. A brief in compliance with 37 CFR 41.37 must be filed within two months of the date of filing the Notice of Appeal (37 CFR 41.37(a)), or any extension thereof (37 CFR 41.37(e)), to avoid dismissal of the appeal. Since a Notice of Appeal has been filed, any reply must be filed within the time period set forth in 37CFR 41.37(a).

AMENDMENTS

3. ☑ The proposed amendments filed after a final rejection, but prior to the date of filing a brief, will not be entered because
   a) ☑ They raise new issues that would require further consideration and/or search (see NOTE below);
   b) ☐ They raise the issue of new matter (see NOTE below);
   c) ☐ They are not deemed to place the application in better form for appeal by materially reducing or simplifying the issues for appeal; and/or
   d) ☐ They present additional claims without canceling a corresponding number of finally rejected claims.
   NOTE: See Continuation Sheet (See 37 CFR 1.116 and 41.33(a)).
4. ☐ The amendments are not in compliance with 37 CFR 1.121. See attached Notice of Non-Compliant Amendment (PTOL-324).
5. ☐ Applicant's reply has overcome the following rejection(s): _____
6. ☐ Newly proposed or amended claim(s) _____ would be allowable if submitted in a separate, timely filed amendment canceling the non-allowable claim(s).
7. ☑ For purposes of appeal, the proposed amendment(s):(a) ☑ will not be entered, or (b) ☐ will be entered, and an explanation of how the new or amended claims would be rejected is provided below or appended.

AFFIDAVIT OR OTHER EVIDENCE

8. ☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____
9. ☐ The affidavit or other evidence filed after final action, but before or on the date of filing a Notice of Appeal will not be entered because applicant failed to provide a showing of good and sufficient reasons why the affidavit or other evidence is necessary and was not earlier presented. See 37 CFR 1.116(e).
10. ☐ The affidavit or other evidence filed after the date of filing the Notice of Appeal, but prior to the date of filing a brief, will not be entered because the affidavit or other evidence failed to overcome all rejections under appeal and/or appellant fails to provide a showing of good and sufficient reasons why it is necessary and was not earlier presented. See 37 CFR 41.33(d)(1).
11. ☐ The affidavit or other evidence is entered. An explanation of the status of the claims after entry is below or attached.

REQUEST FOR RECONSIDERATION/OTHER

12. ☐ The request for reconsideration has been considered but does NOT place the application in condition for allowance because: _____.
13. ☐ Note the attached Information *Disclosure Statement*(s). (PTO/SB/08) Paper No(s). _____
14. ☑ Other: See attached PTO-2323; See attached Interview Summary.

STATUS OF CLAIMS

15. The status of the claim(s) is (or will be) as follows:
    Claim(s) allowed:_____.
    Claim(s) objected to:_____.
    Claim(s) rejected:21-40.
    Claim(s) withdrawn from consideration:_____.

/SARAH SU/ Primary Examiner, Art Unit 2431	

Continuation of 3. NOTE: The element "perform at least one of: identify at least one port on which the vulnerable application is accessible; or use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload" in claim 21, and similarly in claims 33 and 40, would require further consideration and searching. A brief prior art search has found, for example, that Stopel et al. (US 2019/0116199 A1) teaches that at least one port through which an application is accessible is identified where vulnerability exposure is determined (abstract, lines 2-14).

Applicant-Initiated Interview Summary	**Application No.** 17/657,972	**Applicant(s)** SHUA, Avi		
	Examiner SARAH SU	**Art Unit** 2431	**AIA (First Inventor to File) Status** Yes	**Page** 1 of 1

All Participants (applicant, applicants representative, PTO personnel)	**Title**	**Type**
SARAH SU	Primary Examiner	Telephonic
Christopher Johns (68,664)	Attorney	

Date of Interview: 25 October 2022

Issues Discussed:

Other

The submitted after final amendment was discussed. See attached advisory action.

/SARAH SU/ Primary Examiner, Art Unit 2431	

Applicant is reminded that a complete written statement as to the substance of the interview must be made of record in the application file. It is the applicants responsibility to provide the written statement, unless the interview was initiated by the Examiner and the Examiner has indicated that a written summary will be provided. See MPEP 713.04
Please further see:
MPEP 713.04
Title 37 Code of Federal Regulations (CFR) § 1.133 Interviews, paragraph (b)
37 CFR § 1.2 Business to be transacted in writing

Applicant recordation instructions: The formal written reply to the last Office action must include the substance of the interview. (See MPEP section 713.04). If a reply to the last Office action has already been filed, applicant is given a non-extendable period of the longer of one month or thirty days from this interview date, or the mailing date of this interview summary form, whichever is later, to file a statement of the substance of the interview.

Examiner recordation instructions: Examiners must summarize the substance of any interview of record. A complete and proper recordation of the substance of an interview should include the items listed in MPEP 713.04 for complete and proper recordation including the identification of the general thrust of each argument or issue discussed, a general indication of any other pertinent matters discussed regarding patentability and the general results or outcome of the interview, to include an indication as to whether or not agreement was reached on the issues raised.

AFCP 2.0 Decision	Application No.	Applicant(s)	
	17/657,972	SHUA, Avi	
	Examiner	Art Unit	AIA (FITF) Status
	SARAH SU	2431	Yes

This is in response to the After Final Consideration Pilot request filed 13 October 2022.

1. **Improper Request** – The AFCP 2.0 request is improper for the following reason(s) and the after final amendment submitted with the request will be treated under pre-pilot procedure.

   ☐ An AFCP 2.0 request form PTO/SB/434 (or equivalent document) was not submitted.

   ☐ A non-broadening amendment to at least one independent claim was not submitted.

   ☐ The request is not the first proper AFCP 2.0 request submitted in response to the most recent final rejection.

   ☐ Other: _____

2. **Proper Request**

   A. After final amendment submitted with the request will not be treated under AFCP 2.0.
   The after final amendment cannot be reviewed and a search conducted within the guidelines of the pilot program.

   ☐ The after final amendment will be treated under pre-pilot procedure.

   B. Updated search and/or completed additional consideration.
   The examiner performed an updated search and/or completed additional consideration of the after final amendment within the time authorized for the pilot program. The result(s) of the updated search and/or completed additional consideration are:

   ☐ 1. All of the rejections in the most recent final Office action are overcome and a Notice of Allowance is issued herewith.

   ☐ 2. The after final amendment would not overcome all of the rejections in the most recent final Office action . See attached interview summary for further details.

   ☑ 3. The after final amendment was reviewed, and it raises a new issue(s). See attached interview summary for further details.

   ☐ 4. The after final amendment raises new issues, but would overcome all of the rejections in the most recent final Office action. A decision on determining allowability could not be made within the guidelines of the pilot. See attached interview summary for further details, including any newly discovered prior art.

   ☐ 5. Other: _____

   Examiner Note: Please attach an interview summary when necessary as described above.

PATENT
Attorney Docket No.15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.:17/657,972	)	Examiner: Sarah Su
	)	
Filed: April 5, 2022	)	
	)	
For: **Systems and Methods for Managing Cyber Vulnerabilities**	)	Confirmation No.: 2547
	)	

DO NOT ENTER: /S.S/

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

AMENDMENT AFTER FINAL UNDER AFCP 2.0

In reply to the Final Office Action mailed on September 1, 2022, Applicant responds the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims in this paper.

Remarks follow the amendment sections of this paper.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371(C) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
17/657,972	04/05/2022	Avi SHUA	15635.0005-00000

CONFIRMATION NO. 2547

PUBLICATION NOTICE

22852
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

OC00000136698054

Title:SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Publication No.US-2022-0345480-A1
Publication Date:10/27/2022

NOTICE OF PUBLICATION OF APPLICATION

The above-identified application will be electronically published as a patent application publication pursuant to 37 CFR 1.211, et seq. The patent application publication number and publication date are set forth above.

The publication may be accessed through the USPTO's publically available Searchable Databases via the Internet at www.uspto.gov. The direct link to access the publication is currently http://www.uspto.gov/patft/.

The publication process established by the Office does not provide for mailing a copy of the publication to applicant. A copy of the publication may be obtained from the Office upon payment of the appropriate fee set forth in 37 CFR 1.19(a)(1). Orders for copies of patent application publications are handled by the USPTO's Public Records Division. The Public Records Division can be reached by telephone at (571) 272-3150 or (800) 972-6382, by facsimile at (571) 273-3250, by mail addressed to the United States Patent and Trademark Office, Public Records Division, Alexandria, VA 22313-1450 or via the Internet.

In addition, information on the status of the application, including the mailing date of Office actions and the dates of receipt of correspondence filed in the Office, may also be accessed via the Internet through the Patent Electronic Business Center at www.uspto.gov using the public side of the Patent Application Information and Retrieval (PAIR) system. The direct link to access this status information is currently https://portal.uspto.gov/pair/PublicPair. Prior to publication, such status information is confidential and may only be obtained by applicant using the private side of PAIR.

Further assistance in electronically accessing the publication, or about PAIR, is available by calling the Patent Electronic Business Center at 1-866-217-9197.

Office of Data Managment, Application Assistance Unit (571) 272-4000, or (571) 272-4200, or 1-888-786-0101

INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				***Complete if Known***	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	1	*Attorney Docket Number*	**15635.0005-00000**

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-8,621,073-B2	12-31-2013	McClure et al.	

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2018/0288087-A1	10-04-2018	HITTEL et al.	
		US-2017/0012998-A1	01-12-2017	Thornbury et al.	

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Foreign Patent Document Country Code Number Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	Translation

NON PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Final Office Action Application No. 17/658,156 dated September 13, 2022, in the United States Patent and Trademark Office. (15635.0010-00000)	N/A
		Notice of Allowance No. 17/716,491 dated October 11, 2022, in the United States Patent and Trademark Office. (15635.0006-00000)	N/A
		Notice of Allowance No. 17/716,231 dated September 6, 2022, in the United States Patent and Trademark Office. (15635.0004-00000)	N/A
		Ex parte Quayle Action No. 17/716,231 dated July 26, 2022, in the United States Patent and Trademark Office. (15635.0004-00000)	N/A

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

PATENT
Attorney Docket No. 15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.: 17/657,972	)	Examiner: SU, SARAH
	)	
Filed: April 5, 2022	)	Confirmation No.: 2547
	)	
For: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT UNDER 37 C.F.R. § 1.97(d)

Pursuant to 37 C.F.R. §§ 1.56 and 1.97(d), Applicant brings to the attention of the Examiner the documents on the attached listing. This Information Disclosure Statement is being filed after a Final Office Action and is accompanied by a fee of $260.00 as specified under § 1.17(p) and a statement as specified under § 1.97(e).

Statement under 37 C.F.R. § 1.97(e)

The following constitutes the statement specified by 37 C.F.R. § 1.97(e)(2). No item of information contained in the IDS was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the IDS was known to any individual designated in 37 CFR 1.56(c) more

than three months prior to the filing of this Information Disclosure Statement. See 37 CFR 1.97(e)(2).

A copy of the non-patent literature documents is attached.

Applicant respectfully requests that the Examiner consider the listed documents on the attached Form PTO/SB/08 and indicate that they were considered by making appropriate notations on the attached form.

This submission does not represent that a search has been made or that no better art exists and does not constitute an admission that each or all of the listed documents are material or constitute "prior art." If the Examiner applies any of the documents as prior art against any claim in the application and Applicant determines that the cited documents do not constitute "prior art" under United States law, Applicant reserves the right to present to the U.S. Patent and Trademark Office the relevant facts and law regarding the appropriate status of such documents.

Applicant further reserves the right to take appropriate action to establish the patentability of the disclosed invention over the listed documents, should one or more of the documents be applied against the claims of the present application.

If there is any fee due in connection with the filing of this Statement, please charge the fee to deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: 25 October 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4155

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**10/25/2022 05:19:19 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61105060	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 6

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
IDS after FINAL Office Action.pdf	3	Transmittal Letter	19 KB
sb08.pdf	1	Information Disclosure Statement (IDS) Form (SB08)	26 KB
Final Office Action dated 09-13-2022 -17 658,156_NPL.pdf	21	Non Patent Literature	1317 KB
Notice of Allowance dated 10-11-22 - 17 716,491_NPL.pdf	11	Non Patent Literature	829 KB
Notice of Allowability 09-06-	9	Non Patent Literature	357 KB

22 -17 716,231_NPL.pdf			
Ex Parte Quayle Action dated - 07-26-22 - 17 716,231_NPL.pdf	9	Non Patent Literature	452 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
IDS after FINAL Office Action.pdf	C4CCB7BC88B8C0EF14743EA778E06EE6EE40B8DBFEB6B623 35F24FACB0055B086C5AB03759338B9B11D53FDE1688C6F8E 5DD0072282DD57C5A7F0D8BEAE63BFC
sb08.pdf	56CDCEC8778AAFDFAEECA2337080B8008A42D043E4DB128B 9CC6D63B4420FCB96C078E8A4DD15D0BEE0F5DCE82CDD8A EA750BEA4EF9CC62BA1FB9FF469995BEB
Final Office Action dated 09-13-2022 -17 658,156_NPL.pdf	DA1C8F743A4E567DE0EF1F34C03940DBB730169C3E11A55F0 8F83505D531BE3C9A4CD41D9A442832F450CA757975D9CB74 BF3441128B2DEA2E27200F08FFBDFD
Notice of Allowance dated 10-11-22 - 17 716,491_NPL.pdf	E04EC1C51D367450510673181F166FBFEE4E783B0131F6CBC F744BE53F831D3C5DDAC81D6B5B7EAB913882D52683D25B3 FF2EE32B2285FC9D8E035363068DAA5
Notice of Allowability 09-06-22 - 17 716,231_NPL.pdf	85978E8718401FC06B9DB93C55A71FFD05C96B93C07D10ABB C35435058EBDDE1529863E85EFB5802DD17FF96B585B121D5 74D8C913E76CE550A1DA73A4D4CF1D
Ex Parte Quayle Action dated - 07-26-22 - 17 716,231_NPL.pdf	B5F09983761C777884AC43965FDFF4AB54B40B08A36BB6F37 C56E106364BCB7FEC045AFCA9BA9915DD28195C916D226042 FD35EA803A9A112348CD01D2504926

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	10/25/2022 05:19:19 PM ET	15635.0005-00000

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	61105060	AUTHORIZED BY	Christopher Johns
CUSTOMER #	22852	FILING DATE	04/05/2022
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Avi SHUA

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
CARD / 1007	E20220OH20344614	Michelle Daniels

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
1806	SUBMISSION OF AN INFORMATION DISCLOSURE STATEMENT	260.00	1	260.00
			TOTAL AMOUNT:	$260.00

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PATENT
Attorney Docket No.15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.:17/657,972	)	Examiner: Sarah Su
	)	
Filed: April 5, 2022	)	
	)	
For: **Systems and Methods for Managing Cyber Vulnerabilities**	)	Confirmation No.: 2547
	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

AMENDMENT AFTER FINAL UNDER AFCP 2.0

In reply to the Final Office Action mailed on September 1, 2022, Applicant responds the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims in this paper.

Remarks follow the amendment sections of this paper.

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1-20. (canceled)

21. (Currently Amended) A cyber security system for a cloud environment, the system comprising:

at least one processor configured to:

using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment;

identify an installed software application in the accessed block storage volume;

analyze the identified installed software application to determine an associated software version;

access a data structure of known software vulnerabilities for a plurality of versions of software applications;

perform a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and

perform at least one of:

~~query the cloud provider API to determine network accessibility information related to the workload;~~

identify at least one port on which the vulnerable application is accessible; or

use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

22. (Previously Presented) The system of claim 21, wherein the at least one processor is further configured to implement a remedial action in response to the identified one or more vulnerabilities.

23. (Previously Presented) The system of claim 22, wherein the remedial action includes transmitting an alert to a device associated with an administrator.

24. (Currently Amended) The system of claim 21, wherein:

querying the cloud provider API to determine network accessibility information related to the workload further comprises examining data sources associated with the workload; and

determining the network accessibility information based on the examined data sources.

25. (Currently Amended) The system of claim 21, wherein the at least one processor is further configured to ~~querying~~ query the cloud provider API to determine network accessibility information related to the workload and further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.

26. (Previously Presented) The system of claim 21, wherein to identify the installed software application, the at least one processor is configured to:

extract data from at least one of operating system packages, libraries, or program language libraries; and

identify the installed software application based on the extracted data.

27. (Previously Presented) The system of claim 21, wherein the at least one processor is further configured to identify a version of the installed software application.

28. (Previously Presented) The system of claim 21, wherein the identified installed software application includes one or more scripts.

29. (Previously Presented) The system of claim 21, wherein the data structure includes aggregated vulnerability data.

30. (Previously Presented) The system of claim 29, wherein the aggregated vulnerability data includes data from one or more third-party vendors.

31. (Currently Amended) The system of claim 29, wherein the aggregated vulnerability data includes data collected by a scanner.

32. (Currently Amended) The system of claim 29, wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug.

33. (Currently Amended) A method, comprising:

using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment;

identifying an installed software application in the accessed block storage volume;

analyzing the identified installed software application to determine an associated software version;

accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;

performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and

perform at least one of the following:

~~query the cloud provider API to determine network accessibility information related to the workload;~~

identify at least one port on which the vulnerable application is accessible; or

use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

34. (Previously Presented) The method of claim 33, further comprising implementing a remedial action in response to the identified one or more vulnerabilities.

35. (Previously Presented) The method of claim 34, wherein the remedial action includes transmitting an alert to a device associated with an administrator.

36. (Currently Amended) The method of claim 33, wherein the at least one processor is further configured to ~~querying~~ query the cloud provider API to determine the network accessibility information related to the workload further comprises:

examining data sources associated with the workload; and

determining the network accessibility information based on the examined data sources.

37. (Previously Presented) The method of claim 33, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external source, cloud provider information, or at least one network capture log.

38. (Previously Presented) The method of claim 33, wherein identify the installed software application comprises:

extracting data from at least one of OS packages, libraries, or program language libraries; and

identifying the installed software application based on the extracted data.

39. (Previously Presented) The method of claim 38, wherein the at least one processor is further configured to identify a version of the installed software application.

40. (Currently Amended) A non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising:

using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment;

identifying an installed software application in the accessed block storage volume;

analyzing the identified installed software application to determine an associated software version;

accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;

perform at least one of the following:

~~query the cloud provider API to determine network accessibility information related to the workload;~~

identify at least one port on which the vulnerable application is accessible; or

use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

41-159 (Cancelled)

REMARKS

The present response is intended to be fully responsive to all points of rejection raised by the instant Office Action mailed September 1, 2022, and is believed to place the application in condition for allowance.

In the Office Action, the Examiner:

a) Rejected claims 21, 22, 26, 27, 29, 30, 32-34, and 38-40 as being anticipated under 35 U.S.C. § 102(a)(1) by U.S. Patent Pub. No. 2020/0389483 A1 to Dye et al. ("Dye").

b) Rejected claims 23, 31, and 35 as being unpatentable under 35 U.S.C. § 103 over Dye and further in view of U.S. Patent Pub. No. 2018/0063182 A1 to Jones et al. ("Jones").

c) Rejected claims 24, 25, 36, and 37 as being unpatentable under 35 U.S.C. § 103 over Dye and further in view of U.S. Patent No. 9,356,942 to Joffe ("Joffe").

d) Rejected claim 28 as being unpatentable under 35 U.S.C. § 103 over Dye and further in view of U.S. Patent No. 9,413,774 to Liu et al. ("Liu").

In response, Applicant proposes to amend claims 21-40. No new matter has been added. Applicants respectfully request that the Examiner consider this Response After Final based on the After Final Consideration Pilot (AFCP) 2.0. A Certification and Request for Consideration under the After Final Consideration Pilot Program 2.0 (Form PTO/SB/434) is being filed concurrently herewith.

.

35 U.S.C. § 102 REJECTIONS

Claims 21 (and dependent claims 22, 26, 27, 29, 30, and 32) and claims 33 (and dependent claims 34 and 38-39) and 40 are rejected under 35 U.S.C. § 102 as anticipated by US 20200389483 to Dye. The Applicant respectfully traverses this rejection in view of the remarks that follow.

Claim 21, as amended, recites, *inter alia*:

> "…*perform at least one of the following: identify at least one port on which the vulnerable application is accessible; or use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload*."

Applicant respectfully maintains that nowhere does Dye teach each of these elements of the claims as amended. The Examiner asserts that these elements are allegedly disclosed by Dye at [0030, lines 7-16] and [0062, lines 2-13]. Nowhere in this disclosure does Dye mention, *inter alia*, all of these elements.

With respect to an identification of "at least one port on which [a] vulnerable application is accessible" and identifying one or more vulnerabilities using "network accessibility information and at least one port," Dye fails to identify or utilize a "port" for any purpose, let alone to "identify at least one port on which [a] vulnerable application is accessible" or to "identify one or more vulnerabilities" using "at least one port." Indeed, Dye does not disclose a port. Other aspects of Dye, such as a "file path 113," Dye at [0026]; [0030]; [0035]; [0037], do not constitute the claimed "port." Nor has the Examiner made the case that they do. Indeed, the Examiner recognizes that independent claim 21, as amended, was not disclosed by Dye or any other reference discovered during the search process. Neither "identify at least one port on which the vulnerable application is accessible" nor "use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload" were elements disclosed. Instead, the Examiner relies on the phrase "perform at least one of" with a claim element no longer present in claim 21 to deny allowance. Office Action at 3. Therefore, the claim as amended is thus allowable over Dye.

For at least the foregoing reasons, independent claims 21, 33, and 40, and dependent claims 22, 26, 27, 29, 30, 32, 34, and 38-39, as amended, is allowable over this rejection and the Applicant respectfully requests reconsideration based on the above.

35 U.S.C. § 103 REJECTIONS

Claims 23-25, 28, 31, and 35-37 are rejected under 35 U.S.C. § 103 as being unpatentable over various combinations of Dye, Jones, Joffe, and Liu. As discussed above, Dye fails to teach elements of the claims. The Examiner does not assert that any of Jones, Joffe, or Liu make up for these deficiencies, nor do they. Therefore, the rejection of claims 23-25, 28, 31, and 35-37 should be withdrawn.

CONCLUSION

Applicant respectfully requests that this Amendment under 37 C.F.R. § 1.116 be entered by the Examiner, placing claims 21-40 in condition for allowance. Applicant submits that the proposed amendments of claims 21-40 do not raise new issues or necessitate the undertaking of any additional search of the art by the Examiner, since all of the elements and their relationships claimed were either earlier claimed or inherent in the claims as examined. The claims before the Examiner have been previously examined and considered in their entirety, and the Examiner admits to examining the claims as presented with no new material, and therefore an additional search is not required. Therefore, this Amendment should allow for immediate action by the Examiner.

Furthermore, Applicant respectfully points out that the final action by the Examiner presented new arguments as to the application of the references against the pending claims. It is respectfully submitted that the entering of the Amendment would allow the Applicant to reply to the final rejections and place the application in condition for allowance.

Finally, Applicant submits that the entry of the amendment would place the application in better form for appeal, should the Examiner dispute the patentability of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: October 13, 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**10/13/2022 11:29:21 AM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Wanda Whittington
PATENT CENTER #	61062128	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 4

DOCUMENT		PAGES	DESCRIPTION	SIZE (KB)
15635-0005-00000AFCP 2.0-Orca15635-0005.pdf		2	After Final Consideration Program Request	164 KB
15635-0005-00000Response to Final OAOrca.pdf		10	-	50 KB
15635-0005-00000Response to Final OAOrca-A.NE.pdf	(1-1)	1	Response After Final Action	13 KB
15635-0005-00000Response to Final OAOrca-CLM.pdf	(2-6)	5	Claims	28 KB

15635-0005-00000Response to Final OAOrca-REM.pdf	(7-10)	4	Applicant Arguments/Remarks Made in an Amendment	28 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
15635-0005-00000AFCP 2.0-Orca15635-0005.pdf	55EDC7C57B0FF1650468F74DA9827780EA5547AD7037DD31C58A5814DD2C21E3A3A1A927CF5EDC8C53D9688F11346B1CE0EA8DFC79CE5A2E83D0309ED5F4EE50
15635-0005-00000Response to Final OAOrca.pdf	6F3BC8095F45F6E93F6F11336AC002A0E301E35C6535DAB46B0BCB1622F55FA6D2DE4F72B37B51A6CEAA63D1AF4C8697A59C960AA7F9DD9D79F2600A663F4375
15635-0005-00000Response to Final OAOrca-A.NE.pdf	5B2712D22B7E5CEBA52EF7C3B95AC0746F394CA7E33C6C965B440429DFB70D2A8534D46AE526B0FB7305EECB2587720793092059610875OAEA148ECEDE3A6240
15635-0005-00000Response to Final OAOrca-CLM.pdf	E8312F48D4CEE910DE6FDEAC52F95F3EC7B1DB2E9318A1CC443E07ACDB680F9FFD92EFF6E6C071B643A40F8C189BF63D527BE4BA88EA9CE6855F53C2CA0B03AB
15635-0005-00000Response to Final OAOrca-REM.pdf	C8810C56A7426B770D9CD1A882D72C9D3A0C0E28F7CEDFC32E5F17A757E6A150828FFB19D9295005DCBA24596689766719BCEFC541A508B126845DA8E359B7AC

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Doc Code: A.NE.AFCP
Document Description: After Final Consideration Pilot Program Request

PTO/SB/434 (05-13)

CERTIFICATION AND REQUEST FOR CONSIDERATION UNDER THE AFTER FINAL CONSIDERATION PILOT PROGRAM 2.0

Practitioner Docket No.:	Application No.:	Filing Date:
15635.0005-00000	17/657,972	April 5, 2022
First Named Inventor:	**Title:**	
Avi SHUA	Systems and Methods for Managing Cyber Vulnerabilities	

APPLICANT HEREBY CERTIFIES THE FOLLOWING AND REQUESTS CONSIDERATION UNDER THE AFTER FINAL CONSIDERATION PILOT PROGRAM 2.0 (AFCP 2.0) OF THE ACCOMPANYING RESPONSE UNDER 37 CFR 1.116.

1. The above-identified application is (i) an original utility, plant, or design nonprovisional application filed under 35 U.S.C. 111(a) [a continuing application (*e.g.*, a continuation or divisional application) is filed under 35 U.S.C. 111(a) and is eligible under (i)], or (ii) an international application that has entered the national stage in compliance with 35 U.S.C. 371(c).
2. The above-identified application contains an outstanding final rejection.
3. Submitted herewith is a response under 37 CFR 1.116 to the outstanding final rejection. The response includes an amendment to at least one independent claim, and the amendment does not broaden the scope of the independent claim in any aspect.
4. This certification and request for consideration under AFCP 2.0 is the only AFCP 2.0 certification and request filed in response to the outstanding final rejection.
5. Applicant is willing and available to participate in any interview requested by the examiner concerning the present response.
6. This certification and request is being filed electronically using the Office's electronic filing system (EFS-Web).
7. Any fees that would be necessary consistent with current practice concerning responses after final rejection under 37 CFR 1.116, *e.g.*, extension of time fees, are being concurrently filed herewith. [There is no additional fee required to request consideration under AFCP 2.0.]
8. By filing this certification and request, applicant acknowledges the following:
   - Reissue applications and reexamination proceedings are not eligible to participate in AFCP 2.0.
   - The examiner will verify that the AFCP 2.0 submission is compliant, *i.e.*, that the requirements of the program have been met (see items 1 to 7 above). For compliant submissions:
     - The examiner will review the response under 37 CFR 1.116 to determine if additional search and/or consideration (i) is necessitated by the amendment and (ii) could be completed within the time allotted under AFCP 2.0. If additional search and/or consideration is required but cannot be completed within the allotted time, the examiner will process the submission consistent with current practice concerning responses after final rejection under 37 CFR 1.116, *e.g.*, by mailing an advisory action.
     - If the examiner determines that the amendment does not necessitate additional search and/or consideration, or if the examiner determines that additional search and/or consideration is required and could be completed within the allotted time, then the examiner will consider whether the amendment places the application in condition for allowance (after completing the additional search and/or consideration, if required). If the examiner determines that the amendment does not place the application in condition for allowance, then the examiner will contact the applicant and request an interview.
       - The interview will be conducted by the examiner, and if the examiner does not have negotiation authority, a primary examiner and/or supervisory patent examiner will also participate.
       - If the applicant declines the interview, or if the interview cannot be scheduled within ten (10) calendar days from the date that the examiner first contacts the applicant, then the examiner will proceed consistent with current practice concerning responses after final rejection under 37 CFR 1.116.

Signature	Date
/Christopher C. Johns/	13 October 2022
Name (Print/Typed) Christopher C. Johns	**Practitioner Registration No.** 68,664

Note: *This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4(d) for signature requirements and certifications. Submit multiple forms if more than one signature is required, see below*.*

☐ * Total of _____ forms are submitted.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

PTO/SB/06 (09-11)
Approved for use through 1/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 17/657,972	Filing Date 04/05/2022	☐ To be Mailed

ENTITY: ☑ LARGE ☐ SMALL ☐ MICRO

APPLICATION AS FILED - PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A		N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A		N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A		N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*		x $100 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*		x $480 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $310 ($155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).				
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))					
* If the difference in column 1 is less than zero, enter "0" in column 2.				TOTAL	

APPLICATION AS AMENDED - PART II

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT	10/13/2022	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0		x $100 =	0
	Independent (37 CFR 1.16(h))	* 3	Minus	*** 3	= 0		x $480 =	0
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)			
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=		x $0 =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=		x $0 =	
	☐ Application Size Fee (37 CFR 1.16(s))							
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							
							TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.	LIE
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".	/DORIS M KING/
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".	
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.	

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/657,972	04/05/2022	Avi SHUA	15635.0005-00000	2547

22852 7590 09/01/2022

FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

EXAMINER
SU, SARAH

ART UNIT	PAPER NUMBER
2431	

NOTIFICATION DATE	DELIVERY MODE
09/01/2022	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

regional-desk@finnegan.com

Office Action Summary	Application No. 17/657,972	Applicant(s) SHUA, Avi	
	Examiner SARAH SU	Art Unit 2431	AIA (FITF) Status Yes

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTHS FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2a)☑ This action is **FINAL.** 2b) ☐ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims*

5) ☑ Claim(s) 21-40 is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6) ☐ Claim(s) _____ is/are allowed.

7) ☑ Claim(s) 21-40 is/are rejected.

8) ☐ Claim(s) _____ is/are objected to.

9) ☐ Claim(s) _____ are subject to restriction and/or election requirement

* If any claims have been determined allowable, you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to **PPHfeedback@uspto.gov.**

Application Papers

10)☐ The specification is objected to by the Examiner.

11)☑ The drawing(s) filed on 8/19/22 is/are: a)☑ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

a)☐ All b)☐ Some** c)☐ None of the:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

** See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☑ Notice of References Cited (PTO-892)

2) ☑ Information Disclosure Statement(s) (PTO/SB/08a and/or PTO/SB/08b) Paper No(s)/Mail Date _____.

3) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

4) ☐ Other: _____.

FINAL ACTION

Notice of Pre-AIA or AIA Status

1. The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

2. Amendment A, received on 19 August 2022, has been entered into record. In this amendment, claims 21, 23, 31, 32, 35, and 40 have been amended.

3. Claims 21-40 are presented for examination.

4. It is noted that claims 23, 31, 32, and 40 have been erroneously labeled “Previously Presented” and should be labeled as “Currently Amended”.

Response to Arguments

5. With regards to the objection to the specification, the applicant has submitted a substitute specification, and the examiner hereby withdraws the objection.

6. With regards to the objection to the claims, the applicant has submitted amendments, and the examiner hereby withdraws the objection.

7. Applicant's arguments filed 19 August 2022 have been fully considered but they are not persuasive.

Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

Nevertheless, as to the claim 21, it is argued by the applicant that Dye does not disclose querying a cloud provider API to determine network availability, particularly a cloud provider API. The

examiner respectfully disagrees. Dye discloses that binary data, file paths, known vulnerabilities (i.e. network accessibility information) is returned via an API of a server which is also disclosed as being able to be implemented on a cloud network, therefore being a cloud API (0030, lines 7-16; 0062, lines 2-13).

With regards to the applicant's argument that Dye fails to disclose an identification of at least one port and utilizing a port, the examiner notes that the claim recites "perform at least one of:" the querying, identifying, or using steps. Therefore, since it is the examiner's position that Dye discloses the querying step, Dye is shown to disclose "performing at least one of: query…; identify…; or use…"

Information Disclosure Statement

8. The information disclosure statement (IDS) submitted on 11 August 2022 is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner.

Drawings

9. The drawings were received on 19 August 2022. These drawings are acceptable.

Claim Rejections - 35 USC § 102

10. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis for the rejection will not be considered a new ground of rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

11. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a)(1) the claimed invention was patented, described in a printed publication, or in public use, on sale, or otherwise available to the public before the effective filing date of the claimed invention.

12. Claim(s) 21, 22, 26, 27, 29, 30, 32-34, and 38-40 is/are rejected under 35 U.S.C. 102(a)(1) as being anticipated by Dye et al. (US 2020/0389483 A1 and Dye hereinafter).

As to claims 21, 33, and 40, Dye discloses a system and method for computer security vulnerability assessment, the system and method having:

at least one processor configured to: using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment (0062, lines 6-13)**;**

identify an installed software application in the accessed block storage volume (0024, lines 1-11)**;**

analyze the identified installed software application to determine an associated software version (0024, lines 11-17)**;**

access a data structure of known software vulnerabilities for a plurality of versions of software applications (0004, lines 16-19; 0021, lines 3-11)**;**

perform a lookup of the identified installed software version in the data structure to identify known vulnerabilities (0004, lines 19-26)**;**

perform at least one of: query the cloud provider API to determine network accessibility information related to the workload; identify at least one port on which the vulnerable application is accessible; or use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload (0030, lines 7-16; 0062, lines 2-13)**.**

As to claims 22 and 34, Dye discloses:

wherein the at least one processor is further configured to implement a remedial action in response to the identified one or more vulnerabilities (0046, lines 22-29)**.**

As to claims 26 and 38, Dye discloses:

extract data from at least one of operating system packages, libraries, or program language libraries (0018, lines 1-9)**; and**

identify the installed software application based on the extracted data (0018, lines 1-18)**.**

As to claims 27 and 39, Dye discloses:

identify a version of the installed software application (0024, lines 11-17)**.**

As to claim 29, Dye discloses:

wherein the data structure includes aggregated vulnerability data (0004, lines 16-19; 0021, lines 3-11)**.**

As to claim 30, Dye discloses:

wherein the aggregated vulnerability data includes data from one or more third-party vendors (0025, lines 1-15)**.**

As to claim 32, Dye discloses:

wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug (0017, lines 21-26)**.**

Claim Rejections - 35 USC § 103

13. The following is a quotation of 35 U.S.C. 103 which forms the basis for all obviousness rejections set forth in this Office action:

> A patent for a claimed invention may not be obtained, notwithstanding that the claimed invention is not identically disclosed as set forth in section 102, if the differences between the claimed invention and the prior art are such that the claimed invention as a whole would have been obvious before the effective filing date of the claimed invention to a person having ordinary skill in the art to which the claimed invention pertains. Patentability shall not be negated by the manner in which the invention was made.

14. Claim(s) 23, 31, and 35 is/are rejected under 35 U.S.C. 103 as being unpatentable over Dye as applied to claims 22 and 34 above, and further in view of Jones et al. (US 2018/0063182 A1 and Jones hereinafter).

As to claims 23 and 35, Dye fails to specifically disclose:

wherein the remedial action includes transmitting an alert to a device associated with an administrator.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Jones.

Jones discloses a system and method for identifying and mapping sensitive data on an enterprise, the system and method having:

wherein the remedial action includes transmitting an alert to a device associated with an administrator (0040, lines 7-12)**.**

Given the teaching of Jones, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Jones by transmitting an alert. Jones recites motivation by disclosing that an alert is transmitted in order to remediate a potential threat as soon as possible to ensure the least amount of damage or compromise to the enterprise (0040, lines 7-12). It is obvious that

the teachings of Jones would have improved the teachings of Dye by transmitting an alert in order to minimize damage and protect the enterprise.

As to claim 31, Dye fails to specifically disclose:

wherein the aggregated vulnerability data includes data collected by a scanner.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Jones.

Jones discloses:

wherein the aggregated vulnerability data includes data collected by a scanner (0082, lines 2-6; 0090, lines 5-7)**.**

Given the teaching of Jones, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Jones by using data collected by a scanner. Please refer to the motivation recited above with respect to claims 23 and 35 as to why it is obvious to apply the teachings of Jones to the teachings of Dye.

15. Claim(s) 24, 25, 36, and 37 is/are rejected under 35 U.S.C. 103 as being unpatentable over Dye as applied to claims 21 and 33 above, and further in view of Joffe (US Patent 9,356,942 B1 and Joffe hereinafter).

As to claims 24 and 36, Dye fails to specifically disclose:

examining data sources associated with the workload; and

determining the network accessibility information based on the examined data sources.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Joffe.

Joffe discloses a system and method for detecting network compromise, the system and method having:

examining data sources associated with the workload (col. 10, lines 8-13)**; and**

determining the network accessibility information based on the examined data sources (col. 10, lines 8-13)**.**

Given the teaching of Joffe, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Joffe by determining network accessibility. Joffe recites motivation by disclosing that network accessibility information such as log information is collected in order to perform an analysis to determine if a security event is imminent or has occurred, thus providing security to the system (col. 10, lines 8-13, 53-67). It is obvious that the teachings of Joffe would have improved the teachings of Dye by determining network accessibility information in order to provide security.

As to claims 25 and 37, Dye fails to specifically disclose:

wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Joffe.

Joffe discloses a system and method for detecting network compromise, the system and method having:

wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log (col. 10, lines 8-13)**.**

Given the teaching of Joffe, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Joffe by determining network accessibility information including network capture log data. Please refer to the motivation recited above with respect to claims 24 and 36 as to why it is obvious to apply the teachings off Joffe to the teachings of Dye.

16. Claim(s) 28 is/are rejected under 35 U.S.C. 103 as being unpatentable over Dye as applied to claim 21 above, and further in view of Liu et al. (US Patent 9,413,774 B1 and Liu hereinafter).

As to claims 28, Dye fails to specifically disclose:

wherein the identified installed software application includes one or more scripts.

Nonetheless, this feature is well known in the art and would have been an obvious modification of the teachings disclosed by Dye, as taught by Liu.

Liu discloses a system and method for dynamic malware analysis of a URL, the system and method having:

wherein the identified installed software application includes one or more scripts (col. 5, lines 58-62)**.**

Given the teaching of Liu, a person having ordinary skill in the art before the effective filing date of the claimed invention would have readily recognized the desirability and advantages of modifying the teachings of Dye with the teachings of Liu by using scripts. Liu recites motivation by disclosing that scripts are executable content that can be malware or facilitate the delivery of malware and thus may be included as installed software application which must be analyzed (col. 5, lines 58-62). It is obvious that the teachings of Liu would have improved the teachings of Dye by including a script as the installed software application since scripts also can be malware or facilitate the delivery of malware and must also be analyzed.

Prior Art Made of Record

17. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

a. Chhabra et al. (US 2022/0124110 A1) discloses a system and method for anomaly detection using an ensemble of detection models.

b. Murchison et al. (US Patent 9,686,296 B1) discloses a system and method for providing network security monitoring.

c. Murthy et al. (US 2020/0228560 A1) discloses a system and method for automated vulnerability assessment with policy-based mitigation.

d. Newman et al. (US 2022/0239681 A1) discloses a system and method for public cloud database activity monitoring.

e. Seetharamaiah et al. (US 2021/0099478 A1) discloses a system and method for identifying and reporting cloud-based security vulnerabilities.

Conclusion

18. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing

date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

19. Any inquiry concerning this communication or earlier communications from the examiner should be directed to SARAH SU whose telephone number is (571)270-3835. The examiner can normally be reached 7:30 AM - 4:00 PM.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lynn Feild can be reached on 571-272-2092. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of published or unpublished applications may be obtained from Patent Center. Unpublished application information in Patent Center is available to registered users. To file and manage patent submissions in Patent Center, visit: https://patentcenter.uspto.gov. Visit https://www.uspto.gov/patents/apply/patent-center for more information about Patent Center and https://www.uspto.gov/patents/docx for information about filing in DOCX format. For additional questions, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/SARAH SU/
Primary Examiner, Art Unit 2431

Notice of References Cited	Application/Control No. 17/657,972	Applicant(s)/Patent Under Reexamination SHUA, Avi	
	Examiner SARAH SU	Art Unit 2431	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-20220239681-A1	07-2022	Newman; Aaron	G06F11/3065	1/1
*	B	US-20220124110-A1	04-2022	CHHABRA; Jasmeet	H04L63/1425	1/1
*	C	US-20210099478-A1	04-2021	Seetharamaiah; Sriranga	H04L63/1425	1/1
*	D	US-20200228560-A1	07-2020	Murthy; Varun	H04L63/20	1/1
*	E	US-9686296-B1	06-2017	Murchison; Jon	H04L63/08	1/1
	F					
	G					
	H					
	I					
	J					
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	Art Unit
	SARAH SU	2431

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

CLAIMS

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIM		DATE								
Final	Original	06/14/2022	08/26/2022							
	1	-	-							
	2	-	-							
	3	-	-							
	4	-	-							
	5	-	-							
	6	-	-							
	7	-	-							
	8	-	-							
	9	-	-							
	10	-	-							
	11	-	-							
	12	-	-							
	13	-	-							
	14	-	-							
	15	-	-							
	16	-	-							
	17	-	-							
	18	-	-							
	19	-	-							
	20	-	-							
	21	✓	✓							
	22	✓	✓							
	23	✓	✓							
	24	✓	✓							
	25	✓	✓							
	26	✓	✓							
	27	✓	✓							
	28	✓	✓							
	29	✓	✓							
	30	✓	✓							
	31	✓	✓							
	32	✓	✓							
	33	✓	✓							
	34	✓	✓							
	35	✓	✓							
	36	✓	✓							
	37	✓	✓							
	38	✓	✓							
	39	✓	✓							
	40	✓	✓							
	41	-	-							
	42	-	-							

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	**Art Unit**
	SARAH SU	2431

CLAIM		DATE								
Final	Original	06/14/2022	08/26/2022							
	43	-	-							
	44	-	-							
	45	-	-							
	46	-	-							
	47	-	-							
	48	-	-							
	49	-	-							
	50	-	-							
	51	-	-							
	52	-	-							
	53	-	-							
	54	-	-							
	55	-	-							
	56	-	-							
	57	-	-							
	58	-	-							
	59	-	-							
	60	-	-							
	61	-	-							
	62	-	-							
	63	-	-							
	64	-	-							
	65	-	-							
	66	-	-							
	67	-	-							
	68	-	-							
	69	-	-							
	70	-	-							
	71	-	-							
	72	-	-							
	73	-	-							
	74	-	-							
	75	-	-							
	76	-	-							
	77	-	-							
	78	-	-							
	79	-	-							
	80	-	-							
	81	-	-							
	82	-	-							
	83	-	-							
	84	-	-							
	85	-	-							
	86	-	-							
	87	-	-							
	88	-	-							
	89	-	-							
	90	-	-							
	91	-	-							
	92	-	-							
	93	-	-							
	94	-	-							

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	**Art Unit**
	SARAH SU	2431

CLAIM		DATE								
Final	Original	06/14/2022	08/26/2022							
	95	-	-							
	96	-	-							
	97	-	-							
	98	-	-							
	99	-	-							
	100	-	-							
	101	-	-							
	102	-	-							
	103	-	-							
	104	-	-							
	105	-	-							
	106	-	-							
	107	-	-							
	108	-	-							
	109	-	-							
	110	-	-							
	111	-	-							
	112	-	-							
	113	-	-							
	114	-	-							
	115	-	-							
	116	-	-							
	117	-	-							
	118	-	-							
	119	-	-							
	120	-	-							
	121	-	-							
	122	-	-							
	123	-	-							
	124	-	-							
	125	-	-							
	126	-	-							
	127	-	-							
	128	-	-							
	129	-	-							
	130	-	-							
	131	-	-							
	132	-	-							
	133	-	-							
	134	-	-							
	135	-	-							
	136	-	-							
	137	-	-							
	138	-	-							
	139	-	-							
	140	-	-							
	141	-	-							
	142	-	-							
	143	-	-							
	144	-	-							
	145	-	-							
	146	-	-							

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	**Art Unit**
	SARAH SU	2431

CLAIM		DATE								
Final	Original	06/14/2022	08/26/2022							
	147	-	-							
	148	-	-							
	149	-	-							
	150	-	-							
	151	-	-							
	152	-	-							
	153	-	-							
	154	-	-							
	155	-	-							
	156	-	-							
	157	-	-							
	158	-	-							
	159	-	-							

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	Art Unit
	SARAH SU	2431

CPC - Searched*		
Symbol	Date	Examiner
H04L 63/1433, 1441, 14$2	06/14/2022	SS

CPC Combination Sets - Searched*		
Symbol	Date	Examiner

US Classification - Searched*			
Class	Subclass	Date	Examiner

* See search history printout included with this form or the SEARCH NOTES box below to determine the scope of the search.

Search Notes		
Search Notes	Date	Examiner
PLUS Search	06/14/2022	SS
Inventor Search	06/14/2022	SS
IP.com Search	06/14/2022	SS
PE2E Search (see attached)	06/14/2022	SS
H04L 63/1433, 1441, 14$2 (see search history notes to determine scope)	06/14/2022	SS
PE2E Search (see attached)	08/26/2022	SS

Interference Search			
US Class/CPC Symbol	US Subclass/CPC Group	Date	Examiner

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	17/657,972	SHUA, Avi
	Examiner	Art Unit
	SARAH SU	2431

PE2E SEARCH - Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	British Equivalents	Time Stamp
L1	9	((("SHUA") near3 ("Avi"))).INV.	(US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT)	ADJ	ON	ON	2022/06/09 01:43 PM
L2	7	L1 AND version	(US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT)	ADJ	ON	ON	2022/06/09 01:43 PM
L3	0	((("Orca") near3 ("Security"))).AS,AANM.	(USPAT)	ADJ	ON	ON	2022/06/09 01:44 PM
L4	182419	(H04L63/1433 OR H04L63/1441 OR H04L63/14$2).cpc.	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/09 01:53 PM
L5	2266	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/09 01:53 PM
L6	170	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat) SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO,	ADJ	ON	ON	2022/06/09 01:54 PM

			RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L7	50	("10539275" "20170314745" "9109162" "9646093" "20050089749" "20150130346" "20150269204" "4591689" "5186249" "7392163" "20010046558" "20050106574" "20110003808" "5774249" "5999117" "6749768" "6749894" "8098282" "8868379" "10444489" "20020137625" "20040001966" "20040069969" "20050106712" "20060212502" "20080069735" "20090015670" "20100069660" "20110151019" "20120147382" "20130261976" "20140322337" "20160215211" "20170003497" "3554073" "3554085" "3554088" "3554091" "3554093" "3554095" "3554098" "3554099" "3554120" "3554135" "3554137" "3554147" "3554153" "3554155" "3554165" "3554167").pn.	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:33 AM
L8	5	L7 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:34 AM
L9	4	L7 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:34 AM
L10	50	("4770410" "5637144" "6145625" "4246184" "4269744" "4286481" "4301570" "4302939" "4436326" "4448396" "4551623" "4555258" "4796926" "4806203" "4836893" "4921707" "5188692" "5202667" "5250146" "5406914"	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:35 AM

		"5494546" "6036446" "4243070" "4257482" "4269611" "4272640" "4273374" "4281805" "4288558" "4299530" "4310334" "4320333" "4324078" "4324921" "4330553" "4334908" "4336606" "4338499" "4338737" "4347679" "4352957" "4355357" "4359761" "4360187" "4362072" "4365112" "4366353" "4367229" "4375510" "4381127").pn.					
L11	3	L10 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:35 AM
L12	0	L10 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2022/06/13 09:35 AM
L13	584	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/13 09:37 AM
L14	355	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version WITH (software OR app OR application OR software OR program OR code)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/13 09:37 AM
L15	335	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU,	ADJ	ON	ON	2022/06/13 09:37 AM

		WITH (software OR app OR application OR software OR program)	CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L16	239	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version near2 (software OR app OR application OR software OR program)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/13 09:38 AM
L17	376	L4 AND (query$3 OR request$3) WITH cloud WITH source	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 08:46 AM
L18	47	L4 AND (query$3 OR request$3) WITH cloud WITH source WITH access$5 WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN,	ADJ	ON	ON	2022/06/14 09:12 AM

			WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L19	1219	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH access$5 WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:16 AM
L22	420	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:17 AM
L23	75	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:27 AM
L24	44	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 WITH network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB,	ADJ	ON	ON	2022/06/14 09:28 AM

			HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L26	768	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:54 AM
L27	171	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log near3 network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 09:54 AM
L28	1094	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT;	ADJ	ON	ON	2022/06/14 10:39 AM

			IBM_TDB)				
L29	181	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 10:39 AM
L30	69	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 10:40 AM
L31	4	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4 WITH (add$3 OR creat$3)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 10:41 AM
L32	38	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV,	ADJ	ON	ON	2022/06/14 10:42 AM

		owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH (add$3 OR creat$3)	MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L33	1815	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 12:05 PM
L34	701	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 12:06 PM
L35	95	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/06/14 12:06 PM
L37	9	((("SHUA") near3	(US-PGPUB; USPAT;	ADJ	ON	ON	2022/08/25

		("Avi"))).INV.	USOCR; EPO; JPO; DERWENT)				02:36 PM
L38	7	L1 AND version	(US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT)	ADJ	ON	ON	2022/08/25 02:36 PM
L39	0	((("Orca") near3 ("Security"))).AS,AANM.	(USPAT)	ADJ	ON	ON	2022/08/25 02:36 PM
L40	191135	(H04L63/1433 OR H04L63/1441 OR H04L63/14$2).cpc.	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L41	2337	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat)	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L42	180	L4 AND (software OR program OR app OR application) WITH version WITH (vulnerab$5 OR risk OR threat) SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AU, AP, AT, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, OA, RU, SU, WO, MC, MD, MY, NL, NO, NZ, PH, PL, PT, RO, RS, SE, SG, SI, SK, TH, TN, TR, TW, UA, VN); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L43	50	("10539275"	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/25

		"20170314745" "9109162" "9646093" "20050089749" "20150130346" "20150269204" "4591689" "5186249" "7392163" "20010046558" "20050106574" "20110003808" "5774249" "5999117" "6749768" "6749894" "8098282" "8868379" "10444489" "20020137625" "20040001966" "20040069969" "20050106712" "20060212502" "20080069735" "20090015670" "20100069660" "20110151019" "20120147382" "20130261976" "20140322337" "20160215211" "20170003497" "3554073" "3554085" "3554088" "3554091" "3554093" "3554095" "3554098" "3554099" "3554120" "3554135" "3554137" "3554147" "3554153" "3554155" "3554165" "3554167").pn.					02:36 PM
L44	5	L7 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/25 02:36 PM
L45	4	L7 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/25 02:36 PM
L46	50	("4770410" "5637144" "6145625" "4246184" "4269744" "4286481" "4301570" "4302939" "4436326" "4448396" "4551623" "4555258" "4796926" "4806203" "4836893" "4921707" "5188692" "5202667" "5250146" "5406914" "5494546" "6036446" "4243070" "4257482" "4269611" "4272640" "4273374" "4281805" "4288558" "4299530" "4310334" "4320333"	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/25 02:36 PM

		"4324078" "4324921" "4330553" "4334908" "4336606" "4338499" "4338737" "4347679" "4352957" "4355357" "4359761" "4360187" "4362072" "4365112" "4366353" "4367229" "4375510" "4381127").pn.					
L47	3	L10 AND (vulnerab$5 OR threat OR risk)	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/25 02:36 PM
L48	0	L10 AND cloud	(US-PGPUB; USPAT)	OR	ON	ON	2022/08/25 02:36 PM
L49	610	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L50	367	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version WITH (software OR app OR application OR software OR program OR code)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L51	346	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version WITH (software OR app OR application OR software OR program)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL,	ADJ	ON	ON	2022/08/25 02:36 PM

			PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L52	246	L4 AND (vulnerab$5 OR threat OR risk) WITH (database OR record) WITH version near2 (software OR app OR application OR software OR program)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L53	399	L4 AND (query$3 OR request$3) WITH cloud WITH source	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L54	51	L4 AND (query$3 OR request$3) WITH cloud WITH source WITH access$5 WITH network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L55	1267	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY,	ADJ	ON	ON	2022/08/25 02:36 PM

		access$5 WITH network	CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L56	434	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L57	81	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 near3 network	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L58	46	L4 AND (examin$5 OR analyz$3 OR analysis) near3 source WITH access$7 WITH network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH,	ADJ	ON	ON	2022/08/25 02:36 PM

			TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L59	804	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L60	180	L4 AND (examin$5 OR analyz$3 OR analysis) WITH source WITH log near3 network AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L61	1134	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L62	188	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA,	ADJ	ON	ON	2022/08/25 02:36 PM

		OR alarm$3 OR message) WITH (admin OR administrator OR owner)	EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)				
L63	74	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L64	4	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH scan$4 WITH (add$3 OR creat$3)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L65	39	L4 AND (vulnerab$5 OR threat OR risk) WITH (remediat$3 OR fix$3 OR patch$3 OR clean$3) WITH (alert$3 OR alarm$3 OR message) WITH (admin OR administrator OR owner) AND (vulnerab$5 OR threat OR risk) WITH (record OR database) WITH (add$3 OR creat$3)	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO;	ADJ	ON	ON	2022/08/25 02:36 PM

			JPO; DERWENT; IBM_TDB)				
L66	1859	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L67	727	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script AND cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L68	98	L4 AND (vulnerab$5 OR threat OR risk) SAME (software OR app OR application OR program) WITH script SAME cloud	(US-PGPUB; USPAT; USOCR; FIT (AP, AT, AU, BE, BG, BR, BY, CA, CH, CN, CS, CU, CZ, DD, DE, DK, EA, EE, EP, ES, FI, FR, GB, HR, HU, ID, IE, IL, IS, IT, JP, KR, LT, LU, LV, MA, MC, MD, MY, NL, NO, NZ, OA, PH, PL, PT, RO, RS, RU, SE, SG, SI, SK, SU, TH, TN, TR, TW, UA, VN, WO); FPRS; EPO; JPO; DERWENT; IBM_TDB)	ADJ	ON	ON	2022/08/25 02:36 PM
L69	2	("20190281115" OR "20200137097").pn.	(US-PGPUB; USPAT)	ADJ	ON	ON	2022/08/26 09:08 AM
L70	393	L40 AND cloud WITH (api OR application programming interface) WITH network	(US-PGPUB; USPAT)	ADJ	ON	ON	2022/08/26 11:29 AM

L71	95	L40 AND cloud WITH (api OR application programming interface) WITH network WITH access$5	(US-PGPUB; USPAT)	ADJ	ON	ON	2022/08/26 11:29 AM
L72	2091	L40 AND port WITH access$5 WITH (program OR software OR app OR application)	(US-PGPUB; USPAT)	ADJ	ON	ON	2022/08/26 11:39 AM
L73	203	L40 AND port WITH access$5 WITH (program OR software OR app OR application) WITH (vulnerable OR risk$3 OR threat$4)	(US-PGPUB; USPAT)	ADJ	ON	ON	2022/08/26 11:39 AM
L74	74	L40 AND port WITH access$5 WITH (program OR software OR app OR application) near3 (vulnerable OR risk$3 OR threat$4)	(US-PGPUB; USPAT)	ADJ	ON	ON	2022/08/26 11:39 AM
L75	79	L40 AND (identify$3 OR identification OR determin$3 OR determination) WITH (vulnerab$5 OR threat OR risk) WITH port WITH network near3 access$7	(US-PGPUB; USPAT)	ADJ	ON	ON	2022/08/26 12:20 PM

PE2E SEARCH - Search History (Interference)

There are no Interference searches to show.

Doc code: IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (01-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	17657972
Filing Date	2022-04-05
First Named Inventor	Avi SHUA
Art Unit	2431
Examiner Name	Sarah Su
Attorney Docket Number	15635.0005-00000

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20200137097	A1	2020-04-30	Gil ZIMMERMAN et al.	
	2	20190281115	A1	2019-09-12	Douglas SWALLOW et al.	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

NON-PATENT LITERATURE DOCUMENTS Remove

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	17657972
Filing Date	2022-04-05
First Named Inventor	Avi SHUA
Art Unit	2431
Examiner Name	Sarah Su
Attorney Docket Number	15635.0005-00000

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	International Search Report and Written Opinion for PCT International Application No. PCT/IB2022/52670 filed March 23, 2022; pgs. 1-14; mailed July 26, 2022.	

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE

Examiner Signature	/SARAH SU/	Date Considered	08/26/2022

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	17657972
Filing Date	2022-04-05
First Named Inventor	Avi SHUA
Art Unit	2431
Examiner Name	Sarah Su
Attorney Docket Number	15635.0005-00000

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

☒ That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

See attached certification statement.

The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Christopher C. Johns/	Date (YYYY-MM-DD)	2022-08-11
Name/Print	Christopher C. Johns	Registration Number	68664

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

EFS Web 2.1.17

ALL REFERENCES CONSIDERED EXCEPT WHERE LINED THROUGH. /S.S/

INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				***Complete if Known***	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	1	of	2	*Attorney Docket Number*	**15635.0005-00000**

U.S. PATENTS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Issue Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-11,036,856-B2	06-15-2021	Graun et al.	
		US-9,794,289-B1	10-17-2017	Banerjee et al.	
		US-7,013,395-B1	03-14-2006	Swiler et al.	
		US-10,868,825-B1	12-15-2020	Dominessy et al.	
		US-10,152,487-B1	12-11-2018	Patwardhan et al.	
		US-7,143,143-B1	11-28-2006	Thompson	

U.S. PUBLISHED PATENT APPLICATIONS

Examiner Initials*	Cite No.	Document Number Number-Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-2020/0244692-A1	07-30-2020	SHUA	
		US-2019/0327272-A1	10-24-2019	Narayanaswamy	
		US-2020/0097662-A1	03-26-2020	Hufsmith et al.	
		US-2013/0262801-A1	10-03-2013	Sancheti et al.	
		US-20220210195-A1	06-30-2022	Parekh et al.	
		US-2019/0250857-A1	08-15-2019	Reddy et al.	
		US-2022/0191250-A1	06-16-2022	Laliberte	
		US-2014/0366118-A1	12-11-2014	Yin	
		US-2020/0220887-A1	07-09-2020	A. K.	
		US-2018/0367515-A1	12-20-2018	Anandam et al.	
		US-2022/0182294-A1	06-09-2022	Li et al.	
		US-2022/0100869-A1	03-31-2022	Berger et al.	
		US-2020/0396254-A1	12-17-2020	Crabtree et al.	
		US-2016/0330237-A1	11-10-2016	Edlabadkar	
		US-2018/0205755-A1	07-19-2018	Kavi et al.	
		US-2021/0342297-A1	11-04-2021	Gupta et al.	

Note: Submission of copies of U.S. Patents and published U.S. Patent Applications is not required.

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				***Complete if Known***	
				Application Number	17/657,972
				Filing Date	April 5, 2022
				First Named Inventor	Avi SHUA
				Art Unit	2431
				Examiner Name	SU, SARAH
Sheet	2	of	2	*Attorney Docket Number*	**15635.0005-00000**

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.	Foreign Patent Document Country Code Number Kind Code *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	Translation
		WO 2020/106973 A1	05-28-2020	ARAALI NETWORKS INC.		N/A
		WO 2019/051507 A1	03-14-2019	CARBON BLACK INC.		N/A
		WO 2017/153983 A1	09-14-2017	B. G. NEGEV TECHNOLOGIES AND APPLICATIONS LTD.		N/A

NON PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	Translation
		Non-Final Office Action Application No. 17/716,491 dated August 3, 2022, in the United States Patent and Trademark Office.	N/A
		Non-Final Office Action Application No. 17/657,986 dated August 17, 2022, in the United States Patent and Trademark Office.	N/A
		Non-Final Office Action Application No. 17/658,156 dated July 18, 2022, in the United States Patent and Trademark Office.	N/A
		Non-Final Office Action Application No. 17/716,583 dated August 18, 2022, in the United States Patent and Trademark Office.	N/A

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

PATENT
Attorney Docket No. 15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.: 17/657,972	)	Examiner: SU, SARAH
	)	
Filed: April 5, 2022	)	Confirmation No.: 2547
	)	
For: SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT UNDER 37 C.F.R. § 1.97(c)

Pursuant to 37 C.F.R. §§ 1.56 and 1.97(c), Applicant brings to the attention of the Examiner the documents on the attached listing. This Information Disclosure Statement is being filed after the events recited in Section 1.97(b) but, to the undersigned's knowledge, before the mailing date of either a Final action, *Quayle* action, or a Notice of Allowance. Under the provisions of 37 C.F.R. § 1.97(c), this Information Disclosure is accompanied by a fee of $260.00 as specified by Section 1.17(p).

Copies of the listed foreign and non-patent literature documents are attached.

Applicant respectfully requests that the Examiner consider the listed documents on the attached Form PTO/SB/08 and indicate that they were considered by making appropriate notations on the attached form.

This submission does not represent that a search has been made or that no better art exists and does not constitute an admission that each or all of the listed documents are material or constitute "prior art." If the Examiner applies any of the documents as prior art against any claim in the application and Applicant determines that the cited documents do not constitute "prior art" under United States law, Applicant reserves the right to present to the U.S. Patent and Trademark Office the relevant facts and law regarding the appropriate status of such documents.

Applicant further reserves the right to take appropriate action to establish the patentability of the disclosed invention over the listed documents, should one or more of the documents be applied against the claims of the present application.

If there is any fee due in connection with the filing of this Statement, please charge the fee to deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: August 31, 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4155

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC PAYMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	08/31/2022 04:21:24 PM ET	15635.0005-00000

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	60927351	AUTHORIZED BY	Christopher Johns
CUSTOMER #	22852	FILING DATE	04/05/2022
CORRESPONDENCE ADDRESS	-	FIRST NAMED INVENTOR	Avi SHUA

Payment Information

PAYMENT METHOD	PAYMENT TRANSACTION ID	PAYMENT AUTHORIZED BY
CARD / 1007	E20228UG22203011	Michelle Daniels

FEE CODE	DESCRIPTION	ITEM PRICE($)	QUANTITY	ITEM TOTAL($)
1806	SUBMISSION OF AN INFORMATION DISCLOSURE STATEMENT	260.00	1	260.00
			TOTAL AMOUNT:	$260.00

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
28 May 2020 (28.05.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/106973 A1

(51) International Patent Classification:
G06F 21/50 (2013.01) *H04L 29/06* (2006.01)

(21) International Application Number:
PCT/US2019/062641

(22) International Filing Date:
21 November 2019 (21.11.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/770,538 21 November 2018 (21.11.2018) US

(71) Applicant: **ARAALI NETWORKS, INC.** [US/US]; 39812 Mission Boulevard, Suite 224, Fremont, California 94539 (US).

(72) Inventors: **SINGH, Abhishek**; 3051 E. Ruby Hill Drive, Pleasanton, California 94566 (US). **GOPALASETTY, Bhanu**; 5611 Wells Lane, San Ramon, California 94582 (US). **PANI, Ayaskanta**; 33463 Bronco Loop, Fremont, California 94555 (US). **NULU, Deepak**; 696 Pilgrim Loop, Fremont, California 94539 (US). **KAR, Ashish**; 733 Asbury Place, Santa Clara, California 95051 (US). **AGASTHY, Rachita**; 1528 Vista Club Circle, Apt. 202, Santa Clara, California 95054 (US).

(74) Agent: **ENG, U.P. Peter**; Wilson Sonsini Goodrich & Rosati, 650 Page Mill Road, Palo Alto, California 94304 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report (Art. 21(3))*

(54) **Title:** SYSTEMS AND METHODS FOR SECURING A WORKLOAD

(57) **Abstract:** The present disclosure provides systems and methods for securing a computer workload. The method may comprise: receiving a workload; embedding a secure agent into the workload, wherein the secure agent comprises (i) a shim layer located between an application libraries layer of the workload and an operating system service layer and (ii) a security policy repository; and implementing security policies based at least in part on application programming interface (API) calls intercepted by the shim layer.

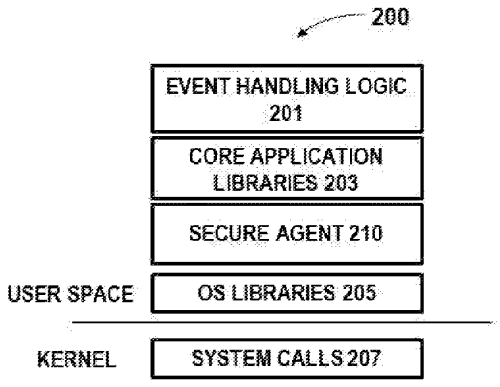


FIG. 2

WO 2020/106973 A1

SYSTEMS AND METHODS FOR SECURING A WORKLOAD

CROSS-REFERENCE

[0001] This application claims priority to U.S. Provisional Application No. 62/770,538 filed on November 21, 2018, which is entirely incorporated herein by reference

BACKGROUND

[0002] The security of computing devices against internal and external threats such as viruses, malware, network intrusions is a challenge in almost all networked computing environments. To protect computing devices against such threats, networked computing devices are configured with various security settings to prevent unwanted behavior. For example, firewalls are a traditional approach to providing communication security. These firewalls are network devices deployed on customer premises, and configured with access control lists (ACLs), or policies to govern allowed in and out of an endpoint.

[0003] With the fast development of IT, the trend towards public cloud, the move to microservices, as well as new application delivery and orchestration mechanisms (e.g., Kubernetes, Docker Swarm® containers) have created new challenges for securing the workloads. For example, virtualization has impacted how virtual machines connect with each other as well as how these communications are secured. In another example, a popular networking technology is creating overlays of multi-tenant network, where each tenant is given an illusion of a virtual private network. On the security front, various approaches have emerged where the firewall function is subsumed by a virtual appliance that is distributed in the network such that it is able to intercept and inspect packets and enforce communication policies. Another security approach is using host native firewalls (e.g., IPTables on Linux) instead of using a separate virtual firewall appliance. In some cases, new firewall paradigms are provided by using host/operating system primitives like eBPF (extended berkeley packet filters) or exploiting system (operating system's) call trapping. Such approaches apply policies at the network layer, although the intent is to control communication between applications and not network endpoints (or IP addresses). However, in the network, identifying the original application from the IP address (e.g., from network header) and port number (e.g., from transport header) is often times error prone. In some cases, secure software development has also been adopted for providing security. Tools and vendors may perform static analysis on the workload and report potential vulnerabilities, insert themselves in the test cycles to perform certain penetration testing to qualify the security readiness of the

application, or mitigate risk at runtime by embedding themselves into the application. These approaches, however, may be limited by being difficult to be applied to the policy or access control list approach of securing network communications of applications.

[0004] A recent trend in IT is using containers instead of or in conjunction with virtual machines (VMs). Containers are a means of specifying and packaging application workloads such that they can be reliability re-created, run in isolation from each other, and scaled at will. However, the security paradigm for containers is not well established. For instance, container workloads are often short lived (sub-second) and are created on demand in response to input bursts for elastic scaling. A considerable amount of coordination is needed to maintain agility, consistency and speed for managing such workloads with the existing security solutions (e.g., security bolted into hypervisor, host, or network). Another technology is unikernel which builds a unified kernel and application binary with tighter functionality support can be adapted for microservice deployments in place of containers.

[0005] The security issues for microservices are also challenging. Modern applications are provided by breaking down monolithic applications into several internal micro-services that can each be individually scaled, secured and evolved. This allows for micro-service reuse, modularity of architecture, and the independence of evolution as well as of upgrades (apart from improving the agility of upgrades). These microservices endpoints may be implemented for other micro-services to interact with, and may not be required to know the end-to-end architecture, resulting in a complex labyrinth of inter-application interactions that is difficult to secure. Given then trend that operations that were once internal function calls have now become network events, the attack surface is greatly increased.

SUMMARY

[0006] Recognized herein is a need for methods and systems capable of providing security to ephemeral or unpredictable workloads at fine grained level. In particular, security is provided without increasing maintenance or management cost. Additionally, security is desired to be applied close to the workload allowing for less permissive and granular policies.

[0007] The present disclosure provides a built-in security solution that a workload (e.g., application service) can be secured by embedding a security layer into the workload dynamically. This may be used to provide security that is always available regardless of where and how the security solution is deployed. The built-in security solution may allow for fine grained security control. In some cases the security policy may be applied based on an

identity of the entity requesting an access to a resource or an identity/type of the resource to be accessed. For example, instead of allowing access to all the APIs (application programming interfaces), access may be authorized at per API level. For instance, engineers may have read access to one API, whereas human resource may have write access to another API even when both APIs are exposed by the same application. Such per API level policies may be difficult to enforce in a traditional security model (e.g., bolt-on security model) since the communication is usually end-to-end encrypted. Furthermore, application context aware policy which captures current state of an application (e.g., changes to application's authorization level by means of obtained newer credentials or authorization tokens) can also be utilized in the provided system.

[0008] Systems and methods may be capable of providing security to ephemeral or unpredictable workload with improved efficiency. For example, modern cloud technologies are running serverless customer workloads, where the placement of workload is usually provided and managed by the cloud service provider that the workload is not known or controlled by the customer. The provided systems and methods may secure the workload by embedding a security layer in the workload thereby eliminating the burden of making assumptions on where the workload might get scheduled to ensure the security fall in the path leading to/from the workload. Additionally, traffic with a built-in security layer may allow for improved network efficiency which is achieved by reducing the necessity of routing the traffic through intermediaries or encrypting the traffic.

[0009] In an aspect, a system is provided for securing a computer workload within a network . The system may comprise: a secure agent, and the secure agent comprises an interfacing layer embedded into the computer workload for intercepting application programming interface (API) calls that the secure agent is also configured to apply a security policy based at least in part on the API calls intercepted by the interfacing layer; and a server in communication with the secure agent, and the server is configured to analyze telemetry data captured by the secure agent to secure the computer workload. In some embodiments, the computer workload is a container application or a computing processing that is performed by a bare metal, virtual machine or a combination of Kubernetes and server.

[0010] In some embodiments, the interfacing layer is a shim layer that is located between an application libraries layer of the computer workload and an operating system service layer. In some embodiments, the interfacing layer comprises a dynamically linked library.

[0011] In some embodiments, the telemetry data comprises application contextual data, metadata or flow telemetry data. In some embodiments, the server is configured to perform behavioral analysis based on the telemetry data. In some embodiments, the server is configured to monitor and record a history of the telemetry data at an API level. In some embodiments, the security policy is applied based on an identity of the entity requesting the access to the resource or an identity of the resource to be accessed.

[0012] In some embodiments, the server is configured to monitor and record a history of the telemetry data at an API level. In some embodiments, the secure agent is cryptographically secured.

[0013] In some embodiments, the secure agent is in communication with an interceptor, and wherein the interceptor is configured to intercept a data flow and apply the security policy on the data flow. In some cases the interceptor and the secure agent forms a master-slave relationship. In some embodiments, the secure agent comprises a memory unit to store data related to a policy state, a firewall state and the telemetry data. In some embodiments, the secure agent is configured to perform packet interception or packet inspection.

[0014] In some embodiments, the security policy is applied at Layer-7 of the OSI (Open Systems Interconnection) Network model. In some cases, the security policy is a location-transparent policy. In some cases, the security policy specifies a connection between two entities in the network. In some cases, the security policy comprises a service policy specifying a connection in a service or an environment policy specifying a connection permitted in an environment. In some embodiments, the security policy is automatically derived from the telemetry data.

[0015] In another aspect, a method is provided for securing a computer workload within a network. The method may comprise: receiving the computer workload in the network; embedding a secure agent into the computer workload, and the secure agent comprises an interfacing layer for intercepting application programming interface (API) calls; and applying, based at least in part on the API calls intercepted by the interfacing layer, a security policy to secure the computer workload.

[0016] In some embodiments, the computer workload is a container application or a computing processing that is performed by a bare metal, virtual machine or a combination of Kubernetes and server. In some embodiments, the interfacing layer is a shim layer that is

located between an application libraries layer of the computer workload and an operating system service layer.

[0017] In some embodiments, the secure agent is configured to capture telemetry data and transmit the telemetry data to a server for further analysis. For instance, the telemetry data comprises application contextual data, metadata or flow telemetry data. In some cases, the method further comprises processing the telemetry data and granting an access to a resource within the network based on the security policy. For example, the security policy is applied based on an identity of the entity requesting the access to the resource or an identity of the resource to be accessed.

[0018] In some embodiments, the security policy is automatically derived from telemetry data captured by the secure agent. In some embodiments, the server is configured to perform behavior analysis on the computer workload based on the telemetry data or monitor and record a history of the telemetry data at an API level.

[0019] In some embodiments, embedding the secure agent comprises using a dynamic linking approach, a lazy binding technique or a dynamic binary modification method. In some embodiments, the method further comprises registering and verifying the secure agent cryptographically via a handshake process. In some cases, the method may use a nonce to protect the handshake process against a replay attack. In some cases, the handshake process comprises verifying the secure agent by a peer node.

[0020] In some embodiments, the interfacing layer comprises a dynamically linked library. In some embodiments, the security policy is applied at Layer-7 of the OSI (Open Systems Interconnection) Network model. In some embodiments, the security policy specifies a connection between two entities in the network. In some embodiments, the security policy comprises a service policy specifying a connection in a service. In some embodiments, the security policy comprises an environment policy specifying a connection permitted in an environment.

[0021] In some embodiments, the secure agent is in communication with an interceptor, and wherein the interceptor is configured to intercept a data flow and apply the security policy on the data flow. In some cases, the interceptor and the secure agent forms a master-slave relationship. In some cases, the data flow comprises application contextual data.

[0022] In some embodiments, the secure agent comprises a memory unit to store data related to a policy state, a firewall state and a telemetry data. In some embodiments, the secure agent

is configured to perform packet interception or packet inspection. In some embodiments, the method further comprises attesting the workload using a nonce and a security key through an inter-process communication socket. In some embodiments, the method further comprises performing one or more actions upon executing the security policy, and wherein the one or more actions comprises alerting a malware.

[0023] Additional aspects and advantages of the present disclosure will become readily apparent to those skilled in this art from the following detailed description, wherein only illustrative embodiments of the present disclosure are shown and described. As will be realized, the present disclosure is capable of other and different embodiments, and its several details are capable of modifications in various obvious respects, all without departing from the disclosure. Accordingly, the drawings and description are to be regarded as illustrative in nature, and not as restrictive.

INCORPORATION BY REFERENCE

[0024] All publications, patents, and patent applications mentioned in this specification are herein incorporated by reference to the same extent as if each individual publication, patent, or patent application was specifically and individually indicated to be incorporated by reference. To the extent publications and patents or patent applications incorporated by reference contradict the disclosure contained in the specification, the specification is intended to supersede and/or take precedence over any such contradictory material.

BRIEF DESCRIPTION OF THE DRAWINGS

[0025] The novel features of the invention are set forth with particularity in the appended claims. A better understanding of the features and advantages of the present invention will be obtained by reference to the following detailed description that sets forth illustrative embodiments, in which the principles of the invention are utilized, and the accompanying drawings (also "Figure" and "FIG." herein), of which:

[0026] FIG. 1 schematically shows an example of a workload.

[0027] FIG. 2 schematically illustrates a workload integrated with a secure agent, in accordance with some embodiments of the invention.

[0028] FIG. 3 schematically illustrates an environment in which a secure management system and secure agents are operated.

[0029] FIG. 3A schematically illustrates an example of a secure management system employing a fog layer.

[0030] FIG. 4 schematically shows a block diagram of an exemplary secure agent, in accordance with various embodiments of the invention.

[0031] FIG. 5 shows an exemplary internal process of implementing security policies with a secure agent.

[0032] FIG. 6 shows an example of converting insecure workload into a secure workload.

[0033] FIG. 7 shows a computer system that is programmed or otherwise configured to implement a security management system.

[0034] FIG. 8 schematically shows various deployments of a secure agent and an interceptor process.

DETAILED DESCRIPTION

[0035] While various embodiments of the invention have been shown and described herein, it will be obvious to those skilled in the art that such embodiments are provided by way of example only. Numerous variations, changes, and substitutions may occur to those skilled in the art without departing from the invention. It should be understood that various alternatives to the embodiments of the invention described herein may be employed.

Certain definitions

[0036] Unless otherwise defined, all technical terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which this invention belongs.

[0037] Reference throughout this specification to "some embodiments," or "an embodiment," means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment. Thus, the appearances of the phrase "in some embodiment," or "in an embodiment," in various places throughout this specification are not necessarily all referring to the same embodiment. Furthermore, the particular features, structures, or characteristics may be combined in any suitable manner in one or more embodiments.

[0038] As utilized herein, terms "component," "system," "interface," "unit" and the like are intended to refer to a computer-related entity, hardware, software (e.g., in execution), and/or firmware. For example, a component can be a processor, a process running on a processor, an

object, an executable, a program, a storage device, and/or a computer. By way of illustration, an application running on a server and the server can be a component. One or more components can reside within a process, and a component can be localized on one computer and/or distributed between two or more computers.

[0039] Further, these components can execute from various computer readable media having various data structures stored thereon. The components can communicate via local and/or remote processes such as in accordance with a signal having one or more data packets (e.g., data from one component interacting with another component in a local system, distributed system, and/or across a network, e.g., the Internet, a local area network, a wide area network, etc. with other systems via the signal).

[0040] As another example, a component can be an apparatus with specific functionality provided by mechanical parts operated by electric or electronic circuitry; the electric or electronic circuitry can be operated by a software application or a firmware application executed by one or more processors; the one or more processors can be internal or external to the apparatus and can execute at least a part of the software or firmware application. As yet another example, a component can be an apparatus that provides specific functionality through electronic components without mechanical parts; the electronic components can include one or more processors therein to execute software and/or firmware that confer(s), at least in part, the functionality of the electronic components. In some cases, a component can emulate an electronic component via a virtual machine, e.g., within a cloud computing system.

[0041] Moreover, the word “exemplary” is used herein to mean serving as an example, instance, or illustration. Any aspect or design described herein as “exemplary” is not necessarily to be construed as preferred or advantageous over other aspects or designs. Rather, use of the word exemplary is intended to present concepts in a concrete fashion. As used in this application, the term “or” is intended to mean an inclusive “or” rather than an exclusive “or.” That is, unless specified otherwise, or clear from context, “X employs A or B” is intended to mean any of the natural inclusive permutations. That is, if X employs A; X employs B; or X employs both A and B, then “X employs A or B” is satisfied under any of the foregoing instances. In addition, the articles “a” and “an” as used in this application and the appended claims should generally be construed to mean “one or more” unless specified otherwise or clear from context to be directed to a singular form.

[0042] Embodiments of the invention may be used in a variety of applications. Some embodiments of the invention may be used in conjunction with various devices and systems, for example, a personal computer (PC), a desktop computer, a mobile computer, a laptop computer, a notebook computer, a tablet computer, a server computer, a handheld computer, a handheld device, a personal digital assistant (PDA) device, a handheld PDA device, a wireless communication station, a wireless communication device, a wireless access point (AP), a modem, a network, a wireless network, a local area network (LAN), a virtual local area network (VLAN), a wireless LAN (WLAN), a metropolitan area network (MAN), a wireless MAN (WMAN), a wide area network (WAN), a wireless WAN (WWAN), a personal area network (PAN), a wireless PAN (WPAN), a virtual private network (VPN), a storage area network (SAN), a frame relay connection, an Advanced Intelligent Network (AIN) connection, a synchronous optical network (SONET) connection, devices and/or networks operating in accordance with existing IEEE 802.11, 802.11a, 802.11b, 802.11e, 802.11g, 802.11h, 802.11i, 802.11n, 802.16, 802.16d, 802.16e standards and/or future versions and/or derivatives and/or long term evolution (LTE) of the above standards, units and/or devices which are part of the above networks, one way and/or two-way radio communication systems, cellular radio-telephone communication systems, a cellular telephone, a wireless telephone, a personal communication systems (PCS) device, a PDA device which incorporates a wireless communication device, a multiple input multiple output (MIMO) transceiver or device, a single input multiple output (SIMO) transceiver or device, a multiple input single output (MISO) transceiver or device, or the like.

[0043] It is noted that various embodiments can be used in conjunction with one or more types of wireless or wired communication signals and/or systems, for example, radio frequency (RF), infrared (IR), frequency-division multiplexing (FDM), orthogonal FDM (OFDM), time-division multiplexing (TDM), time-division multiple access (TDMA), extended TDMA (E-TDMA), general packet radio service (GPRS), extended GPRS, code-division multiple access (CDMA), wideband CDMA (WCDMA), CDMA 2000, multi-carrier modulation (MDM), discrete multi-tone (DMT), Bluetooth®, ZigBee™, or the like. Embodiments of the invention may be used in various other devices, systems, and/or networks.

[0044] Systems and methods of the present disclosure may provide a built-in security model. In some embodiments, the built-in security model may include a secure agent embedded into a workload (e.g., application service) and a backend system in communication with the

secure agent. This may be used to provide security that is always available and with fine control.

[0045] Systems and methods may be capable of providing security to short-lived or ephemeral workload with improved efficiency. The provided systems and methods may be applied to various types of workloads. A workload may also be referred to as computer workload throughout the specification.

[0046] In some embodiments, a computer workload (e.g., application, server software, software development environment, software test environment) is a unit of computing processing that is performed via an IaaS, PaaS, or SaaS. For example, IaaS may comprise instances of Microsoft, Windows or Linux running on a virtual computer, or a Desktop-as-a-service (DaaS) provided by Citrix.RTM. or VMWare.RTM.; a PaaS may comprise a database server (e.g., MySQL.RTM. server), Samba server, Apache.RTM. server, Microsoft.RTM. IIS.NET server, Java.RTM. runtime, or Microsoft.RTM. .NET.RTM. runtime, Linux-Apache-MySQL-PHP (LAMP) server, Microsoft.RTM. Azure, or Google.RTM. AppsEngine; a SaaS may comprise SalesForce.RTM., Google.RTM. Apps, or other software application that can be deployed as a cloud service, such as in a web services model. A cloud-computing resource may be a physical or virtual computing resource (e.g., virtual machine). In some embodiments, the cloud-computing resource is a storage resource (e.g., Storage Area Network (SAN), Network File System (NFS), or Amazon S3.RTM.), a network resource (e.g., firewall, load-balancer, or proxy server), an internal private resource, an external private resource, a secure public resource, an infrastructure-as-a-service (IaaS) resource, a platform-as-a-service (PaaS) resource, or a software-as-a-service (SaaS) resource. Hence, in some embodiments, a cloud-computing service provided may comprise an IaaS, PaaS, or SaaS provided by private or commercial (e.g., public) cloud service providers. In some cases, a computer workload may be a container application, a virtual machine-based application, or any computing processing that is performed by a bare metal, virtual machine, a combination of virtual machine or kubernetes-api-server.

[0047] In some cases, a computer workload may be a containerized application (e.g., application container or service containers). The application container may provide tooling for applications and batch processing such as web servers with Python or Ruby, JVMs, or even Hadoop or HPC tooling. Application containers are what developers are trying to move into production or onto a cluster to meet the needs of the business. Methods and systems of the invention will be described with reference to embodiments where container-based

virtualization (containers) is used. However, it is to be understood that the container application is just an example workload, without suggesting any limitation as to the scope of the invention. The methods and systems can be applied to any type of workload provided by any type of systems (e.g., containerized application, unikernel adapted application, operating-system-level virtualization or machine level virtualization).

[0048] In some cases, a workload may be integrated with an embedded secure agent. The secure agent may interact with a backend security management system to provide telemetry and/or to fetch the security policy for the workload. A secure agent can be referred to as an embedded secure agent which are interchangeably used throughout the specification.

[0049] **FIG. 1** schematically shows an example of a workload 100. The workload 100 can be any workload as described above where a security agent can be embedded. For example, a workload may be an application consisting of either a single process or a collection of processes working together to provide a service. User space can be a system memory allocated to running applications while the kernel provides certain services. The user space may be used by libraries and application programs. In an example, programs in user space may contain system calls that ask the kernel to perform certain functions, and the kernel may perform the functions as requested or return an error code. The kernel services and core of the kernel may or may not run in a higher privileged mode.

[0050] An application or workload may have layers of software as shown in **FIG. 1**. As illustrated in **FIG. 1**, a workload such as an application 100 may comprise an event handling logic 101 (e.g., application event handling logic core) that processes incoming events. The event handling logic 101 may be coupled to the core application libraries layer 103 that helps process the events such as storing data in a database, querying a database, or communicating with another process and the like. The core application libraries layer 103 may be statically linked into the process binary. The core application library may comprise application code.

[0051] The OS (operating system) libraries layer 105 may provide access to system resources such as memory, storage, and network. The OS libraries layer 105 may comprise, for example, the Secure Socket Layer (SSL) APIs for cryptographic operations. The OS libraries layer 105 may comprise (C Runtime) libraries such as GLibC and OpenSSL on Unix like Operating Systems, and msvcrt (Microsoft Visual C/C++ Runtime) and OpenSSL on the Windows Operating Systems. The OS libraries layer 105 may provide a set of standards, backward compatible, clean APIs for applications programs. Besides the OS provided system

libraries, other important libraries (e.g., 'matlab') may also be included in the user space. It should be noted that the provided security model can be integrated with other entities such as DPDK (Intel's DataPath Development Kit) which is a mechanism to bypass/augment the kernel networking stack. These APIs may call kernel provided System Calls 107 to access the system resources. Both the OS libraries and system call layer code may still run in the context of the application process, although the system calls may be executed in privileged mode. In some cases, libraries in the OS (operating system) libraries layer 105 may be dynamically linked to the process during its execution.

[0052] The workloads can be ephemeral, dynamic, or unpredictable. In some cases, the workload may be a container application. Container applications are a type of computer application that leverages a form of virtualization. Unlike virtual machines, in which both a guest operating system kernel and the user space are virtualized on a per-virtual machine basis, container applications utilize separate user spaces with a shared host operating system kernel space.

[0053] Container based microservices are often used as modular, open source pieces that may be combined and shared to build an application. For example, container applications for performing specific or common tasks may be created and then shared in repositories so that unrelated entities may take and incorporate the container applications into their own computing environments.

[0054] The user space may affect applications. The user space is where the language runtimes exist. For example, the user space may be packaged up inside the container image and shared between developers, architects, and system administrators. In some cases, a host operating system may allow for multiple isolated user-space instances (e.g., containers using isolated kernel namespaces) to run in host operating system (e.g., a single operating system instance).

[0055] The provided security solution may integrate firewall functions with the workload (e.g., container application), where the data is clean. Clean data may refer to clean and unencrypted data. In some cases, clean data may include, without limitation, application contextual data and/or associated metadata. For example, the clean data may include process lineage, process's executable a.k.a binary image version/checksum, operating-system type/version, environment variables of an application, container or virtual-machine metadata

(e.g., container's image id/version), zone or vCenter tags associated with the application and/or kubernetes selectors associated with the application's running instance and the like.

[0056] A network firewall may enforce a set of rules that govern what data streams, or packets, are permitted to enter or leave a network based on one or more fields in the packet. Such firewall function is traditionally performed by optimized independent entities that sit in the path of the data packet, such as Palo Alto Next Generation Firewall, Cisco Advanced Security Appliance, or OS native packet filters (IPTables in Linux, or NetSh Firewall in Windows). However, such traditional approach may not work well with workloads that are ephemeral, dynamic, or unpredictable.

[0057] The present disclosure provides a built-in security layer (i.e., secure agent) in a workload providing an interface where security policies can be applied, and which further interacts with a backend security management system to provide telemetry and/or fetch the security policy for the workload. The firewall policy enforced by the secure agent may be the same as existing firewall policies. For example, the firewall policy can include any security policies including but not limited to, a simple access policy (permit/deny, or rate-limiting), or a simple anomaly engine that works in concert with the backend system (e.g. detect/alert/mitigate data exfiltration when more than normal amount of data is seen to be transferred outside). Alternatively or in addition to, the firewall policy can include policies based on data of layer-7 (Application Layer of Open Systems Interconnection Network model) information such as http headers, URL (RPC/REST-api endpoint, topics, or application signatures, metadata and other data extracted from the workload.

[0058] FIG. 2 schematically illustrates a workload with a secure agent 200. The secure agent 210 may be embedded into the workload by injecting a shim layer into the workload. A shim can be a small library that transparently intercepts API calls and changes the arguments passed, handles the operation itself or redirects the operation elsewhere. In some embodiments, the shim layer may be located between an application libraries layer of the workload and an operating system service layer. For example, the secure agent 210 may be injected at the interface between the core application libraries layer 203 and the operating system libraries layer 205. The core application libraries layer 203 and the operating system libraries layer 205 can be the same as the core application libraries layer and OS libraries layer as described in FIG. 1. Similarly, event handling logic 201 and system calls 207 can be the same as the event handling logic and system calls as described in FIG. 1.

[0059] The secure agent 210 or shim layer may function as the OS libraries layer to interface the core application libraries layer, providing the same API constructs, and call the OS services APIs on the application's behalf and the like. Alternatively or in addition to, the secure agent may be a process intercepting network connection sourced from or destined to the application.

[0060] The secure agent 210 may be used to implement network firewall functionality to provide security in a network. For example, the secure agent 210 or shim layer may provide wrapper APIs/routines to create and destroy network sockets, and to send and receive data to/from these sockets. In some cases, the secure agent may be a library that is built into the application binary. Since the shim layer sits above the OpenSSL library as well, it may have access to the clean data where the data is clean and/or unencrypted. As described above, clean data may include, for example, application contextual data and/or associated metadata. For instance, the clean data may include process lineage, process's executable a.k.a binary image version/checksum, operating-system type/version, environment variables of an application, container or virtual-machine metadata (e.g., container's image id/version), zone or vCenter tags associated with the application and/or kubernetes selectors associated with the application's running instance and the like. The shim layer may apply policies at Layer-7 of the OSI (Open Systems Interconnection) Network model, such as at a REST API level, at gRPC (Remote Procedure Calls) transaction level, at the level of Kafka® topics, SQL queries and various others. Components of the secure agent and the firewall functions performed by the secure agent are described later herein.

[0061] The provided security solution may be implemented by employing a plurality of built-in secure agents in communication with a backend secure management system. **FIG. 3** schematically illustrates an environment 300 in which a secure management system and secure agents are operated. As described above, a workload may be integrated with a secure agent 303-1, 303-2, 303-3. The secure agent(s) may communicate with the secure management system 301 over a network 310. In some cases, the secure agent may be configured to apply security policies, provide real-time telemetry to the secure management system 301 for further analysis and/or fetch security policies from the secure management system 301. Real-time as used herein may generally refer to a response time of less than 1 second, tenths of a second, hundredths of a second, or a millisecond, such as by a computer processor.

[0062] In some cases, realtime telemetry data may be captured by the secure agent. The telemetry data may be used to derive security policies in an automatic fashion. The telemetry data may be application contextual data and/or flow telemetry data that captures rich contextual information about the communicating peers. Flow telemetry data may contain ephemeral information such as flow time stamp and/or statistics, environment, process binary or parent binary, process arguments and other associated run-time tags. The telemetry data such as application contextual data can be fetched using various mechanisms. For example, the secure agent may utilize the netlink and linux /proc/<pid> tree to find out about an intercepted flow's origination or destination process (application) and use that information to glean context about the process from the system. In another example, the secure agent may fetch its own application/process context in a native manner as the secure agent is co-resident and runs inside the process run-time space. In another example, the secure agent may be in communication with another process such as interceptor process which provides data path redirect for intercepting flows and applying policy on these flows. For instance, the secure agent may establish a side channel to the interceptor and send its local process application context in a serialized manner to the interceptor which caches this data stream. The secure agent may also send flow tuple information (e.g. TCP/UDP flow case local and remote ip address and layer-4 port numbers) in a serialized manner to the interceptor process via the side channel. The information on the side channel may be sent using a dedicated connection to the interceptor and can be different from the application's socket (flow channel). The interceptor may then retrieve the tuple information from the intercepted flow and look-up the information gleaned from the side channel to derive corresponding process's application contextual data. In a further example, the application context data may be sent in-band on the flow socket channel such as via the application data channel without using a side channel, to the interceptor. The interceptor may glean the application contextual data from this flow's byte stream, use it for authentication/authorization and remove it from the stream before forwarding the rest of the streamed data on this data flow socket channel to its destination.

[0063] In certain scenarios such as when employing Kubernetes, a set of containers may share the network namespace but not share the application process and root-file-system where the interception cannot be applied directly. In such cases, a slave agent can be employed which shares the namespace of the application process. The slave agent may establish a control channel communication with the master secure agent and provide the application contextual data on an on-demand basis in response to a request from the master

agent. The above-mentioned slave agent and/or method can be well-applied to a distributed set of compute nodes in addition to the namespace and Kubernetes scenarios. For example, in the case of virtual machine/bare metal (VM/BM), network flow interception may be performed by providing a default gateway in the application server network, requesting the application contextual data from another remote VM/BM server, and providing data-path functionality along with enhanced security based on the application contextual data (i.e., telemetry data).

[0064] In some embodiments, the secure management system 301 may be configured to provide security policy management, behavior analysis of workloads, audit and alerting system to an operator/administrator, and various others. For example, the secure management system 301 may update Blacklist signatures on the fly, manage Blacklist policies (e.g., traditional firewall policies, per API policies, or customized policies) or Whitelist Policies (e.g., Zero Trust policies), discover new signatures from an external threat feed, or update policies based on user (e.g., operator/administrator) provided changes. In some cases, one or more actions may be performed by the secure agent upon executing a security policy. The one or more actions can be performed may include, for example, blocking, alerting, and/or sending related information to secure a host, a virtual machine (VM), and/or a container. The one or more actions may include a list of signaling actions such as: block and/or alert on all traffic and isolate host or VM from the network; block and/or alert on only malware traffic from the host or VM; block and/or alert on malware lateral movement on the network; and send specific security configuration to secure the host, virtual machine or container.

[0065] In some cases, the secure management system 301 may monitor and record a history of flow data at the API/transaction level (e.g., byte count, packet count type information) for audit or forensics. In some cases, the flow data may be tracked by the secure agent and reported/transmitted to the secure management system 301 periodically. In some cases, the secure management system 301 may perform analytics and forensics of the telemetry provided by the embedded secure agents. For instance, the secure management system 301 may extract forensic intelligence from data transmitted by the secure agents. The security management system may, for example, identify anomalous patterns based on baseline data where the baseline data may be application behavior extracted from historical data. The secure management system 301 may utilize any suitable techniques such as statistical clustering/anomaly/correlation, machine-learning or artificial intelligence-based techniques for performing the afore-mentioned tasks.

[0066] The security management system may receive telemetry data transmitted from a secure agent. The telemetry data may be transmitted on pre-determined schedule (e.g., periodically) or upon detection of an event. In some cases, the secure agent may be configured to transmit telemetry data that includes at least flow data and metadata. The flow data may include parsed data such as the destination the application is trying to reach, a network source address, Internet Protocol (IP) address, Media Access Control (MAC) address, Domain Name System (DNS) name, source port, destination address, destination port, protocol type, class of service, and Layer-7 information such as http headers, URL (RPC/REST-api endpoint, topics, or application signatures). In some cases, metadata such as application type, Kubernetes labels, process lineage, binary/executable checksum, environment variables, and the like may also be included in the telemetry data and transmitted to the security management system for further analysis.

[0067] In some cases, the secure management system 301 may be configured to manage the secure agents such as deploying the secure agents and/or registering secure agents. Secure agent may first register with the secure management system 301 to establish a channel for communication with the secure management system 301. During the registration or verification process, system and method of the disclosure may utilize cryptographic technologies such as Transport Layer Security (TLS) or Secure Sockets Layer (SSL) to provide secure connections over the network. According to SSL protocols, session information between endpoints of a connection such as an SSL client and an SSL server are negotiated through a handshake phase and the identity of the SSL server is verified by the SSL client. For example, the initial certificate for this handshake may be embedded into the secure agent and may be refreshed periodically. During the deployment, the secure management system 301 may keep track of all workloads being spawned. In some cases, if a secure agent does not register or is not verified, it may be flagged and quarantined immediately. The secure agent may then store and manage the signed certificate associated with the endpoint or workload in a local database. In some cases, the certificate may be used for establishing secure channel among workloads or secure agents. For instance, the secure agent may verify if the certificate with another workload is a valid certificate based on the digital signature included in the certificate. The handshake can be used to exchange information in a secure way such that the identities and information that policy relies upon cannot be spoofed. This beneficially provides non-repudiation of transferring policy and related information. In an exemplary process, the initial certificate for the handshake maybe

provided to an agent after attestation that agent is a legitimate application allowed to run in the environment where it is currently running. The cryptographic keys associated with the certificate can be generated by the secure-agent and the certificate issued is digitally signed by the secure management system with its own root certificate.

[0068] In some cases, the process of attestation can be done by utilizing host-agents and/or deployment specific agents (e.g., kubernetes-api-server, kubectl agent) or vCenter information. In general, an attestation is a statement by an attestor that a container image is ready for deployment and the attestor may sign the image if it passes the attestation. In some cases, metadata for attestation maybe gathered. The digital signing of the certificate by the secure management system 301 can be achieved by using any suitable techniques such as Intel-SGX or other techniques that provide high level protection for cryptographic function implementation by providing a secure enclave even when such signing application runs in a cloud provider environment (e.g., Amazon-AWS, Microsft-Azue or Google-GCE).Unlike conventional attestation mechanism, the provided attestation may not rely on IP address such that it may be conveniently applied to the scenarios wherein shared-IP is used. Details about attestation are described later herein.

[0069] Generation and verification of the handshake signature (e.g., including at least part of application contextual data and security credentials) can be a time-consuming process. In some cases, the provided system may employ caching to improve the exchange and verification process. For instance, a sender may, for a first time, create a cached entry of the handshake payload and store it for sending in future to a peer node that may or may not in communication with before. This may beneficially allow for a faster exchange of handshake payload as the process of application context fetch, security key generation and attestation with management module may not be required for the future handshake. A receiving peer node may cache the handshake payload after an initial verification. In the subsequent exchange and verification process, the receiving peer may simply compare the cached handshake payload to incoming payload to determine if the handshake payload sent is trust worthy that may not require modification. Alternatively or in addition to, in the subsequent exchange and verification process, a sender may send a session-id (e.g., an assigned ID for a given peer's handshake) and only that session-id may be exchanged instead of the entire payload.

[0070] In some cases, the handshake process may be followed by a short challenge authorization process in order to protect the handshake process or the verification process

against replay attacks by an eavesdropper. For example, the handshake message may be used to send the application context and associated public certificate payload, as well as a randomly generated nonce value. Each peer node may send a digital signature of the sender's nonce and send it back in the final message of handshake process thus proving to the other peer nodes that it has the private key corresponding to the public key sent in the handshake payload's public key section. The public private key can be any type of asymmetric cipher such as RSA or Elliptical curve. The public private key can also be based on a shared-secret based symmetric cipher. The digital signature can utilize any suitable mechanism such as symmetric (shared secret) or asymmetric cipher mechanism using the public/private key based mechanism.

[0071] In some embodiments, the secure agent may employ various mechanisms to identify if the peer side is verified by the security system. For example, the secure agent's IP Addresses may be distributed to the plurality of secure agents of the system by using a backend fanout mechanism. Alternately, flow's data may be analyzed using a carrier sensing approach. For instance, prior to establishing a TCP connection, a server may examine the received payload's digital signature, application context and determine if they match the signature protocol (e.g., proper-prefix of a signature generated by the secure management system) and IP address.

[0072] The security agent may be injected in the application stack in a seamless and frictionless manner. The secure management system 301 may receive a request for transforming/converting a workload into a secure workload. In some cases, the request is initiated by a service provider or endpoint of the network. The secure management system 301 may receive a workload (provided by the service provider) as input and output a secure workload embedded with a secure agent. As described above, the secure agent may be an embedded shim library which is a dynamically linked library that becomes the gateway or access layer to the original OS services library. The shim layer or secure agent can be injected using various suitable approaches.

[0073] For example, the shim layer may be embedded into a workload using a dynamic linking approach. For instance, the shim layer may be injected as a new dynamically linked library into the input container application. The API names and signatures provided by the shim layer should be compliant and imitate the real services library so that changes to the application code are not required. The shim layer may be a replacement for the original OS services library that the container application may go through the shim layer to access the

services offered by the platform (OS) service. For instance, the shim layer may be injected by lazy binding technique. The container application may first bind with the shim layer and then delegate to the system provided library for APIs that are not trapped by the shim layer. The shim layer, after applying prescribed policies and validating the parameters, calls the original APIs for the APIs that are being trapped. This dynamic linking approach works for applications written in various interpreted language such as Python, Perl, Ruby, Java Script, PHP or compiled languages such as C, C++, Java and the like. This dynamic linking approach works well with most of these languages typically run on top of a C runtime which is usually glibc and openssl.

[0074] The secure agent or shim layer may also be injected into statically linked applications (workloads). Various suitable methods may be employed to embed a secure agent into statically linked applications in a seamless or frictionless manner. For example, the secure agent may be embedded by dynamic binary modification method. This method may require minimal changes to the application source code, or in the process by the application is built. For example, since the OS Service library is built statically into the application binary itself, the dynamic binary modification method may be performed by parsing the binary (ELF formatted in case of Linux operating systems) executable, identifying the respective call to the underlying system calls, e.g. kernel services or the equivalents and replacing them with the secure shim wrapper function call. This is beneficial for improving performance by instrumenting at a minimum level in order to provide a firewall function. In another example, an application writer may be provided with a secure agent library and get involved during the application building process. This process may require a re-compile and/or re-linking of application code.

[0075] In some cases, an additional agent functioning as guarantor or co-resident agent may be provided. The co-resident agent may be configured to process workload that does not have an embedded secure agent in order to ensure that policies can be applied to all traffic. The co-resident agent may allow workload without embedded secure agent can also benefit from the distributed firewall and policy paradigm.

[0076] Referring back to FIG. 3, the secure management system 301 may comprise one or more servers and one or more database systems 305 which may be configured for storing or retrieving relevant data. Relevant data may comprise the telemetry data provided by secure agents (e.g., destination the application is trying to reach, the port number, layer-7 information such as http headers, URL, RPC/REST-api endpoint, topics, or application

signatures, metadata such as user), secure agents data (e.g., registration information, verification information, certificates, etc) and various other data as described elsewhere herein. In some instances, the secure management system 301 may receive data from the database systems 305 which are in communication with the one or more external systems.

[0077] In some embodiments, the database systems 305 may comprise a policy database as a repository for network policy or security policy. The policy database may utilize various database structures such as a normalized relational database or NoSQL database. Policies can be established external to a network system. In some cases, a network administrator can manually change the policies. Policies can dynamically change and be conditional on events or workload. In some cases, the policy (table) may be pushed to selected secure agents or all of the secure agents in communication with the security management system.

[0078] The one or more databases 305 may utilize any suitable database techniques. For instance, structured query language (SQL) or "NoSQL" database may be utilized for storing telemetry data provided by secure agents, security policies, certificates or other data. Some of the databases may be implemented using various standard data-structures, such as an array, hash, (linked) list, struct, structured text file (e.g., XML), table, JavaScript Object Notation (JSON), NOSQL and/or the like. Such data-structures may be stored in memory and/or in (structured) files. In another alternative, an object-oriented database may be used. Object databases can include a number of object collections that are grouped and/or linked together by common attributes; they may be related to other object collections by some common attributes. Object-oriented databases perform similarly to relational databases with the exception that objects are not just pieces of data but may have other types of functionality encapsulated within a given object. If the database of the present invention is implemented as a data-structure, the use of the database of the present invention may be integrated into another component such as the component of the present invention. Also, the database may be implemented as a mix of data structures, objects, and relational structures. Databases may be consolidated and/or distributed in variations through standard data processing techniques. Portions of databases, e.g., tables, may be exported and/or imported and thus decentralized and/or integrated.

[0079] Each of the components (e.g., servers, database systems, computer devices, external systems, and the like) may be operatively connected to one another via one or more networks 310 or any type of communication links that allows transmission of data from one component to another. For example, the respective hardware components may comprise network

adaptors allowing unidirectional and/or bidirectional communication with one or more networks. For instance, the servers and database systems may be in communication—via the one or more networks 310—with the endpoint devices, service providers, and/or data sources to transmit and/or receive workloads and relevant data.

[0080] The security management system 301 may be implemented on a server. A server may include a web server, a mobile application server, an enterprise server, or any other type of computer server, and can be computer programmed to accept requests (e.g., HTTP, or other protocols that can initiate data transmission) from a computing device (e.g., user device, other servers) and to serve the computing device with requested data. A server may be a unitary server or a distributed server spanning multiple computers or multiple datacenters. The servers may be of various types, such as, for example and without limitation, web server, news server, mail server, message server, advertising server, file server, application server, exchange server, database server, proxy server, another server suitable for performing functions or processes described herein, or any combination thereof. In addition, a server can be a broadcasting facility, such as free-to-air, cable, satellite, and other broadcasting facility, for distributing data. A server may also be a server in a data network (e.g., a cloud/fog computing network).

[0081] A server may include various computing components, such as one or more processors, one or more memory devices storing software instructions executed by the processor(s), and data. A server can have one or more processors and at least one memory for storing program instructions. The processor(s) can be a single or multiple microprocessors, field programmable gate arrays (FPGAs), or digital signal processors (DSPs) capable of executing particular sets of instructions. Computer-readable instructions can be stored on a tangible non-transitory computer-readable medium, such as a flexible disk, a hard disk, a CD-ROM (compact disk-read only memory), and MO (magneto-optical), a DVD-ROM (digital versatile disk-read only memory), a DVD RAM (digital versatile disk-random access memory), or a semiconductor memory. Alternatively, the methods can be implemented in hardware components or combinations of hardware and software such as, for example, ASICs, special purpose computers, or general purpose computers.

[0082] In some embodiments, the security management system 301 may construct the database 305 in order to deliver the data (e.g., policy data) to the secure agents or users efficiently. For example, the security management system 301 may provide customized algorithms to extract, transform, and load (ETL) the data. In some embodiments, the security

management system 301 may construct the databases using proprietary database architecture or data structures to provide an efficient database model that is adapted to large scale databases, is easily scalable, is efficient in query and data retrieval, or has reduced memory requirements in comparison to using other data structures.

[0083] The security management system may be implemented anywhere in the network. The security management system may be implemented on one or more servers in the network, in one or more databases in the network, one or more service providers or one or more endpoint devices. For example, the security management system may be implemented in a distributed architecture (e.g., a plurality of devices collectively performing together to implement or otherwise execute the security management system or its operations) or in a duplicate manner (e.g., a plurality of devices each implementing or otherwise executing the security management system or its operations as a standalone system). The security management system may be implemented using software, hardware, or a combination of software and hardware in one or more of the above-mentioned components within the network environment 300. The security management system may be implemented as a standalone system or integral to an endpoint of the network (e.g., service provider).

[0084] The network 310 may be a communication pathway between the security management system 301, a workload provider (e.g., endpoints of the network, container host platform, cloud service provider, or other platforms), or other components of the network. The network may comprise any combination of local area and/or wide area networks using both wireless and/or wired communication systems. For example, the network 310 may include the Internet, as well as mobile telephone networks. In one embodiment, the network 310 uses standard communications technologies and/or protocols. Hence, the network 310 may include links using technologies such as Ethernet, 802.11, worldwide interoperability for microwave access (WiMAX), 2G/3G/4G or Long Term Evolution (LTE) mobile communications protocols, Infra-Red (IR) communication technologies, and/or Wi-Fi, and may be wireless, wired, asynchronous transfer mode (ATM), InfiniBand, PCI Express Advanced Switching, or a combination thereof. Other networking protocols used on the network 310 can include multiprotocol label switching (MPLS), the transmission control protocol/Internet protocol (TCP/IP), the User Datagram Protocol (UDP), the hypertext transport protocol (HTTP), the simple mail transfer protocol (SMTP), the file transfer protocol (FTP), and the like. The data exchanged over the network can be represented using technologies and/or formats including image data in binary form (e.g., Portable Networks

Graphics (PNG)), the hypertext markup language (HTML), the extensible markup language (XML), etc. In addition, all or some of links can be encrypted using conventional encryption technologies such as secure sockets layers (SSL), transport layer security (TLS), Internet Protocol security (IPsec), etc. In another embodiment, the entities on the network can use custom and/or dedicated data communications technologies instead of, or in addition to, the ones described above. The network may be wireless, wired, or a combination thereof.

[0085] In another deployment scenario, the secure management system can be a multi-tenant aware providing afore-mentioned services to multiple tenants simultaneously while providing secure isolation of information related to policy and telemetry data among the multiple tenants. For example, the secure management system may allow for automatic policy derivation for each tenant and/or workload individually. Details about deriving policies based on realtime telemetry data are described later herein.

[0086] As mentioned above, a secure management system may include services or applications that run in the cloud or on-premises environment. In some cases, the secure management system may comprise a software-based solution based on fog computing concepts which extends some of the tasks performed by the secure management system closer to the edge (e.g., workloads). Maintaining close proximity to the edge (e.g., workloads, virtual machines, data centers, etc) may significantly reduce overall bandwidth requirements and the cost of managing widely distributed networks. The provided secure management system may employ fog computing paradigm that at least a portion of secure tasks can be performed at the edge. For example, tasks performed by a fog layer between the cloud/SaaS and the secure agent may include, but not limited to, security policy management, behavior analysis of workloads, workload attestation for non-repudiation to attest workloads against known sources of truth (e.g., kubernetes master, VmWare, vCenter), proxying of traffic out of demilitarized zone, sandboxing and live baselining to help with attesting workloads as belonging to the original package of application delivery, co-resident with existing firewall, manage or update the secure agent, and various others as described elsewhere herein. Alternatively, one or more of the aforementioned tasks can be performed by a management console of the secure management system running in the cloud (e.g., cloud application).

[0087] In some cases, the secure agent upgrades can be performed without bringing the workload down or bringing the host link down, thus improving the performance of the network or traffic. The secure agents may be upgraded dynamically without causing service

disruption. In some cases, the upgrade of the secure agent may be performed automatically according to a scheduled upgrade plan. The upgrade plan may comprise, for example, a scheduled date, plan description and one or more entries describing how the upgrade may affect various secure agents on workload side or an action to be performed by the secure agent. The upgrade plan can be managed, modified, created by a user and may be saved in a database accessible to the secure management system.

[0088] The secure agent may be upgraded in a seamless manner without interrupting the traffic or service. The upgrade process may follow a hand off mechanism to ensure the order of packet being sent. For example, the secure agent to be upgraded to a new version may download the newer version based on an event trigger (e.g., a direct service command, a scheduled event according to the upgrade plan). The running version of the guarantor process may fork a child process and execute the new version of guarantor binary on the child process. It may also pass down all the existing file-descriptors corresponding to intercepted socket flows and/or forward socket flows to child process. The child process may be attested with the fog CA and built its internal data-structures based on the file-descriptors to be passed from the parent to the child process, and the child process may declare it is ready to handle flow processing by signaling the parent process. The parent process may not perform read actions from any of the socket file-descriptors during this process while the parent process may finish writing to all the file-descriptors before signaling the child process to take over. Upon receiving the signal from parent, the child process may resume all read/write on the socket's normal manner. At this point the parent process may proceed to close all its file-descriptors and exit to complete the seamless upgrade process.

[0089] FIG. 3A schematically illustrates an example of the secure management system employing a fog layer 321, 323. In the illustrated example, a certificate authority (CA) 321 may be implemented by the fog layer. The certificate authority (CA) certificate may be configured to authenticate the CA signature on the certificate, as part of the authorizations before launching a secure connection or transferring data. A certificate authority or certification authority (CA) is an entity that issues digital certificates. A digital certificate may certify the ownership of a public key by the named subject of the certificate. The certificates may have system defined formats or can be proprietary certificates as described later herein. The fog layer may comprise other fog applications 323 to perform one or more tasks as described above. In some cases, the fog applications, fog CA may be managed by a management console 325 running on the cloud.

[0090] In some cases, the process of attesting that a workload is ready for deployment can be done by utilizing host-agents and/or deployment specific agents in the fog layer (e.g., fog CA 321, 323, kubernetes-api-server, kubectl agent) or vCenter information. For instance, the fog layer may verify the guarantor instance running on a container of a given Pod (basic execution unit of a Kubernetes application) and then check if the container and guarantor conforms to the secure management system fortified case. In an example, a guarantor may perform fog attestation by sending container-id, pid, binary path, bin cksum, and process start time. The fog layer may run a command to verify a guarantor's report. In some cases, a fraudulent guarantor may be detected by catching it sending an invalid container-id multiple times, which does not match the cid/pid/startup-time. In some cases, to provide additional security, a two-factor mechanism may be provided. For example, a guarantor process may open a UDS (Unix domain socket) or inter-process communication socket after sending an initial attestation request, the fog layer may "kubectl exec" to write a random number (e.g., nonce) to the UDS socket, the guarantor may then report the nonce in its final attestation request for authentication. The attestation method can be conveniently extended to VM scenario. For instance, the fog CA may generate a random number to the VM guarantor (AWS EC2 scenario), the VM guarantor may utilize its AWS IID Spec PKCS7 private key to sign it and send it back in the final attestation message, then the fog CA may verify its validity based on public key of the EC2 instance.

[0091] In alternative cases, the guarantor can be attested by the infrastructure and assigned a verifiable cryptographic identity after successful attestation and the trust boundary may be extended to the secure agent process. Subsequently any flow that is intercepted by this secure agent process can provide more granular verification of the co-resident agent.

[0092] During the fortification process, various binaries and libraries installed in the container may be scanned. In some cases, the fortification process may be improved by caching previous layers of scanned binary/library information and overlaying the previous layers with higher layered changes to effectively derive the final container image's scanned binary/library information. The scanned container data base may have a mapping between the a file's absolute path to the associated metadata. The metadata may contain the creation/last-update timestamp, size, checksum of the file content, CVE data-base known vulnerability mapping and the like. A fortified container's scanned database can be stored in the backend.

[0093] During the attestation process, the fog layer may retrieve the scanned data information pertaining to the attesting container and upon successful attestation, the fog layer may pass this information to the guarantor process in the container. For an intercepted flow, when the guarantor process fetches the process information, it may cross-verify the process by checking the process' dynamic binary information such as current binary file's checksum against information retrieved from the scanned-database. When there is a mismatch, the guarantor may alert the end-user about any malicious activity, quarantine the process from communicating or stop the process.

[0094] In some scenarios, a malicious code may not present or create new binaries or spawn a new process from that malicious binary using fork/exec mechanism. Rather, the malicious code may modify the current binary. To prevent such attacks, the memory mapped instruction segments of the running processes can be periodically scanned to determine if any modification has generated by a malicious code.

[0095] FIG. 4 schematically shows a block diagram of an example secure agent 400, in accordance with various embodiments of the invention. In some embodiments, the secure agent 400 may be in communication with a security management system. The secure agent may comprise a shim layer cleanly intercepts all the API calls that the application intends to make to the C runtime (such as libc). The shim layer may be capable of intercepting the communication APIs. Since the C runtime provides access to the resources of the operating system such as memory, storage, network and the like, the secure agent may process all the triggers and provide policy-based access to any of these resources the workload (e.g., application) requests for. The secure agent may secure network access and provide network firewall functionality embedded in the application.

[0096] As described above, the secure agent may be cryptographically secured during the process of embedding in the application binary. Authentication information (e.g., certificates) may be used to verify authenticity by the backend security management system. The authentication information (e.g., certificates) may also be used to establish secure communication channels between secure agents or workloads. For example, an application with embedded secure agent communicating with another application with embedded secure agent, may verify each other by validating the respective cryptographic signatures. In another example, metadata such as process lineage, security labels and/or binary checksum or other selectors such as kubernetes labels or vCenter tags may be digitally signed by each communicating agent's cryptographic keys and may be verified by the peer agent. The

signature may be verified based on a common trusted ancestor certificate which belongs to a digital certificate signed by the secure management system. Failure to verify may result in raising an alert. In some cases, the secure agent may verify if the certificate is a valid certificate based on the digital signature included in the certificate. The secure agent may then store and manage the signed certificate associated with the endpoint/workload in a local database.

[0097] In some embodiments, the secure agent may comprise a compatibility layer such as an API shim layer 409. The API shim layer may be a C Runtime shim layer that emulates the C runtime API provided by libraries such as libc and OpenSSL. This may advantageously allow for minimal change to the application code and can be embedded into the application binary without involvement of the application owner/writer. In some cases, the shim layer may review all API requests, records relevant pieces of data for visibility, monitoring or analytics and/or blocks a set of API requests that violate a specified policy.

[0098] The secure agent may comprise a memory unit to store data related to policy state, firewall state and telemetries. The policy state 403 may hold the policies, including but not limited to whitelist, blacklist, or threat signatures and the like. Traffic originating and destined to the application may be validated against this policy state and corresponding action may be taken. The policy state may be configured at application instantiation time. The policy state may be instantly updated by the backend security management system based on new threat discovery or upon detection of configuration change.

[0099] The firewall state 405 may hold the application security posture and the active firewall state. The firewall state 405 may include flows and statistics, and may perform the firewall function based on the policy configured for the application. This firewall state 405 may be built (at runtime) as the application starts to interact with other applications in the network. This firewall state 405 may also perform analysis such as IDS/IPS (Intrusion Detection/Prevention Systems) analysis to detect and mitigate things like SQL Injection or buffer overflow attacks. In some cases, the firewall state may be transferred using a handshake/challenge protocol.

[00100] The telemetry information 407 may include flow data and metadata (e.g., logs, destination, port number, Layer-7 information such as http headers, URL, RPC/REST-api endpoint, topics, or application signatures, metadata) and other data as described elsewhere

herein. The telemetry information may be pushed to the backend security management systems for further analysis.

[00101] The secure agent 400 may comprise a backend interfacing unit 401 for establishing and maintaining communication with the security management system. For example, the secure agent 400 may discover the security management system on the network and maintain a perpetual secure connection with the security management system so that it can stream telemetry and fetch updated security policy. In the case where the security management system is not reachable, the secure agent 400 may continue to apply the policy it previously had and reconciles with the security management system once the connection is restored. Alternatively or in addition to, policies can be pushed down to agents using a distributed consensus database such as CoreOS-Etcd, Apache-ZooKeeper, Consul or by utilizing a message stream distribution solution such as Apache-Kafka.

[00102] **FIG. 5** shows an example internal process 500 of implementing security policies with a secure agent. The process 500 can provide security policies at the shim layer. The process 500 can be adapted for any policy-based operations (e.g. various security operations). In the example, the workload may be an application that is to send a payload to the network. The application may first make a call (e.g., send()") to the API shim. Next, the API shim's send()" API may be invoked. The API shim may harness the data and metadata such as the destination the application is trying to reach or the port number. The API shim may also parse Layer-7 information such as headers, URL, RPC/REST-api endpoint, topics, or application signatures, and metadata such as application type, Kubernetes labels, and the like. The secure agent may then look up firewall policy and runtime state to determine if this payload is allowed. If it is denied based on the policy, the call may be returned with an error. A firewall policy may include but not limited to, a simple access policy (permit/deny), or a simple anomaly engine that works in concert with the backend system (e.g. detect/alert/mitigate data exfiltration when more than normal amount of data is seen to be transferred outside). The firewall policy may include any other policies as described elsewhere herein. Actions may be taken upon executing the firewall policy. For instance, an error may be returned and/or an alert may be generated when the action is to deny. In case the call passes all checks, the real send() routine implemented by OS service may be invoked to send the payload out. Telemetry data may be recorded and pushed to the telemetry queue. The telemetry data may be transmitted to the backend security management system based on a pre-determined schedule (e.g., periodically) or upon detection of an event. In some cases,

the policy application can be performed during TCP/UDP socket establishment time such as by trapping socket connect() call.

[00103] The provided method and systems may have the ability to apply existing firewall policies as well as firewall policies at a fine-grained level such as per API level, per resource served by the application and obeying principles of least privilege, at an application level, or at a group of applications level. An application group can be identified by a higher level entity such as a Kubernetes label. Unlike the conventional method using IP Addresses or Port Numbers, the provided method and system advantageously allows for applying the policies at the fine controlled level.

[00104] Traditional firewall may apply policies based on contents of the payload data that is being transferred such as Layer-3 (IP addresses) and Layer-4 (TCP/UDP port numbers). For instance, the secure agent may execute an action according to the policy determined based on the source and destination to, for example, permit or allow the flow described in the policy (i.e., forward the communication), block or deny the flow described in the policy (i.e., drop the communication), limit the bandwidth consumed by the flow (i.e., rate-limit), log the flow, "mark" the flow for quality of service (QoS) (e.g., set a lower or higher priority for the flow), redirect the flow (e.g., to avoid critical paths), copy the flow, and various other actions.

[00105] With the application of the provided systems, the security policy can be based on data of Layer-7, other attributes or metadata. The security policy may be applied based on an identity of the entity requesting an access to a resource and/or an identity or type of the resource to be accessed. For example, metadata may include data that is available on the client side that may be lost in translation. For instance, source IP is usually overloaded with data related to location, user, application, and the like when the packet is processed at the server/service side. Such metadata or attributes (e.g., user, process, subnet, geographical location and the like) may be sent as labels in a proprietary encapsulation and used in making policy decisions on the receiving side. In some cases, such metadata or attributes may be encapsulated and transmitted to the security management system by the secure agents. In an example, consider a case where Application-A invokes a Restful API of an endpoint provided by Application-B to retrieve a set of data and the data requested by the application-A is the result of a request coming from User-1. If the policy of allowing the rest API to retrieve data is based on the requestor (i.e., Application-A), it may always be allowed. However, if the policy to retrieve data taking into account the fact that User-1 is the entity

requesting this data, the policy paradigm may allow for fine control that takes into account of useful factors (e.g., user, type of user, identity of user, etc). In the above example, the secure agent of Application-1 may embed the metadata information in the request (e.g., by means of adding http headers for example), which may then be leveraged for policy application by the secure agent on Application-2, before removing them prior to delivering it to the upper layers of Application-2.

[00106] Below is a list of examples of policies can be applied by the provided systems or methods.

[00107] Example 1: Traditional policies

[00108] src ip="1.1.1.1", dst_ip="2.2.2.2", dst_port="3300", action= "permit"

[00109] deny all

[00110] In this example, only one source IP (1.1.1.1) is allowed to communicate to the service running on 2.2.2.2 (represented by port 3300). Everything else is denied.

[00111] Example 2: Deep policies

[00112] src ip="sap-app", dst ="database", action= "permit"

[00113] deny all

[00114] In this example, the policy is based on application/service rather than an IP address. The policy may allow communication between the two applications (and instances thereof). In the previous example, any application on sap-app would have been able to access the database service.

[00115] Example 3: Tight policies

[00116] src ip="sap-app", dst ="database", dst_resource= "employees", action= "permit"

[00117] deny all

[00118] In this example, the policy may restrict further to define which topic/table within database is being authorized for sap-app.

[00119] Example 4: Policies using other attributes

[00120] user=="userA", src ip="sap-app", dst ="database", dst_resource= "employees", action= "permit"

[00121] deny all

[00122] In some embodiments, the secure management system may employ a policy paradigm comprising location-transparent service-policies that can be deployed in multiple environments/locations with minimal input from users. In some cases, the location-transparent service-policies may specify different types of connections such as east-west connections and external north/south connections, permitted between two or more entities. The policies may specify which connections between entities are permitted. If a connection is permitted, an entity may connect to and communicate with its dependency. If a connection is not permitted, then an entity may not connect to or communicate with its dependency. In some cases, in the context of a collection of entities (e.g., a service is a collection of processes), east-west connection may refer to a connection within the collection, a north connection may refer to a connection from outside the collection to inside the collection, and a south connection may refer to a connection from inside the collection to outside the collection. The connection may be unidirectional. For example, when two entities need to initiate connections with each other, two connections may need to be permitted between them as specified by the policies.

[00123] A service may be a collection of communicating processes/nested-services. The processes/nested-services in a service may have north, south, and east-west connections. A service-policy may specify the connections permitted in a service. A service may also be referred to as an application. For east-west connections, the entities are all within the service and may be identified by the service-policy. For north and south connections, one entity is in the service thus can be identified by the service-policy while the external entity may not be identified by the service-policy, the service-policy may use placeholders for such external entities.

[00124] In some embodiments, the provided policy paradigm may be whitelist policies that are used to enforce legitimate communication within a network/system. The location-transparent service policies can be reused in various environments. For example, services can run in various environments such as production, staging, test, per-developer, and other different stages. Environments may also be different based on geographic regions, or subdivided by geographic region such as ‘production.east’, ‘production.west’, ‘staging.us’, ‘staging.eu’, and the like. A service can be instantiated in multiple environments, and set up to have dependencies in the same environment or in different environments. An environment may be a collection of services. The collection services in an environment may have north, south, and east-west connections. An environment-policy may specify the connections

permitted in an environment. Environment-policies can be used to create environments. The name of the environment policy may not need to match the name of the created environment. An environment-policy can be used to create multiple environments. For example, developers may recreate their own instance of a production environment for development/debugging.

[00125] The location-transparent service-policies can be retrieved and deployed in an automated fashion. For example, users may specify an environment tag and a service-name tag when deploying their services. Interceptor-processes report may be generated with these two tags as well as process-related information discovered by the interceptor-processes. Using these tags, the secure management system may be able to monitor all flows within the service (e.g., east-west flows), flows from outside the service to inside the service (e.g., north flows), and flows from inside the service to outside the service (e.g., south flows). Placeholders may be created for north and south flows.

[00126] The provided policy paradigm may allow for dynamic composition of location-transparent policies based on existing or deployed policies/services. The location-transparent policies can be composed to conform to an existing policy or deployed service. For instance, a set of location-transparent policies may be composed dynamically to create a new location-transparent policy that mirrors the composing of deploy services. The newly composed service policies may be flattened and sent to the interceptor-processes for enforcement.

[00127] The secure agent may interact with a backend security management system to provide telemetry and/or to fetch the security policy for the workload. In some cases, the security policy may be derived automatically based on realtime telemetry data. For example, the contextual information about the communication peers extracted from the flow telemetry data may be used to derive a whitelist policy which can then be used to enforce legitimate communication with the network. As described above, the telemetry data may be application contextual data and/or flow telemetry data that captures rich contextual information about the communicating peers. Flow telemetry data may contain ephemeral information such as flow time stamp and/or statistics, environment, process binary or parent binary, process arguments and other associated run-time tags. In an example process of fetching the security policy, a pair of flow telemetry (coming from sender and receiver side of a secure agent) may be captured and normalized to remove the ephemeral information or ephemeral selectors. A set of sender selectors and receiver selectors may be created to form a mapping from one set of selectors associated with the sender (e.g., source selectors) to a set of selectors associated

with the receiver (e.g., destination selectors). The occurrence of the same or similar mappings may be counted. In some cases, the mappings (source-destination selector mappings) may be directly converted into a policy where the source-destination selector mapping may become an ingress policy at the destination and/or egress policy at the source. These policies may be layer-7 policy that may contain rich contextual information (e.g. source/dest container image, source/dest container's kubernetes pod selectors, source/dest process binary, path, cmd line arguments) compared to conventional policies. Next, data mining algorithms such as Apriori algorithm may be applied to create strong policies using association which have minimum support and use minimum threshold for admissible policies. This mechanism can be useful to identify strong association in a communication pattern.

[00128] The provided systems and methods may be capable of converting insecure workload into secure workload seamlessly and dynamically. For example, application services that interact with other services over open channels may automatically be converted to secure/encrypted communications without the need to change the application core logic or the source code. **FIG. 6** shows an example of converting insecure workload 610 into a secure workload 620. In this case, an application making a connect() API call is intercepted and translated to an OpenSSL API to invoke a secure socket (which also acts as a digital signature for the purposes of non-repudiation). Alternatively, the secure agent could potentially leverage advanced functionality such as socket-redirect and Kernel TLS to offload the encryption work. The secure agent may be configured to decide such functionalities based on the platform (e.g., workload provider) capabilities and ability to offload. If the other end of the socket is an application with a secure agent, it may accept the secure connection, decrypt the traffic and provide clear data to the receiving application without having to change the application.

[00129] **FIG. 8** schematically shows various deployments of the secure agent and an interceptor. In the examples, the workload is an application. As illustrated in the first scenario **810**, a secure agent shim **811** may be embedded in the application. The secure agent may perform various functions such as fetching the context of the application, enforcing a policy, data path functionalities such as performing deep-packet operation (e.g., packet interception, packet inspection) as described elsewhere herein. In the second scenario **820**, the secure agent is implemented as a secure agent interceptor process **821** which may intercept networks connections, apply policies and fetch the application contextual data using

suitable techniques (e.g., netstat and /proc walks). The interceptor process can be the same as the interceptor process as described elsewhere herein. In the third scenario **830**, a combination of the secure agent shim **831** and a secure agent interceptor process **833** may be employed in the security management system. The data path functionality (e.g., performing deep-packet operation) may be performed by the secure agent interceptor process **833** and control path functionalities such as application contextual data fetching may be performed by the embedded secure agent shim **831.** In a multi container scenario (e.g. Kubernetes pod with multiple containers) **840**, a master-slave secure agent mechanism may be employed. the data path functionality may be performed by the master secure agent **841** (e.g., master secure agent interceptor process) which shares the network namespace with the application. The control path functionalities such as fetching application context may be performed via a control channel by the slave secure agent **843** (e.g., slave secure agent interceptor process).

Computer systems

[00130] The security management system or processes described herein can be implemented by one or more processors. In some embodiments, the processor may be a processing unit of a computer system. **FIG.** 7 shows a computer system 701 that is programmed or otherwise configured to implement the security management system. The computer system 701 can regulate various aspects of the present disclosure. The computer system 701 can be an electronic device of a user or a computer system that is remotely located with respect to the electronic device. The electronic device can be a mobile electronic device.

[00131] The computer system 701 includes a central processing unit (CPU, also "processor" and "computer processor" herein) 705, which can be a single core or multi core processor, or a plurality of processors for parallel processing. The computer system 701 also includes memory or memory location 710 (e.g., random-access memory, read-only memory, flash memory), electronic storage unit 715 (e.g., hard disk), communication interface 720 (e.g., network adapter) for communicating with one or more other systems, and peripheral devices 725, such as cache, other memory, data storage and/or electronic display adapters. The memory 710, storage unit 715, interface 720 and peripheral devices 735 are in communication with the CPU 705 through a communication bus (solid lines), such as a motherboard. The storage unit 715 can be a data storage unit (or data repository) for storing data. The computer system 701 can be operatively coupled to a computer network ("network") 730 with the aid of the communication interface 720. The network 730 can be

the Internet, an internet and/or extranet, or an intranet and/or extranet that is in communication with the Internet. The network 730 in some cases is a telecommunication and/or data network. The network 730 can include one or more computer servers, which can enable distributed computing, such as cloud computing. The network 730, in some cases with the aid of the computer system 701, can implement a peer-to-peer network, which may enable devices coupled to the computer system 701 to behave as a client or a server.

[00132] The CPU 705 can execute a sequence of machine-readable instructions, which can be embodied in a program or software. The instructions may be stored in a memory location, such as the memory 710. The instructions can be directed to the CPU 705, which can subsequently program or otherwise configure the CPU 705 to implement methods of the present disclosure. Examples of operations performed by the CPU 705 can include fetch, decode, execute, and writeback.

[00133] The CPU 705 can be part of a circuit, such as an integrated circuit. One or more other components of the system 701 can be included in the circuit. In some cases, the circuit is an application specific integrated circuit (ASIC).

[00134] The storage unit 715 can store files, such as drivers, libraries and saved programs. The storage unit 715 can store user data, e.g., user preferences and user programs. The computer system 701 in some cases can include one or more additional data storage units that are external to the computer system 701, such as located on a remote server that is in communication with the computer system 701 through an intranet or the Internet.

[00135] The computer system 701 can communicate with one or more remote computer systems through the network 730. For instance, the computer system 701 can communicate with a remote computer system of a user (e.g., a user device). Examples of remote computer systems include personal computers (e.g., portable PC), slate or tablet PC's (e.g., Apple® iPad, Samsung® Galaxy Tab), telephones, Smart phones (e.g., Apple® iPhone, Android-enabled device, Blackberry®), or personal digital assistants. The user can access the computer system 701 via the network 730.

[00136] Methods as described herein can be implemented by way of machine (e.g., computer processor) executable code stored on an electronic storage location of the computer system 701, such as, for example, on the memory 710 or electronic storage unit 715. The machine executable or machine readable code can be provided in the form of software. During use, the code can be executed by the processor 705. In some cases, the code can be

retrieved from the storage unit 715 and stored on the memory 710 for ready access by the processor 705. In some situations, the electronic storage unit 715 can be precluded, and machine-executable instructions are stored on memory 710.

[00137] The code can be pre-compiled and configured for use with a machine having a processer adapted to execute the code, or can be compiled during runtime. The code can be supplied in a programming language that can be selected to enable the code to execute in a pre-compiled or as-compiled fashion.

[00138] Aspects of the systems and methods provided herein, such as the computer system 701, can be embodied in programming. Various aspects of the technology may be thought of as "products" or "articles of manufacture" typically in the form of machine (or processor) executable code and/or associated data that is carried on or embodied in a type of machine readable medium. Machine-executable code can be stored on an electronic storage unit, such as memory (e.g., read-only memory, random-access memory, flash memory) or a hard disk. "Storage" type media can include any or all of the tangible memory of the computers, processors or the like, or associated modules thereof, such as various semiconductor memories, tape drives, disk drives and the like, which may provide non-transitory storage at any time for the software programming. All or portions of the software may at times be communicated through the Internet or various other telecommunication networks. Such communications, for example, may enable loading of the software from one computer or processor into another, for example, from a management server or host computer into the computer platform of an application server. Thus, another type of media that may bear the software elements includes optical, electrical and electromagnetic waves, such as used across physical interfaces between local devices, through wired and optical landline networks and over various air-links. The physical elements that carry such waves, such as wired or wireless links, optical links or the like, also may be considered as media bearing the software. As used herein, unless restricted to non-transitory, tangible "storage" media, terms such as computer or machine "readable medium" refer to any medium that participates in providing instructions to a processor for execution.

[00139] Hence, a machine readable medium, such as computer-executable code, may take many forms, including but not limited to, a tangible storage medium, a carrier wave medium or physical transmission medium. Non-volatile storage media include, for example, optical or magnetic disks, such as any of the storage devices in any computer(s) or the like, such as may be used to implement the databases, etc. shown in the drawings. Volatile storage

media include dynamic memory, such as main memory of such a computer platform. Tangible transmission media include coaxial cables; copper wire and fiber optics, including the wires that comprise a bus within a computer system. Carrier-wave transmission media may take the form of electric or electromagnetic signals, or acoustic or light waves such as those generated during radio frequency (RF) and infrared (IR) data communications. Common forms of computer-readable media therefore include for example: a floppy disk, a flexible disk, hard disk, magnetic tape, any other magnetic medium, a CD-ROM, DVD or DVD-ROM, any other optical medium, punch cards paper tape, any other physical storage medium with patterns of holes, a RAM, a ROM, a PROM and EPROM, a FLASH-EPROM, any other memory chip or cartridge, a carrier wave transporting data or instructions, cables or links transporting such a carrier wave, or any other medium from which a computer may read programming code and/or data. Many of these forms of computer readable media may be involved in carrying one or more sequences of one or more instructions to a processor for execution.

[00140] The computer system 701 can include or be in communication with an electronic display 735 that comprises a user interface (UI) 740 for providing, for example, a graphical user interface for displaying security analytics captured by one or more secure agents or the security management system. Examples of UI's include, without limitation, a graphical user interface (GUI) and web-based user interface.

[00141] Methods and systems of the present disclosure can be implemented by way of one or more algorithms. An algorithm can be implemented by way of software upon execution by the central processing unit 705. The algorithm can, for example, trained models such as interpreters, sentimental analysis module.

[00142] While preferred embodiments of the present invention have been shown and described herein, it will be obvious to those skilled in the art that such embodiments are provided by way of example only. It is not intended that the invention be limited by the specific examples provided within the specification. While the invention has been described with reference to the aforementioned specification, the descriptions and illustrations of the embodiments herein are not meant to be construed in a limiting sense. Numerous variations, changes, and substitutions will now occur to those skilled in the art without departing from the invention. Furthermore, it shall be understood that all aspects of the invention are not limited to the specific depictions, configurations or relative proportions set forth herein which depend upon a variety of conditions and variables. It should be understood that various

alternatives to the embodiments of the invention described herein may be employed in practicing the invention. It is therefore contemplated that the invention shall also cover any such alternatives, modifications, variations or equivalents. It is intended that the following claims define the scope of the invention and that methods and structures within the scope of these claims and their equivalents be covered thereby.

CLAIMS

WHAT IS CLAIMED IS:

1. A system for securing a computer workload within a network comprising:
   (a) a secure agent, wherein the secure agent comprises an interfacing layer embedded into the computer workload for intercepting application programming interface (API) calls and wherein the secure agent is configured to apply a security policy based at least in part on the API calls intercepted by the interfacing layer; and
   (b) a server in communication with the secure agent, wherein the server is configured to analyze telemetry data captured by the secure agent to secure the computer workload.

2. The system of claim 1, wherein the computer workload is a container application or a computing processing that is performed by a bare metal, virtual machine or a combination of Kubernetes and server.

3. The system of claim 1, wherein the interfacing layer is a shim layer that is located between an application libraries layer of the computer workload and an operating system service layer.

4. The system of claim 1, wherein the interfacing layer comprises a dynamically linked library.

5. The system of claim 1, wherein the telemetry data comprises application contextual data, metadata or flow telemetry data.

6. The system of claim 1, wherein the server is configured to perform behavioral analysis of the computer workload based on the telemetry data.

7. The system of claim 1, wherein the server is configured to monitor and record a history of the telemetry data at an API level.

8. The system of claim 1, wherein the security policy is applied based on an identity of an entity requesting an access to a resource or an identity of the resource to be accessed.

9. The system of claim 1, wherein the security policy specifies a connection between two entities in the network.

10. The system of claim 1, wherein the security policy comprises a service policy specifying a connection in a service.

11. The system of claim 1, wherein the security policy comprises an environment policy specifying a connection permitted in an environment.

12. The system of claim 1, wherein the security policy is applied at Layer-7 of the OSI (Open Systems Interconnection) Network model.

13. The system of claim 1, wherein the security policy is automatically derived from the telemetry data.

14. The system of claim 1, wherein the secure agent is cryptographically secured.

15. The system of claim 1, wherein the secure agent is in communication with an interceptor, and wherein the interceptor is configured to intercept a data flow and apply the security policy on the data flow.

16. The system of claim 15, wherein the interceptor and the secure agent forms a master-slave relationship.

17. The system of claim 1, wherein the secure agent comprises a memory unit to store data related to a policy state, a firewall state and the telemetry data.

18. The system of claim 1, wherein the secure agent is configured to perform packet interception or packet inspection.

19. A method for securing a computer workload within a network comprising:

(a) receiving the computer workload in the network;

(b) embedding a secure agent into the computer workload, wherein the secure agent comprises an interfacing layer for intercepting application programming interface (API) calls; and

(c) applying, based at least in part on the API calls intercepted by the interfacing layer, a security policy to secure the computer workload.

20. The method of claim 19, wherein the computer workload is a container application or a computing processing that is performed by a bare metal, virtual machine or a combination of Kubernetes and server.

21. The method of claim 19, wherein the interfacing layer is a shim layer that is located between an application libraries layer of the computer workload and an operating system service layer.

22. The method of claim 19, wherein the secure agent is configured to capture telemetry data and transmit the telemetry data to a server for further analysis.

23. The method of claim 22, wherein the telemetry data comprises application contextual data, metadata or flow telemetry data.

24. The method of claim 22, further comprising processing the telemetry data and granting an access to a resource within the network based on the security policy.

25. The method of claim 24, wherein the security policy is applied based on an identity of the entity requesting the access to the resource or an identity of the resource to be accessed.

26. The method of claim 19, wherein the security policy is automatically derived from telemetry data captured by the secure agent.

27. The method of claim 22, wherein the server is configured to perform behavior analysis on the computer workload based on the telemetry data.

28. The method of claim 22, wherein the server is configured to monitor and record a history of the telemetry data at an API level.

29. The method of claim 19, wherein embedding the secure agent comprises using a dynamic linking approach, a lazy binding technique or a dynamic binary modification method.

30. The method of claim 19, further comprising registering and verifying the secure agent cryptographically via a handshake process.

31. The method of claim 30, further comprising using a nonce to protect the handshake process against a replay attack.

32. The method of claim 30, wherein the handshake process comprises verifying the secure agent by a peer node.

33. The method of claim 19, wherein the interfacing layer comprises a dynamically linked library.

34. The method of claim 19, wherein the security policy is applied at Layer-7 of the OSI (Open Systems Interconnection) Network model.

35. The method of claim 19, wherein the security policy specifies a connection between two entities in the network.

36. The method of claim 35, wherein a placeholder is used for specifying a connection with an entity outside of the network.

37. The method of claim 19, wherein the security policy comprises a service policy specifying a connection in a service.

38. The method of claim 19, wherein the security policy comprises an environment policy specifying a connection permitted in an environment.

39. The method of claim 19, wherein the security policy is a location-transparent policy.

40. The method of claim 19, wherein the security policy is composed of a set of security policies based on an existing service.

41. The method of claim 19, wherein the secure agent is in communication with an interceptor, and wherein the interceptor is configured to intercept a data flow and apply the security policy on the data flow.

42. The method of claim 41, wherein the interceptor and the secure agent forms a master-slave relationship.

43. The method of claim 41, wherein the data flow comprises application contextual data.

44. The method of claim 19, wherein the secure agent comprises a memory unit to store data related to a policy state, a firewall state and a telemetry data.

45. The method of claim 19, wherein the secure agent is configured to perform packet interception or packet inspection.

46. The method of claim 19, further comprising attesting the workload using a nonce and a security key through an inter-process communication socket.

47. The method of claim 19, further comprising performing one or more actions upon executing the security policy, and wherein the one or more actions comprises alerting a malware.

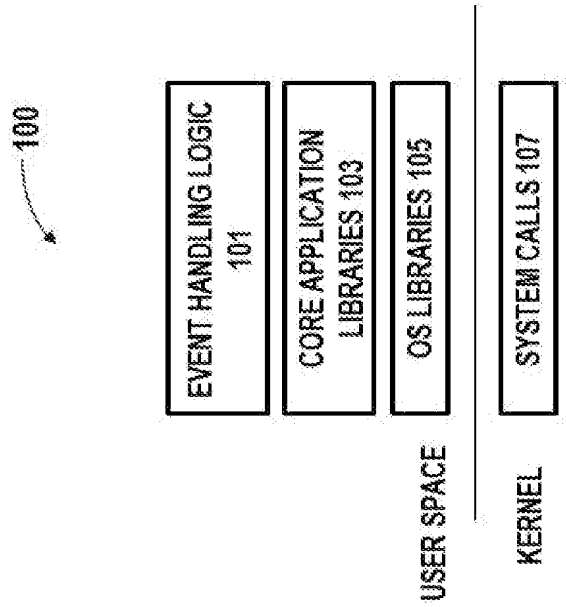

100
EVENT HANDLING LOGIC 101
CORE APPLICATION LIBRARIES 103
OS LIBRARIES 105
SYSTEM CALLS 107
USER SPACE
KERNEL

FIG. 1

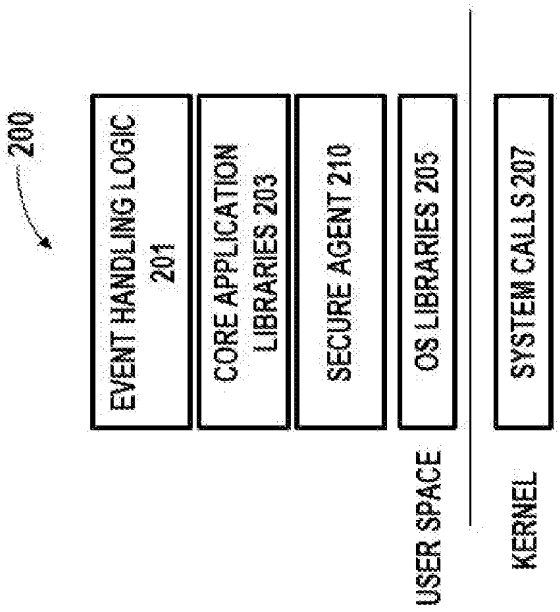

200
EVENT HANDLING LOGIC 201
CORE APPLICATION LIBRARIES 203
SECURE AGENT 210
OS LIBRARIES 205
SYSTEM CALLS 207
USER SPACE
KERNEL

FIG. 2

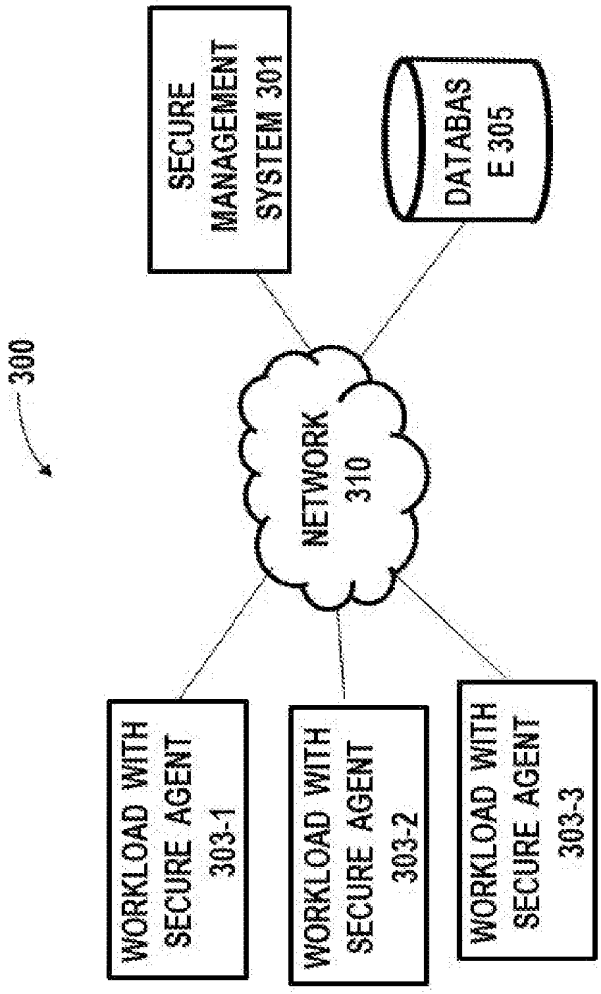

300
SECURE MANAGEMENT SYSTEM 301
DATABAS E 305
NETWORK 310
WORKLOAD WITH SECURE AGENT 303-1
WORKLOAD WITH SECURE AGENT 303-2
WORKLOAD WITH SECURE AGENT 303-3

FIG. 3

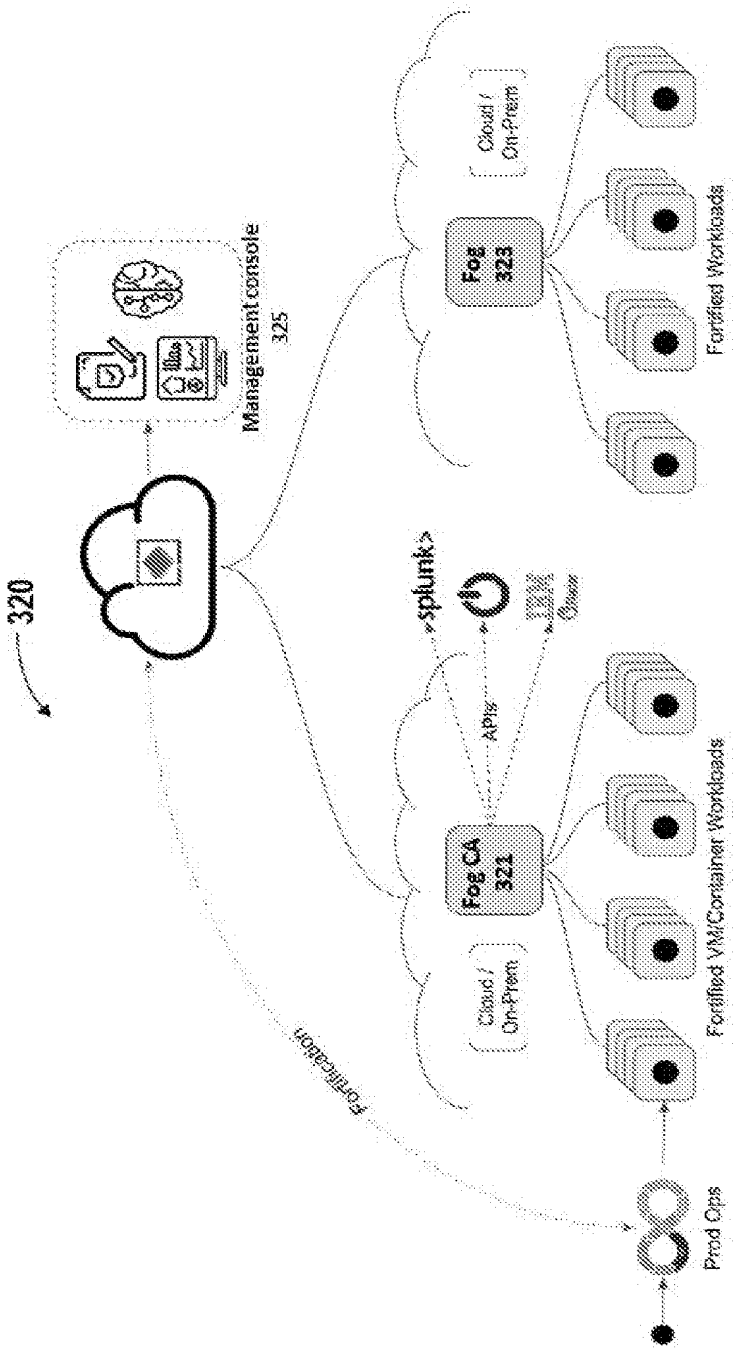

320
Management console
325
Fog
323
Cloud /
On-Prem
Fortified Workloads
splunk>
APIs
Fog C4
321
Cloud /
On-Prem
Fortified VM/Container Workloads
Prod Ops
Fortification

FIG. 3A

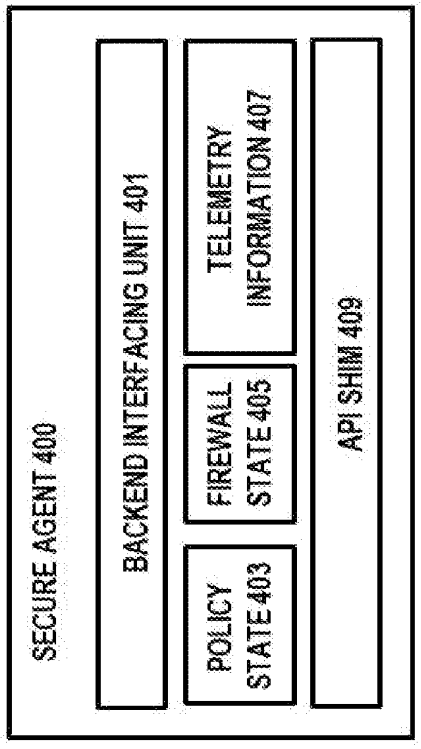

SECURE AGENT 400
BACKEND INTERFACING UNIT 401
POLICY STATE 403
FIREWALL STATE 405
TELEMETRY INFORMATION 407
API SHIM 409

FIG. 4

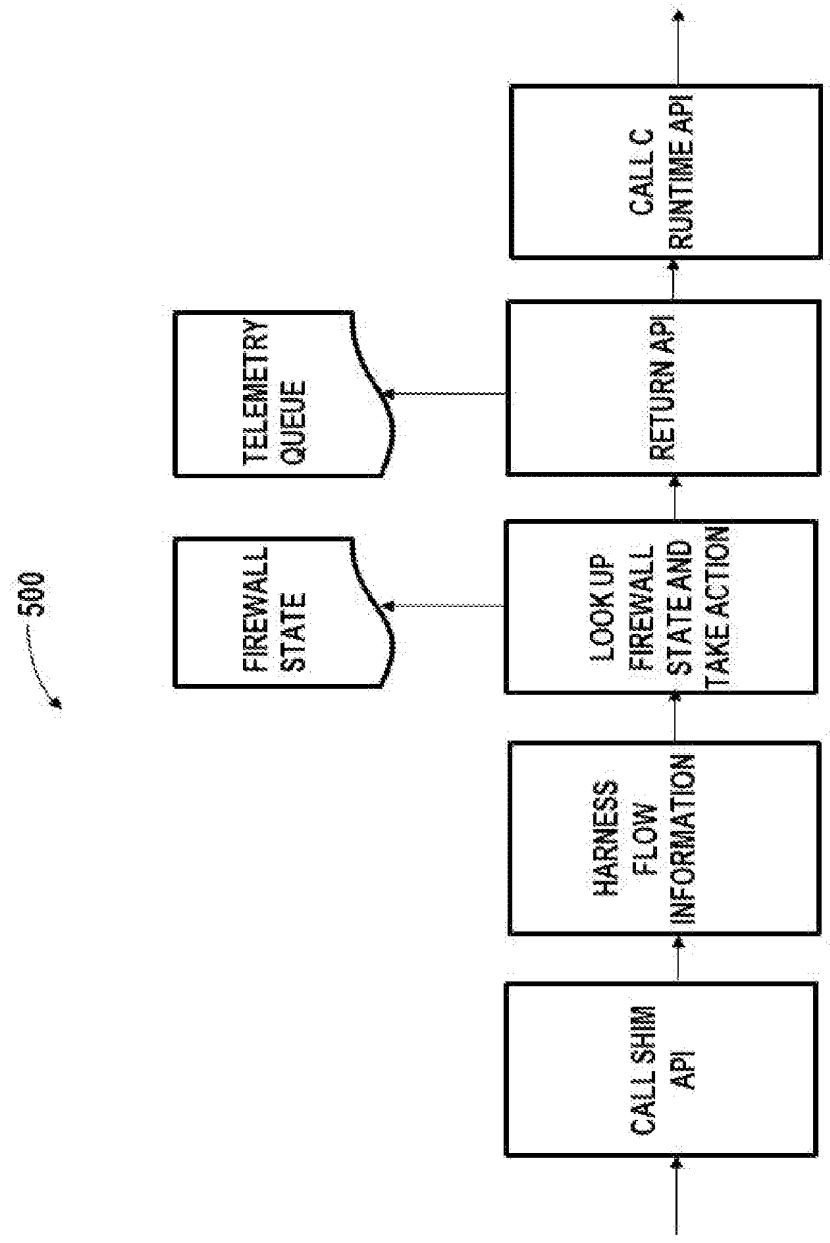

500
CALL SHIM API
HARNESS FLOW INFORMATION
LOOK UP FIREWALL STATE AND TAKE ACTION
FIREWALL STATE
RETURN API
TELEMETRY QUEUE
CALL C RUNTIME API

FIG. 5

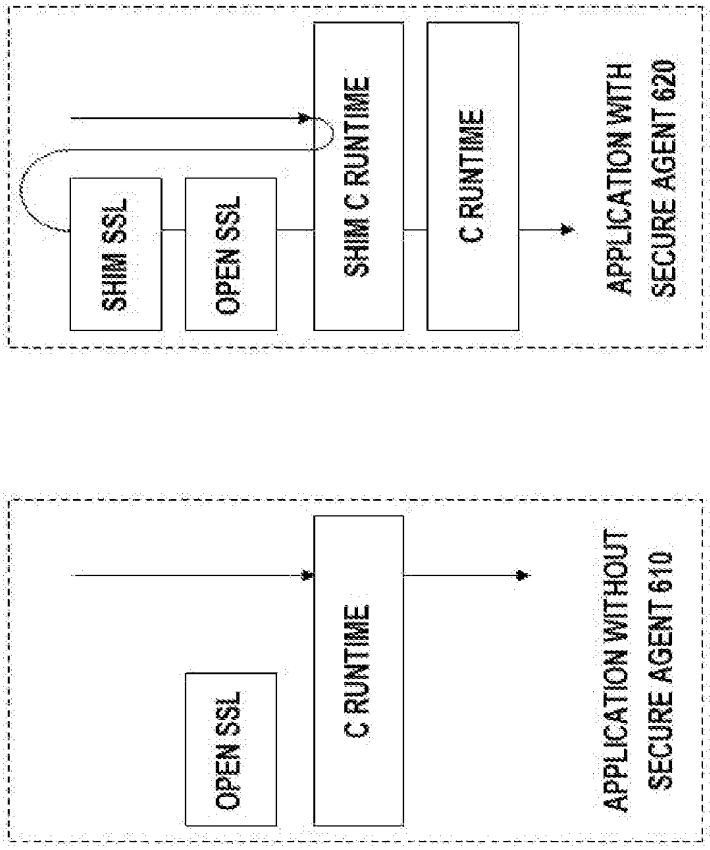

APPLICATION WITHOUT SECURE AGENT 610
OPEN SSL
C RUNTIME
APPLICATION WITH SECURE AGENT 620
SHIM SSL
OPEN SSL
SHIM C RUNTIME
C RUNTIME

FIG. 6

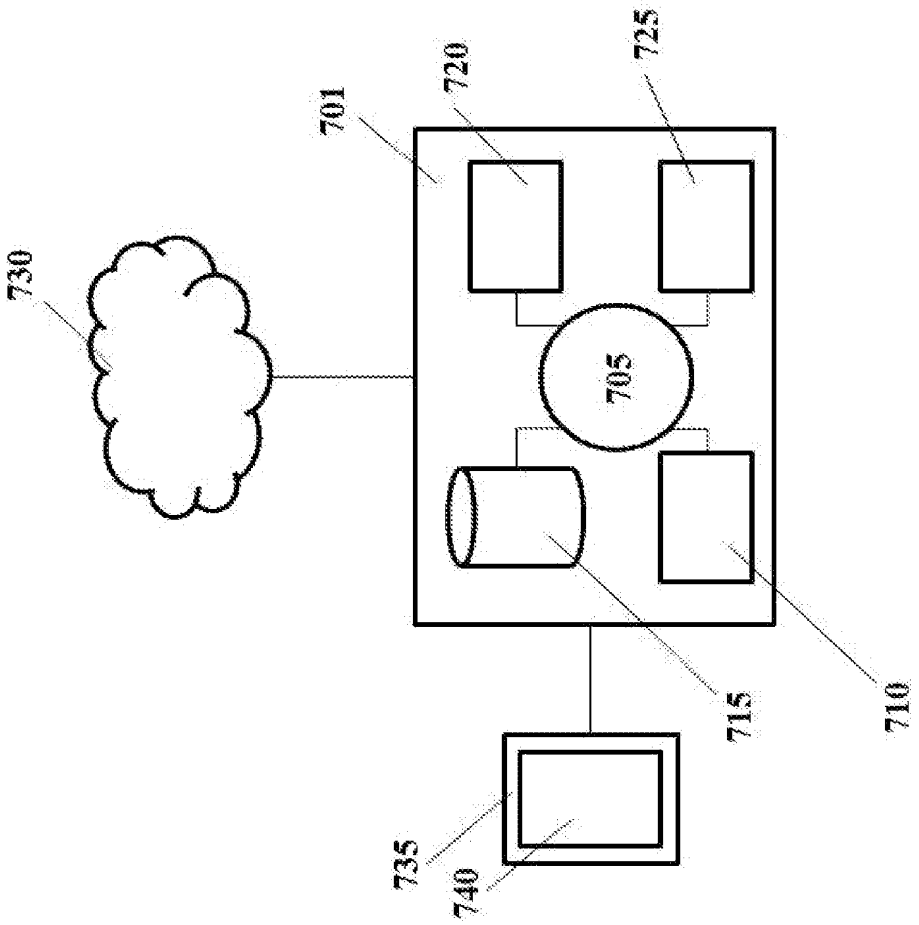


FIG. 7

811
APPLICATION
EMBEDDED SHIM
DATA PATH
EMBEDDED SECURITY AGENT SHIM
810
APPLICATION
DATA PATH
SECURE AGENT INTERCEPTOR PROCESS
821
OSINFRA (NETSTAT, /PROC
SECURITY AGENT PROCESS
820
APPLICATION
EMBEDDED SHIM
CONTROL PATH - APP CONTEXT
DATA PATH
SECURE AGENT INTERCEPTOR PROCESS
833
831
APPLICATION CONTEXT HELPER AND SECURITY AGENT PROCESS
830
DATA PATH
843
841
APPLICATION
SECURE AGENT INTERCEPTOR PROCESS
CONTROL PATH - APP CONTEXT
SECURE AGENT INTERCEPTOR PROCESS
OSINFRA (NETSTAT, /PROC
(MULTI-INSTANCE CASE) SECURITY AGENT PROCESS
840

FIG. 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2019/062641

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/50(2013.01)i, H04L 29/06(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F 21/50; G06F 21/00; G06F 9/44; G06F 9/455; H04L 29/06; H04L 29/08; H04Q 11/00

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: intercept, API (application programming interface) call, security policy, analyze, telemetry data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2014-0108558 A1 (CITRIX SYSTEMS, INC.) 17 April 2014 Paragraphs [0095], [0125]-[0129], [0270]-[0291], [0384]-[0388], [0423], [0449], [0518], [0527], [0532], [0576]; and figure 4	1-47
Y	US 2015-0106932 A1 (GLIMMERGLASS NETWORKS, INC.) 16 April 2015 Paragraphs [0032]-[0033], [0095]; claim 15; and figure 1	1-47
A	US 2015-0207816 A1 (SHAPE SECURITY, INC.) 23 July 2015 Paragraphs [0078], [0120], [0177]; and figure 2	1-47
A	US 2012-0185913 A1 (FRANK R. MARTINEZ et al.) 19 July 2012 Paragraphs [0010], [0035], [0071]; and figure 2B	1-47
A	KR 10-2018-0019070 A (GIGAVATION, INC.) 23 February 2018 Paragraphs [0008], [0058]-[0060]; and figure 3	1-47

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"D" document cited by the applicant in the international application
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
12 March 2020 (12.03.2020)	**12 March 2020 (12.03.2020)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
KIM, Sung Hee
Telephone No. +82-42-481-3516

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2019/062641

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2014-0108558 A1	17/04/2014	CN 104798355 A	22/07/2015
		CN 104823189 A	05/08/2015
		CN 104838630 A	12/08/2015
		CN 104854561 A	19/08/2015
		EP 2898652 A1	29/07/2015
		EP 2907003 A1	19/08/2015
		EP 2907073 A1	19/08/2015
		EP 2907076 A1	19/08/2015
		EP 2907289 A1	19/08/2015
		JP 2016-517977 A	20/06/2016
		JP 2016-519817 A	07/07/2016
		JP 2016-519818 A	07/07/2016
		JP 2016-522472 A	28/07/2016
		JP 2016-526223 A	01/09/2016
		JP 2016-530582 A	29/09/2016
		KR 10-2016-0005112 A	13/01/2016
		KR 10-2016-0010600 A	27/01/2016
		KR 10-2016-0018474 A	17/02/2016
		KR 10-2016-0018475 A	17/02/2016
		KR 10-2016-0018476 A	17/02/2016
		KR 10-2016-0019410 A	19/02/2016
		US 2014-0006347 A1	02/01/2014
		US 2014-0006772 A1	02/01/2014
		US 2014-0007048 A1	02/01/2014
		US 2014-0007182 A1	02/01/2014
		US 2014-0007183 A1	02/01/2014
		US 2014-0007192 A1	02/01/2014
		US 2014-0007193 A1	02/01/2014
		WO 2014-047168 A1	27/03/2014
		WO 2014-058640 A1	17/04/2014
		WO 2014-058722 A1	17/04/2014
		WO 2014-062337 A1	24/04/2014
		WO 2014-062395 A1	24/04/2014
		WO 2014-062420 A1	24/04/2014
US 2015-0106932 A1	16/04/2015	US 9705906 B2	11/07/2017
US 2015-0207816 A1	23/07/2015	EP 3097509 A1	30/11/2016
		US 10044753 B2	07/08/2018
		US 2016-0261629 A1	08/09/2016
		US 2017-318052 A1	02/11/2017
		US 2018-227325 A1	09/08/2018
		US 8954583 B1	10/02/2015
		US 9356954 B2	31/05/2016
		US 9712561 B2	18/07/2017
		WO 2015-109321 A1	23/07/2015
US 2012-0185913 A1	19/07/2012	EP 2316071 A1	04/05/2011
		EP 2972858 A1	20/01/2016

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		EP 2972975 A1	20/01/2016
		EP 2973116 A1	20/01/2016
		US 2010-0027552 A1	04/02/2010
		US 2011-0231899 A1	22/09/2011
		US 2013-0326516 A1	05/12/2013
		US 2014-0201017 A1	17/07/2014
		WO 2009-155574 A1	23/12/2009
		WO 2011-091056 A1	28/07/2011
		WO 2012-100092 A2	26/07/2012
		WO 2014-145658 A1	18/09/2014
KR 10-2018-0019070 A	23/02/2018	CN 107820604 A	20/03/2018
		EP 3284003 A1	21/02/2018
		US 2016-0306966 A1	20/10/2016
		WO 2016-168487 A1	20/10/2016

UNITED STATES
PATENT AND TRADEMARK OFFICE

P.O. Box 1450
Alexandria, VA 22313 - 1450
www.uspto.gov

ELECTRONIC ACKNOWLEDGEMENT RECEIPT

APPLICATION #	RECEIPT DATE / TIME	ATTORNEY DOCKET #
17/657,972	**08/31/2022 04:21:24 PM ET**	**15635.0005-00000**

Title of Invention

SYSTEMS AND METHODS FOR MANAGING CYBER VULNERABILITIES

Application Information

APPLICATION TYPE	Utility - Nonprovisional Application under 35 USC 111(a)	PATENT #	-
CONFIRMATION #	2547	FILED BY	Michelle Daniels
PATENT CENTER #	60927351	FILING DATE	04/05/2022
CUSTOMER #	22852	FIRST NAMED INVENTOR	Avi SHUA
CORRESPONDENCE ADDRESS	-	AUTHORIZED BY	Christopher Johns

Documents

TOTAL DOCUMENTS: 9

DOCUMENT	PAGES	DESCRIPTION	SIZE (KB)
IDS before FINAL Office Action.pdf	2	Transmittal Letter	18 KB
sb08.pdf	2	Information Disclosure Statement (IDS) Form (SB08)	32 KB
WO2020106973A1.pdf	56	Foreign Reference	4243 KB
WO2019051507A1.pdf	69	Foreign Reference	4705 KB
WO2017153983A1.pdf	16	Foreign Reference	869 KB
Non-Final Office Action dated 08-03-2022 - 17 716,491.pdf	8	Non Patent Literature	429 KB

Non-Final Office Action dated 08-17-2022 - 17 657,986.pdf	13	Non Patent Literature	758 KB
Non-Final Office Action dated 07-18-2022 - 17 658,156.pdf	16	Non Patent Literature	915 KB
Non-Final Office Action dated 08-18-2022 - 17 716,583.pdf	13	Non Patent Literature	716 KB

Digest

DOCUMENT	MESSAGE DIGEST(SHA-512)
IDS before FINAL Office Action.pdf	57B53B87FD22E42C257CC9A9D257C77B0BE4D7572D30847B39395ABAA4DF9ECC37D13E5D583C0C466979D4FCDB7AFF13EE74D856C191220954678CF2C8D601C3
sb08.pdf	9B4B7A55B6F9220254F60B4FB8B185381BEBC939464288FEDBCA61D63C8B991CC35C210B08449A3797F950A734AC126EE52EC3672AAC9C98BA02C13AA3FBE43F
WO2020106973A1.pdf	E11441120E6AF16378204B5DB1DB9C959C897EF13EB82278ED00352710F8416CFA18F1920B2637826471BE70B378983A7776EDDD5A74B0FBAB020D4CC33FD7D2
WO2019051507A1.pdf	3FA4E891E8A81CA5667740B392A17A40B94D8FC139EF6BDE24AAA3DE00FDC11DB2E055EED6AC7B1A07144F25317AB7A704145699529F2E650840391637554 4DB
WO2017153983A1.pdf	8FC6DCDF5F6EDECA183538C43C4DC946428926DBE35508AB770DDFECA2285080947FC35FD9B15DD76F0EC7C61434FED3C0D6F6159B2A9DD4FD5006EF5C2E4D6B
Non-Final Office Action dated 08-03-2022 - 17 716,491.pdf	BE91EA18215C04454FFE0E953D7B71DF9BFBD950621446233D41AE62FE0BF3D0B1BE898310FFE2EE32E6B3448AC5F3B52DFBD7A59ACEF66DEE61B4FC1BCF37C6
Non-Final Office Action dated	3F17A9071D8FA03B72416E7D75C3EF081442DFBDC3A035353

08-17-2022 - 17 657,986.pdf	89794BE137078687C103D1D42F08010A213E573C386E7EC6EC50A6CF0E9EF3C5AF5E709F6A96360
Non-Final Office Action dated 07-18-2022 - 17 658,156.pdf	389051A0BCA5A56E649D2310E05863D9CBB3ED1A8ED1559EC2527F93770A48444A186BE8AC581BA3B424D2739C031B2D08D7446D0C729D043E32FB1502F75612
Non-Final Office Action dated 08-18-2022 - 17 716,583.pdf	54DE2F031B7EDE5354937C74B4E034272295DD5F865686AE9F3327E0C29EE370CC226BA43F81FF2F6F389A5BD82D72DC0A78048342AD5A134C7FAD98BA36C33A

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
14 September 2017 (14.09.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/153983 A1

(51) International Patent Classification:
G06F 21/56 (2013.01) *G06F 21/55* (2013.01)
G06F 21/60 (2013.01) *G06F 21/50* (2013.01)

(21) International Application Number:
PCT/IL2017/050277

(22) International Filing Date:
7 March 2017 (07.03.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/304,958 8 March 2016 (08.03.2016) US

(71) Applicant: **B. G. NEGEV TECHNOLOGIES AND APPLICATIONS LTD., AT BEN-GURION UNIVERSITY** [IL/IL]; P.O.B. 653, 8410501 Beer Sehva (IL).

(72) Inventor: **GURI, Mordechai**; 28 Emek Ayalon Street, 7170656 Modi'in (IL).

(74) Agents: **FUERST, Zadok** et al.; Luzzatto & Luzzatto, P.O. Box 5352, 8415202 Beer Sheva (IL).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

— *with international search report (Art. 21(3))*

(54) **Title**: SYSTEM AND METHOD FOR PERFORMING IN-CLOUD SECURITY OPERATIONS ON CONNECTED DEVICES

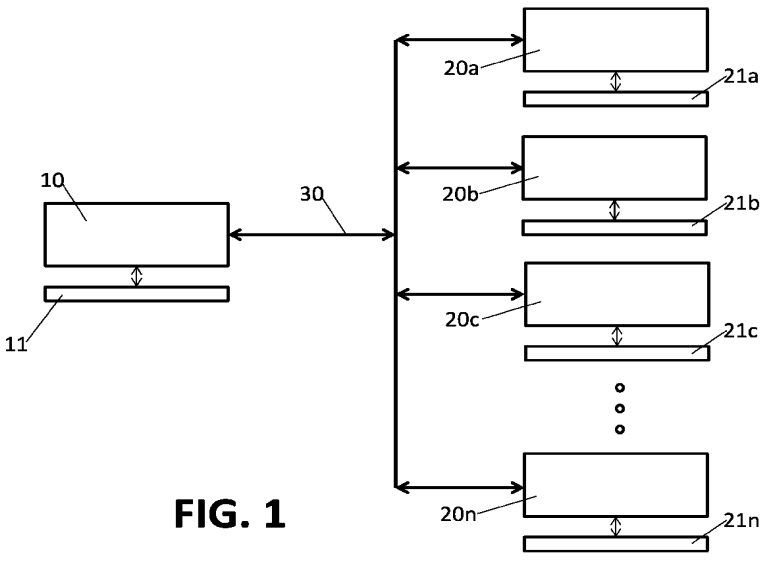


FIG. 1

(57) **Abstract**: The invention relates to a system for protecting IoT devices from malicious code, which comprises: (a) a memory extracting module at each of said IoT devices, for extracting a copy of at least a portion of the memory content from the IoT device, and sending the same to an in-cloud server; and (b) an in-cloude server for receiving said memory content, and performing an integrity check for a possible existance of malicious code within said memory content.

WO 2017/153983 A1

SYSTEM AND METHOD FOR PERFORMING IN-CLOUD SECURITY OPERATIONS ON CONNECTED DEVICES

Field of Invention

The field of the invention relates in general to methods and systems for securing computerized environments and devices. More specifically, the invention relates to a method and a system for checking the integrity and authenticity of "Internet of Things" (IoT) type devices, thereby detecting whether they have been infected by malicious code.

Background of the Invention

The "Internet of Things (IoT)" is the internetworking of physical devices, vehicles (also referred to as "connected devices" and "smart devices"), buildings, and other items embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data. The IoT allows objects to be sensed or controlled remotely across existing network infrastructure, creating opportunities for more direct integration of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit in addition to reduced human intervention. When IoT is augmented with sensors and actuators, the technology becomes an instance of the more general class of cyber-physical systems, which also encompasses technologies such as smart homes, intelligent transportation and smart cities. Each thing is uniquely identifiable through its embedded computing system and is able to interoperate within the existing Internet infrastructure. According to Wikipedia, Experts estimate that the IoT will consist of almost 50 billion objects ("things") by 2020.

Typically, IoT offers advanced connectivity of devices, systems, and services that goes beyond machine-to-machine (M2M) communications and covers a variety of protocols, domains, and applications. The interconnection of these embedded

devices (including smart objects), is expected to be adopted in in nearly all fields of automation.

For example, the range of IoT devices includes heart monitoring implants, biochip transponders on farm animals, electric clams in coastal waters, automobiles with built-in sensors, or field operation devices that assist firefighters in search and rescue operations. These devices use sensors to collect useful data with the help of various existing technologies, and then autonomously flow the data between other devices. Other examples include home automation (smart home devices) such as the control and automation of lighting, heating (like smart thermostat), ventilation, air conditioning (HVAC) systems, and appliances such as washer/dryers, robotic vacuums, air purifiers, ovens or refrigerators/freezers that use Wi-Fi for remote monitoring.

As discussed, the IoT devices that are used in many fields and structures, are manufactured by a huge number of different manufacturers. While the protocol for interconnectivity and communication between various of these device has been defined and standartized, still a vast majority of these devices are characterized by:

a. They apply a great variety of proprietary operating systems (OS), software and hardware;

b. They have a limited, in many cases very limited processing power; and

c. The devices are in many cases very cheap.

Although the IoT devices are typically characterized by said (a) – (c) above, still these devices in many cases are used to control or sense very critical elements. The exploitation of these devices by hostile entities, for example, by injection of malicious code, can cause very significant damages, such as stealing of data, manipulation of the operation of the devices, or using the attacked device as a platform for attacking other device (e.g., bot). Therefore, in contrast to (a) the low cost of each of said devices; (b) the fact that the devices may have a very limited

processing power; and (c) the fact that each of the devices may have a different proprietary oprating system that may require a dedicated protection software; there is still a real need to protect the IoT devices from damages resulting from malicious code. This is particularly due to the fact that all these devices are accessible to hackers via the Internet.

The prior art techniques that have been so far adopted for protecting IoT devices from malicious code are typically tradional, for example:

a. Use of a firewall, which substantially isolates the internal network of the IoT devices from the external domain;
b. The use of conventional anti-virus software, whenever possible. This is, however, impractical in IoT devices having proprietary operating systems, in view of the huge variety of the proprietary operating systems involved. Furthermore, this is in many cases impractical in view of the high difference between the value of the IoT device and the cost of developing such software to account for this large variety of proprietary operating systems. Finally, the limited processing power of the IoT devices does not always enable the use of anti-virus software.
c. The use of network based security systems such as firewalls, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). These systems are monitoring the network traffic and try to detect attacks or malicious activities. However, this approach is limited only to attacks that can be detect from the network traffic. Moreover, this approach is very limited in detecting attacks that are hiding within encrypted traffic.

More specifically, the prior art has typically applies the classic approach that is typically used for end-points and workstations, which involves running detection programs on the IoT devices to detect, scan and analyze for malicious code within the (1) persistent storage (e.g., disk) and (2) non-persistent storage (e.g., memory).

However, this approach is not applicable in many case, in view of the limited computational power of the IoT device.

In another aspect, cloud computing is a type of Internet-based computing that provides shared computer processing resources and data to computers and other devices on demand. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g., computer networks, servers, storage, applications and services) which can be rapidly provisioned and released with minimal management effort. Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in either privately owned, or third-party data centers that may be located far from the user–ranging in distance from across a city to across the world. Cloud computing relies on sharing of resources to achieve coherence and economy of scale, similar to a utility (like the electricity grid) over an electricity network.

It is therefore an object of the present invention to provide a method and system for protecting IoT devices from malicious code.

It is another object of the present invention to provide a method and system that can protect IoT devices, that wereotherwise remained unprotected.

It is still another object of the present invention to provide such a system that can detect malicious code in IoT devices, provide an alert, and remove the malicious code.

It is still another object of the present invention to provide upgrades and updates to the operating systems of the IoT devices, whether these operating systems are standard, or proprietary.

Other objects and advantages of the invention will become apparent as the description proceeds.

Summary of the Invention

The invention relates to a system for protecting IoT devices from malicaious code, which comprises: (a) a memory extracting module at each of said IoT devices, for extracting a copy of at least a portion of the memory content from the IoT device, and sending the same to an in-cloud server; and (b) an in-cloude server for receiving said memory content, and performing an integrity check for a possible existance of malicious code within said memory content.

In an embodiment of the invention, said in-cloud server performs one or more of the following: (a) analysis of the memory to find malware using static analysis methods; analysis of the memory to find malicious behavior using behavioral and heuristics methods; (b) reconstruction of the state of an OS of the IoT to determine and report important structural elements; (c) check of the integrity of the OS and its memory to possibly find fault in the integrity or hidden processes; (d) a cross-view check on resources to find rootkits and hidden operations; and (e) a cross-view check and validation of memory contents of plurality of IoT devices.

In an embodiment of the invention, and following said integrity check, said in-cloud server performs one or more of the following:
(1) logging of the results;
(2) reporting the results, raising a warning or an alert in a case of detection of an unexpected code or behaveior; or
(3) communicating and responding to an IoT request.

In an embodiment of the invention, the memory, a copy of which is sent to the in-cloud server, is either a persistent memory or a non-persistent memory.

In an embodiment of the invention, said memory extraction module is embedded within a kernel of a respective operating system of the IoT device.

In an embodiment of the invention, said memory extraction module is positioned within a trusted layer at the IoT device.

In an embodiment of the invention, said memory extraction module is positioned within a Trusted Execution Environment at the processor of the IoT device.

Brief description of the Drawings

In the drawings:

- Fig. 1 illustrates the general structure of the system of the present invention.

Detailed Description of Preferred Embodiments of the Invention

As noted, there are many cases in which the classical techniques are not applicable for protecting IoT devices from damages of malicious codes. The present invention provides alternative system and method for protecting IoT devices.

Fig. 1 shows a general structure of the system of the present invention. The system comprises in general an in-cloud protection server 10, which is used to protect a plurality of remote IoT devices 20a – 20n from malicious code. Several of the devices 20a- 20n may be stand-alone devices, others may be a part of IoT networks. In general, the in-cloud server 10 may serve a very large number of IoT devices. For example, a single in-cloud server may serve hundreds of thouthands, even millions of IoT devices 20, that are spread along the globe.

Each of the devices 20a-20n comprises a memory extraction modul 21a-21n, respectively. In one embodiment, the memory extraction module 21 is embedded within the kernel of the respective operating system of device 20. In another embodiment, the memory extraction module 21 is positioned within a trusted layer, such as hypervisor, at device 20. In still another embodiment, the memory extraction module may be positioned within a Trusted Execution Environment (TEE) at the processor of the IoT device.

The memory extraction modules 21 at each of the devices 20 extracts, either upon demand from server 10, or independently the memory content (persistent and/or non persistent), and transmits the same to the in-cloud server 10 for inspection and verification.

Upon receipt of the memory content (or a portion thereof) of a device 20 from a respective memory extraction module 21, the in-cloud server 10 performs one or more of the following operations:

(1) Analysis of the memory to find malware using static analysis methods;

(2) Analysis of the memory to find malicious behavior using behavioral and heuristics methods;

(3) Reconstruction of the state of the OS to determine and report important structural elements, such as:

a. Process/thread list
b. Communication ports
c. Kernel modules
d. Objects in memory
e. Object in cache
f. Open/close files
g. System status
h. Bootstrap information
i. Memory corruptions

(4) Check of the integrity of the OS and its memory to find fault in the integrity or hidden processes;

(5) A cross-view check on resources to find rootkits and hidden operations. The cross-view collaborates with components as reported from within the OS; and.

(6) A cross-view check and validation of memory contents of plurality of IoT devices.

Following the above perations, the in-cloud server may perform one ore more of the following operations:

(1) Logging of the results;
(2) Reporting the results, raising a warning or an alert in a case of detection of an unexpected code or behaveior; or
(3) Communicating and responding to an IoT request.

For its proper operation, server 10 has a database 11 that contains authentic and reliable data for comparison and verification with the memory content which is received from the IoT devices. For example, the database 11 may contain a copy of the authentic OS which is used in each IoT device 20 and its version number. The database content, whether relating to standard data or proprietary data, is accumulated by the operator of server 10, for example, by contacting the manufacturers of devices 20. The fact that a single server 10 can serve a huge number of devices 20 significantly reduces the costs of obtaining such data. Moreover, the communication between server 10 and devices 20 is typically performed over a secured channel 30. The system of the present invention also overcomes the problem which is associated with the typical IoT devices, namely the lack of sufficient processing power to perform the integrity check in the classical approach.

Server 10 may also perform an update of the OS of each of the IoT devices, when it becomes necessary. The integrity check and/or update may be performed from time to time, or periodically.

As shown, the in-cloud system of the present invention provides protection to IoT devices from damages resulting from malicious code. The system of the present invention provides such a protection in cases where the classical approach is either inapplicable (for example due to lack of sufficient processing power), or impractical (for example, due to the relatively high costs involved in providing of such protection).

While some embodiments of the invention have been described by way of illustration, it will be apparent that the invention can be carried into practice with many modifications, variations and adaptations, and with the use of numerous equivalents or alternative solutions that are within the scope of persons skilled in the art, without departing from the spirit of the invention or exceeding the scope of the claims.

CLAIMS

1. A system for protecting IoT devices from malicious code, which comprises:
   a. a memory extracting module at each of said IoT devices, for extracting a copy of at least a portion of the memory content from the IoT device, and sending the same to an in-cloud server; and
   b. an in-cloude server for receiving said memory content, and performing an integrity check for a possible existance of malicious code within said memory content.

2. A system according to claim 1, wherein said in-cloud server performs one or more of the following:
   a. analysis of the memory to find malware using static analysis methods;
   b. analysis of the memory to find malicious behavior using behavioral and heuristics methods;
   c. reconstruction of the state of an OS of the IoT to determine and report important structural elements;
   d. check of the integrity of the OS and its memory to possibly find fault in the integrity or hidden processes;
   e. a cross-view check on resources to find rootkits and hidden operations; and
   f. a cross-view check and validation of memory contents of plurality of IoT devices.

3. A system according to claim 1, wherein following said integrity check, said in-cloud server performs one or more of the following:
   (1) logging of the results;
   (2) reporting the results, raising a warning or an alert in a case of detection of an unexpected code or behaveior; or
   (3) communicating and responding to an IoT request.

4. A system according to claim 1, wherein the memory, a copy of which is sent to the in-cloud server, is either a persistent memory or a non-persistent memory.

5. A system according to claim 1, wherein said memory extraction module is embedded within a kernel of a respective operating system of the IoT device.

6. A system according to claim 1, wherein said memory extraction module is positioned within a trusted layer at the IoT device.

7. A system according to claim 1, wherein said memory extraction module is positioned within a Trusted Execution Environment at the processor of the IoT device.

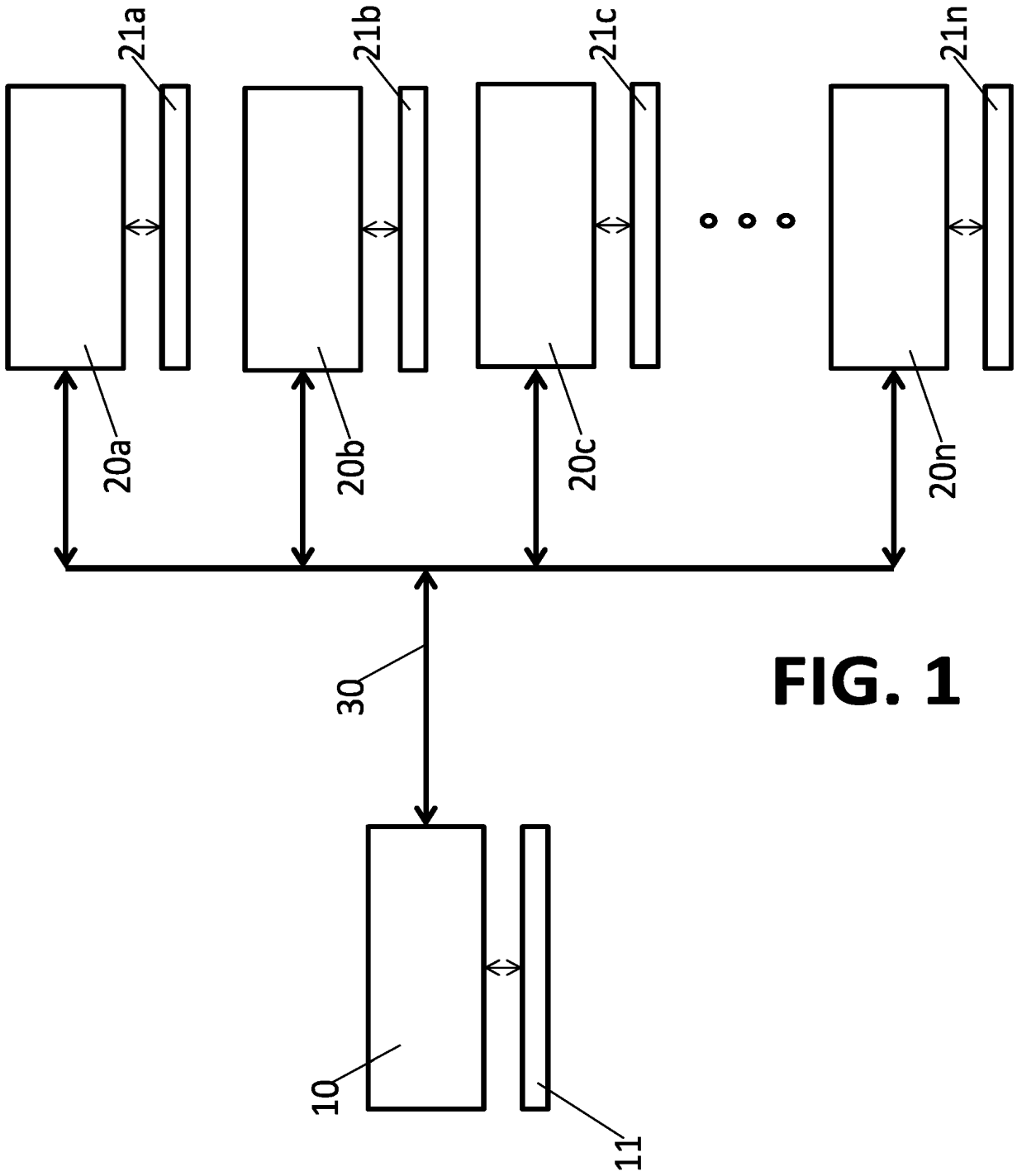


FIG. 1

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL2017/050277

A. CLASSIFICATION OF SUBJECT MATTER
IPC (2017.01) G06F 21/56, G06F 21/60, G06F 21/55, G06F 21/50

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC (2017.01) G06F 21/56, G06F 21/60, G06F 21/55, G06F 21/50, H04L 29/00, G06F 11/00, G06F 12/14

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
See extra sheet.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2007277241 A1 SYMANTEC CORPORATION 29 Nov 2007 (2007/11/29) Abstract, ¶¶ 0008, 0022, 0023, 0061, 0062	1-7
Y	US 2014047544 A1 JAKOBSSON BJORN MARKUS 13 Feb 2014 (2014/02/13) Abstract, ¶¶0024, 0025, 0028, 0029, 0039, 0040, 0046, 0051-0055, 0063, Fig 6	1-7
A	US 8904525 B1 Hodgman et al. 02 Dec 2014 (2014/12/02) Entire Document	1-7
A	US 2013227636 A1 APPTHORITY INC 29 Aug 2013 (2013/08/29) Entire Document	1-7
A	US 2013111547 A1 SCARGO INC 02 May 2013 (2013/05/02) Entire Document	1-7

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
11 Jun 2017	14 Jun 2017
Name and mailing address of the ISA: Israel Patent Office Technology Park, Bldg.5, Malcha, Jerusalem, 9695101, Israel Facsimile No. 972-2-5651616	Authorized officer COPPENHAGEN Uri Telephone No. 972-2-5657811

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/IL2017/050277

Patent document cited search report	Publication date	Patent family member(s)	Publication Date
US 2007277241 A1	29 Nov 2007	US 2007277241 A1	29 Nov 2007
		US 7870394 B2	11 Jan 2011
US 2014047544 A1	13 Feb 2014	US 2014047544 A1	13 Feb 2014
		US 9411955 B2	09 Aug 2016
US 8904525 B1	02 Dec 2014	US 8904525 B1	02 Dec 2014
US 2013227636 A1	29 Aug 2013	US 2013227636 A1	29 Aug 2013
		US 8918881 B2	23 Dec 2014
		US 2015143455 A1	21 May 2015
		US 9438631 B2	06 Sep 2016
		WO 2013126259 A1	29 Aug 2013
US 2013111547 A1	02 May 2013	US 2013111547 A1	02 May 2013
		US 9223978 B2	29 Dec 2015
		US 2015350237 A1	03 Dec 2015
		US 9460285 B2	04 Oct 2016
		US 2016373486 A1	22 Dec 2016
		WO 2013063474 A1	02 May 2013

B. FIELDS SEARCHED:

* Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Databases consulted: Esp@cenet, Google Patents
Search terms used: malicious code, malware, internet of things device, iot device, device, cloud, network, extract, copy, audit, check, verify, screen, scan, monitor, memory, cache, data, integrity

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
14 March 2019 (14.03.2019)

WIPO | PCT

(10) International Publication Number
WO 2019/051507 A1

(51) International Patent Classification:
G06F 21/56 (2013.01) *G06F 21/55* (2013.01)
H04L 29/06 (2006.01)

(21) International Application Number:
PCT/US2018/050528

(22) International Filing Date:
11 September 2018 (11.09.2018)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/557,132 11 September 2017 (11.09.2017) US

(71) Applicant: **CARBON BLACK, INC.** [US/US]; 1100 Winter Street, Waltham, MA 02451 (US).

(72) Inventors: **KRAEMER, Jeffrey Albin**; 27 Kenilworth Road, Wellesley, MA 02482 (US). **MALINOWSKI, Adam Karol**; 46 Mary Street, Apt. 1, Arlington, MA 02474 (US).

(74) Agent: **LEHRER, Joel E.** et al.; Goodwin Procter LLP, 100 Northern Avenue, Boston, MA 02210 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report (Art. 21(3))*
— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

(54) **Title:** METHODS FOR BEHAVIORAL DETECTION AND PREVENTION OF CYBERATTACKS, AND RELATED APPARATUS AND TECHNIQUES

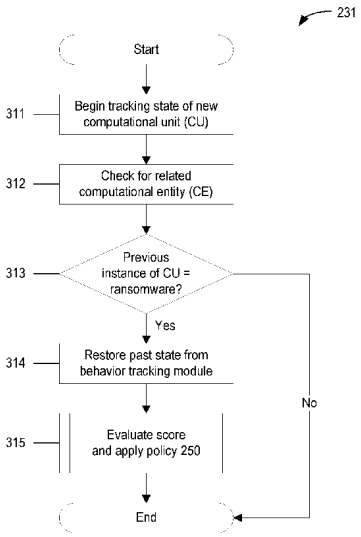


FIGURE 3A

(57) **Abstract:** A security engine may use event-stream processing and behavioral techniques to detect ransomware. The engine may detect process behavior associated with encrypting a file, encrypting a storage device, or disabling a backup file, and may assign a ransomware category to the process based thereon. The engine may initiate protection actions to protect system resources from the process, which may continue to execute. The engine may monitor the process for specific behavior corresponding to its ransomware category. Based on the extent to which such specific behavior is detected, the engine may determine that the process is not ransomware, assign a ransomware subcategory to the process, or adjust the process's threat score. Monitoring of the process may continue, and the threat score may be updated based on the process's behavior. If the threat score exceeds a threshold corresponding to the ransomware category (or subcategory), a corresponding policy action may be initiated.

WO 2019/051507 A1

METHODS FOR BEHAVIORAL DETECTION AND PREVENTION OF CYBERATTACKS, AND RELATED APPARATUS AND TECHNIQUES

CROSS-REFERENCE TO RELATED APPLICATION(S)

This application claims priority to and the benefit of U.S. Provisional Patent Application Serial Number 62/557,132, titled "Methods for Behavioral Detection and Prevention of Cyberattacks, and Related Apparatus and Methods" and filed under Attorney Docket No. BIT-015PR on September 11, 2018, which is hereby incorporated by reference herein in its entirety.

FIELD OF INVENTION

The present disclosure relates generally to computer security systems and techniques. Some embodiments described herein relate specifically to detection and prevention of ransomware-based attacks on endpoint devices (e.g., desktops, laptops, terminals, servers, embedded systems, etc.).

BACKGROUND

As the Internet and other networked computer systems become increasingly integrated into public activities (e.g., management and operation of governmental organizations) and private activities (e.g., personal activities, management and operation of households and businesses, etc.), cybersecurity threats pose an increasingly significant threat to such pursuits. Some examples of cybersecurity threats include malicious software ("malware") and ransomware (collectively, "threatware"). Malware generally operates to disrupt operation of computer systems (e.g., by taking control of computational resources and using those resources for unauthorized purposes, by disabling individual computers or entire networks, by damaging or otherwise sabotaging system components, etc.) and/or to steal resources from computer systems (e.g., by gathering sensitive data). Malware can be deployed in many forms, including computer viruses, worms, trojan horses, spyware, adware, scareware, keystroke loggers, rootkits, bots, crimeware, phishing scams, etc.

Computer users devote significant resources to detecting cybersecurity threats and preventing such threats from disrupting the operations of their computer systems or stealing their computer-based resources. Conventional cybersecurity engines have relied extensively on static, signature-based techniques for detecting malware. In general, static, signature-based malware detection involves obtaining a copy of a file that is known to contain threatware, analyzing the static features of the file (e.g., the sequence of bytes contained in the file) to extract a static signature that is characteristic of the threatware, and adding the threatware's static signature to a

database (often referred to as a "blacklist") of known cybersecurity threats. When a user attempts to access (e.g., download, open, or execute) a file, the cybersecurity engine scans the file and extracts the file's static signature. If the file's static signature matches a signature on the blacklist, the cybersecurity engine detects the presence of a threat and intervenes to prevent the threatware from executing (e.g., by quarantining or deleting the file).

Static, signature-based techniques for detecting threatware are generally useful for quickly detecting known threats. However, these techniques can generally be circumvented by new threats that are not yet blacklisted (e.g., zero-day threatware or next-generation threatware) or by threatware that modifies itself to avoid matching a static signature on the blacklist (e.g., oligomorphic, polymorphic, or metamorphic threatware).

Some cybersecurity engines rely on behavior-based techniques for detecting cybersecurity threats. In general, behavior-based threat detection involves monitoring the execution of a computational unit (e.g., a thread, a process, an application, etc.) and identifying suspicious features of the computational unit's execution (e.g., suspicious behaviors, which may include unpacking code, modifying the host file, logging keystrokes, etc.). When suspicious behaviors are identified, the cybersecurity engine intervenes to protect the computer system (e.g., by terminating or quarantining the computational unit) and assess the threat (e.g., by initiating a forensic investigation of the computational unit, etc.). Behavior-based threatware detection techniques can protect computer systems from certain types of threats (for example, zero-day threatware and self-modifying threatware) that are difficult to detect using blacklists of static signatures.

The spectrum of potential threats ("threatscape") posed by threatening software (e.g., malware, ransomware, etc.) continues to evolve rapidly. Ransomware, for example, has quickly evolved to become one of the most threatening attack methods used by modern cybercriminals and one of the most significant cybersecurity challenges facing organizations that rely on enterprise computing. "Ransomware" generally refers to a type of software that threatens to publish or perpetually block access to a computer user's data unless the user pays a ransom to an attacker. Access to the user's data is generally blocked by encrypting the user's files or storage devices without providing the decryption key to the user.

SUMMARY OF THE INVENTION

Behavior-based threatware detection techniques can protect computer systems from certain types of threatware (for example, zero-day threatware and self-modifying threatware) that are difficult to detect using blacklists of static signatures. However, behavior-based threatware detection techniques can place significant demands on the host computer system's resources, and

can therefore substantially degrade the host computer system's performance. For example, a cybersecurity engine that performs behavior-based threatware detection may track a substantial portion of the behavior of a substantial fraction of the computational units executing on a computer system (e.g., by monitoring the execution of the computational units, detecting behaviors of interest, and/or generating tracking data indicative of such behaviors). Such tracking can consume a significant portion of the computer system's computing resources (e.g., processor cycles, storage space, communication bandwidth, etc.).

Another problem associated with behavior-based threatware detection techniques relates to a tradeoff between false negative outcomes (e.g., erroneously classifying threatware as benign software) and false positive outcomes (e.g., erroneously classifying benign software as threatware). A cybersecurity engine that produces too many false negative outcomes may fail to detect threatware, rendering the engine ineffective and exposing the computer system to dangerous attacks. On the other hand, an engine that produces too many false positive outcomes may identify legitimate software as suspicious and initiate unnecessary interventions (e.g., forensic investigations) that disrupt the legitimate activities of a user or an organization. Initiating unnecessary interventions also wastes resources, including computing resources, energy resources (e.g., electrical power), human resources (e.g., the time and attention of security experts), and others.

Thus, techniques are needed for (1) reducing the amount of computing resources used for behavioral detection of ransomware, and/or (2) reducing the false positive rate of behavioral ransomware detection techniques without increasing (or without significantly increasing) the false negative rate of ransomware detection. The inventors have recognized and appreciated that the aforementioned improvement(s) in the efficiency of behavioral ransomware detection can be achieved using the following, adaptive technique. Initially, a security engine may monitor a computational entity for behavior associated with encrypting a file, encrypting a storage device, and/or disabling a backup file. If such behavior is detected, the security engine may assign a potential ransomware category to the computational entity based on the detected behaviors. Optionally, after assigning a potential ransomware category, the security engine may initiate protection actions to protect storage resources of the computer system from potentially malicious behaviors of the computational entity, which may continue to execute. The security engine may then monitor the computational entity for specific behavior corresponding to the assigned ransomware category. Based on the extent to which such specific behavior is (or is not) detected, the security engine may determine that the computational entity is not ransomware, or may assign a potential ransomware subcategory and a ransomware score to the computational entity. In the latter case, monitoring of the computational entity for behavior specific to the

assigned ransomware category (and/or subcategory) may continue, and the ransomware score may be updated from time to time based on the extent to which such behavior is (or is not) detected. If the ransomware score exceeds a threshold corresponding to the ransomware category (and/or subcategory), a mitigation action associated with the ransomware subcategory may be initiated.

In this way, the tracking of the computational entity's behavior and the interpretation of the tracked behavior (e.g., the determination as to whether the software is ransomware or not) may both be tailored based on what is already known about the computational entity, thereby achieving lower resource utilization and/or a lower rate of false positives than conventional techniques.

In general, one innovative aspect of the subject matter described in this specification can be embodied in a computer-implemented method for ransomware detection, including: based on first events initiated by a computational entity in a computer system, detecting one or more instances of the computational entity engaging in one or more first behaviors associated with encrypting a file, encrypting a storage device, and/or disabling a backup of at least a portion of the computer system; assigning at least one potential ransomware category to the computational entity based on the detected instances of the first behaviors; and after assigning the potential ransomware category to the computational entity, initiating one or more protection actions to protect one or more storage resources against one or more modification operations, and based on second events initiated by the computational entity, detecting one or more instances of the computational entity engaging in one or more second behaviors corresponding to the potential ransomware category.

Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the method. A system of one or more computers can be configured to perform particular actions by virtue of having software, firmware, hardware, or a combination of them installed on the system (e.g., instructions stored in one or more storage devices) that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

The foregoing and other embodiments can each optionally include one or more of the following features, alone or in combination. In some embodiments, the first behaviors associated with encrypting a file include enumerating directories of a file system and/or modifying a private file, the first behaviors associated with encrypting a storage device include modifying the storage device, and the first behaviors associated with disabling a backup of at

least a portion of the computer system include disabling and/or deleting a backup file, wherein the backup file stores backup data for a plurality of files and/or settings of the computer system.

In some embodiments, detecting the instance of the computational entity enumerating directories of a storage device includes detecting an enumeration operation initiated by the computational entity on a specified first directory, wherein the specified first directory is (1) a top-level directory of the storage device, (2) a user directory in a file system of the storage device, or (3) an immediate subdirectory of a user directory in the file system of the storage device. In some embodiments, a second directory is nested within the first directory, and detecting the instance of the computational entity enumerating directories of the storage device further includes: detecting an enumeration operation initiated by the computational entity on the second directory; and detecting a file modification operation initiated by the computational entity, wherein the file modification operation targets a particular file in the second directory. In some embodiments, detecting the instance of the computational entity enumerating directories of the storage device further includes: for each intermediate directory in a nesting hierarchy between the first directory and the second directory, detecting an enumeration operation initiated by the computational entity on the intermediate directory. In some embodiments, the top-level directory of the storage device is a mount point of the storage device. In some embodiments, the storage device is an external storage device or a network storage device.

In some embodiments, detecting the instance of the computational entity modifying a storage device includes detecting the computational entity (1) bypassing an operating system of the computer system when accessing the storage device, (2) writing to one or more specified regions of the storage device, and/or (3) modifying one or more file system structures of the storage device. In some embodiments, the specified regions of the storage device include a master boot record (MBR) sector of the storage device and/or a partition table of the storage device. In some embodiments, the file system structures of the storage device include a file system structure pointer and/or an inode. In some embodiments, detecting the instance of the computational entity modifying a private file includes detecting a file modification operation initiated by the computational entity, wherein the file modification operation targets a particular file that is private to a different computational entity.

In some embodiments, assigning the potential ransomware category to the computational entity based on the detected instances of the first behaviors includes: assigning a first category to the computational entity if the detected instances include an instance of the first behavior associated with encrypting a file; assigning a second category to the computational entity if the detected instances include an instance of the first behavior associated with encrypting a storage device; and assigning a third category to the computational entity if the detected instances

include an instance of the first behavior associated with disabling a backup of at least a portion of the computer system. In some embodiments, the potential ransomware category assigned to the computational entity is the first category, the modification operations include one or more file modification operations targeting one or more files, and the protection actions include intercepting the file modification operations and creating backup copies of the files before permitting the file modification operations to modify the files. In some embodiments, permitting the file modification operations to modify the files includes permitting the file modification operations to encrypt the files.

In some embodiments, the potential ransomware category assigned to the computational entity is the second category, the modification operations include one or more storage device modification operations targeting one or more storage devices, and the protection actions include preventing the storage device modification operations from executing. In some embodiments, preventing the storage device modification operations from executing includes preventing the storage device modification operations from (1) bypassing an operating system of the computer system when accessing a storage device, (2) writing to one or more specified regions of the storage device, and/or (3) modifying one or more file system structures of the storage device. In some embodiments, the specified regions of the storage device include a master boot record (MBR) sector of the storage device and/or a partition table of the storage device. In some embodiments, the file system structures of the storage device include a file system structure pointer and/or an inode.

In some embodiments, the potential ransomware category assigned to the computational entity is the third category, the modification operations include one or more file modification operations targeting one or more backup files, and the protection actions include intercepting the file modification operations and creating backup copies of the backup files before permitting the file modification operations to modify the backup files.

The actions of the method may further include assigning at least one potential ransomware subcategory to the computational entity based on the detected instances of the second behaviors. In some embodiments, the potential ransomware category assigned to the computational entity is the first category, and the second behaviors corresponding to the first category include a second behavior associated with encrypting files in-place, a second behavior associated with encrypting files out-of-place, a second behavior associated with encrypting files using a single computational unit, and/or a second behavior associated with encrypting files using a plurality of related computational units. In some embodiments, the second behavior associated with encrypting files in-place includes modifying a file not created by the computational entity. In some embodiments, the second behavior associated with encrypting

files out-of-place includes creating a copy of a file not created by the computational entity, modifying the copy, and deleting the file not created by the computational entity. In some embodiments, the potential ransomware category assigned to the computational entity is the second category, and the second behaviors corresponding to the second category include (1) attempting to modify a master boot record (MBR) of the storage device wherein the storage device is a bootable storage device, and/or (2) attempting to modify a file system structure of the storage device wherein the storage device is a non-bootable storage device.

The actions of the method may further include assigning a score to the computational entity based on the assigned potential ransomware category, the assigned potential ransomware subcategory, and/or the detected instances of the second behaviors. In some embodiments, assigning the score to the computational entity includes setting the score to an initial value based on the assigned potential ransomware category. In some embodiments, the potential ransomware category assigned to the computational entity is the first category, detecting the computational entity engaging in the instance of the second behavior includes detecting an increase in a number of files modified by the computational entity, an increase in a number of distinct file type classes associated with files modified by the computational entity, and/or an increase in a rate of file modification by the computational entity, and assigning the score to the computational entity includes increasing the score based on the increased number of modified files, the increased number of distinct file type classes, and/or the increased rate of file modification.

In some embodiments, the potential ransomware category assigned to the computational entity is the first category, the second behavior includes the computational entity modifying a name of a file, modifying a header of a file, and/or modifying a file such that a file header and a file extension of the modified file are inconsistent with each other, and assigning the score to the computational entity includes increasing the score based on the detected instances of the second behaviors. In some embodiments, the potential ransomware category assigned to the computational entity is the first category and the potential ransomware subcategory assigned to the computational entity indicates that the computational entity performs in-place encryption, the second behavior includes modifying a particular amount of data in a plurality of files, modifying data at a particular offset in a plurality of files, and/or writing particular data to a plurality of files, and assigning the score to the computational entity includes increasing the score based on the detected instances of the second behaviors.

In some embodiments, the potential ransomware category assigned to the computational entity is the first category and the potential ransomware subcategory assigned to the computational entity indicates that the computational entity performs out-of-place encryption,

the second behavior includes specifying file extensions of a plurality of files such that the specified file extensions are all identical to each other or all different from each other, and assigning the score to the computational entity includes increasing the score based on the detected instances of the second behaviors. In some embodiments, the second behavior includes persistently scheduling the computational entity, creating a network connection, querying the computer system for system information, exhibiting suspicious provenance, obfuscating the computational entity, and/or parallelizing the computational entity, and assigning the score to the computational entity includes increasing the score based on the detected instances of the second behaviors.

In some embodiments, the potential ransomware category assigned to the computational entity is the first category, the second behavior includes modifying files of a sanctioned file type, modifying files in a temporary file storage area, and/or modifying private files of the computational entity, and assigning the score to the computational entity includes decreasing the score based on the detected instances of the second behaviors. In some embodiments, determining whether the assigned score exceeds a threshold score associated with the assigned potential ransomware subcategory, and if so, initiating a mitigation action associated with the ransomware subcategory. In some embodiments, the mitigation action includes terminating execution of the computational entity, preventing the computational entity from accessing a communication network, preventing the computational entity from modifying the storage resources, and/or replacing one or more files modified by the computational entity with respective backup files.

In some embodiments, the computational entity includes a single computational unit or a plurality of related computational units, wherein each computational unit is a thread, a process, or an application. In some embodiments, the modification operations are initiated by the computational entity.

In general, another innovative aspect of the subject matter described in this specification can be embodied in a computer-implemented method for ransomware detection, including: based on first events initiated by a computational entity in a computer system, detecting one or more instances of the computational entity engaging in one or more first behaviors associated with encrypting a file, encrypting a storage device, and/or disabling a backup of at least a portion of the computer system; assigning at least one potential ransomware category to the computational entity based on the detected instances of the first behaviors; and after assigning the potential ransomware category to the computational entity, based on second events initiated by the computational entity, detecting one or more instances of the computational entity engaging in one or more second behaviors corresponding to the potential ransomware category, assigning at

least one potential ransomware subcategory to the computational entity based on the detected instances of the second behaviors, assigning a score to the computational entity based on the assigned potential ransomware category, the assigned potential ransomware subcategory, and/or the detected instances of the second behaviors, and determining whether the assigned score exceeds a threshold score associated with the assigned potential ransomware subcategory, and if so, initiating a mitigation action associated with the ransomware subcategory.

Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the method. A system of one or more computers can be configured to perform particular actions by virtue of having software, firmware, hardware, or a combination of them installed on the system (e.g., instructions stored in one or more storage devices) that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

The foregoing and other embodiments can each optionally include one or more of the following features, alone or in combination. In some embodiments, each of the first behaviors is selected from a group including enumerating directories of a storage device, modifying a private file, modifying a storage device, and disabling a backup of at least a portion of the computer system.

The actions of the method may further include, after assigning the potential ransomware category to the computational entity, initiating one or more protection actions to protect one or more storage resources against one or more modification operations. In some embodiments, the modification operations are initiated by the computational entity. In some embodiments, assigning the potential ransomware category to the computational entity based on the detected instances of the first behaviors includes: assigning a first category to the computational entity if the detected instances include an instance of the first behavior associated with encrypting a file; assigning a second category to the computational entity if the detected instances include an instance the first behavior associated with encrypting a storage device; and assigning a third category to the computational entity if the detected instances include an instance of the first behavior associated with disabling a backup of at least a portion of the computer system.

In some embodiments, the potential ransomware category assigned to the computational entity is the first category, the modification operations include one or more file modification operations targeting one or more files, and the protection actions include intercepting the file modification operations and creating backup copies of the files before permitting the file modification operations to modify the files. In some embodiments, the potential ransomware

category assigned to the computational entity is the second category, the modification operations include one or more storage device modification operations targeting one or more storage devices, and the protection actions include preventing the storage device modification operations from executing. In some embodiments, the potential ransomware category assigned to the computational entity is the third category, the modification operations include one or more file modification operations targeting one or more backup files, and the protection actions include intercepting the file modification operations and creating backup copies of the backup files before permitting the file modification operations to modify the backup files. In some embodiments, the mitigation action includes terminating execution of the computational entity, preventing the computational entity from accessing a communication network, preventing the computational entity from modifying one or more storage resources of the computer system, and/or replacing one or more files modified by the computational entity with respective backup files.

Particular embodiments of the subject matter described in this specification can be implemented so as to realize one or more of the following advantages. In some embodiments, a behavioral security engine leverages event-stream processing to update (e.g., continually update) risk profiles of computational entities based on a stream of computer system activity. When one or more potentially threatening events are detected (e.g., when a cluster of multiple potentially threatening events is detected), the behavioral security engine defends against (e.g., blocks, prevents, or mitigates) an attack by the potential threatware (e.g., malware or ransomware). The threatware can be file-based or fileless.

In some embodiments, a behavioral security engine detects fast-evolving threats (e.g., ransomware threats, malware threats, etc.), including fileless attacks and unknown (e.g., zero-day) attacks, and protects computer systems (e.g., endpoints) from such threats. In some embodiments, a behavioral security engine detects streaming ransomware, defends against streaming ransomware, and/or prevents streaming ransomware attacks. In some embodiments, a behavioral security engine detects and/or prevents ransomware attacks, even if the attack uses an unknown file (e.g., an unknown executable file) or no file at all. In some embodiments, a behavioral security engine is able to detect and/or prevent ransomware attacks even if the security engine is offline (e.g., disconnected from the Internet, from the cloud, from the enterprise network, etc.). In some embodiments, a behavioral security engine detects and/or prevents ransomware attacks with little or no performance impact on the computer systems protected by the security engine. In some embodiments, a behavioral security engine detects and/or prevents ransomware attacks using little or no additional security resources and/or information technology resources beyond those already allocated to cybersecurity applications.

In some embodiments, a behavioral security engine protects files and/or boot sectors from encryption by ransomware.

In some embodiments, a behavioral security engine detects ransomware attacks with a low rate of false negative detection events and a low rate of false positive detection events. In some embodiments, the behavioral security engine functions as a complete, non-disruptive prevention solution effective against data loss resulting from techniques used by existing and emerging ransomware attacks, including "zero-day" attacks. In some embodiments, a behavioral security engine provides protection against ransomware-induced data loss on two or more levels, including: (1) file-system level and (2) raw storage device level. At the file-system level, some embodiments of the security engine can protect user data files (e.g., user documents or media files), application data, and system files. The scope of file-system level protection can span across local file systems, external drives, network shares, and backups. At the raw storage device level, some embodiments of the security engine can protect system boot records, partition tables, entire physical devices, and file-system structures.

The details of one or more embodiments of the subject matter described in this specification are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages of the subject matter will become apparent from the description, the drawings, and the claims.

The foregoing Summary, including the description of some embodiments, motivations therefor, and/or advantages thereof, is intended to assist the reader in understanding the present disclosure, and does not in any way limit the scope of any of the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

Certain advantages of some embodiments may be understood by referring to the following description taken in conjunction with the accompanying drawings. In the drawings, like reference characters generally refer to the same parts throughout the different views. Also, the drawings are not necessarily to scale, emphasis instead generally being placed upon illustrating principles of some embodiments of the invention.

FIG. 1 is a block diagram of a behavioral security engine, according to some embodiments.

FIG. 2 is a flowchart of a ransomware detection method, according to some embodiments.

FIG. 3A is a flowchart of a method for handling an event associated with creating a new computational unit, according to some embodiments.

FIG. 3B is a flowchart of a method for handling an event associated with accessing a raw storage device, according to some embodiments.

FIG. 3C is a flowchart of a method for handling an event associated with accessing a backup file, according to some embodiments.

FIG. 3D is a flowchart of a method for handling an event associated with enumerating a directory, according to some embodiments.

FIG. 3E is a flowchart of a method for handling an event associated with accessing a file, according to some embodiments.

FIG. 3F is a flowchart of a method for handling an event associated with accessing a network, according to some embodiments.

FIG. 3G is a flowchart of a method for handling an event associated with performing an application programming interface (API) call, according to some embodiments.

FIG. 3H is a flowchart of a method for evaluating the threat score of a computational entity and applying a security policy to the computational entity, according to some embodiments.

FIG. 4 is a flowchart of another ransomware detection method, according to some embodiments.

FIG. 5 is a block diagram of an implementation of a computer system.

DETAILED DESCRIPTION

Threat Detection and Response

Referring to FIG. 1, a behavioral security engine 100 may include a tracking module 110, a heuristics module 120, a behavior scoring module 130, a configuration module 140, a policy module 150, and a protection and mitigation module 160. The tracking module 110 may include an event monitoring module 112 and a behavior tracking module 114. The heuristics module 120 may include a categorization module 122, a subcategorization module 124, a score increasing module 126, and a score decreasing module 128.

The behavioral security engine 100 may be a component of a cybersecurity engine, and its modules may cooperate detect potential threats from threatware, including file-based threats and/or fileless (e.g., streaming) threatware threats, by monitoring and analyzing events (e.g., a stream of events) on a computer system (e.g., an endpoint computer system). The behavioral security engine's components may also cooperate to respond to detected threats. Some embodiments of the components of the behavior security engine 100 are described in further detail below.

In some embodiments, the behavioral security engine 100 may perform the following actions: (1) the event monitoring module 112 may track low-level events in a computer system; (2) the behavior tracking module 114 may map events to states and track them; (3) the categorization module 122 may assign a threat category (e.g., a ransomware category) to a potential threat and update the threat category as more events in the computer system are tracked; (4) the subcategorization module 124 may assign a threat subcategory (e.g., a ransomware category) to a potential threat and update the threat subcategory as more events in the computer system are tracked; (5) the behavior scoring module 130 may assign a threat score to the potential threat adjust the t threat score as more events in the computer system are tracked; and (6) if the threat score exceeds a threshold associated with the threat category (or subcategory) assigned to the potential threat, the policy module 150 and protection and mitigation module 160 may cooperate to perform a threat response action specified by the configuration module 140 for the detected type of threat. The configuration module 140 may be configured (e.g., by an administrator) with a security policy that defines actions to be taken upon detection of particular categories of threats (e.g., particular categories of ransomware threats).

Tracking:

The tracking module 110 may monitor events occurring in the computer system and track the behavior of computational entities in the computer system based on the monitored events. A computational entity may be, for example, a thread, a process, an application, or a related set of two or more threads, processes, and/or applications. The tracking module 110 may include an event monitoring module 112, which may monitor low-level events in the computer system. In some embodiments, the event monitoring module 112 tracks events from one or more layers of the operating system ("OS") of a computer system in real-time, within the context of a computational entity, and analyzes the monitored events. The types of events monitored may include, without limitation: (1) computational unit creation events, (2) script and/or library load events, (3) file input/output ("I/O") events (e.g., directory read / enumerate; file read, write, create, delete, rename / move; etc.), (4) network events (e.g., create incoming and/or outgoing network connections, etc.), and (5) system calls and/or calls made by a process to an application programming interface ("API") of the OS. Some examples of API calls that may be monitored include, without limitation, (1) API calls related to searching a filesystem, gathering system information, disabling backups, or mounting a disk partition; (2) API calls characteristic to code injection (e.g., from a malicious actor into ransomware host process); and (3) API calls querying system information.

The tracking module 110 may also include a behavior tracking module 114, which may receive data relating to monitored events from the event monitoring module 112. The behavior tracking module 114 may track the behavior of computational entities and maintain threat profiles for the computational entities. Each computational entity's threat profile may include data indicating one or more profile states of the computational entity. The behavior tracking module 114 may update a computational entity's threat profile (e.g., update the profile states in the threat profile) based on monitored events associated with the computational entity, thereby mapping the events to corresponding states.

In some embodiments, the behavior tracking module 114 augments data corresponding to tracked events with additional static and/or dynamic information including, without limitation, context details and/or state information associated with the event. The behavior tracking module 114 may map events to internally tracked states that include additional metadata (e.g., counters and/or timestamps). Such states may be tracked on different levels, for example, (1) per computational entity performing the action, for example, a process enumerating directories, reading and writing files; (2) per unique application identifier (e.g., sha256 file hash), to track behavior of an application across processes, and (3) on system level (e.g., system states impacting or responding to application behavior).

In some embodiments, the behavior tracking module 114 extends the notion of behavioral state tracking from individual computational units (e.g., individual processes) to sets of related computational units (e.g., multiple-process chains). The behavior tracking module 114 may determine whether suspicious behaviors of one computational unit are likely performed on behalf of another computational unit (e.g., a main process in control). In addition, the behavior tracking module 114 may determine such likelihood, identify the effective actor and the extent of the set of related computational units, and track behavior across the entire set of related computational units, ultimately enabling the behavioral security engine 100 to enforce policies for all computational units in a set responsible for malicious behavior (e.g., ransomware behavior). In addition, the behavior tracking module 114 may use heuristics (described below) to differentiate potentially malicious behaviors from safe behaviors of computational units that are part of the OS, thereby reducing false positive detection of threatware.

Heuristics:

In some embodiments, the heuristics module 120 implements processes (e.g., heuristic processes) for assigning threatware categories (e.g., ransomware categories) to computational entities, assigning threatware subcategories (e.g., ransomware subcategories) to computational entities, and/or adjusting threat scores of computational entities. From time to time (e.g.,

periodically or in response to specific events), the heuristics module 120 may query the behavior tracking module 114 to determine the profile states of a computational entity, and may notify the behavior scoring module 130 when a heuristic that affects a computational entity's category, subcategory, and/or score is triggered (e.g., by a change in the state of the computational entity). In some embodiments, the heuristics module 120 may notify the protection and mitigation module 160 when a heuristic associated with initiation a protection or mitigation action (e.g., initiating a backup process) is triggered. The heuristics module 120 may track the values of heuristic states and categories internally.

Categorization and Subcategorization:

The heuristic module's categorization module 122 may implement categorization heuristics suitable for assigning threat categories to computational entities based on their behavior. In some embodiments, the categorization module 122 may assign a computational entity the threat category that matches (e.g., most closely matches) the computational entity's current set of profile states.

In some embodiments, the categorization module 122 uses the categorization heuristics to detect ransomware-specific behaviors, at least one of which is generally characteristic to any type of ransomware. These categorization heuristics may act as first-level filters and triggers. These first-level filters may facilitate the maintenance of a low false-positive ransomware detection rate, and/or may narrow the monitoring scope of the tracking module 110, thereby lowering overall impact on system performance. Suspicious computational entities may be assigned ransomware categories (for example, the following ransomware categories) based on detection of certain behaviors exhibited by the applications:

Category A: Encrypting files on local, remote (network), or external drives;

Category B: Encrypting raw storage device or parts of it (e.g., master boot record (MBR) sector on boot drive, or entire backup drive); and/or

Category C: Disabling system backups.

A suspicious computational entity that exhibits behaviors associated with two or ransomware categories may be assigned to all of the corresponding ransomware categories at the same time.

A computational entity may be assigned to ransomware Category A if the entity exhibits behaviors associated with encrypting files, including, without limitation, (1) enumerating a storage device or a directory of a file system, and/or (2) modifying private files.

Storage Device and/or Directory Enumeration: Heuristics for detecting enumeration of a storage device (e.g., disk) or directory may take advantage of the observation that ransomware generally has a starting point during a search to locate data files. The starting point

generally includes, but is not limited to, well-known top-level directories (e.g., the driver letter root, user folder, user's documents and desktop, etc.). In some embodiments, several top level directories on the endpoint device are monitored for enumeration, including the root of each mount point (internal, external drives, network shares, backups), user folders, and several well-known subfolders under the user folders. Additional high-value directories may be monitored (e.g., high-value directories that could be derived by ransomware in some other way than using recursive enumeration, such as by querying the system state for mount points or user names). For example, network share mounts that often use directory structure conventions may be monitored.

In some embodiments, descendant directories deeper in the directory tree are not monitored for the starting point directory enumeration. This technique may help the behavioral security engine 100 maintain a low false positive rate, without increasing false negatives, because real ransomware generally does not possess a knowledge of target machine directory structure, beyond the starting point directories, without performing the initial discovery.

In some embodiments, after a computational entity enumerates a starting point directory, the descendant directories may be recursively monitored for enumeration to establish whether a continuous recursive directory enumeration occurred all the way down to a directory in which a potential file encryption operation is detected. This heuristic can further reduce false positives in cases in which a legitimate application enumerates one of the known starting point directories and independently, often for other reasons, enumerates a directory known to that application, which may be several levels down from the starting point directory, but without enumerating any directories in between.

On some operating systems, false positive mitigation techniques may be used to rule out directory enumeration by a computational unit while the computational unit is loading and enumerating folders to locate dynamic libraries to be linked. For example, a useful false positive mitigation technique in this case may be to discard events initiated by a computational unit prior to initiating the execution of the computational unit's main method.

Modifying Private Files: Some files are known to be private only to a specific application and are known to be maintained only by that application. Examples may include document files created by an installer of another application. When a computational entity attempts to modify a file that is private to a separate computational entity, that behavior may be indicative of ransomware. (In addition or in the alternative, modification of private files may be used by the subcategorization module 124 to subcategorize suspected malware as "file-encrypting.")

A computational entity may be assigned to ransomware Category B if the entity exhibits behaviors associated with encrypting a raw storage device or a portion thereof, including, without limitation, modifying a raw storage device.

Modifying Raw Device: Examples of raw device modification operations may include attempting to access a raw storage device and write to specific regions thereof (e.g., master boot record (MBR) sector in case of the boot device, partition table or arbitrary region of non-bootable data drive, etc.), bypassing the OS. In case of a computational entity that includes a set of related computational units, the effective actor may be derived by traversing the process hierarchy. To reduce or avoid potential false positives during an OS upgrade, raw device modification operations initiated by computational units that are well known OS components and are not invoked by an external computational unit may be discarded.

Raw device modification heuristics may extend to detection of modification of low-level file system structures (e.g., inodes). For example, ransomware may traverse and encrypt chosen inode data directly, without going through file-system virtualization layer, like most user-space applications. Ransomware could also modify file system structure pointers to hide files from user, without actually encrypting the inode data, thereby achieving a similar end-effect.

In some embodiments, the categorization module 122 may assign separate categories to computational entities that attempt to write to specific regions of the raw device (e.g., the MBR or partition table) and computational entities that attempt to modify low-level file system structures. For example, the categorization module 122 may assign Category B1 to the former computational entities, and Category B2 to the latter computational entities.

A computational entity may be assigned to ransomware Category C if the entity exhibits behaviors associated with disabling a computer system's backup capabilities, including, without limitation, deleting / disabling backup files and/or disabling backup / restore functionality.

Disabling System Backups: Examples of behaviors associated with disabling system backups may include (1) deleting, encrypting, or otherwise modifying backup files, and (2) disabling a computer system's ability to create new backups or restore data and settings from existing backups. In the case of a computational entity that includes a related set of computational units, the effective actor may be determined to derive intent and avoid triggering false positives in legitimate use-cases (e.g., when a user is re-configuring backups).

Since some of these behaviors could be also exhibited by legitimate software, additional heuristic techniques (described below) may be by the categorization module 122 to further increase detection and prevention accuracy and reduce the false positive rate.

When a ransomware category is assigned to a computational entity, the behavior scoring module 130 may use scoring heuristics specific to that category to determine an initial threat

score for the computational entity. Suitable scoring heuristics for computational entities assigned to ransomware Categories A, B, B1, B2, and/or C are described in further detail below.

After a ransomware category is assigned to a computational entity, the subcategorization module 124 may use subcategorization heuristics specific to the assigned category to further classify potential ransomware into subcategories. These heuristic methods may factor in complexity and uncertainty characteristics to detection of different malicious techniques. For example, Category A ransomware may be divided into subcategories based on any suitable criteria, including but not limited to: (1) whether the computational entity encrypts files in-place, out-of-place, or both, (2) whether the computational entity has multiple computational units or a single computational unit, and (3) whether the computational entity utilizes built-in system utilities.

When a ransomware subcategory is assigned to a computational entity, the behavior scoring module 130 may use scoring heuristics specific to that subcategory to adjust the threat score for the computational entity. Suitable scoring heuristics for computational entities assigned to the various ransomware subcategories are described in further detail below.

After a ransomware category is assigned to a computational entity, the categorization module 122 may adjust the category assigned to the computational entity dynamically as more behavior is processed and more is known about the entity. Likewise, after a ransomware subcategory is assigned to a computational entity, the subcategorization module 124 may adjust the subcategory assigned to the computational entity dynamically as more behavior is processed and more is known about the entity.

In general, categorization and subcategorization may allow for better control of prevention sensitivity for a given category or subcategory of ransomware, without affecting scoring of other categories or subcategories.

In some embodiments, one or more of the following heuristic techniques may be used by the categorization module 122 for categorization, by the subcategorization module 124 for subcategorization, and/or by the scoring module 130 for initial scoring:

- Tracking of a computational entity's use of low level IO operations and pathways that can lead to net-resulting file modification.
- Tracking of file modifications initiated by a computational entity's use of, in relation to file IO path, file node attributes and content.
- Tracking of a computational entity's modifications of files in directories and subdirectories that the entity previously enumerated during discovery stage.
- Tracking of number of files modified by the computational entity.

- Filtering file modification events based on file type. Modifications of all files may be accounted for and/or prevented, however, modifications of particular file types may carry more weight than others. For example, data files may be assigned more weight.
- Filtering file modifications based on file location. The system may not track modifications of files located in specific directories (e.g., user folders, on network shares or backup devices, etc.) when the modifications are performed by specific computational entities. For example, events associated with known backup software modifying files on backup store may be filtered out.
- Filtering file modifications based on file location and file type. More weight may be assigned to modification of non-data files on non-local drives (e.g., network drives, external drives, etc.). This method may increase detection sensitivity and reduce data-loss when ransomware attempts to encrypt non-data files on non-local drives. Non-local drives can be particularly vulnerable to ransomware because (1) ransomware can generally encrypt such drives entirely without taking away a mechanism to deliver a ransom message to the user, and (2) network shares and backup drives may be of especially high value to the user.
- Tracking of different types of file modifications, by keeping track of counters and ratios of different IO operations (e.g., reads, writes, creates and/or deletes) on files. These counter values and ratios may be updated regularly (e.g., after every IO operation). This data may be used to determine whether (1) a file of interest is being directly modified (in-place encryption), or (2) a copy of the file is being modified, followed by deletion of the original file (out-of-place encryption). This technique may play a role in assigning Category A ransomware to the corresponding subcategory (e.g., the in-place encrypting subcategory or the out-of-place encrypting subcategory). Once the computational entity is subcategorized, the tracking module 110 may track the file encryption behaviors specific to the subcategory more closely, and may assign a more accurate scoring heuristic and more appropriate threat score thresholds specific to the subcategory.
- Tracking of file IO operation patterns indicating legitimate applications, to lower the false positive rate. If a computational entity exhibits specific sequences or patterns of file IO operations (e.g., read, create, write, and/or delete), the heuristics module 120 may determine that the entity is non-threatening, and the tracking module 110 may filters out IO operations of such entities. For example, software installers may modify a small number of pre-existing files, while also generating a larger number of new files. Such a pattern does not fit the behavioral signature of out-of-place encryption and can therefore be rejected to reduce false positives.

- Tracking of file modification and disk enumeration spread across a related set of computational units. This technique may be used for subcategorization and for tracking the behavior of a computational entity that includes multiple computational units. For example, an entity assigned to a ransomware category (e.g., Category A) may be subcategorized further as a single-process entity or a multi-process entity (independently from being subcategorized as an in-place encrypting entity or an out-of-place encrypting entity) for more accurate scoring and independent threshold assignment. In addition, after the entity is categorized as a multi-process entity, the behavior (e.g., heuristic ransomware behavior) of the entity's computational units may be tracked and attributed to the entire entity. The entity's computational units may include known built-in OS tools, in which case invocation of the tool is treated similarly to invocation of an API call. The effective actor among the entity's computational units may be identified by walking the process hierarchy and identifying the process higher in the hierarchy that is not part of the OS and is not a known verified application. In this way, ransomware behavior may be tracked across processes spawned by the actor process or its descendants. For example, a malicious actor may delegate enumeration or encryption tasks to one or multiple sub-processes. When the entity's threat score reaches the assigned threshold, a prevention policy may be applied to the entire entity to ensure that no dangling processes continue executing sub-tasks after terminating only part of the process chain. Due to higher complexity in detection of multi-process ransomware and the intent not being always clear, scoring heuristics and thresholds may be tuned accordingly to maintain a low false positive rate.
- For detection and categorization of Class B ransomware, the categorization module 122 may determine the region of the storage device being modified and correlate that region with the storage device type. For a boot-type storage device, attempts to modify the master boot record (MBR) region may be highly suspicious. For a non-boot-type storage device (e.g., a backup device), modification of any portion of the raw device may be suspicious. These heuristics take advantage of the observation that ransomware that encrypts data at the raw device level generally still needs to be able to execute some piece of code to display the ransomware message with instructions.

Scoring:

The behavior scoring module 130 may maintain a threat score for each computational entity of interest, and the threat score may be persisted across application invocations. The behavior scoring module 130 may receive score adjustment notifications from the heuristics

module 120, and may adjust the threat scores for the affected computational entities accordingly. When a computational entity's threat score reaches a corresponding threshold associated with a protection or mitigation policy, the behavior scoring module 130 may signal the protection and mitigation module 160 to apply the corresponding protection or mitigation policy.

As described above, the scoring techniques used by the behavior scoring module 130 for a given computational entity may be specific to the ransomware category and/or subcategory assigned to the entity. For example, the scoring algorithm, scoring units, and/or scoring weights (e.g., multipliers) used by the behavior scoring module 130 to calculate a computational entity's threat score may be specific to the entity's ransomware category and/or subcategory. In some embodiments, the behavior scoring module 130 may also adjust the scoring units and/or scoring weights for a computational entity in response to score adjustment notifications received from the heuristics module 120. As another example, the threshold at which a computational entity's threat score triggers application of a protection or mitigation policy may be specific to the entity's ransomware category and/or subcategory.

In some embodiments, the scoring algorithm (e.g., scoring progression), scoring units, scoring weights, and/or thresholds specific to the different threat classes may be configurable for flexibility and tuning. For example, the rate of increase of a threat score may be configurable to facilitate better tuning of the scoring algorithm and reflect how the various score increasing and score decreasing heuristics can influence the score progression, also accounting for different weights. In some cases, a score increasing or decreasing heuristic may adjust the score by a number of units multiplied by the current multiplier assigned to the entity. In some cases, a score increasing or decreasing heuristic may adjust the heuristic multiplier for that entity, thereby allowing the heuristic to affect the score progression when other scoring heuristics trigger.

In some embodiments, the behavior scoring module 130 may receive a notification from the score increasing module 126 of the heuristics module 120 when one or more of the following score-increasing heuristics triggers (e.g., for a computational entity that has already been assigned a potential ransomware category):

- Computational entity modifies multiple files. In some embodiments, the higher the number of files modified by the potential ransomware entity, the greater the increase in the threat score.
- Computational entity modifies files of multiple file type classes, which distinguishes the entity from an entity that modifies only files of a single, specific file type class. For example, an unknown computational entity modifying Office documents, text, images, video, and audio files is unlikely to be a file editor. In some embodiments, for every file

modified by a potential ransomware entity, the tracking module 110 detects and tracks the original file type and assigns it a file type class (e.g., document, image, video, audio, text, etc.). In some embodiments, the number of file type classes the entity modifies is tracked, and when the number of file type classes modified by the entity reaches a specified threshold, the threat score of the entity is increased.

- Detection of file content, name and attribute modification indicating a more definite encryption vs. a non-malicious file content modification, using the following heuristics:
  - Comparison of new file names compared to old file names for files with modified content. File name change patterns could indicate clear intent to encrypt as some ransomware uses this technique as a simple way of communicating to the endpoint user which files have been encrypted,
  - file magic header after modification different than original magic header,
  - file magic header after modification not matching file extension.
  - file magic matching original but modification of every file follows a pattern: similar data size modified, or at similar offset, or similar content written to each file.
- Rate of file modification: application modifies (in-place or not-in-place) many files within a small time window.
- File extension naming patterns for new files created as a result of not-in-place encryption. Once application is sub-categorized as not-in-place encrypting, file extensions of new files created by its process are tracked per every new file created by the process. The method looks for continuity in file extension names generically, rather than only relying on static signatures of extension names commonly used for encrypted files. Typical ransomware is more likely to use either 1) the same arbitrary file extension string for every new encrypted file; often a human readable word used as a hint to the ransomware victim or 2) a different arbitrary file extension string for every new encrypted file, such as a string containing some pseudo-randomness. Potential not-in-place encrypting ransomware that follow either of the 2 these file extension patterns get an additional heuristic score boost, while the ones that break these patterns are not affected by this method.
- Persistence – application configures itself as a service or schedules itself as a recurring job.
- Tracking application behavior across process instances, i.e. handling ransomware that is re-run. If previous instance of the application exhibited ransomware behavior, the behavior is tracked per hash and then inherited by the newer process instance. This

method allows for applying policy sooner on next execution, preventing ransomware from achieving incremental progress as it is being re-executed. On next process execution, new process is either blocked from running, or it is terminated on one of the first file modification attempts. The resulting action depends on accumulation of behavior and confidence during the previous process run. The goal is to prevent ransomware from making incremental progress, while reducing a chance of recurring false positive.

- Application attempts a network connection (such as to connect to potential C2). This heuristic is of more value if the application connects to server before modifying data.
- Application queries unique system information, such as hardware serial number, UUID, MAC address, IP address, hostname.
- Application queries CPU information, such as to detect number of cores and whether the environment is virtualized.
- Application invokes randomization API, such as to generate a unique encryption key.
- Applications invoked cryptographic API.
- Application invokes multiple instances of itself – to achieve parallelization. In addition to increasing the score, the system tracks multiple instances in order to apply policy to all instances of the process, much like in case of generic multi process ransomware chain..
- Application vector – applications originating from email applications, web browsers, chat clients, external or network drives.
- Date/Time when application was dropped before exhibiting ransomware behavior – applications that appeared on endpoint more recently (threshold is configurable), are treated as more suspect.
- Application obfuscates itself by either: running out of unusual folder (not where programs are typically installed), unusual name or extension (such as a document extension), with unusual attributes (such as hidden), by deleting or moving the file the code was loaded from.
- Application links to a known encryption library (dynamically or statically)
- Detection of IPC and data passed between processes involved in a potential multi-process ransomware chain. This method helps determine intent and whether process is performing a specific task on behalf of ancestor process or on its own. For example, if a sub-process is spawned, which then performs disk enumeration, heuristic looks for data being passed back to the potential process in control to establish clear intent and whether this behavior should be attributed to that process, effectively extending the multi-process chain.

- Application reputation: applications that are not yet widely known and appear to be potential 0-day attacks, or applications that are not signed by a trusted and verified authority.
- Application has been previously detected as compromised by one of code injection mechanism.
- Application known to process in memory data content that is more prone to a vulnerability and in-memory exploit, or application that has already undergone buffer overflow.
- Commonly known application exhibiting unusual behavior, outside of established norms, indicating potential compromise, even without earlier detection by the system of the actual compromise vector. This method accounts for new, unknown vulnerabilities in core applications. For example, calculator app communicating on the network.

In some embodiments, the behavior scoring module 130 may receive a notification from the score decreasing module 128 of the heuristics module 120 when one or more of the following score-decreasing heuristics triggers (e.g., for a computational entity that has already been assigned a potential ransomware category):

- Verified application that is a known modifier of files of certain sanctioned type (for example, a known photo editor application is modifying image files). Such sanctioned file modifications are bypassed by the system, while still keeping track of modification of other file types to account for a case of application being compromised.
- Application modifying files in temporary file area - such file modifications are ignored.
- Application modifying its own private folder data files, such as an email application modifying e-mail files in data folder corresponding to the actor process.
- In multi-process chain, a method is used to determine if the effective actor in control is part of the operating system. The method uses a combination of file path, verified certificate and file reputation. If the effective process in control is an operating system component, the behavior is not policy-enforced, but it can be optionally tracked and reported. This technique reduces false positives with built-in backup utilities, during OS upgrade and operations driven by GUI controls, such as re-partitioning.
- While calculating multi-process chain, system takes into account standard operating system services that play a role of "launchers", i.e. execute a process on behalf of another process (e.g., WMI, launchd, etc.). In such case, behavior is conducted across such chain, but policy enforcement excludes those standard services that are long-lived and spawned by the OS, and not by the malicious actor.

In some embodiments, the behavior scoring module 130 uses additional heuristic methods to analyze isolated "interesting" behaviors, and the results of such analysis may be used to increase the score of an entity that has been previously classified as potential ransomware. In turn, behaviors that are characteristic of legitimate applications may decrease the current score or bypass one of other heuristics that would otherwise contribute to a false positive. These methods may be helpful to increase certainty.

In some embodiments, the scoring increment associated with each of the foregoing scoring heuristics is relatively low. In some embodiments, none of the scoring heuristics alone may increase the score decisively enough to reach the assigned threshold and trigger the application of a protection or mitigation policy. However, having more scoring methods in the system may allow for higher thresholds to be set and therefore better separation between a real threat and a false positive without sacrificing threat detection.

Configuration, Policy and Protection / Mitigation:

The configuration module 140 may configure (or permit a user or administrator to configure) various parameters and settings of the behavioral security engine, including but not limited to (1) units and weights of the scoring heuristics, (2) categorization and subcategorization parameters, (3) threshold scores that trigger application of protection or mitigation actions for different ransomware categories and subcategories, and (4) protection and mitigation policies (e.g., the particular policy that is triggered when a computational entity's score reaches a particular threshold).

A protection or mitigation policy may define action(s) to be taken against a malicious entity when the entity's threat score reaches the corresponding threshold. Options for responding to a threat may include, without limitation (1) terminating the malicious entity, (2) isolating (e.g., "sandboxing") the malicious entity (e.g., allowing the entity to run, but without allowing the entity to modify data), or (3) allowing the entity to perform all operations and logging the entity's operations.

The policy module 150 may store policy settings retrieved from the configuration module 140. The policy module 150 may receive a notification from the behavior scoring module 130 when an entity's threat score exceeds and associated threshold. In some embodiments, the policy module 150 determines which protection / mitigation action (e.g., terminate, isolate, or log) to perform based on the policy settings and/or the entity's reputation. In some embodiments, the policy module 150 notifies the protection and mitigation module 160 to perform the selected protection / mitigation action.

The protection and mitigation module 160 may enforce protection and mitigation policies in response to receipt of a corresponding notification from the policy module 150. In some embodiments, the protection and mitigation module 160 has the ability to enforce a policy action by denying completion of a file operation, network operation, and/or process creation operation in-line with the operation being requested. In some embodiments, the protection and mitigation module 160 may communicate with the OS to terminate or isolate a running process.

In some embodiments, the protection and mitigation module 160 may optionally (e.g., using a setting) enable the techniques described below in the Section titled "Data Backup and Recovery" (e.g., after a computational entity has been categorized as potential ransomware). These techniques may serve two main purposes: (1) to reduce any data loss that may occur during the transient state of low threat score, e.g., after low number of files are potentially encrypted and before the policy action is taken, and (2) to allow for additional flexibility in increasing the heuristic thresholds, leading to lower false positives.

As just one example, after a computational entity is categorized as Category A ransomware, the protection and mitigation module may begin backing up all files modified by the entity, in-line with and prior to completion of the entity's attempt to modify the files. If the protection and mitigation module 160 subsequently receives a notification from the policy module to restore the files modified by the entity (e.g., in response to the behavior scoring module 130 indicating that the entity's threat score has exceeded the corresponding threshold), the module 160 may then restore the backed-up files.

Data Backup and Recovery

The techniques described in this Section may be used to enhance a behavioral security engine 100 for endpoint devices (e.g., desktops, laptops, terminals, servers, embedded systems) against data loss (e.g., via ransomware) or attempts to make the underlying operating system inoperative (e.g., via destructive malware). The techniques described in this Section may enhance such systems in one or both of the following ways:

1. Protection against file data loss: Behavioral detection systems are generally characterized by a lag between the initial malicious behavior taking place (e.g., data file encryption or destruction) by a suspect application and the time when policy action is taken (e.g., the malicious application is terminated). During this time window, the system can be prone to allow for some data loss, before action is taken to terminate or restrict the potentially malicious application. For example, to maintain low false positives, a conventional behavioral detection system may choose to gather more evidence and gain sufficient confidence. By

contrast, the techniques described in this Section may protect the file data integrity during the period of lower confidence.

2. Reduction of false positives: Since the technique protects against data loss during the period of low confidence, the technique may allow for additional flexibility in tuning sensitivity until the state of higher confidence is reached, consequently reducing (e.g., minimizing) the false positive rate. When policy action is taken, data files at risk that were modified or deleted during the period of low confidence can be completely restored.

A data backup and recovery module may be incorporated into the behavioral security engine 100 as follows:

- The data backup and recovery module receives notifications about file and directory I/O operations (e.g., create, delete, move / rename, read, write). The notifications may be received in-line with the operations taking place.
- The data backup and recovery module receives notifications from the behavioral security engine 100 on interesting state changes of monitored applications.

The security engine 100 may notify the data backup and recovery module (which may be embedded within or external to the security engine 100) when to start performing backups of files affected by a potentially malicious application. For example, the security engine 100 can send the notification early, when a sample has been pre-categorized as potentially malicious, but before any data loss occurred. From that point onwards, the data backup and recovery module may monitor file operations performed by the application / process (or process chain). The data backup and recovery module may be capable of stalling the original file I/O operation (operation requested by the monitored application), for example write or delete, creating a copy of each file that the monitored application is preparing to modify or delete, and storing the copies in a backup area. The file backup operation can be implemented either directly in the file-filter driver or in a user-space agent, as long as the original operation is stalled until the file backup is complete.

The backup area may be protected by file-filter driver, as such, applications that are not part of the security engine are not allowed to modify it. When the prevention system reaches threshold and applies policy at higher certainty point, the files are available for restoration. The restore functionality can be triggered automatically by the prevention system, or by user action.

The techniques described herein may be applicable to Windows, macOS, Linux, and/or other software platforms. There may be differences in implementation details across platforms (e.g., because some data point inputs associated with specific detection methods may not be available on every platform, and/or because low-level operating system features or services available to ransomware differ slightly across different platforms). For example, some of the

techniques described herein may differ in implementation and/or score weighting on each platform.

Some Embodiments of Ransomware Detection Methods

Referring to FIG. 2, a ransomware detection method 200 may include steps of detecting 210 an event, determining 220 whether the detected event is a trigger event, and if so, handling the event in accordance with its type. For example, if the trigger event is the creation of a new computational unit, the event may be handled by performing a process 231 for handling creation of a new computational unit. If the trigger event is a computational entity accessing a raw storage device, the event may be handled by performing a process 232 for handling an access of a raw storage device. If the trigger event is a computational entity accessing a backup file, the event may be handled by performing a process 233 for handling an access of a backup file. If the trigger event is a computational entity enumerating a directory, the event may be handled by performing a process 234 for handling directory enumeration. If the trigger event is a computational entity accessing a file, the event may be handled by performing a process 235 for handling an access of a file. If the trigger event is a computational entity accessing a network, the event may be handled by performing a process 237 for handling an access of a network. If the trigger event is a computational entity performing an API call, the event may be handled by performing a process 238 for handling an API call.

The foregoing examples of trigger events and corresponding handler processes is non-limiting; other trigger events may be detected and handled.

If the detected event is not a trigger event, the event detection step 210 may be repeated. Likewise, after a trigger is handled, the event detection step 210 may be repeated. Some techniques for detecting events (e.g., process creation and termination events; file read / write / create / delete / move / mmap events; network activity events; API and system call events, etc.) are described above.

Non-limiting embodiments of the handler processes 231-235, 237, and 238 are illustrated in FIGS. 3A-3G, respectively.

Referring to FIG. 3A, a process 231 for handling an event associated with creating a new computational unit may include steps 311-315. In step 311, the tracking module 110 begins tracking the profile state of a new computation unit (CU). In some embodiments, the profile state of the new CU is initialized such that all elements of the profile state (e.g., file IO counters, network state, system call state, threat score, score weights, CU category, CU subcategory, etc.) are initialized to default values. In some embodiments, the tracking module 110 obtains reputation information indicating a reputation of an application A1 that created the new CU, and

stores that reputation information in the CU's state profile. Some examples of suitable techniques for tracking the profile state of a CU during the CU's execution are described above.

In step 312, the tracking module 110 determines whether the new CU is a single-CU computational entity or part of a multi-CU computational entity. If the new CU is part of a multiple-CU computational entity, the new CU is added to the computational entity's state profile, thereby linking the new CU to the computational entity. Some examples of suitable techniques for determining whether a CU is part of a related set of CUs forming a computational entity are described above.

In step 313, the tracking module 110 determines whether a previous instance of a CU corresponding to the application A1 was categorized as ransomware. If not, the process 231 terminates. If so, the tracking module 110 restores (314) the state of the previous instance of the CU in the state profile of the new CU and the behavioral security engine 100 performs (315) a process 250 for evaluating the CU's threat score and applying a corresponding policy, if any.

Referring to FIG. 3B, a process 232 for handling an event associated with accessing a raw storage device may include steps 321-328. At step 321, the tracking module 110 determines whether the event indicates that a computational entity (CE) is attempting to modify a master boot record (MBR) of a boot-type storage device.

At step 321, if the event indicates that the CE is attempting to modify the MBR, the process 232 proceeds from step 321 to step 322. At step 322, the tracking module 110 determines whether the CE is part of a larger computational entity, and if so, links the computational entities. At step 323, the categorization module 122 assigns potential ransomware Category B (or B1) to the CE. At step 324, the behavioral security engine 100 updates the CE's threat score. At step 325, the behavioral security engine 100 performs a process 250 for evaluating the CE's threat score and applying a corresponding policy, if any. Process 232 then ends.

At step 321, if the event does not indicate that the CE is attempting to modify the MBR, the process 232 proceeds from step 321 to step 326. At step 326, if the event indicates that the CE is attempting to modify the file system of a storage device, the process 232 proceeds to step 327. At step 327, the tracking module 110 determines whether the CE is part of a larger computational entity, and if so, links the computational entities. At step 328, the categorization module 122 assigns potential ransomware Category B (or B2) to the CE. At step 324, the behavioral security engine 100 updates the CE's threat score. At step 325, the behavioral security engine 100 performs a process 250 for evaluating the CE's threat score and applying a corresponding policy, if any. Process 232 then ends.

At step 326, if the event does not indicate that the CE is attempting to modify the file system of a storage device, the process 232 ends.

Referring to FIG. 3C, a process 233 for handling an event associated with accessing a backup file may include steps 331-335. At step 331, if the event indicates that the CE is attempting to modify the backup capabilities of the computer system (e.g., by disabling the backup functionality, deleting a backup file, modifying a backup file, etc.), the process 233 proceeds from step 331 to step 332. At step 332, the tracking module 110 determines whether the CE is part of a larger computational entity, and if so, links the computational entities. At step 333, the categorization module 122 assigns potential ransomware Category C to the CE. At step 334, the behavioral security engine 100 updates the CE's threat score. At step 335, the behavioral security engine 100 performs a process 250 for evaluating the CE's threat score and applying a corresponding policy, if any. Process 233 then ends.

At step 331, if the event does not indicate that the CE is attempting to modify the backup capabilities of the computer, the process 233 ends.

Referring to FIG. 3D, a process 234 for handling an event associated with enumerating a directory may include steps 341-347. At step 341, if the directory being enumerated by the CE is one of a specified set of suspicious enumeration starting points, the process 234 proceeds from step 341 to step 342. At step 342, the categorization module 122 assigns potential ransomware Category A to the CE. At step 343, the behavioral security engine 100 adds the directory to a set of enumerated directories in the CE's profile state. At step 344, the behavioral security engine 100 activates the scoring and subcategorization heuristics associated with Category A ransomware for the CE. Process 234 then ends.

At step 341, if the directory being enumerated by the CE is not one of the specified set of suspicious enumeration starting points, the process 234 proceeds from step 341 to step 345. At step 345, if the directory being enumerated by the CE is a child (e.g., a direct child) of a directory that is already in the set of enumerated directories in the CE's profile state, the process 234 proceeds from step 345 to step 346. At step 346, the behavioral security engine 100 adds the directory to the set of enumerated directories in the CE's profile state. At step 347, the tracking module 110 determines whether the CE is part of a larger computational entity, and if so, links the computational entities. Process 234 then ends.

At step 345, if the directory being enumerated by the CE is not a child (e.g., a direct child) of a directory that is already in the set of enumerated directories in the CE's profile state, the process 234 ends.

Referring to FIG. 3E, a process 235 for handling an event associated with accessing a file may include steps 351-363. At step 351, if the event indicates that the CE is attempting to

modify a file that is private to a different computational entity, the process 235 proceeds from step 351 to step 352. At step 352, the tracking module 110 determines whether the CE is part of a larger computational entity, and if so, links the computational entities. At step 353, the categorization module 122 assigns potential ransomware Category A to the CE. At step 354, the behavioral security engine 100 updates the CE's threat score. At step 355, the behavioral security engine 100 performs a process 250 for evaluating the CE's threat score and applying a corresponding policy, if any. Process 235 then proceeds to step 356. Likewise, at step 351, if the event indicates that the CE is attempting to modify a file that is not private to a different computational entity, the process 235 proceeds from step 351 to step 356.

At step 356, if the CE has already been assigned a potential malware category, the process 235 proceeds from step 356 to step 357. At step 357, if the file is in a directory in the CE's set of enumerated directories (as indicated in the CE's profile state), the process 235 proceeds from step 357 to step 358. On the other hand, if the CE has not been categorized as potential ransomware category or the file is not in one of the CE's enumerated directories, process 235 ends.

At step 358, the tracking module 110 maps attributes of the file and the event (e.g., file type, etc.) to heuristic states. At step 359, the behavioral security engine 100 runs the subcategorization heuristics associated with the CE's ransomware category (e.g., Category A). At step 360, the behavioral security engine 100 runs the score adjustment heuristics. At step 361, the behavioral security engine 100 updates the CE's threat score. At step 362, the behavioral security engine 100 performs a process 250 for evaluating the CE's threat score and applying a corresponding policy, if any. At step 363, the protection and mitigation module 160 backs up the file. Process 235 then ends.

Referring to FIG. 3F, a process 237 for handling an event associated with accessing a network may include steps 371-374. At step 371, if the CE attempting to access the network has been categorized as potential Class A ransomware, the process 237 proceeds from step 371 to step 372. At step 372, the behavioral security engine 100 runs the score adjustment heuristics. At step 373, the behavioral security engine 100 updates the CE's threat score. At step 374, the behavioral security engine 100 performs a process 250 for evaluating the CE's threat score and applying a corresponding policy, if any. Process 237 then ends.

At step 371, if the CE attempting to access the network has not been categorized as potential Class A ransomware, the process 237 ends.

Referring to FIG. 3G, a process 238 for handling an event associated with performing an application programming interface (API) call (or system call) may include steps 381-385. At step 381, if the API call (or system call) is one of a specified set of API calls (or system calls)

associated with ransomware, the process 238 proceeds from step 381 to step 382. At step 382, if the CE attempting to access the network has been categorized as potential Class A ransomware, the process 238 proceeds from step 382 to step 383. On the other hand, if the API call (or system call) is not one of the specified set of calls or the CE has not been categorized as potential Class A ransomware, process 238 ends.

At step 383, the behavioral security engine 100 runs the score adjustment heuristics. At step 384, the behavioral security engine 100 updates the CE's threat score. At step 385, the behavioral security engine 100 performs a process 250 for evaluating the CE's threat score and applying a corresponding policy, if any. Process 238 then ends.

Referring to FIG. 3H, a process 250 for evaluating the threat score of a computational entity and applying a security policy to the entity may include steps 391-398. At step 391, the behavioral security engine 100 determines whether the CE's threat score exceeds a corresponding threshold. If not, the process 250 ends.

If the CE's threat score exceeds the threshold, the process 250 proceeds to step 392. At step 392, if a prevention policy is enabled, the behavioral security engine 100 terminates the CE (step 395) and restores the files modified by the CE (step 398). If the prevention policy is not enabled, the process 250 proceeds to step 393. At step 393, if an isolation policy is enabled, the behavioral security engine isolates the CE (step 396). At step 394, if a logging policy is enabled, the behavioral security engine initiates logging of the CE's behavior. At step 398, the behavioral security engine restores the files modified by the CE.

Referring to FIG. 4, a ransomware detection method 400 may include steps 410-470. The steps of the ransomware detection method 400 may be performed, for example, by the behavioral security engine 110, using the techniques described above.

At step 410, one or more instances of a computational entity (CE) engaging in one or more first behaviors associated with encrypting a file, encrypting a storage device, and/or disabling a backup of at least a portion of a computer system are detected (e.g., by the tracking module 110). The detected instances of the first behaviors may be detected based on first events initiated by the CE in the computer system. Such events may be detected, for example, by the event monitoring module 114.

At step 420, a potential ransomware category is assigned to the CE based, at least in part, on the detected instances of the first behaviors. The ransomware category may be assigned to the CE, for example, by the categorization module 122.

At step 430, after the potential ransomware category is assigned to the CE, a protection action may be initiated to protect resources of the computer system against ransomware attacks. The protection action may be initiated, for example, by the behavior scoring module 130, the

policy module 150, and/or the protection and mitigation module 160. The protection action may, for example, protect storage resources of the computer system against modification operations.

At step 440, after the potential ransomware category is assigned to the CE, one or more instances of the CE engaging in one or more second behaviors corresponding to the ransomware category may be detected. The instances of the second behaviors may be detected, for example, by the behavior tracking module 114. In some embodiments, the instances of the second behaviors are detected based on second events initiated by the CE in the computer system. Such events may be detected, for example, by the event monitoring module 112. In some cases, the tracking module 110 may be configured to ignore the CE initiating the second events and/or engaging in the instances of the second behaviors unless the ransomware category has been assigned to the CE.

At step 450, a potential ransomware subcategory is assigned to the CE based on the detected instances of the second behaviors. The ransomware subcategory may be assigned to the CE, for example, by the subcategorization module 124.

At step 460, a threat score may be assigned to the CE (e.g., the threat score may be initialized or adjusted) based on the potential ransomware category, the potential ransomware subcategory, and/or the detected instances of the second behaviors. The threat score may be determined, for example, by the behavior scoring module 130 based on data provided by the heuristics module 120.

At step 470, a determination is made as to whether the threat score exceeds a threshold score associated with the ransomware category or subcategory assigned to the CE. This determination may be made, for example, by the behavior scoring module 130. If the threat score exceeds the corresponding threshold score, a policy action (e.g., a mitigation action or a protection action) associated with the ransomware category or subcategory assigned to the CE may be initiated. The policy action may be initiated, for example, by the behavior scoring module 130, the policy module 150, and/or the protection and mitigation module 160.

One of ordinary skill in the art will appreciate that the ransomware detection and prevention methods described herein use a variety of concepts to attain low a false negative rate, a low false positive rate, and a very low impact on computer system performance. The term "ransomware" generally refers to a large and diverse set of data-encrypting malware. The techniques used by ransomware may vary greatly: MBR and raw disk encrypting; various strains of file-encrypting; strains targeting specific file types, strains utilizing different file IO and encryption techniques, single CU or multi CU strains, strains utilizing built-in OS tools, strains using different file system discovery techniques, etc. Thus, ransomware detection and protection

techniques that are robust and do not significantly degrade system performance are very difficult to devise.

The techniques described herein address this tension by (1) targeting foundational behaviors that are highly likely to be used by any data-encrypting malware, (2) assigning potential ransomware categories and subcategories to computational entities (CEs) based on their behavior and customizing (or tailoring) the detection, protection, and mitigation policies applied to each CE based on those categories and subcategories, and (3) using a rich set of category-specific and subcategory-specific heuristics to increase and decrease the CE's threat score based on the CE's behavior. In this way, the techniques described herein can further increase overall detection confidence, yielding a balanced system suitable for effective ransomware prevention.

Some embodiments of the behavioral security engine 100 may exhibit the following characteristics:

(1) A multi-tier architecture may be used, with a rich set of heuristic methods present in each tier.

(2) An initial categorization tier may be used to analyze a limited, well-defined set of behavioral events for every CE and to categorize potential ransomware into specific high-level categories. This tier may act as pre-filter that filters out obvious non-ransomware while continuing to track any potential ransomware. The pre-filtering may have a variety of benefits. For example, by monitoring high-level types of behavior any file-encrypting malware is highly likely to exhibit, the initial categorization tier ensures that a very high percentage of potential ransomware (e.g., all ransomware) (including zero-day ransomware) passes through the filter, yielding a low false negative rate. Filtering out the non-ransomware also lowers the false positive rate and increases the effectiveness of the more specialized heuristics in the other tiers by filtering out irrelevant events that could otherwise confuse the heuristics or thwart their analysis. As another example, this approach conserves system resources (and reduces the impact on system performance) by reducing the number and types of events that are collected before the first trigger point and limiting the scope of computational analysis done for every CE.

(3) Once the initial high-level category is determined and if further behavioral analysis is warranted to gain confidence, the secondary categorization tier opens up the event stream relevant to that high-level category and adapts the behavioral analysis by selecting algorithms specific to that category to establish a high-confidence sub-category. For example, the actual encryption technique may be identified in this step. The benefit is an adaptive choice of specific lower-level encryption-detection algorithms that are well-suited for the encryption method, which can be an important factor in achieving high accuracy and both low false negative and false positives. This tier can also detect a starting point in a potential encryption of the first data

unit, which can trigger the engine to start backing up files and prevent data loss until a prevention action is applied.

(4) Once the specific encryption techniques are identified, another tier of specialized heuristics (for the specific encryption technique) may be used to increase and decrease the CE's threat score. A rich set of heuristics may be used to monitor specific behaviors, patterns and anti-patterns that have a net effect of balancing each other. The use of such methods may increase the confidence in the engine's analysis, further decreasing false positives and false negatives. Also, the system architecture may facilitate easy extensibility and addition of novel methods in this category.

Further Description of Some Embodiments

In some examples, some or all of the processing described above can be carried out on an endpoint device (e.g., a personal computing device, laptop device, tablet device, smartphone device, etc.), on one or more centralized computing devices, or via cloud-based processing by one or more servers. In some examples, some types of processing occur on one device and other types of processing occur on another device. In some examples, some or all of the data described above can be stored on a personal computing device, in data storage hosted on one or more centralized computing devices, or via cloud-based storage. In some examples, some data are stored in one location and other data are stored in another location. In some examples, quantum computing can be used. In some examples, functional programming languages can be used. In some examples, electrical memory, such as flash-based memory, can be used.

FIG. 5 is a block diagram of an example computer system 500 that may be used in implementing the technology described in this document. General-purpose computers, network appliances, mobile devices, or other electronic systems may also include at least portions of the system 500. The system 500 includes a processor 510, a memory 520, a storage device 530, and an input/output device 540. Each of the components 510, 520, 530, and 540 may be interconnected, for example, using a system bus 550. The processor 510 is capable of processing instructions for execution within the system 500. In some implementations, the processor 510 is a single-threaded processor. In some implementations, the processor 510 is a multi-threaded processor. The processor 510 is capable of processing instructions stored in the memory 520 or on the storage device 530.

The memory 520 stores information within the system 500. In some implementations, the memory 520 is a non-transitory computer-readable medium. In some implementations, the memory 520 is a volatile memory unit. In some implementations, the memory 520 is a non-volatile memory unit.

The storage device 530 is capable of providing mass storage for the system 500. In some implementations, the storage device 530 is a non-transitory computer-readable medium. In various different implementations, the storage device 530 may include, for example, a hard disk device, an optical disk device, a solid-date drive, a flash drive, or some other large capacity storage device. For example, the storage device may store long-term data (e.g., database data, file system data, etc.). The input/output device 540 provides input/output operations for the system 500. In some implementations, the input/output device 540 may include one or more of a network interface devices, e.g., an Ethernet card, a serial communication device, e.g., an RS-232 port, and/or a wireless interface device, e.g., an 802.11 card, a 3G wireless modem, or a 4G wireless modem. In some implementations, the input/output device may include driver devices configured to receive input data and send output data to other input/output devices, e.g., keyboard, printer and display devices 260. In some examples, mobile computing devices, mobile communication devices, and other devices may be used.

In some implementations, at least a portion of the approaches described above may be realized by instructions that upon execution cause one or more processing devices to carry out the processes and functions described above. Such instructions may include, for example, interpreted instructions such as script instructions, or executable code, or other instructions stored in a non-transitory computer readable medium. The storage device 530 may be implemented in a distributed way over a network, for example as a server farm or a set of widely distributed servers, or may be implemented in a single computing device.

Although an example processing system has been described in FIG. 5, embodiments of the subject matter, functional operations and processes described in this specification can be implemented in other types of digital electronic circuitry, in tangibly-embodied computer software or firmware, in computer hardware, including the structures disclosed in this specification and their structural equivalents, or in combinations of one or more of them. Embodiments of the subject matter described in this specification can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions encoded on a tangible nonvolatile program carrier for execution by, or to control the operation of, data processing apparatus. Alternatively or in addition, the program instructions can be encoded on an artificially generated propagated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal that is generated to encode information for transmission to suitable receiver apparatus for execution by a data processing apparatus. The computer storage medium can be a machine-readable storage device, a machine-readable storage substrate, a random or serial access memory device, or a combination of one or more of them.

The term "system" may encompass all kinds of apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. A processing system may include special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit). A processing system may include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them.

A computer program (which may also be referred to or described as a program, software, a software application, a module, a software module, a script, or code) can be written in any form of programming language, including compiled or interpreted languages, or declarative or procedural languages, and it can be deployed in any form, including as a standalone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A computer program may, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

The processes and logic flows described in this specification can be performed by one or more programmable computers executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit).

Computers suitable for the execution of a computer program can include, by way of example, general or special purpose microprocessors or both, or any other kind of central processing unit. Generally, a central processing unit will receive instructions and data from a read-only memory or a random access memory or both. A computer generally includes a central processing unit for performing or executing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Moreover, a computer can be embedded in another device, e.g., a mobile telephone, a personal digital assistant (PDA), a mobile audio or video player, a game console, a

Global Positioning System (GPS) receiver, or a portable storage device (e.g., a universal serial bus (USB) flash drive), to name just a few.

Computer readable media suitable for storing computer program instructions and data include all forms of nonvolatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD-ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

To provide for interaction with a user, embodiments of the subject matter described in this specification can be implemented on a computer having a display device, e.g., a CRT (cathode ray tube) or LCD (liquid crystal display) monitor, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse or a trackball, by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with a user by sending documents to and receiving documents from a device that is used by the user; for example, by sending web pages to a web browser on a user's user device in response to requests received from the web browser.

Embodiments of the subject matter described in this specification can be implemented in a computing system that includes a back end component, e.g., as a data server, or that includes a middleware component, e.g., an application server, or that includes a front end component, e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the subject matter described in this specification, or any combination of one or more such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data communication, e.g., a communication network. Examples of communication networks include a local area network ("LAN") and a wide area network ("WAN"), e.g., the Internet.

The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

While this specification contains many specific implementation details, these should not be construed as limitations on the scope of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments. Certain features that are described in

this specification in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

Particular embodiments of the subject matter have been described. Other embodiments are within the scope of the following claims. For example, the actions recited in the claims can be performed in a different order and still achieve desirable results. As one example, the processes depicted in the accompanying figures do not necessarily require the particular order shown, or sequential order, to achieve desirable results. In certain implementations, multitasking and parallel processing may be advantageous. Other steps or stages may be provided, or steps or stages may be eliminated, from the described processes. Accordingly, other implementations are within the scope of the following claims.

Terminology

The phraseology and terminology used herein is for the purpose of description and should not be regarded as limiting.

The term “approximately”, the phrase “approximately equal to”, and other similar phrases, as used in the specification and the claims (e.g., “X has a value of approximately Y” or “X is approximately equal to Y”), should be understood to mean that one value (X) is within a predetermined range of another value (Y). The predetermined range may be plus or minus 20%, 10%, 5%, 3%, 1%, 0.1%, or less than 0.1%, unless otherwise indicated.

The indefinite articles “a” and “an,” as used in the specification and in the claims, unless clearly indicated to the contrary, should be understood to mean “at least one.” The phrase “and/or,” as used in the specification and in the claims, should be understood to mean “either or

both" of the elements so conjoined, i.e., elements that are conjunctively present in some cases and disjunctively present in other cases. Multiple elements listed with "and/or" should be construed in the same fashion, i.e., "one or more" of the elements so conjoined. Other elements may optionally be present other than the elements specifically identified by the "and/or" clause, whether related or unrelated to those elements specifically identified. Thus, as a non-limiting example, a reference to "A and/or B", when used in conjunction with open-ended language such as "comprising" can refer, in one embodiment, to A only (optionally including elements other than B); in another embodiment, to B only (optionally including elements other than A); in yet another embodiment, to both A and B (optionally including other elements); etc.

As used in the specification and in the claims, "or" should be understood to have the same meaning as "and/or" as defined above. For example, when separating items in a list, "or" or "and/or" shall be interpreted as being inclusive, i.e., the inclusion of at least one, but also including more than one, of a number or list of elements, and, optionally, additional unlisted items. Only terms clearly indicated to the contrary, such as "only one of or "exactly one of," or, when used in the claims, "consisting of," will refer to the inclusion of exactly one element of a number or list of elements. In general, the term "or" as used shall only be interpreted as indicating exclusive alternatives (i.e. "one or the other but not both") when preceded by terms of exclusivity, such as "either," "one of," "only one of," or "exactly one of." "Consisting essentially of," when used in the claims, shall have its ordinary meaning as used in the field of patent law.

As used in the specification and in the claims, the phrase "at least one," in reference to a list of one or more elements, should be understood to mean at least one element selected from any one or more of the elements in the list of elements, but not necessarily including at least one of each and every element specifically listed within the list of elements and not excluding any combinations of elements in the list of elements. This definition also allows that elements may optionally be present other than the elements specifically identified within the list of elements to which the phrase "at least one" refers, whether related or unrelated to those elements specifically identified. Thus, as a non-limiting example, "at least one of A and B" (or, equivalently, "at least one of A or B," or, equivalently "at least one of A and/or B") can refer, in one embodiment, to at least one, optionally including more than one, A, with no B present (and optionally including elements other than B); in another embodiment, to at least one, optionally including more than one, B, with no A present (and optionally including elements other than A); in yet another embodiment, to at least one, optionally including more than one, A, and at least one, optionally including more than one, B (and optionally including other elements); etc.

The use of "including," "comprising," "having," "containing," "involving," and variations thereof, is meant to encompass the items listed thereafter and additional items.

Use of ordinal terms such as "first," "second," "third," etc., in the claims to modify a claim element does not by itself connote any priority, precedence, or order of one claim element over another or the temporal order in which acts of a method are performed. Ordinal terms are used merely as labels to distinguish one claim element having a certain name from another element having a same name (but for use of the ordinal term), to distinguish the claim elements.

Equivalents

Having thus described several aspects of at least one embodiment of this invention, it is to be appreciated that various alterations, modifications, and improvements will readily occur to those skilled in the art. Such alterations, modifications, and improvements are intended to be part of this disclosure, and are intended to be within the spirit and scope of the invention. Accordingly, the foregoing description and drawings are by way of example only.

What is claimed is:

CLAIMS

1. A computer-implemented method for ransomware detection, comprising:

based on first events initiated by a computational entity in a computer system, detecting one or more instances of the computational entity engaging in one or more first behaviors associated with encrypting a file, encrypting a storage device, and/or disabling a backup of at least a portion of the computer system;

assigning at least one potential ransomware category to the computational entity based on the detected instances of the first behaviors; and

after assigning the potential ransomware category to the computational entity,

initiating one or more protection actions to protect one or more storage resources against one or more modification operations, and

based on second events initiated by the computational entity, detecting one or more instances of the computational entity engaging in one or more second behaviors corresponding to the potential ransomware category.

2. The method of claim 1, wherein:

the first behaviors associated with encrypting a file include enumerating directories of a file system and/or modifying a private file,

the first behaviors associated with encrypting a storage device include modifying the storage device, and

the first behaviors associated with disabling a backup of at least a portion of the computer system include disabling and/or deleting a backup file, wherein the backup file stores backup data for a plurality of files and/or settings of the computer system.

3. The method of claim 2, wherein detecting the instance of the computational entity enumerating directories of a storage device comprises:

detecting an enumeration operation initiated by the computational entity on a specified first directory, wherein the specified first directory is (1) a top-level directory of the storage device, (2) a user directory in a file system of the storage device, or (3) an immediate subdirectory of a user directory in the file system of the storage device.

4. The method of claim 3, wherein a second directory is nested within the first directory, and wherein detecting the instance of the computational entity enumerating directories of the storage device further comprises:

detecting an enumeration operation initiated by the computational entity on the second directory; and

detecting a file modification operation initiated by the computational entity, wherein the file modification operation targets a particular file in the second directory.

5. The method of claim 4, wherein detecting the instance of the computational entity enumerating directories of the storage device further comprises:

for each intermediate directory in a nesting hierarchy between the first directory and the second directory, detecting an enumeration operation initiated by the computational entity on the intermediate directory.

6. The method of claim 3, wherein the top-level directory of the storage device is a mount point of the storage device.

7. The method of claim 2, wherein detecting the instance of the computational entity modifying a storage device comprises detecting the computational entity (1) bypassing an operating system of the computer system when accessing the storage device, (2) writing to one or more specified regions of the storage device, and/or (3) modifying one or more file system structures of the storage device.

8. The method of claim 7, wherein the specified regions of the storage device include a master boot record (MBR) sector of the storage device and/or a partition table of the storage device.

9. The method of claim 8, wherein the file system structures of the storage device include a file system structure pointer and/or an inode.

10. The method of claim 2, wherein detecting the instance of the computational entity modifying a private file comprises: detecting a file modification operation initiated by the computational entity, wherein the file modification operation targets a particular file that is private to a different computational entity.

11. The method of claim 1, wherein assigning the potential ransomware category to the computational entity based on the detected instances of the first behaviors comprises:

assigning a first category to the computational entity if the detected instances include an instance of the first behavior associated with encrypting a file;

assigning a second category to the computational entity if the detected instances include an instance of the first behavior associated with encrypting a storage device; and

assigning a third category to the computational entity if the detected instances include an instance of the first behavior associated with disabling a backup of at least a portion of the computer system.

12. The method of claim 11, wherein:

the potential ransomware category assigned to the computational entity is the first category,

the modification operations comprise one or more file modification operations targeting one or more files, and

the protection actions comprise intercepting the file modification operations and creating backup copies of the files before permitting the file modification operations to modify the files.

13. The method of claim 12, wherein permitting the file modification operations to modify the files comprises permitting the file modification operations to encrypt the files.

14. The method of claim 11, wherein:

the potential ransomware category assigned to the computational entity is the second category,

the modification operations comprise one or more storage device modification operations targeting one or more storage devices, and

the protection actions comprise preventing the storage device modification operations from executing.

15. The method of claim 14, wherein preventing the storage device modification operations from executing comprises preventing the storage device modification operations from (1) bypassing an operating system of the computer system when accessing a storage device, (2) writing to one or more specified regions of the storage device, and/or (3) modifying one or more file system structures of the storage device.

16. The method of claim 15, wherein the specified regions of the storage device include a master boot record (MBR) sector of the storage device and/or a partition table of the storage device.

17. The method of claim 15, wherein the file system structures of the storage device include a file system structure pointer and/or an inode.

18. The method of claim 11, wherein:

the potential ransomware category assigned to the computational entity is the third category,

the modification operations comprise one or more file modification operations targeting one or more backup files, and

the protection actions comprise intercepting the file modification operations and creating backup copies of the backup files before permitting the file modification operations to modify the backup files.

19. The method of claim 11, further comprising:

assigning at least one potential ransomware subcategory to the computational entity based on the detected instances of the second behaviors.

20. The method of claim 19, wherein:

the potential ransomware category assigned to the computational entity is the first category, and

the second behaviors corresponding to the first category include a second behavior associated with encrypting files in-place, a second behavior associated with encrypting files out-of-place, a second behavior associated with encrypting files using a single computational unit, and/or a second behavior associated with encrypting files using a plurality of related computational units.

21. The method of claim 20, wherein the second behavior associated with encrypting files in-place includes modifying a file not created by the computational entity.

22. The method of claim 20, wherein the second behavior associated with encrypting files out-of-place includes creating a copy of a file not created by the computational entity, modifying the copy, and deleting the file not created by the computational entity.

23. The method of claim 19, wherein:

the potential ransomware category assigned to the computational entity is the second category, and

the second behaviors corresponding to the second category include (1) attempting to modify a master boot record (MBR) of the storage device wherein the storage device is a bootable storage device, and/or (2) attempting to modify a file system structure of the storage device wherein the storage device is a non-bootable storage device.

24. The method of claim 19, further comprising:

assigning a score to the computational entity based on the assigned potential ransomware category, the assigned potential ransomware subcategory, and/or the detected instances of the second behaviors.

25. The method of claim 24, wherein assigning the score to the computational entity comprises setting the score to an initial value based on the assigned potential ransomware category.

26. The method of claim 24, wherein:

the potential ransomware category assigned to the computational entity is the first category,

detecting the computational entity engaging in the instance of the second behavior comprises detecting an increase in a number of files modified by the computational entity, an increase in a number of distinct file type classes associated with files modified by the computational entity, and/or an increase in a rate of file modification by the computational entity, and

assigning the score to the computational entity comprises increasing the score based on the increased number of modified files, the increased number of distinct file type classes, and/or the increased rate of file modification.

27. The method of claim 24, wherein:

the potential ransomware category assigned to the computational entity is the first category,

the second behavior includes the computational entity modifying a name of a file, modifying a header of a file, and/or modifying a file such that a file header and a file extension of the modified file are inconsistent with each other, and

assigning the score to the computational entity comprises increasing the score based on the detected instances of the second behaviors.

28. The method of claim 24, wherein:

the potential ransomware category assigned to the computational entity is the first category and the potential ransomware subcategory assigned to the computational entity indicates that the computational entity performs in-place encryption,

the second behavior includes modifying a particular amount of data in a plurality of files, modifying data at a particular offset in a plurality of files, and/or writing particular data to a plurality of files, and

assigning the score to the computational entity comprises increasing the score based on the detected instances of the second behaviors.

29. The method of claim 24, wherein:

the potential ransomware category assigned to the computational entity is the first category and the potential ransomware subcategory assigned to the computational entity indicates that the computational entity performs out-of-place encryption,

the second behavior includes specifying file extensions of a plurality of files such that the specified file extensions are all identical to each other or all different from each other, and

assigning the score to the computational entity comprises increasing the score based on the detected instances of the second behaviors.

30. The method of claim 24, wherein:

the second behavior includes persistently scheduling the computational entity, creating a network connection, querying the computer system for system information, exhibiting suspicious provenance, obfuscating the computational entity, and/or parallelizing the computational entity, and

assigning the score to the computational entity comprises increasing the score based on the detected instances of the second behaviors.

31. The method of claim 24, wherein:

the potential ransomware category assigned to the computational entity is the first category,

the second behavior includes modifying files of a sanctioned file type, modifying files in a temporary file storage area, and/or modifying private files of the computational entity, and

assigning the score to the computational entity comprises decreasing the score based on the detected instances of the second behaviors.

32. The method of claim 24, further comprising:

determining whether the assigned score exceeds a threshold score associated with the assigned potential ransomware subcategory, and if so, initiating a mitigation action associated with the ransomware subcategory.

33. The method of claim 32, wherein the mitigation action comprises terminating execution of the computational entity, preventing the computational entity from accessing a communication network, preventing the computational entity from modifying the storage resources, and/or replacing one or more files modified by the computational entity with respective backup files.

34. The method of claim 1, wherein the computational entity comprises a single computational unit or a plurality of related computational units, wherein each computational unit is a thread, a process, or an application.

35. The method of claim 1, wherein the modification operations are initiated by the computational entity.

36. A computer system comprising:

one or more processing devices and one or more storage devices storing instructions that are operable, when executed by the processing devices, to cause the processing devices to perform operations including:

based on first events initiated by a computational entity in the computer system, detecting one or more instances of the computational entity engaging in one or more first behaviors associated with encrypting a file, encrypting a storage device, and/or disabling a backup of at least a portion of the computer system;

assigning at least one potential ransomware category to the computational entity based on the detected instances of the first behaviors; and

after assigning the potential ransomware category to the computational entity,

initiating one or more protection actions to protect one or more storage resources against one or more modification operations, and

based on second events initiated by the computational entity, detecting one or more instances of the computational entity engaging in one or more second behaviors corresponding to the potential ransomware category.

37. A computer-implemented method for ransomware detection, comprising:

based on first events initiated by a computational entity in a computer system, detecting one or more instances of the computational entity engaging in one or more first behaviors associated with encrypting a file, encrypting a storage device, and/or disabling a backup of at least a portion of the computer system;

assigning at least one potential ransomware category to the computational entity based on the detected instances of the first behaviors; and

after assigning the potential ransomware category to the computational entity,

based on second events initiated by the computational entity, detecting one or more instances of the computational entity engaging in one or more second behaviors corresponding to the potential ransomware category,

assigning at least one potential ransomware subcategory to the computational entity based on the detected instances of the second behaviors,

assigning a score to the computational entity based on the assigned potential ransomware category, the assigned potential ransomware subcategory, and/or the detected instances of the second behaviors, and

determining whether the assigned score exceeds a threshold score associated with the assigned potential ransomware subcategory, and if so, initiating a mitigation action associated with the ransomware subcategory.

38. The method of claim 37, wherein each of the first behaviors is selected from a group comprising enumerating directories of a storage device, modifying a private file, modifying a storage device, and disabling a backup of at least a portion of the computer system.

39. The method of claim 37, further comprising:

after assigning the potential ransomware category to the computational entity, initiating one or more protection actions to protect one or more storage resources against one or more modification operations.

40. The method of claim 39, wherein the modification operations are initiated by the computational entity.

41. The method of claim 39, wherein assigning the potential ransomware category to the computational entity based on the detected instances of the first behaviors comprises:

assigning a first category to the computational entity if the detected instances include an instance of the first behavior associated with encrypting a file;

assigning a second category to the computational entity if the detected instances include an instance the first behavior associated with encrypting a storage device; and

assigning a third category to the computational entity if the detected instances include an instance of the first behavior associated with disabling a backup of at least a portion of the computer system.

42. The method of claim 41, wherein:

the potential ransomware category assigned to the computational entity is the first category,

the modification operations comprise one or more file modification operations targeting one or more files, and

the protection actions comprise intercepting the file modification operations and creating backup copies of the files before permitting the file modification operations to modify the files.

43. The method of claim 41, wherein:

the potential ransomware category assigned to the computational entity is the second category,

the modification operations comprise one or more storage device modification operations targeting one or more storage devices, and

the protection actions comprise preventing the storage device modification operations from executing.

44. The method of claim 41, wherein:

the potential ransomware category assigned to the computational entity is the third category,

the modification operations comprise one or more file modification operations targeting one or more backup files, and

the protection actions comprise intercepting the file modification operations and creating backup copies of the backup files before permitting the file modification operations to modify the backup files.

45. The method of claim 37, wherein the mitigation action comprises terminating execution of the computational entity, preventing the computational entity from accessing a communication network, preventing the computational entity from modifying one or more storage resources of the computer system, and/or replacing one or more files modified by the computational entity with respective backup files.

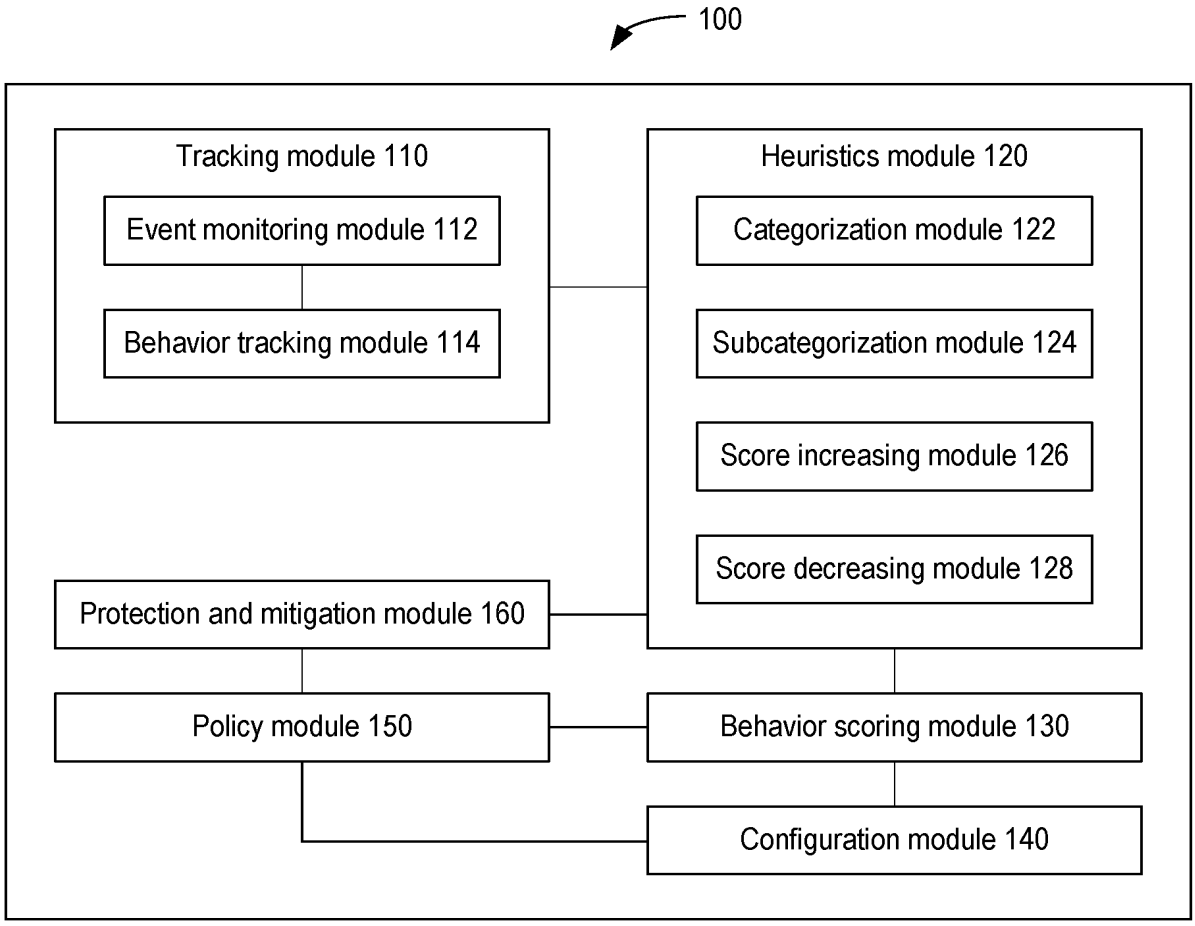

100
Tracking module 110
Event monitoring module 112
Behavior tracking module 114
Heuristics module 120
Categorization module 122
Subcategorization module 124
Score increasing module 126
Score decreasing module 128
Protection and mitigation module 160
Policy module 150
Behavior scoring module 130
Configuration module 140

FIGURE 1

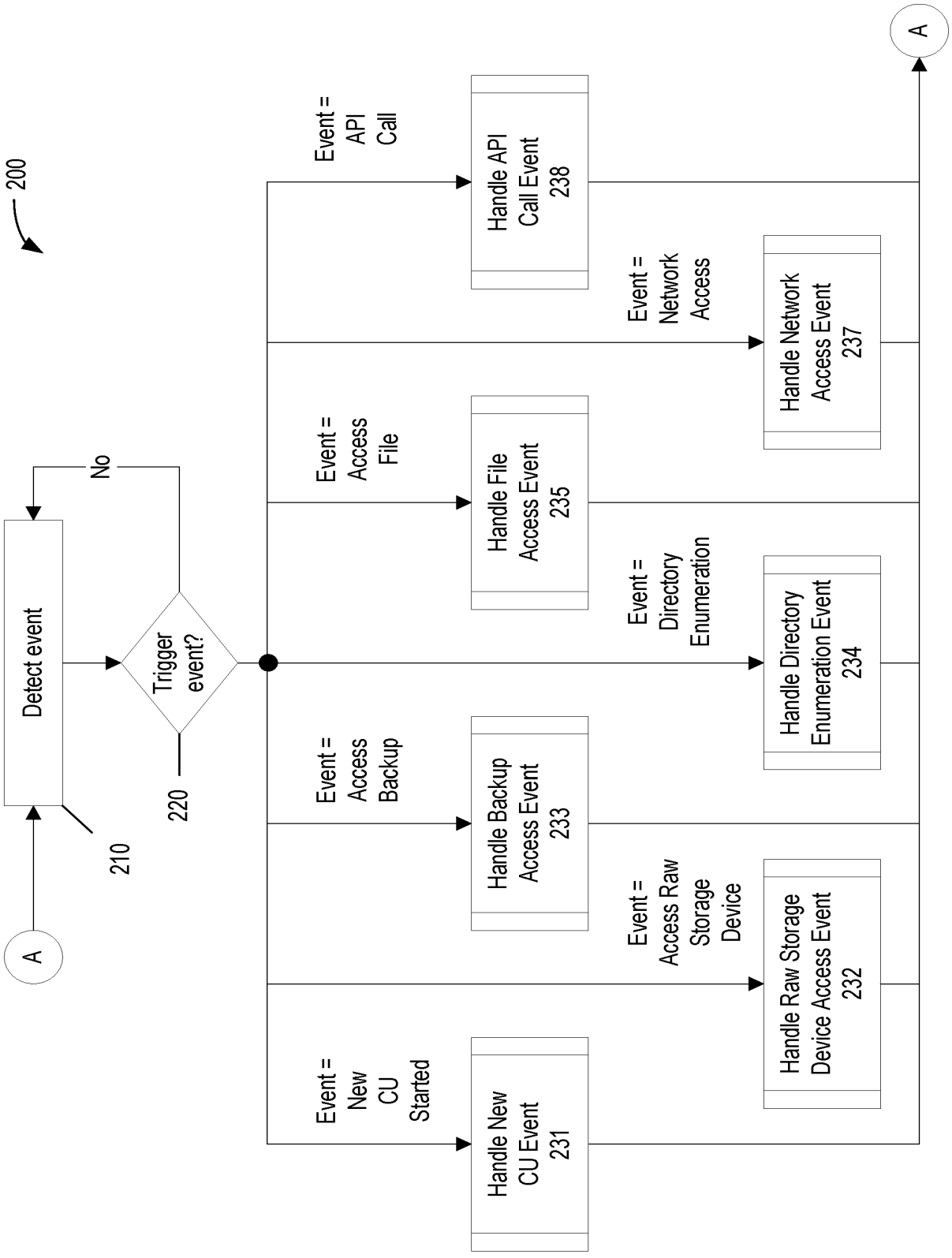

200
A
Detect event
210
Trigger event?
220
No
Event = New CU Started
Handle New CU Event 231
Event = Access Raw Storage Device
Handle Raw Storage Device Access Event 232
Event = Access Backup
Handle Backup Access Event 233
Event = Directory Enumeration
Handle Directory Enumeration Event 234
Event = Access File
Handle File Access Event 235
Event = Network Access
Handle Network Access Event 237
Event = API Call
Handle API Call Event 238
A

FIGURE 2

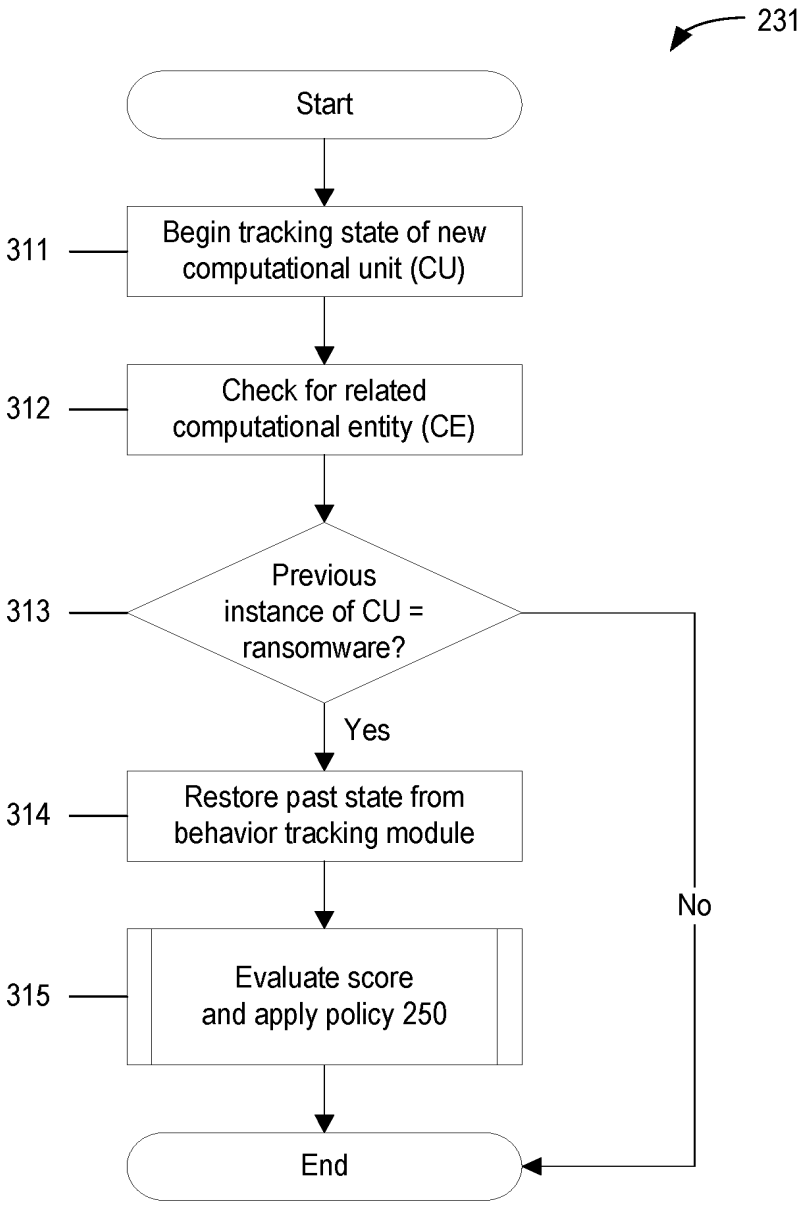

231
Start
311
Begin tracking state of new computational unit (CU)
312
Check for related computational entity (CE)
313
Previous instance of CU = ransomware?
Yes
No
314
Restore past state from behavior tracking module
315
Evaluate score and apply policy 250
End

FIGURE 3A

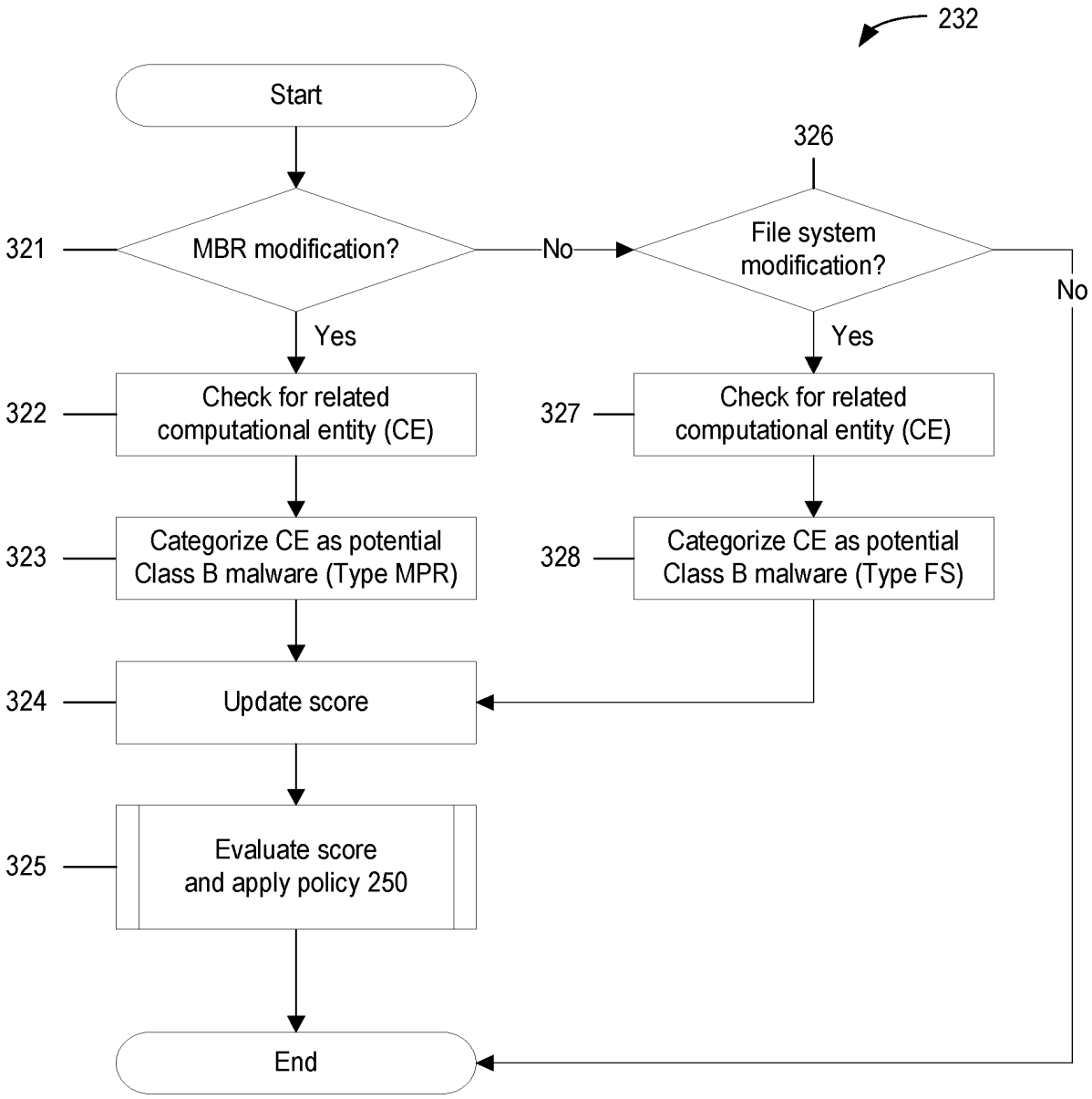

232
Start
326
321
MBR modification?
No
File system modification?
No
Yes
Yes
322
Check for related computational entity (CE)
327
Check for related computational entity (CE)
323
Categorize CE as potential Class B malware (Type MPR)
328
Categorize CE as potential Class B malware (Type FS)
324
Update score
325
Evaluate score and apply policy 250
End

FIGURE 3B

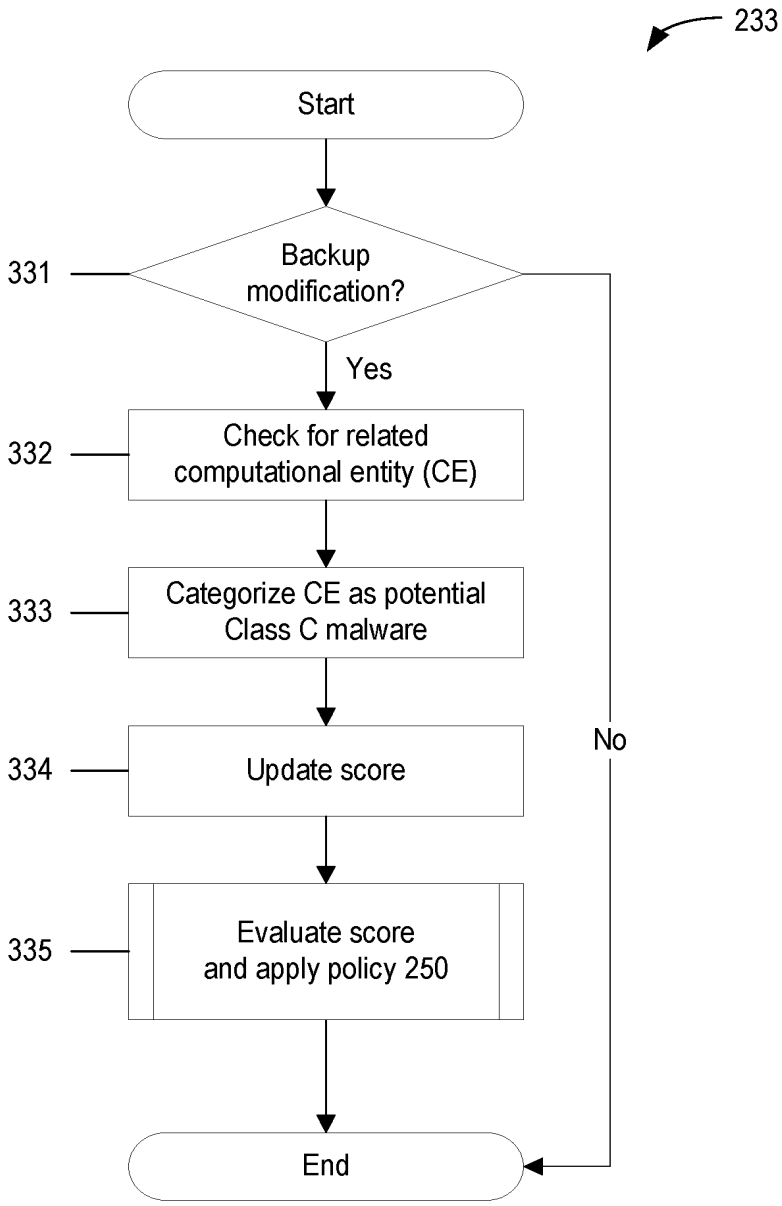

233
Start
331
Backup modification?
Yes
No
332
Check for related computational entity (CE)
333
Categorize CE as potential Class C malware
334
Update score
335
Evaluate score and apply policy 250
End

FIGURE 3C

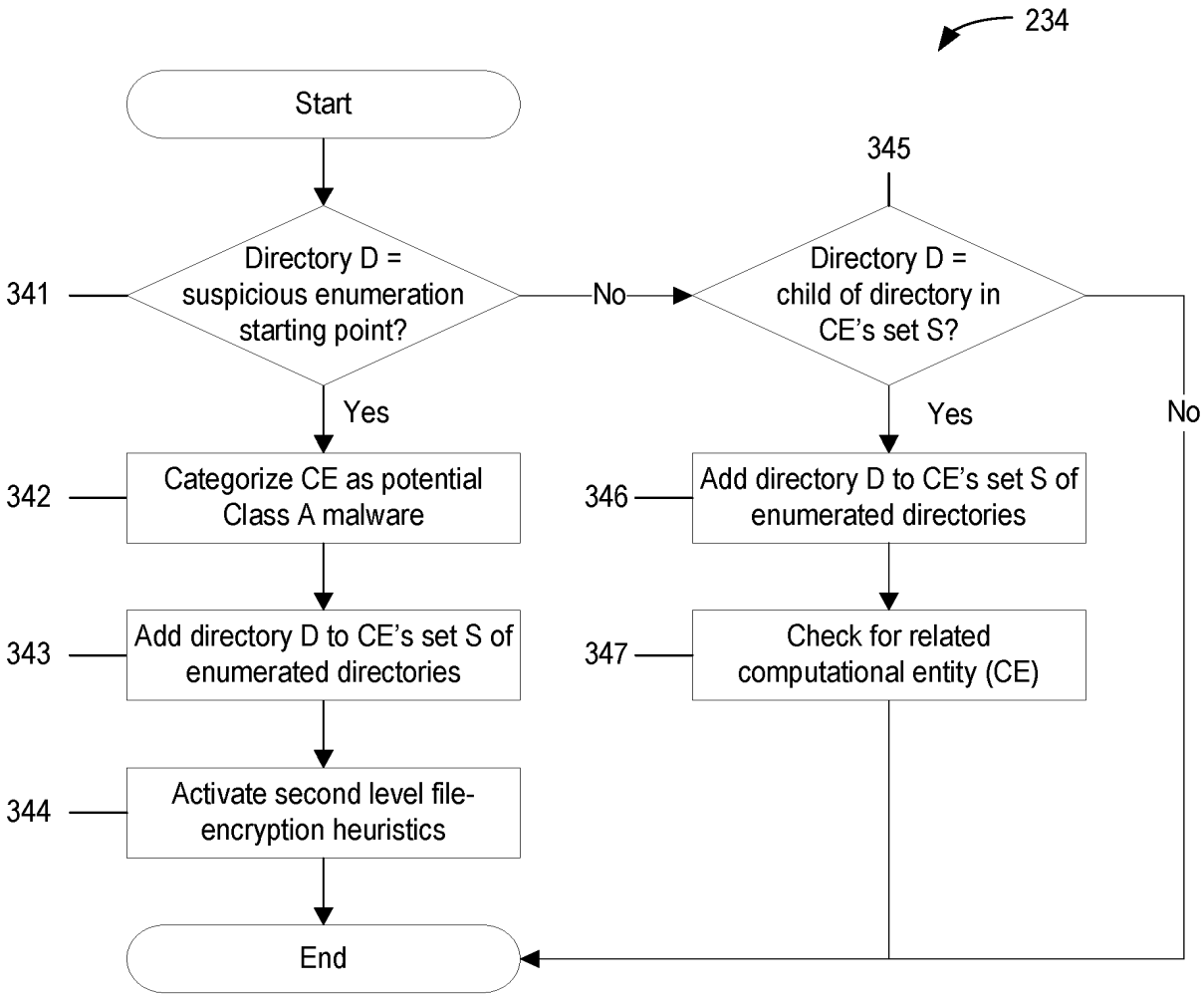

234
Start
345
Directory D = suspicious enumeration starting point?
341
No
Directory D = child of directory in CE's set S?
Yes
Yes
No
Categorize CE as potential Class A malware
342
Add directory D to CE's set S of enumerated directories
346
Add directory D to CE's set S of enumerated directories
343
Check for related computational entity (CE)
347
Activate second level file-encryption heuristics
344
End

FIGURE 3D

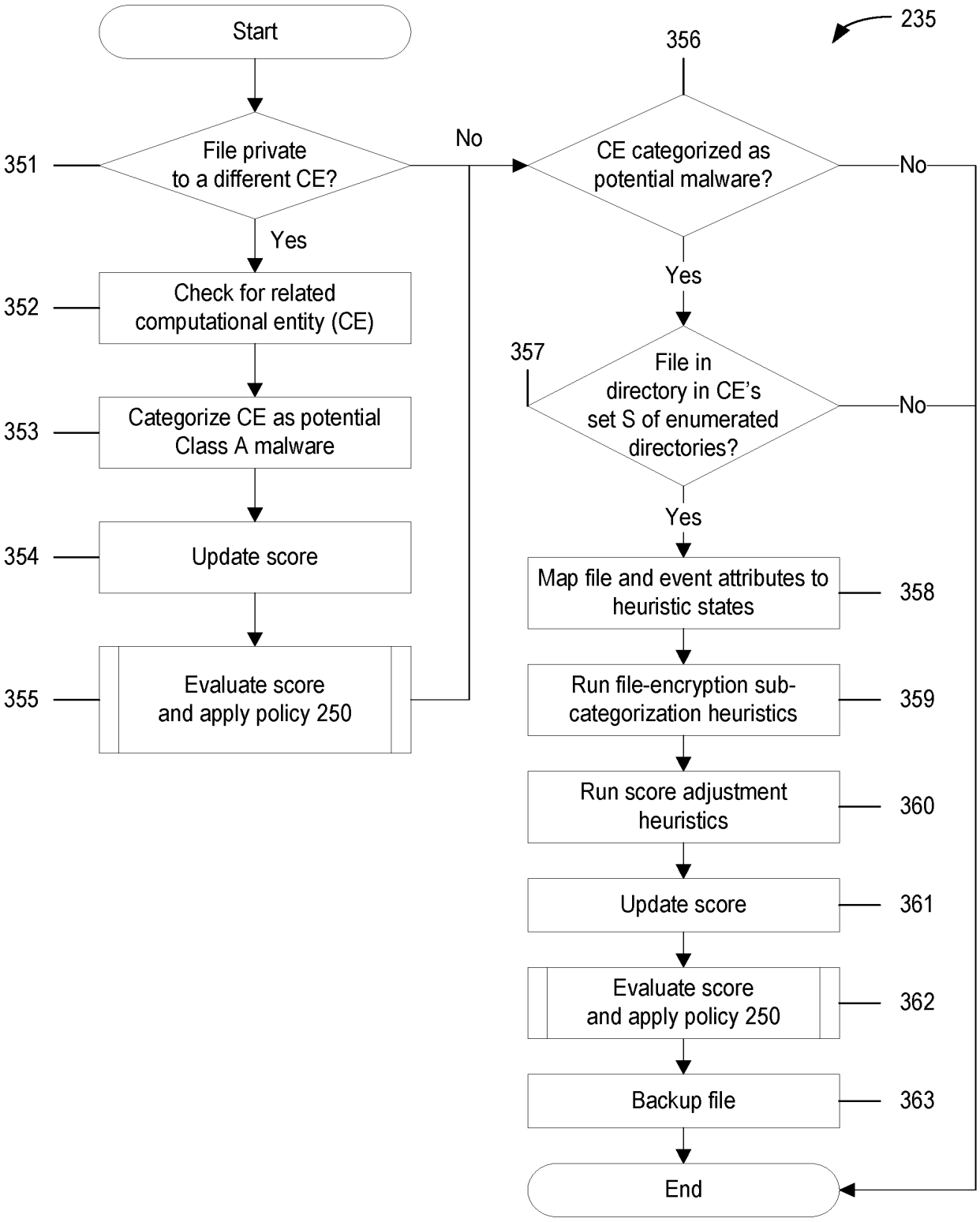

Start
File private to a different CE?
351
No
Yes
Check for related computational entity (CE)
352
Categorize CE as potential Class A malware
353
Update score
354
Evaluate score and apply policy 250
355
235
356
CE categorized as potential malware?
No
Yes
357
File in directory in CE's set S of enumerated directories?
No
Yes
Map file and event attributes to heuristic states
358
Run file-encryption sub-categorization heuristics
359
Run score adjustment heuristics
360
Update score
361
Evaluate score and apply policy 250
362
Backup file
363
End

FIGURE 3E

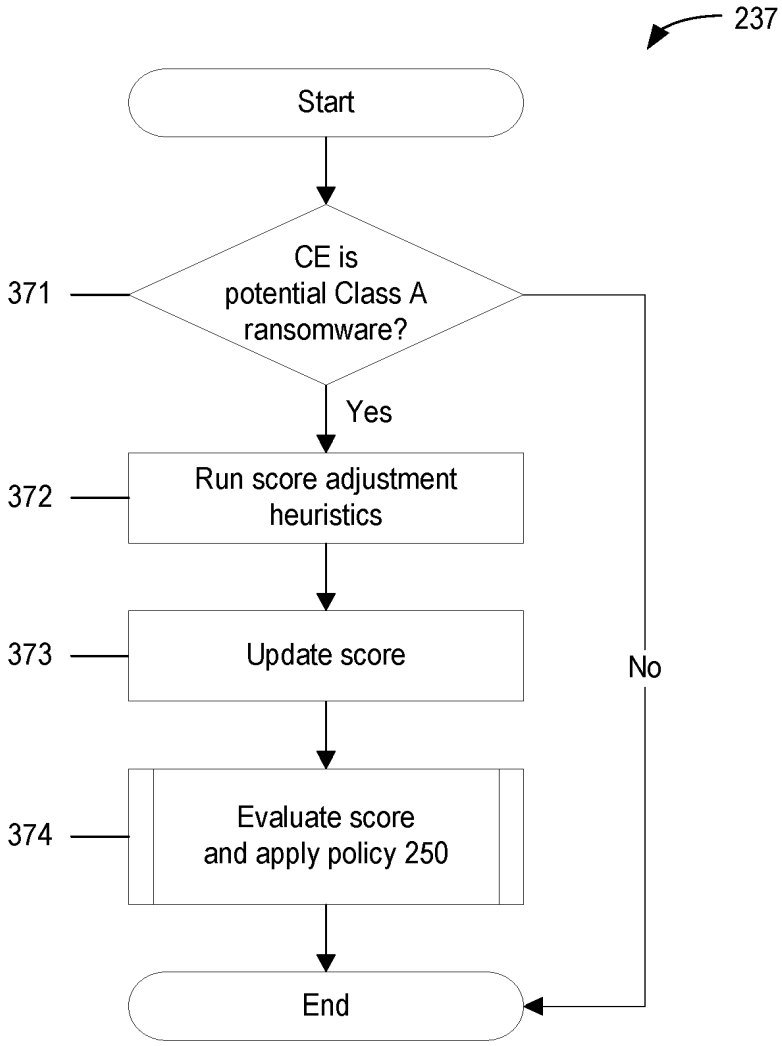

237
Start
CE is potential Class A ransomware?
371
Yes
No
Run score adjustment heuristics
372
Update score
373
Evaluate score and apply policy 250
374
End

FIGURE 3F

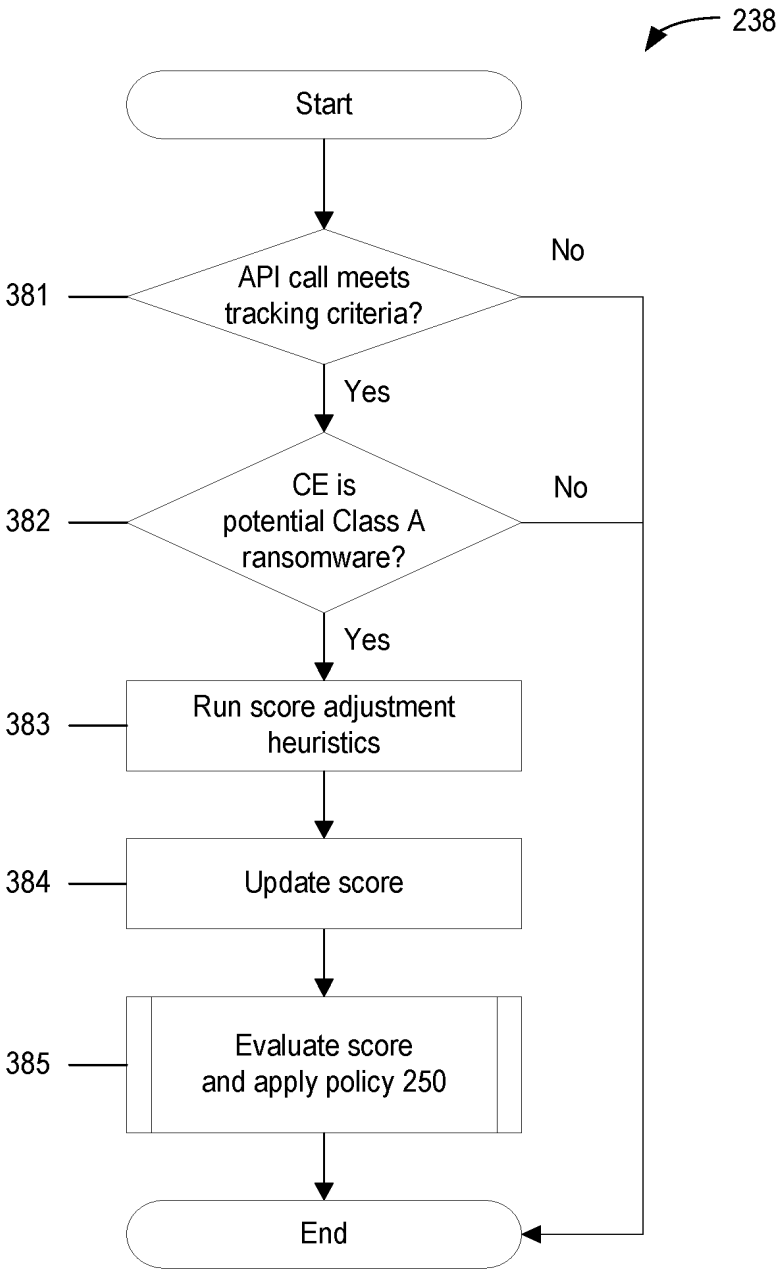

238
Start
API call meets tracking criteria?
381
No
Yes
CE is potential Class A ransomware?
382
No
Yes
Run score adjustment heuristics
383
Update score
384
Evaluate score and apply policy 250
385
End

FIGURE 3G

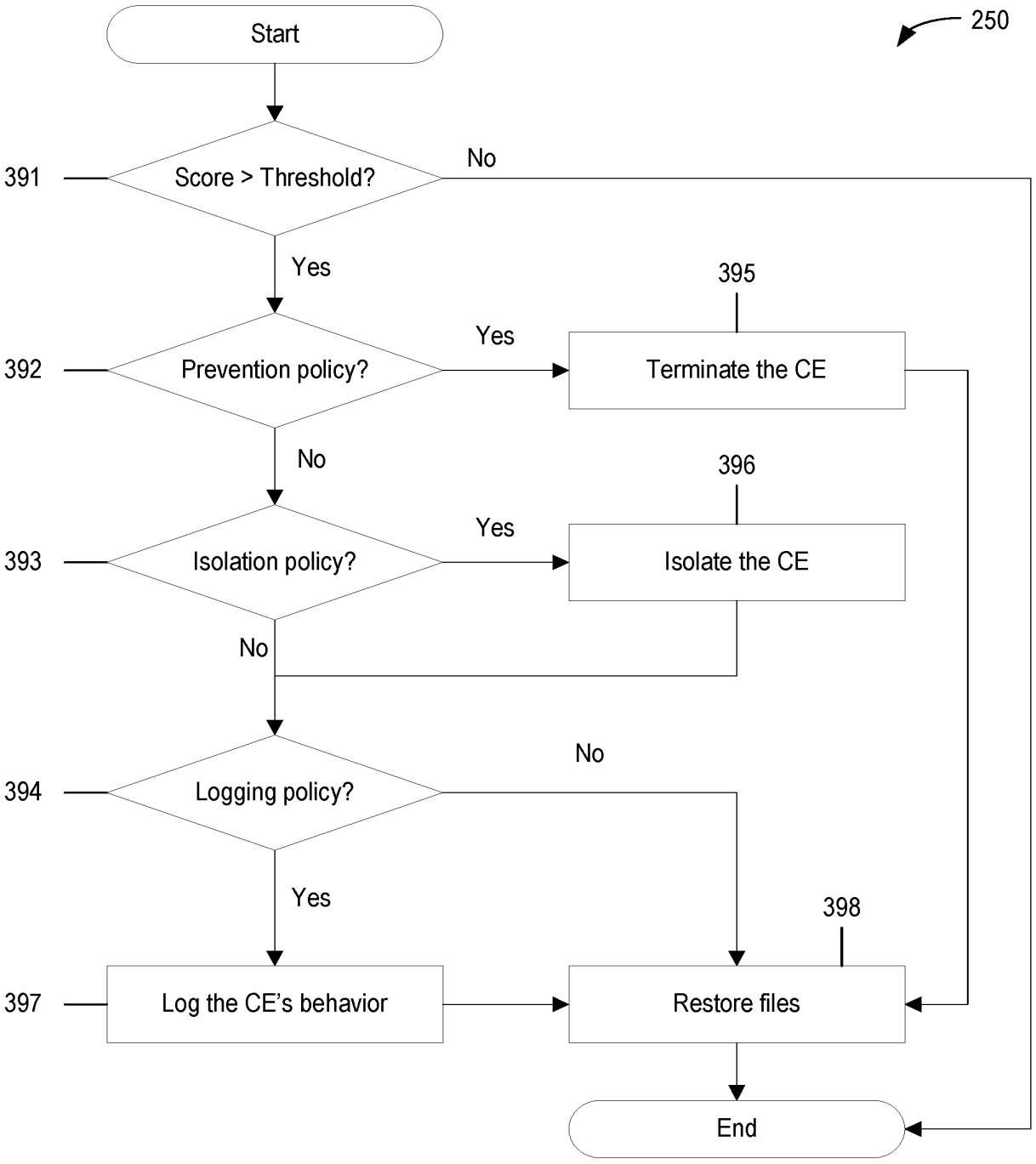

250
Start
391
Score > Threshold?
No
Yes
395
392
Prevention policy?
Yes
Terminate the CE
No
396
393
Isolation policy?
Yes
Isolate the CE
No
No
394
Logging policy?
Yes
398
397
Log the CE's behavior
Restore files
End

FIGURE 3H

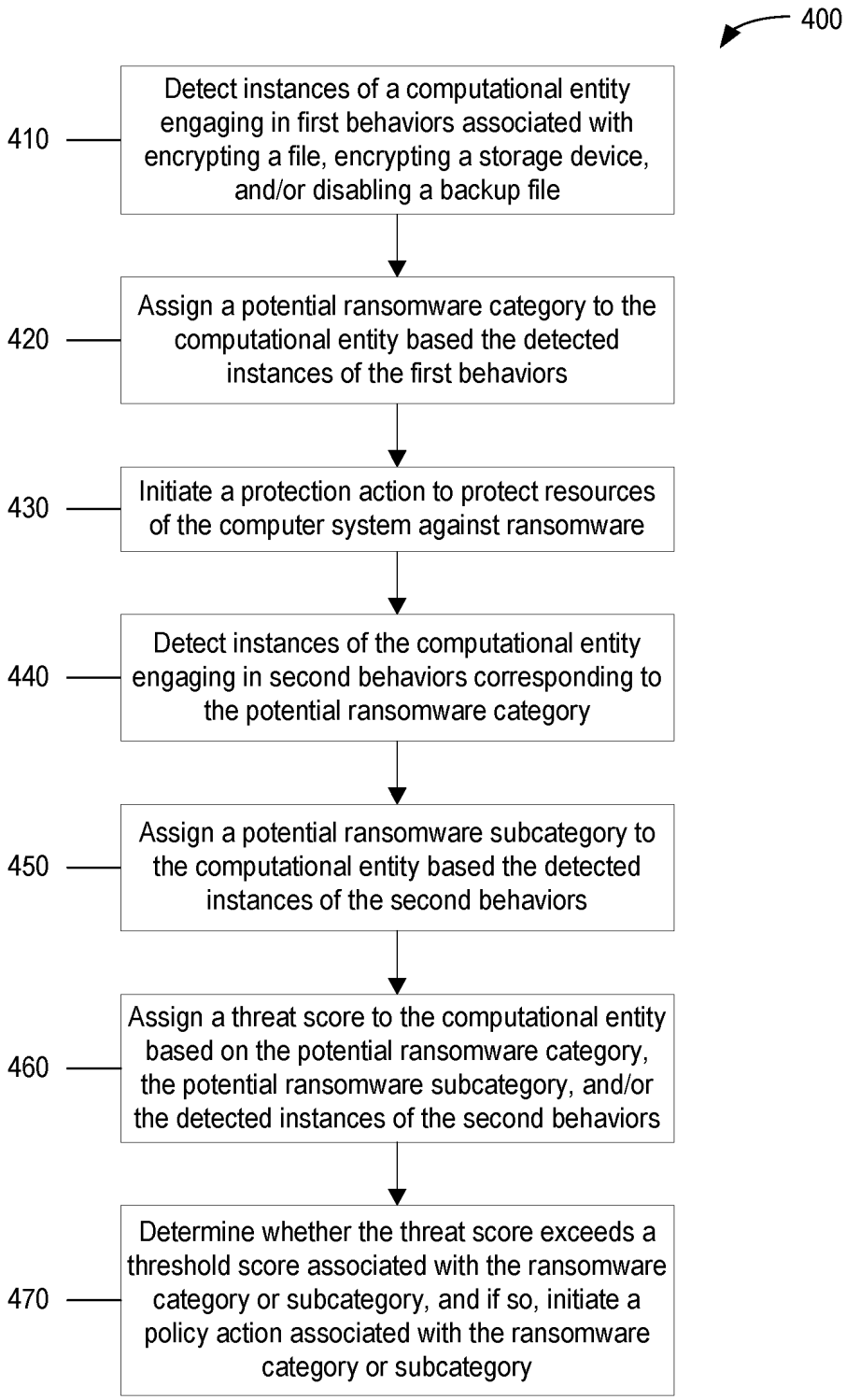

400
410
Detect instances of a computational entity engaging in first behaviors associated with encrypting a file, encrypting a storage device, and/or disabling a backup file
420
Assign a potential ransomware category to the computational entity based the detected instances of the first behaviors
430
Initiate a protection action to protect resources of the computer system against ransomware
440
Detect instances of the computational entity engaging in second behaviors corresponding to the potential ransomware category
450
Assign a potential ransomware subcategory to the computational entity based the detected instances of the second behaviors
460
Assign a threat score to the computational entity based on the potential ransomware category, the potential ransomware subcategory, and/or the detected instances of the second behaviors
470
Determine whether the threat score exceeds a threshold score associated with the ransomware category or subcategory, and if so, initiate a policy action associated with the ransomware category or subcategory

FIGURE 4

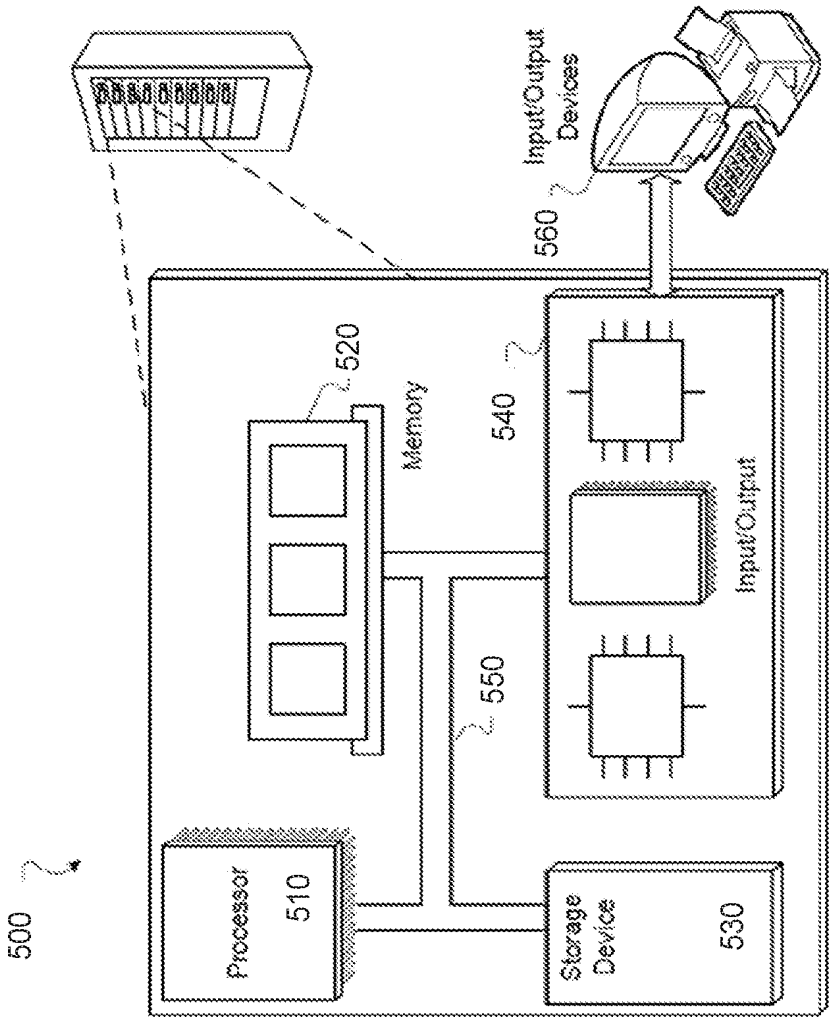

500
Processor
510
520
Memory
540
Input/Output
550
Storage Device
530
560
Input/Output Devices

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2018/050528

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F21/56 H04L29/06
ADD. G06F21/55

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2017/223031 A1 (GU LEI [US] ET AL) 3 August 2017 (2017-08-03)	1,11-13, 18,24, 25,30, 34-36, 39-42, 44,45
A	paragraph [0003] - paragraph [0006] paragraph [0038] - paragraph [0047] figures 1,3,4 claims 1-8	2-6,10, 19-22, 31-33, 37,38

-/--

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
3 December 2018	29/01/2019

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Sauzon, Guillaume

2

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2018/050528

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2017/053745 A1 (UNIV FLORIDA [US]) 30 March 2017 (2017-03-30)	1-6, 10-13, 18-22, 24-29, 31-42, 44,45
A	page 1, line 30 - page 2, line 28 page 5, line 14 - page 7, line 2 page 10, line 26 - page 13, line 27 page 16, line 10 - page 18, line 18 -----	30

2

Form PCT/ISA/210 (continuation of second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2018/050528

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2017223031 A1	03-08-2017	CN 108701188 A	23-10-2018
		EP 3411825 A1	12-12-2018
		US 2017223031 A1	03-08-2017
		WO 2017136073 A1	10-08-2017
WO 2017053745 A1	30-03-2017	NONE	

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2018/050528

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

1, 3-6, 10, 12, 13, 18, 20-22, 26-29, 31, 42, 44(completely); 2, 11, 19, 24 25, 30, 32-41, 45(partially)

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☐ No protest accompanied the payment of additional search fees.

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. claims: 1, 3-6, 10, 12, 13, 18, 20-22, 26-29, 31, 42, 44(completely); 2, 11, 19, 24, 25, 30, 32-41, 45(partially)

   How to detect ransomware encrypting files and associated prevention actions

   ---

2. claims: 7-9, 14-17, 23, 43(completely); 2, 11, 19, 24, 25, 30, 32-41, 45(partially)

   How to detect ransomware modifying a storage device and associated prevention actions

   ---

PATENT
Attorney Docket No.15635.0005-00000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Avi SHUA	)	Group Art Unit: 2431
	)	
Application No.:17/657,972	)	Examiner: Sarah Su
	)	
Filed: April 5, 2022	)	
	)	
For: **Systems and Methods for Managing Cyber Vulnerabilities**	)	Confirmation No.: 2547
	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Commissioner:

AMENDMENT

In reply to the Non-Final Office Action mailed on June 21, 2022, Applicant responds the above-identified application as follows:

Amendments to the Drawings begin on page 2 of this paper

Amendments to the Claims are reflected in the listing of claims in this paper.

Remarks follow the amendment sections of this paper.

Attachments to this amendment include:

- Replacement Drawings Sheet (Fig. 4)
- Substitute Specification (Marked-Up)
- Substitute Specification (Clean Copy)

AMENDMENTS TO THE DRAWINGS:

Please replace existing FIG. 4 with FIG. 4 provided in the attached Replacement Sheet.

Applicant amends figure 4 to correct for a mislabeled item.

Attachments: 1 New Sheet showing Figure 4.

.

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1-20. (canceled)

21. (Currently Amended) A cyber security system for a cloud environment, the system comprising:

at least one processor configured to:

using a cloud provider API, access a block storage volume of a workload maintained in a cloud storage environment;

identify an installed software application in the accessed block storage volume;

analyze the identified installed software application to determine an associated software version;

access a data structure of known software vulnerabilities for a plurality of versions of software applications;

perform a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and

perform at least one of:

query the cloud provider API to determine network accessibility information related to the workload;

identify at least one port on which the vulnerable application is accessible; [[and]] or

use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

22. (Previously Presented) The system of claim 21, wherein the at least one processor is further configured to implement a remedial action in response to the identified one or more vulnerabilities.

23. (Previously Presented) The system of claim 22, wherein the remedial ~~measure~~ action includes transmitting an alert to a device associated with an administrator.

24. (Previously Presented) The system of claim 21, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises examining data sources associated with the workload; and

Determining the network accessibility information based on the examined data sources.

25. (Previously Presented) The system of claim 21, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external data source, cloud provider information, or at least one network capture log.

26. (Previously Presented) The system of claim 21, wherein to identify the installed software application, the at least one processor is configured to:

Extract data from at least one of operating system packages, libraries, or program language libraries; and

Identify the installed software application based on the extracted data.

27. (Previously Presented) The system of claim 21, wherein the at least one processor is further configured to identify a version of the installed software application.

28. (Previously Presented) The system of claim 21, wherein the identified installed software application includes one or more scripts.

29. (Previously Presented) The system of claim 21, wherein the data structure includes aggregated vulnerability data.

30. (Previously Presented) The system of claim 29, wherein the aggregated vulnerability data includes data from one or more third-party vendors.

31. (Previously Presented) The system of claim 2~~8~~9, wherein the aggregated vulnerability data includes data collected by a scanner.

32. (Previously Presented) The system of claim 2~~8~~9, wherein the aggregated vulnerability data includes at least one of an advisory, an exploit, a security announcement, or a known bug.

33. (Previously Presented) A method, comprising:

using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment;

identifying an installed software application in the accessed block storage volume;

analyzing the identified installed software application to determine an associated software version;

accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;

performing a lookup of the identified installed software version in the data structure to identify known vulnerabilities; and

perform at least one of the following:

query the cloud provider API to determine network accessibility information related to the workload;

identify at least one port on which the vulnerable application is accessible; or

use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

34. (Previously Presented) The method of claim 33, further comprising implementing a remedial action in response to the identified one or more vulnerabilities.

35. (Currently Amended) The method of claim 34, wherein the remedial ~~measure~~ action includes transmitting an alert to a device associated with an administrator.

36. (Previously Presented) The method of claim 33, wherein querying the cloud provider API to determine the network accessibility information related to the workload further comprises:

examining data sources associated with the workload; and

determining the network accessibility information based on the examined data sources.

37. (Previously Presented) The method of claim 33, wherein querying the cloud provider API to determine network accessibility information related to the workload further comprises: wherein the network accessibility information includes at least one of: data from an external source, cloud provider information, or at least one network capture log.

38. (Previously Presented) The method of claim 33, wherein identify the stalled software application comprises:

extracting data from at least one of OS packages, libraries, or program language libraries; and

identifying the installed software application based on the extracted data.

39. (Previously Presented) The method of claim 38, wherein the at least one processor is further configured to identify a version of the installed software application.

40. (Previously Presented) A non-transitory computer-readable medium storing instructions that, when executed by at least one processor, are configured to cause the at least one processor to perform operations comprising:

using a cloud provider API, accessing a block storage volume of a workload maintained in a cloud storage environment;

identifying an installed software application in the accessed block storage volume;

analyzing the identified installed software application to determine an associated software version;

accessing a data structure of known software vulnerabilities for a plurality of versions of software applications;

perform at least one of the following:

query the cloud provider API to determine network accessibility information related to the workload;

identify at least one port on which the vulnerable application is accessible; ~~and~~ or

use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.

41-159 (Cancelled)

REMARKS

The present response is intended to be fully responsive to all points of objection and/or rejection raised by the instant Office Action mailed June 21, 2022, and is believed to place the application in condition for allowance.

In the Office Action, the Examiner:

a) Objected to the Specification for containing an embedded hyperlink and/or other form of browser-executable code.

b) Objected to claims 23, 31, 32, and 35 are objected to because of alleged informalities.

c) Objected to the Drawings as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description: 403B, 405B, 405C, 409B, 409C, 411B, 411C (Figure 4); 1305 (Figure 13).

d) Rejected claims 21, 22, 26, 27, 29, 30, 32-34, and 38-40 as being anticipated under 35 U.S.C. § 102(a)(1) by U.S. Patent Pub. No. 2020/0389483 A1 to Dye et al. ("Dye").

e) Rejected claims 23, 31, and 35 as being unpatentable under 35 U.S.C. § 103 over Dye and further in view of U.S. Patent Pub. No. 2018/0063182 A1 to Jones et al. ("Jones").

f) Rejected claims 24, 25, 36, and 37 as being unpatentable under 35 U.S.C. § 103 over Dye and further in view of U.S. Patent No. 9,356,942 to Joffe ("Joffe").

g) Rejected claim 28 as being unpatentable under 35 U.S.C. § 103 over Dye and further in view of U.S. Patent No. 9,413,774 to Liu et al. ("Liu").

Applicant respectfully disagrees in view of the amendments above and the reasons discussed below. Favorable reconsideration and allowance of the application is respectfully requested.

OBJECTIONS

The Office objected to the Specification for containing an embedded hyperlink and/or other form of browser-executable code. A substitute specification is submitted to overcome this objection by removing the hyperlink.

The Office further objected to claims 23, 31, 32, and 35 for various informalities. Amendment to these claims has been addressed above.

Finally, the Office objected to the drawings as failing to address certain elements of the drawings in the specification. A substitute specification and one revised drawing is submitted to overcome this objection.

35 U.S.C. § 102 REJECTIONS

Claims 21 (and dependent claims 22, 26, 27, 29, 30, and 32) and claims 33 (and dependent claims 34 and 38-39) and 40 are rejected under 35 U.S.C. § 102 as anticipated by US 20200389483 to Dye. The Applicant respectfully traverses this rejection in view of the remarks that follow.

Claim 21, as amended, recites, *inter alia*:

> "…pe*rform at least one of: (i)query the cloud provider API to determine network accessibility information related to the workload; (ii) identify at least one port on which the vulnerable application is accessible; or (iii) use network accessibility information and at least one port to identify one or more vulnerabilities susceptible to attack from outside the workload.*"

Applicant respectfully maintains that nowhere does Dye teach at least these elements of the claims. The Examiner asserts that these elements are allegedly disclosed by Dye at [0030, lines 7-16] and [0062, lines 2-13]. Nowhere in this disclosure does Dye mention, *inter alia*, any of these elements.

With respect to "query[ing] [a] cloud provider API to determine network availability." Dye fails to discuss any "cloud provider API" nor has the Examiner asserted correspondence between another element and the claimed cloud provider API.

With respect to an identification of "at least one port on which [a] vulnerable application is accessible" and identifying one or more vulnerabilities using "network accessibility information and at least one port," Dye fails to identify or utilize a "port" for any purpose, let alone to "identify at least one port on which [a] vulnerable application is accessible" or to "identify one or more vulnerabilities" using "at least one port." . Indeed, Dye does not disclose a port. Other aspects of Dye, such as a "file path 113," Dye at [0026]; [0030]; [0035]; [0037], do not constitute the claimed "port." Nor has the Examiner made the case that they do.

For at least the foregoing reasons, independent claims 21, 33, and 40, and dependent claims 22, 26, 27, 29, 30, 32, 34, and 38-39, as amended, is allowable over this rejection and the Applicant respectfully requests reconsideration based on the above.

35 U.S.C. § 103 REJECTIONS

Claims 23-25, 28, 31, and 35-37 are rejected under 35 U.S.C. § 103 as being unpatentable over various combinations of Dye, Jones, Joffe, and Liu. As discussed above, Dye fails to teach elements of the claims. The Examiner does not assert that any of Jones, Joffe, or Liu make up for these deficiencies, nor do they. Therefore, the rejection of claims 23-25, 28, 31, and 35-37 should be withdrawn.

CONCLUSION

In view of the foregoing amendments and remarks, Applicant respectfully requests reconsideration of this application and the timely allowance of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: August 19, 2022

By: /Christopher C. Johns/
Christopher C. Johns
Reg. No. 68,664
(202) 408-4000

Docket Number: 15635.0004-00304

SYSTEMS AND METHODS FOR SIDE SCANNING

PRIORITY

[001] This application claims the benefit of U.S. Provisional Application No. 63/180,048 filed on April 26, 2021, the contents of which are hereby incorporated by reference.

TECHNICAL FIELD

[002] Disclosed embodiments include new systems and methods for securing cloud infrastructure to help meet compliance mandates, without disrupting business operations in live environments.

BACKGROUND

[003] Modern organizations typically depend on cloud infrastructure for data storage and processing. Data storage and processing nodes, among other things, power today's electronic infrastructure. In contrast to early network-centric days, an organization may have little in the way of "on-prem" (on-premises) systems and may run its entire business through systems having shared resources. Such an infrastructure may be the entire inventory of cloud assets for an organization, including running and stopped workloads of all types: virtual machines, containers, storage objects, load balancers, IAM (Identity and Access Management) configurations, and more. Organizations are searching for effective ways to scan their entire cloud estate to look for risks stemming from vulnerabilities, misconfigurations, malware, lateral movement risk, weak and leaked passwords, and improperly secured PII.

[004] The infrastructure of these organizations are plainly different than '90s-style physical servers running on bare metal. Existing agents and scanners from the on-prem days have been simply reused in cloud environments without reconfiguration. Before the cloud, it was required to secure physical hosts. That meant spending time installing multiple security agents—one for each server. IP addresses were assigned to physical assets and they seldom changed. Even then, however, agent integration was tedious and coverage rarely reached 100% of assets. In a cloud environment, scaling up and down is a frequent occurrence—possibly thousands of times per hour across multiple clouds—and all within a CI/CD pipeline that builds an organization's infrastructure. There are containers and virtual machines to contend with, and agents carry high operational costs.

[005] This causes problems of inefficiency and overuse of resources, including excess processor cycles on regular scans. In turn, such problems and inefficiencies can lead to a higher monetary cost for hosted resources as some cloud providers charge per processing cycle.

[006] Relying on agents for security visibility in the cloud also has drawbacks. Visibility is limited to only those assets that are already known and accessible to an agent, requiring that such assets are capable of having the agent installed and maintained in the first place. The assets also must have ongoing network connectivity to the backend, something that makes the

development cycle inefficient as developers tend not to be inclined to deploy agents on virtual machines, in containers, and in serverless configurations—let alone dealing with their never-ending maintenance—thus leading to insecure implementations.

[007] An authenticated scan allows for direct host access using remote protocols such as SSH or RDP. The scanner uses a privileged account to log in and determine how secure each host is from an inside vantage point. While authenticated scans can successfully discover potential vulnerabilities, they are limited as they require a privileged account on each scanned host. Furthermore, scans use significant system resources during the test procedures and require opening ports that by themselves pose a security risk.

[008] An unauthenticated scan can only examine publicly visible information and is not able to provide detailed information about assets. It essentially acts as a friendly attacker. An unauthenticated scan can easily miss identifying some assets and vulnerabilities, making it much less effective. For example, a website at mydomain.com/email_campaign not linked from a main website will not be scanned unless the scanner is manually configured. This can leave organizations exposed to vulnerabilities in areas where the scanner cannot reach.

[009] While unauthenticated scanners act like an attacker, they often get stuck where a real attacker would not. For example, a CAPTCHA can easily prevent automatic mechanisms (such as scanners) from registering. But such techniques would not have any effect on a human attacker.

[0010] Cloud security posture managers (CSPMs) are scanning tools developed specifically for the cloud. Rather than going inside the machine, a CSPM analyzes the cloud configuration itself for weaknesses. CSPMs are used to discover, assess, and solve cloud misconfigurations but provide shallow coverage where cloud security is concerned because CSPMs have difficulty detecting critical risks such as vulnerabilities, malware, and misconfigurations within the workloads themselves. Organizations choosing to combine agent-based solutions with a CSPM end up getting flooded with separate alerts that lack context which results in alert fatigue on behalf of security analysts.

[0011] Therefore, there is a need for improved security scanning systems that solve the problem with existing systems and are more adapted for use in cloud infrastructures.

[0012] The foregoing background is for illustrative purposes and is not intended as a discussion of the scope of the prior art or an admission of prior art.

SUMMARY

[0013] This summary is provided for the convenience of the reader to present a basic understanding of a few exemplary embodiments of this disclosure, and does not wholly define the breadth of the disclosure. This summary is not an extensive overview of all contemplated embodiments and is intended to neither identify key or critical elements of all embodiments nor to delineate the scope of any or all aspects. Its sole purpose is to present some features of one

or more embodiments in a simplified form as a prelude to the more detailed description presented later. For convenience, the term "embodiments" may be used herein to refer to one or more features or combinations of features described herein. A description associated with an embodiment does not necessarily apply to all embodiments.

[0014] Embodiments of the present disclosure may deliver instant-on, workload level visibility across cloud infrastructure assets without running opcodes on the target scanned workloads, helping organizations to: (1) Detect risks such as vulnerabilities, malware, misconfigurations, lateral movement risk, and unsecured sensitive data; (2) See cloud inventory at every layer: infrastructure, operating system, applications, and data; (3) Discover and see previously missed assets.

[0015] Embodiments of the present disclosure may further use an out-of-band process to reach cloud workloads through the runtime storage layer, combining this with information gathered from cloud provider APIs, including but not limited to K8S APIs.

DRAWINGS

[0016] The drawings and figures contained herein are a part of this specification, illustrate several embodiments consistent with the disclosure and, together with the description, serve to explain principles of the disclosure.

[0017] **Fig. 1** is a schematic block diagram illustrating an exemplary embodiment of a network including computerized systems, consistent with disclosed embodiments.

[0018] **Fig. 2A** is a schematic block diagram illustrating an exemplary embodiment of a process for integration, scanning, assessment, analyzing/reporting, and change reviewing, consistent with disclosed embodiments.

[0019] **Fig. 2B** is a schematic block diagram illustrating an exemplary embodiment of a process for integration, consistent with disclosed embodiments.

[0020] **Fig. 2C** is a schematic block diagram illustrating an exemplary embodiment of a process for scanning/mapping, consistent with disclosed embodiments.

[0021] **Fig. 2D** is a schematic block diagram illustrating an exemplary embodiment of a process for assessment, consistent with disclosed embodiments.

[0022] **Fig. 3** is a block diagram of method for insight extraction, consistent with disclosed embodiments.

[0023] **Fig. 4** depicts a cybersecurity system performing a side scanning function to protect against potential vulnerabilities, consistent with disclosed embodiments.

[0024] **Fig. 5** of the disclosed embodiments describes a method of operating a cybersecurity system performing a side scanning function to protect against potential vulnerabilities, consistent with disclosed embodiments.

[0025] **Fig. 6** is a block diagram of method for risk prioritization, consistent with disclosed embodiments.

[0026] **Fig. 7** is a block diagram illustrating an exemplary process of cyber security scanning for a cloud environment, consistent with the disclosed embodiments.

[0027] **Fig. 8** is a schematic block diagram illustrating an exemplary embodiment of a system for performing visualization of forward and backward facing threats, consistent with the disclosed embodiments.

[0028] **Fig. 9** is a block diagram illustrating an exemplary process of matching keys with compute resources in a cloud environment, consistent with the disclosed embodiments.

[0029] **Fig. 10** is a block diagram of a method for deployment of ephemeral scanners, consistent with disclosed embodiments.

[0030] **Fig. 11** represents a schematic block diagram illustrating an exemplary embodiment of a method for providing a dashboard aggregating risk information, consistent with disclosed embodiments.

[0031] **Fig. 12** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of displaying aggregated cybersecurity risk information, consistent with disclosed embodiments.

[0032] **Fig. 13** is a schematic block diagram illustrating an exemplary embodiment of a visual representation of a flow path for aggregated risk information, consistent with disclosed embodiments.

DETAILED DESCRIPTION

[0033] Exemplary embodiments are described with reference to the accompanying drawings. The figures are not necessarily drawn to scale. While examples and features of disclosed principles are described herein, modifications, adaptations, and other implementations are possible without departing from the spirit and scope of the disclosed embodiments. For example, with this detailed description provides a few examples, these implementations are provided as examples only and are not restrictive of the claim concepts that follow or any of the descriptions herein. Also, the words “comprising,” “having,” “containing,” and “including,” and other similar forms are intended to be equivalent in meaning and be open ended in that an item or items following any one of these words is not meant to be an exhaustive listing of such item or items or meant to be limited to only the listed item or items. It should also be noted that as used herein and in the appended claims, the singular forms “a,” “an,” and “the” include plural references unless the context clearly dictates otherwise.

[0034] In the following description, various working examples are provided for illustrative purposes. However, it is to be understood that the present disclosure may be practiced without one or more of these details.

[0035] It is intended that one or more aspects of any mechanism may be combined with one or more aspect of any other mechanisms, and such combinations are within the scope of this disclosure.

[0036] Various embodiments are described herein with reference to a system, method, device, or computer readable medium. It is intended that the disclosure of one is a disclosure of all. For example, it is to be understood that disclosure of a computer readable medium described herein also constitutes a disclosure of methods implemented by the computer readable medium, and systems and devices for implementing those methods, via for example, at least one processor. It is to be understood that this form of disclosure is for ease of discussion only, and one or more aspects of one embodiment herein may be combined with one or more aspects of other embodiments herein, within the intended scope of this disclosure.

[0037] Embodiments described herein may refer to a non-transitory computer readable medium containing instructions that when executed by at least one processor, cause the at least one processor to perform a method. Non-transitory computer readable medium may include any medium capable of storing data in any memory in a way that may be read by any computing device with a processor to carry out methods or any other instructions stored in the memory. The non-transitory computer readable medium may be implemented as hardware, firmware, software, or any combination thereof. Moreover, the software may preferably be implemented as an application program tangibly embodied on a program storage unit or computer readable medium consisting of parts, or of certain devices and/or a combination of devices. The application program may be uploaded to, and executed by, a machine having any suitable architecture. Preferably, the machine may be implemented on a computer platform having hardware such as one or more central processing units ("CPUs"), a memory, and input/output interfaces. The computer platform may also include an operating system and microinstruction code. The various processes and functions described in this disclosure may be either part of the microinstruction code or part of the application program, or any combination thereof, which may be executed by a CPU, whether or not such a computer or processor is explicitly shown. In addition, various other peripheral units may be connected to the computer platform such as an additional data storage unit and a printing unit. Furthermore, a non-transitory computer readable medium may be any computer readable medium except for a transitory propagating signal.

[0038] Memory employed herein may include a Random Access Memory (RAM), a Read-Only Memory (ROM), a hard disk, an optical disk, a magnetic medium, a flash memory, other permanent, fixed, volatile or non-volatile memory, or any other mechanism capable of storing instructions. The memory may include one or more separate storage devices collocated or disbursed, capable of storing data structures, instructions, or any other data. The memory may further include a memory portion containing instructions for the processor to execute. The memory may also be used as a working scratch pad for the processors or as a temporary storage.

[0039] Some embodiments may involve at least one processor. A processor may be any physical device or group of devices having electric circuitry that performs a logic operation on input or inputs. For example, the at least one processor may include one or more integrated circuits (IC), including application-specific integrated circuit (ASIC), microchips, microcontrollers, microprocessors, all or part of a central processing unit (CPU), graphics processing unit (GPU), digital signal processor (DSP), field-programmable gate array (FPGA), server, virtual server, or other circuits suitable for executing instructions or performing logic operations. The instructions executed by at least one processor may, for example, be pre-loaded into a memory integrated with or embedded into the controller or may be stored in a separate memory.

[0040] In some embodiments, the at least one processor may include more than one processor. Each processor may have a similar construction, or the processors may be of differing constructions that are electrically connected or disconnected from each other. For example, the processors may be separate circuits or integrated in a single circuit. When more than one processor is used, the processors may be configured to operate independently or collaboratively. The processors may be coupled electrically, magnetically, optically, acoustically, mechanically or by other means that permit them to interact.

[0041] Consistent with the present disclosure, disclosed embodiments may involve a network. A network may constitute any type of physical or wireless computer networking arrangement used to exchange data. For example, a network may be the Internet, a private data network, a virtual private network using a public network, a Wi-Fi network, a LAN or WAN network, and/or other suitable connections that may enable information exchange among various components of the system. In some embodiments, a network may include one or more physical links used to exchange data, such as Ethernet, coaxial cables, twisted pair cables, fiber optics, or any other suitable physical medium for exchanging data. A network may also include a public switched telephone network ("PSTN") and/or a wireless cellular network. A network may be a secured network or unsecured network. In other embodiments, one or more components of the system may communicate directly through a dedicated communication network. Direct communications may use any suitable technologies, including, for example, BLUETOOTH™, BLUETOOTH LE™ (BLE), Wi-Fi, near field communications (NFC), or other suitable communication methods that provide a medium for exchanging data and/or information between separate entities.

[0042] Certain embodiments disclosed herein may also include a computing device for cloud cybersecurity, the computing device may include processing circuitry communicatively connected to a network interface and to a memory, wherein the memory contains instructions to be executed. The computing devices may be devices such as mobile devices, desktops, laptops, tablets, or any other devices capable of processing data. Such computing devices may include a display such as an LED display, augmented reality (AR), virtual reality (VR) display.

[0043] "Software" as used herein refers broadly to any type of instructions, whether referred to as software, firmware, middleware, microcode, hardware description language, or otherwise. Instructions may include code (e.g., in source code format, binary code format, executable code format, or any other suitable format of code). The instructions, when executed by the one or more processors, may cause the processing system to perform the various functions described in further detail herein.

[0044] The one or more processors may be implemented with any combination of general-purpose microprocessors, microcontrollers, digital signal processors (DSPs), field programmable gate array (FPGAs), programmable logic devices (PLDs), controllers, state machines, gated logic, discrete hardware components, dedicated hardware finite state machines, or any other suitable entities that can perform calculations or other manipulations of information.

[0045] Aspects of this disclosure may provide technical solutions to challenges associated with cloud cybersecurity. Disclosed embodiments include methods, systems, devices, and computer-readable media. For ease of discussion, a system is described below with the understanding that the disclosed details may equally apply to methods, devices, and computer-readable media.

[0046] Embodiments of the present disclosure include technology referred to as "SideScanning." In contrast to some existing systems and techniques, these embodiments may provide a distinct advantage because the technology does not necessarily require entering into each workload to inspect data. Rather, some embodiments use an out-of-band process to reach cloud workloads through a runtime storage layer, combining this with metadata gathered from API provided through a cloud service provider's system, thus providing visibility of cloud environments both at a low level and with context, without the requirement for an agent or network scanner.

[0047] **Fig. 1** is a schematic block diagram 100 illustrating an exemplary embodiment of a network including computerized systems, consistent with the disclosed embodiments. Diagram 100 includes user device 102, network 105, and cloud infrastructure 106.

[0048] Cloud infrastructure 106 includes scanning system 101, databases 103A-103D, virtual machines 107A-107D, databases 109A-109D, storage 111A-111D, keystores 113A-113D, and load balancer 115. While particular numbers and arrangements of devices, systems, and connections, are depicted in exemplary Fig. 1, in some embodiments, each of the devices, systems, or connections may be omitted, duplicated, or modified. For example, in some embodiments, databases 109A-109D may exist as only a single database; in other embodiments, cloud infrastructure 106 may exist as one or more distinct or combined infrastructures (e.g., operated by the same or different cloud services). In some embodiments, scanning system 101 and/or databases 103A-103D may be part of cloud infrastructure 106 (and

may be connected to the various other systems and devices in cloud infrastructure 106); in other embodiments, scanning system 101 and/or databases 103A-103D may be separate from cloud infrastructure 106 (e.g., connected to the systems and devices in cloud infrastructure 106 through network 105).

[0049] Scanning system 101, in some embodiments, may include one or more computer systems. Each of the one or more computer systems may include memory storing instructions and at least one CPU configured to execute those instructions to perform operations as discussed herein. In some embodiments, the instructions cause the CPU to perform scanning operations. In some embodiments, scanning system 101 may perform a scanning operation on one or more workloads (e.g., systems, devices, resources, etc.) in cloud infrastructure 106.

[0050] User device 102, in some embodiments, may include a mechanism operated by a user to control scanning system 101. For example, in some embodiments, user device 102 may be any of a personal computer, a server, a thin client, a tablet, a personal digital assistant, a smartphone, a kiosk, or any other mechanism enabling data input. User device 102 may be operated to instantiate functionality, access data, or otherwise interact with scanning system 101 via network 105, as described herein.

[0051] Databases 103A-103D include data stores for use by scanning system 101. In some embodiments, one or more of databases 103A-103D may be implemented as a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. In some embodiments, one or more of databases 103A-103D may be implemented as flat file stores, data stores, or other non-database storage systems. In some embodiments, databases 103A-103D may be implemented using one or more of ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances.

[0052] Network 105 may be implemented as one or more interconnected data networks. For example, network 105 may include one or more of any type of network (including infrastructure) that provides communications, exchanges information, and/or facilitates the exchange of information, such as the Internet, a Local Area Network, a near field communication (NFC) network, or other suitable connection(s) that enables the sending and receiving of information between the components of system 100. Network 105 may be implemented using wireless connections, wired connections, or both. In some embodiments, one or more components of system 100 can communicate through network 105. In some embodiments, one or more components of system 100 may communicate directly through one or more dedicated communication links. While particular devices and systems are shown as connected to network 105, in some embodiments, more or fewer devices and systems may be connected to network 105.

[0053] Cloud infrastructure 106 may be implemented as a set of devices and systems offered by a single cloud service provider. For example, cloud infrastructure 106 may comprise devices and systems that are part of Amazon Web Services, Microsoft Azure, Google Cloud Platform, IBM Cloud, Alibaba Cloud, or any other cloud platform provider. In some embodiments, one or more of the devices and systems in cloud infrastructure may require authentication or other identity validation for access. For example, to access virtual machine 107A, a user may be required to enter a password or provide a key. Systems (e.g., scanning system 101 or user device 102) may administer or interact with cloud infrastructure 106 using a cloud service provider's system (not pictured).

[0054] Virtual machines 107A-107D may include one or more devices and systems that implement a virtualized/emulated version of a computer. A virtual machine may be implemented as an emulated version of a computer - including an operating system, memory, storage, graphics processing - such that it can be indistinguishable from a standard (non-virtual) machine to a running program. A computer system, referred to as a "host," may operate virtual machines 107A-107D, referred to as "guests," by dividing the resources of the host between the virtual machines such that each virtual machine is isolated from one another. This means that in some embodiments, one virtual machine, and the operating system(s) and application(s) running thereon, is only able to access the resources that are allocated to that virtual machine and cannot access resources allocated to other virtual machines. For example, if a host has 32 gigabytes of random access memory (RAM), and is hosting three virtual machines, the host may segment 8 gigabytes of RAM to each virtual machine such that each virtual machine may only access data in that 8 gigabytes of RAM and not any of the other 24 gigabytes. Examples of commercial virtual machine software and services include VMWare Workstation, VMWare Server, VMWare ESXi, VirtualBox, Parallels Desktop, Parallels RAS, Amazon Machine Image, Amazon ECS, Kubernetes, Microsoft Hyper-V, and Xen.

[0055] Databases 109A-109D may include data stores for use by devices and systems in cloud infrastructure 106. In some embodiments, one or more of databases 109A-109D may be implemented as a NoSQL database, a relational database, a cloud database, a columnar database, a wide column database, a key-value database, an object-oriented database, a hierarchical database, or any other kind of database. In some embodiments, one or more of databases 109A-109D may be implemented as flat file stores, data stores, or other non-database storage systems. In some embodiments, databases 109A-109D may be implemented using one or more of ElasticCache, ElasticSearch, DocumentDb, DynamoDB, Neptune, RDS, Aurora, Redshift clusters, Kafka clusters, or EC2 instances. Databases 109A-109D may store data usable by devices or systems in cloud infrastructure 106. The data, in some embodiments, may include e-commerce data (e.g., shipments, orders, inventory), media data (e.g., pictures, movies, streaming data), financial data (e.g., banking data, investment data), or other data.

[0056] Storage 111A-111D may include storage systems for use by devices and systems in cloud infrastructure 106. In some embodiments, one or more of storage 111A-111D may be implemented as a hard drive, a RAID array, flash memory, optical storage, or any other kind of storage. Each of 111A-111D may include one or more filesystems (e.g., Amazon Elastic File System, GlusterFS, Google File System, Hadoop Distributed File System, OpenZFS, S3, Elastic Block Storage).

[0057] In some embodiments, systems and devices of cloud infrastructure 106 may use databases 109A-109D to store data that is accessed frequently (where, for example, access is required within a few milliseconds), and may use storage 111A-111D to store data that is accessed less frequently (where, for example, access is required within a few minutes or hours).

[0058] Keystores 113A-113D may include systems storing keys for accessing data and functionality. For example, to access certain data or systems, a system may require the use of passwords or keys in keystores 113A-113D for authentication. The data and functionality that the keys grant access to may be part of cloud infrastructure 106 or may be separate from cloud infrastructure 106. For example, keystores 113A-113D may include systems that store public and private keys (e.g., for use via SSH), may store passwords (e.g., login information for websites or programs), may store tokens (e.g., one-time passcodes), or the like. In some embodiments, keystores 113A-113D may be implemented as one or more of Amazon Web Services KMS, Azure Key Vault, or Google KMS.

[0059] Load balancer 115 may include one or more systems that balance incoming requests between the different systems and devices of cloud infrastructure 106. For example, load balancer 115 may be configured to determine usage (e.g., processor load, used storage capacity) of systems or devices in cloud infrastructure 106 to assist in determining where to route an incoming request from network 105 to store data, perform processing, or retrieve data.

[0060] Load balancer 115 may be configured to receive an incoming request from user device 102. Upon receipt of the request, load balancer 115 may consult a data store (part of or separate from load balancer 115; not pictured) to determine usage or forecasted usage of various systems or devices in cloud infrastructure 106, and may forward the request to the systems or devices having the lowest usage or forecasted usage.

[0061] **Fig. 2A** is a schematic block diagram illustrating an exemplary embodiment of a process 200 for integration, scanning, assessment, and review, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2A are executed using software and hardware of scanning system 101. In some embodiments, the steps in Fig. 2A may be performed in an order other than those depicted in Fig. 2A. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2A may be used to provide context or other information for use in another step of Fig. 2A. In other embodiments, the steps in Fig. 2A may be executed by other devices.

[0062] Process 200 begins with step 201. In step 201, scanning system 101 may execute a process of integration. The integration process may be performed by scanning system 101 with cloud infrastructure 106. In some embodiments, the integration process includes creating a connection between an account on scanning system 101 and an account on cloud infrastructure 106. The process of integration in step 201 may, in some embodiments, be implemented as described below with respect to Fig. 2B.

[0063] In step 203, scanning system 101 may execute a process of scanning/mapping. The scanning/mapping process may be performed by scanning system 101 with cloud infrastructure 106. In some embodiments, the scanning and mapping process may include analyzing data relating to cloud infrastructure 106 using scanning system 101, by reading cloud infrastructure 106 through the connection made in step 201 and generate a "map" (e.g., a data structure or data collection) representing systems and devices in cloud infrastructure 106. The process of scanning and mapping in step 203 may, in some embodiments, be implemented as described below with respect to Fig. 2C.

[0064] In step 205, scanning system 101 may execute a process of assessing. The assessing process may be performed by scanning system 101. In some embodiments, the assessing process may include reviewing vulnerabilities, infrastructure, interconnections, data, and other information, using scanning system 101. The process of assessing in step 205 may, in some embodiments, be implemented as described below with respect to Fig. 2D.

[0065] In step 207, scanning system 101 may execute a process of analyzing/reporting. The analyzing/reporting may be performed by scanning system 101. In some embodiments, the change reviewing process comprises scanning system 101 scanning cloud infrastructure 106 again to determine the differences between an earlier observed snapshot of cloud infrastructure 106 and the current state thereof. In some embodiments, scanning system 101 may analyze information (e.g., steps 203 and/or 205) to generate reports. The reports may list what each vulnerability is, where it is located, and its priority. In this way, security engineers and DevOps teams may be able to easily assess how to best allocate their time and attention.

[0066] In some embodiments, analyzing/reporting in step 207 may include scanning system 101 combining conclusions from different environmental perspectives (e.g., metadata) into a single model. For example, scanning system 101 may map the running services on cloud infrastructure 106 and consider collected vulnerability data. Scanning system 101, in some embodiments, may generate a visualization of the map for review, listing where the vulnerabilities are on a two-dimensional graphical representation of cloud infrastructure 106.

[0067] In some embodiments, scanning system 101 may indicate in the map whether an asset (e.g., a system or device) is Internet-facing and easily accessible to attackers, exposed only to internal assets or if it is private altogether (and may be less critical). For example, based on the contextual map, scanning system 101 may perform a "forward" analysis of the specific asset

under identification to identify at least one possible Internet-originating attack vector to the asset. Alternatively or additionally, scanning system 101 may perform a "backward" analysis of the specific asset to identify exposure risk to assets downstream of the specific asset, wherein the downstream exposure risk includes an identification of an exposed asset, an entry point to the exposed asset, and lateral movement risks associated with the exposed asset. Both backward and forward analysis can be recursively, for one or more hops, Scanning system 101 may present these analyses to a user via graphical user interface or expose them via an API.

[0068] As one example, scanning system 101 may determine a vulnerability in a web service. In some embodiments, scanning system 101 may score the vulnerability as: high risk if the web service is connected to the Internet (e.g., has at least one public port forwarded through a firewall to the web service, is able to be accessed through a load balancer, is able to be accessed via reverse proxy); medium risk if the web service is only accessible internally (e.g., because of firewall configuration); and low risk if access to the web server is blocked by a configuration of cloud infrastructure 106. As another example, if a machine is stopped (e.g., turned off, not running, not connected), a vulnerability, but one that is less likely to be exploited because the machine isn't running. This affects its risk score and other mitigating factors.

[0069] In step 209, scanning system 101 may execute a process of change reviewing. The change reviewing may be performed by scanning system 101. In some embodiments, the change reviewing process may include scanning system 101 scanning cloud infrastructure 106 again to determine the differences between an earlier observed snapshot of cloud infrastructure 106 and the current state thereof.

[0070] In some embodiments, scanning system 101 may monitor cloud change, or "delta," logs (e.g., Cloud event logs such as AWS CloudTrail, monitoring a network-related change in state, a trust-related change in state, or a disk configuration-related change in state in at least one of the primary asset group or a secondary asset group) and contextualize associated risks discovered within a customer cloud account. In some embodiments, scanning system 101 may generate a new map of assets based on a review of all systems or devices in cloud infrastructure 106 (e.g., as described above with respect to step 203), and compare the new map to the map generated earlier in step 203 to determine which assets have changed. In some embodiments, this comparison may require comparing data, metadata, network connections, software configurations, firewall configurations, or any other aspects related to cloud infrastructure 106, including, for example, determining new assets that are created or assets for which their configuration has changed.

[0071] After step 209, as depicted in Fig. 2A, process 200 may return to step 205. Scanning system 101 may then perform step 205 again. In some embodiments, steps 205, and 207 may be performed only on the devices or systems that have experienced a change that was

detected by step 209. In some embodiments, process 200 may return to step 203 after step 209.

[0072] **Fig. 2B** is a schematic block diagram illustrating an exemplary embodiment of a process 210 for integration, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2B are executed using software and hardware of scanning system 101.Integration, in some embodiments, is a process for creating a trust relationship between scanning system 101 and cloud infrastructure 106.

[0073] In some embodiments, the steps in Fig. 2B may be performed in an order other than those depicted in Fig. 2B. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2B may be used to provide context or other information for use in another step of Fig. 2B. In other embodiments, the steps in Fig. 2B may be executed by other devices.

[0074] Process 210 may begin with step 211 to initiate a connection to cloud infrastructure 106. In step 211, scanning system 101 may send a message to user device 102 instructing a user to authenticate, or log in, to a cloud service provider's system operating cloud infrastructure 106. For example, the user may use a username, password, one-time password, two-factor authentication, or any other authentication mechanism to gain access to a cloud service provider's system.

[0075] Concurrently with or after the first message, scanning system 101 may send a second message to user device 102, instructing the user to generate a role. The second message may include instructions for the user to follow to generate the role. In step 213, a user may provide (e.g., via a keyboard at user device 102) a role definition to the cloud service provider's system. In some embodiments, the role definition includes read-only permissions and permissions to read a block storage layer (containing block storage volumes). In some embodiments, scanning system 101 provides a role formation template (e.g., an Amazon Web Services CloudFormation Template) for use with cloud infrastructure 106 to create the necessary role. In step 213, the user may utilize user device 102, for example, by copying and pasting a URL of the template, downloading and uploading the template to the cloud service provider's system, or selecting the template from a list of templates.

[0076] In step 215, the cloud service provider's system may generate a string (e.g., a "key" or "resource name") for use by scanning system 101. In some embodiments, this string may be used to enable access by scanning system 101 to the workload of cloud infrastructure as permitted by the generated role. A user using user device 102 may copy the string and paste it into a user interface presented by scanning system 101 on user device 102. Other aspects of transmitting this string to scanning system 101 are possible as well.

[0077] In step 217, scanning system 101 determines that it is able to access cloud infrastructure 106. For example, scanning system 101 may attempt to perform a command such

as authenticating to cloud infrastructure 106 using the string received in step 215 or attempting to perform a command using cloud infrastructure (e.g., requesting a listing of files stored on storage 111A). Once scanning system 101 determines that it is able to access cloud infrastructure 106, the process may return to Fig. 2A and step 203.

[0078] **Fig. 2C** is a schematic block diagram illustrating an exemplary embodiment of a process 220 for scanning/mapping, consistent with the disclosed embodiments. In some embodiments, the steps in Fig. 2C are executed using software and hardware of scanning system 101. In some embodiments, the steps in Fig. 2C may be performed in an order other than those depicted in Fig. 2C. In some embodiments, steps may also be omitted, repeated, or modified. In some embodiments, information gathered in one step of Fig. 2C may be used to provide context or other information for use in another step of Fig. 2C. In other embodiments, the steps in Fig. 2C may be executed by other devices.

[0079] Process 220 may begin with step 221 where scanning system 101 initiates a process to access keys, such as those keys stored in keystores 113A-113D. Scanning system 101 may utilize the string (e.g., a key or resource name) received in process 210 to authenticate in order to retrieve keys stored in keystores 113A-113D. Once authenticated, scanning system 101 may access one, some, or all keys stored in keystores 113A-113D. In some embodiments, step 221 may comprise utilizing pre-established trust relationships, such as AWS trust policies, instead of creating a new relationship.

[0080] In step 223, scanning system 101 may generate a “snapshot” of devices and systems in cloud infrastructure 106. In some embodiments, generating a snapshot may include reading devices and systems in cloud infrastructure, such as storage 111A-111D, databases 109A-109D, and virtual machines 107A-107D, and copying the information read from those devices and systems to storage at scanning system 101. Generating a snapshot, in some embodiments, may include recording a reference count to data blocks in one or more of storage 111A-111D, databases 109A-109D, and virtual machines 107A-107D, and copying each block them like a copy and write operation. In some embodiments, scanning system 101 may generate an Elastic Block Storage snapshot in step 223.

[0081] In some embodiments, scanning system 101 uses one or more keys retrieved from keystores 113A-113D to encrypt “snapshots” of cloud infrastructure.

[0082] In step 225, scanning system 101 may apply “tags” to the snapshot. This process may include, in some embodiments, adding information to the snapshot to identify the snapshot as being associated with scanning system 101. Scanning system 101, in some embodiments, may be configured to delete only snapshots with the associated tags.

[0083] In step 227, scanning system 101 may generate a map of cloud infrastructure 106. In some embodiments, this map may be in a form of graph: a plurality of interconnected vectors connecting the plurality of systems and devices, based on the networking configuration.