

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NORMSHIELD INC. d/b/a BLACK KITE INC.,

Petitioners,

v.

BITSIGHT TECHNOLOGIES, INC.,

Patent Owner.

Case No. IPR2024-01393

U.S. Patent No. 9,973,524

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 9,973,524**

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
II. STANDING	2
III. GROUNDS	2
IV. POSITA	2
V. BACKGROUND	2
A. Technical Background (EX1003, ¶¶50-67)	2
B. Public Availability and Authenticity	14
1. McNab	14
2. Internet Archive Pages	14
3. Public Videos	16
C. '524 patent	17
1. Prosecution History	18
VI. CLAIM CONSTRUCTION	19
VII. GROUNDS	19
A. Ground 1: McNab and McGovern	19
1. Motivation to Combine	19
(a) McGovern	19
(b) Nexus	23
2. Claim 1	24
(a) 1[pre], 1[a]	24
(b) 1[b]	32
(c) 1[c]	40
(d) 1[d]	48
(e) 1[e]	50
3. Claim 2: The method of claim 1 in which the technical assets comprise an IP space.	51

4.	Claim 3: The method of claim 1 in which maintaining an entity map comprises querying a regional Internet registry.	52
5.	Claim 4: The method of claim 3 in which maintaining an entity map comprises submitting a name of one of the entities to the regional Internet registry.	52
6.	Claim 5: The method of claim 4 in which submitting the entity name includes submitting spelling variations of the entity name.	53
7.	Claim 6: The method of claim 1 in which maintaining an entity map comprises using a domain name associated with the entity.	54
8.	Claim 7: The method of claim 6 in which maintaining an entity map comprises using a dig tool on the domain name to determine a published IP address.	54
9.	Claim 8: The method of claim 1 in which maintaining an entity map comprises probing IP addresses around a published IP address.	55
10.	Claim 10: The method of claim 1 in which maintaining an entity map comprises using a name server lookup to verify the derived server name.	56
11.	Claim 11: The method of claim 2 comprising using the IP space for collecting the cyber-security characterizations.	56
12.	Claim 12: The method of claim 1 in which the technical assets that are associated with cyber-security characterizations of the respective entities comprise technical assets that the entity owns, controls, uses, or is affiliated with.	57
13.	Claim 13: The method of claim 1 in which the technical assets comprise ranges of IP addresses.	58
14.	Claim 14: The method of claim 1 in which the technical assets comprise servers.	58
15.	Claim 15: The method of claim 1 in which maintaining an entity map comprises using information received from Internet Service Providers.	58

16.	Claim 16: The method of claim 1 in which maintaining an entity map comprises using heuristics.	58
17.	Claim 18: The method of claim 1, wherein annotating comprises prepending to the domain name at least one of “mail,” “smtp,” and “ftp”.....	59
18.	Claims 19 and 20.....	59
B.	Ground 2: Cole, McNab, and McGovern.....	60
1.	Motivation to Combine	60
	(a) McNab	61
	(b) McGovern.....	63
2.	Claim 1	66
	(a) 1[pre], 1[a]	67
	(b) 1[b].....	68
	(c) 1[c].....	72
	(d) 1[d].....	74
	(e) 1[e].....	75
3.	Claim 2	79
4.	Claim 3	80
5.	Claim 4.....	80
6.	Claim 5.....	80
7.	Claim 6.....	80
8.	Claim 7	80
9.	Claim 8.....	80
10.	Claim 10.....	80
11.	Claim 11	81
12.	Claim 12.....	81
13.	Claim 13.....	81
14.	Claim 14.....	81
15.	Claim 15.....	81
16.	Claim 16.....	81

17.	Claim 18.....	81
18.	Claims 19 and 20.....	82
VIII.	<i>FINTIV</i>	83
IX.	§325(D).....	83
X.	MANDATORY NOTICES AND FEES	83
A.	Real Party-In-Interest	83
B.	Related Matters.....	83
C.	Counsel and Service Information	84
D.	Payment of Fees	85

EXHIBIT LIST

Ex.	Description
1001	U.S. Patent No. 9,973,524 (“’524 patent”)
1002	Prosecution History for U.S. Patent No. 9,973,524
1003	Declaration of Kevin Almeroth, Ph.D.
1004	<i>Curriculum Vitae</i> of Kevin Almeroth Ph.D.
1005	McNab, “Network Security Assessment,” O’Reilly Media, Inc. (2008) (prior art under at least § 102(b))
1006	U.S. Pat. No. 7,257,630 (“Cole”) (prior art under at least § 102(b))
1007	U.S. Pat. Pub. No. 2009/0024663 A1 (“McGovern”) (prior art under at least § 102(b))
1008	Nessus, “Plugins: The remote host is infected by msblast.exe”, available at https://web.archive.org/web/20060924205758/http://www.nessus.org/plugins/index.php?view=single&id=11818
1009	Nessus, “Plugins: The remote host is infected by a virus”, available at https://web.archive.org/web/20070222091638/http://www.nessus.org/plugins/index.php?view=single&id=11329
1010	“Nessus 3.0 Client Guide,” available at https://web.archive.org/web/20070312175637/http://nessus.org:80/documentation/nessus_3.0_client_guide.pdf
1011	“Nessus, Part 3: Analysing Reports,” available at https://web.archive.org/web/20061020202310/http://www.securityfocus.com/infocus/1759
1012	Nessus, “Documentation,” available at https://web.archive.org/web/20070219213924/http://www.nessus.org/documentation/
1013	Declaration of Nathaniel Frank-White

1014	Declaration of Sylvia Hall-Ellis, Ph.D.
1015	Matta Security Limited, “An Introduction to Internet Attack & Penetration,” available at https://web.archive.org/web/20070202020317/http://www.trustmatta.com/downloads/pdf/Matta_Attack_and_Penetration_Introduction.pdf
1016	Stoneburgner, et al., “Risk Management Guide for Information Technology Systems,” NIST, available at https://www.archives.gov/files/era/recompete/sp800-30.pdf
1017	Moore & Valsmith, et al., “Tactical Exploitation,” available at https://web.archive.org/web/20100208161237/https://www.blackhat.com/presentations/bh-usa-07/Moore_and_Valsmith/Whitepaper/bh-usa-07-moore_and_valsmith-WP.pdf
1018	Chris Gates, “New School Information Gathering” (2008) available at https://www.carnal0wnage.com/papers/17_Gates.pdf
1019	Chris Gates, “Toorcon X Gates: New School Information Gathering,” available at http://vimeo.com/2745624
1020	Chris Gates, “Toorcon X Gates: New School Information Gathering,” available at https://web.archive.org/web/20090302045813/https://vimeo.com/2745624
1021	Social-Engineer, LLC, “Social Engineering Using Maltego,” available at https://www.youtube.com/watch?v=qiv4-wy3mxo
1022	“W32.Blaster.Worm: Technical Details” available at https://web.archive.org/web/20070503023514/http://www.symantec.com/security_response/writeup.jsp?docid=2003-081113-0229-99&tabid=2
1023-1028	Intentionally Left Blank
1029	August 29, 2024 Email from Melissa Nezhnik

1030	Metadata for YouTube video ID qiv4-wy3mxo
1031	Relevant screen captures of Exhibit 1021

LIST OF CHALLENGED CLAIMS

Claim	Limitation
1[pre]	A computer implemented method comprising:
1[a]	on one or more computers, maintaining in a database an entity map that maps technical assets to respective companies or other entities with which the assets are associated, wherein mapping technical assets to respective companies or other entities comprises determining IP information by annotating a domain name with naming conventions for at least one of a mail server, an SMTP server, and an FTP server, thereby deriving a likely server name having assigned IP addresses,
1[b]	automatically using [<i>sic</i>] sensors on the Internet to collect externally observable cyber-security characterizations of the technical assets that have been mapped to each of the entities and associating the observable cyber-security characterizations with a respective entity based at least in part on the derived server name,
1[c]	automatically deriving observations about the technical assets from the collected cyber-security characterizations, wherein the derived observations comprise (i) a number of technical assets that have been reported to be malicious and (ii) a duration of detected malicious activity associated with the technical assets,
1[d]	automatically generating a cyber-security rating for each of the entities using the entity map and the derived observations, and
1[e]	through a communication network, exposing to users in a user interface of a portal, information about the cyber-security ratings of the entities that is useful to decisions about entities with which to do business.
2	The method of claim 1 in which the technical assets comprise an IP space.

Claim	Limitation
3	The method of claim 1 in which maintaining an entity map comprises querying a regional Internet registry.
4	The method of claim 3 in which maintaining an entity map comprises submitting a name of one of the entities to the regional Internet registry.
5	The method of claim 4 in which submitting the entity name includes submitting spelling variations of the entity name.
6	The method of claim 1 in which maintaining an entity map comprises using a domain name associated with the entity.
7	The method of claim 6 in which maintaining an entity map comprises using a dig tool on the domain name to determine a published IP address.
8	The method of claim 1 in which maintaining an entity map comprises probing IP addresses around a published IP address.
10	The method of claim 1 in which maintaining an entity map comprises using a name server lookup to verify the derived server name.

Claim	Limitation
11	The method of claim 2 comprising using the IP space for collecting the cyber-security characterizations.
12	The method of claim 1 in which the technical assets that are associated with cyber-security characterizations of the respective entities comprise technical assets that the entity owns, controls, uses, or is affiliated with.
13	The method of claim 1 in which the technical assets comprise ranges of IP addresses.
14	The method of claim 1 in which the technical assets comprise servers.
15	The method of claim 1 in which maintaining an entity map comprises using information received from Internet Service Providers.
16	The method of claim 1 in which maintaining an entity map comprises using heuristics.
18	The method of claim 1, wherein annotating comprises prepending to the domain name at least one of “mail,” “smtp,” and “ftp”.
19 [pre]	A computer implemented method comprising:

Claim	Limitation
19[a]	on one or more computers operating with a database, maintaining an entity map that maps technical assets that comprise an IP space to respective entities with which the assets are associated,
19[b]	the maintaining of the entity map comprising (a) automatically using sensors on the Internet to collect externally observable cyber-security characterizations of the technical assets that have been mapped to each of the entities, (b) associating the observable cyber-security characterizations with a respective entity based at least in part on the derived server name, (c) submitting names of the entities to regional Internet registries using various spellings of the names, (d) using domain names associated with the entities, (e) probing IP addresses around published IP addresses, (f) using a name server lookup to verify IP addresses associated with domain names, (g) using information received from Internet Service Providers, (h) annotating a domain name with naming conventions for at least one of a mail server, an SMTP server, and an FTP server, thereby deriving a likely server name having assigned IP addresses, and (i) applying heuristics,
19[c]	automatically deriving observations about the technical assets from the collected cyber-security characterizations, wherein the derived observations comprise (a) a number of technical assets that have been reported to be malicious and (b) a duration of detected malicious activity associated with the technical assets,
19[d]	automatically generating a cyber-security rating for each of the entities using the entity map and the derived observations, and
19[e]	through a communication network, exposing to users in a user interface the cyber-security ratings of the entities.
20[pre]	A system comprising:

Claim	Limitation
20[a]	one or more computers programmed to facilitate operations comprising:
20[b]	maintaining in a database an entity map that maps technical assets to respective companies or other entities with which the assets are associated, wherein mapping technical assets to respective companies or other entities comprises determining IP information by annotating a domain name with naming conventions for at least one of a mail server, an SMTP server, and an FTP server, thereby deriving a likely server name having assigned IP addresses,
20[c]	collecting, using sensors on the Internet, externally observable cyber-security characterizations of the technical assets that have been mapped to each of the entities and associating the observable cyber-security characterizations with a respective entity based at least in part on the derived server name,
20[d]	deriving observations about the technical assets from the collected cyber-security characterizations, wherein the derived observations comprise (i) a number of technical assets that have been reported to be malicious and (ii) a duration of detected malicious activity associated with the technical assets,
20[e]	generating a cyber-security rating for each of the entities using the entity map and the derived observations, and
20[f]	through a communication network, exposing to users in a user interface of a portal, information about the cyber-security ratings of the entities that is useful to decisions about entities with which to do business.

I. INTRODUCTION

The '524 patent relates to a conventional method for mapping and evaluating the security of a network by performing external cyber-security tests, generating a rating and displaying that rating in a user interface. But by 2010, the earliest priority date of the '524 patent, network security intrusion tests were common-place and well-known. As the '524 patent recognizes, “cyber security assessment[s]” were well-known in the art by the time of the patent. EX1001, 1:29-33. Indeed, the grounds presented in this petition rely on an O'Reilly-published book called “Network Security Assessment” (“McNab”) or alternatively a patent filed by McAfee.

The McNab book discloses numerous methods for mapping and evaluating the security of a network. In particular, it discloses the capability of mapping the assets of a network by, for example, SMTP probing and using DNS name servers. It also discloses numerous methods for evaluating the security of the assets within the network, including the use of various well-known software that can perform vulnerability checks from outside of the network.

McAfee's Cole reference similarly discloses a testing system allowing highly automated testing of vulnerabilities to intrusion on a target network including through the use of creating a network topology, *i.e.* a network mapping. The testing

results in a quantitative assessment that can be displayed via a user interface. Thus, Cole provides an alternative basis for invalidating the challenged claims.

II. STANDING

Petitioner certifies the '524 patent is available for IPR and they are not estopped.

III. GROUNDS

Petitioner presents the below grounds.

Ground	Basis	Reference(s)	Claims
1	§103	McNab, McGovern	1-8, 10-16, 18-20
2	§103	Cole, McNab, McGovern	1-8, 10-16, 18-20

IV. POSITA

A POSITA would have a Bachelor's degree in Computer Science or similar field and at least two years of industry experience in network security. EX1003, ¶49. More education could compensate for less experience and vice versa. *Id.* Petitioner's expert surpassed this level of skill in the art at the earliest filing date of the '524 patent. *Id.*

V. BACKGROUND

A. Technical Background (EX1003, ¶¶50-67)

The '524 Patent is directed to collecting information about network resources, generally from external information sources. Prior to 2010, ordinary artisans in the field were aware of numerous techniques for collecting information about a network and its resources using what are known as Open Source Intelligence ("OSINT")

tools, “open source” indicating that such information could be collected without access to or permission from the target organization.

For example, there were a number of well-known OSINT tools for collecting IP address information about assets. The McNab reference devotes an entire chapter to such methods for “Internet Host and Network Enumeration,” *i.e.* enumerating the various hosts within a network, including their IP addresses.

Specifically, McNab describes a tool called “whois,” which can look up various information about a domain name, including its authoritative name servers. EX1005, 22.

Example 3-1. Obtaining the domain WHOIS record for blah.com

\$ whois blah.com

```
Domain Name: BLAH.COM
Registrar: NETWORK SOLUTIONS, LLC.
Whois Server: whois.networksolutions.com
Referral URL: http://www.networksolutions.com
Name Server: NS1.BLAH.COM
Name Server: NS2.BLAH.COM
Name Server: NS3.BLAH.COM
Status: clientTransferProhibited
Updated Date: 04-oct-2006
Creation Date: 20-mar-1995
Expiration Date: 21-mar-2009
```

McNab also describes a well-known, free tool called “dig” which can perform a number of functions related to DNS queries. McNab describes using the “dig” command to retrieve a list of hostnames for a specified domain name (“ucia.gov” in

the example below) from a specified name server (“relay2.ucia.gov” in the example below), using a technique called a “DNS zone transfer.” EX1005, 33.

Example 3-8. Using dig to perform a DNS zone transfer

\$ dig @relay2.ucia.gov ucia.gov axfr

```
; <<>> DiG 9.2.4 <<>> ucia.gov @relay2.ucia.gov axfr
;; global options: printcmd
ucia.gov.          3600    IN      SOA     relay1.ucia.gov. root.ucia.gov.
511210023 7200 900 604800 900
ucia.gov.          3600    IN      NS      relay1.ucia.gov.
ucia.gov.          3600    IN      NS      relay7.ucia.gov.
ucia.gov.          3600    IN      NS      auth100.ns.uu.net.
ucia.gov.          3600    IN      MX      5 mail2.ucia.gov.
ain.ucia.gov.      3600    IN      A       198.81.128.68
ain-relay.ucia.gov. 3600    IN      CNAME   relay1.ucia.gov.
ain-relay-int.ucia.gov. 3600    IN      CNAME   ain-relay1-int.ucia.gov.
ain-relay1.ucia.gov. 3600    IN      CNAME   relay1.ucia.gov.
ain-relay1-ext.ucia.gov. 3600    IN      CNAME   relay1.ucia.gov.
ain-relay1-int.ucia.gov. 3600    IN      A       192.168.64.2
ain-relay2.ucia.gov. 3600    IN      CNAME   relay2.ucia.gov.
ain-relay2-ext.ucia.gov. 3600    IN      CNAME   relay2.ucia.gov.
ain-relay2-int.ucia.gov. 3600    IN      A       192.168.64.3
ain-relay7.ucia.gov. 3600    IN      CNAME   relay7.ucia.gov.
ain-relay7-ext.ucia.gov. 3600    IN      CNAME   relay7.ucia.gov.
ain-relay7-int.ucia.gov. 3600    IN      A       192.168.64.67
ex-rtr.ucia.gov.   3600    IN      CNAME   ex-rtr-129.ucia.gov.
ex-rtr-129.ucia.gov. 3600    IN      A       198.81.129.222
ex-rtr-129.ucia.gov. 3600    IN      HINFO   "Cisco 4000 Router" "NP-1E Board"
ex-rtr-191-a.ucia.gov. 3600    IN      A       192.103.66.58
ex-rtr-191-b.ucia.gov. 3600    IN      A       192.103.66.62
foia.ucia.gov.     3600    IN      NS      relay1.ucia.gov.
foia.ucia.gov.     3600    IN      NS      auth100.ns.uu.net.
mail1.ucia.gov.    3600    IN      A       198.81.129.68
mail1out.ucia.gov. 3600    IN      A       198.81.129.71
mail2.ucia.gov.    3600    IN      A       198.81.129.148
mail2out.ucia.gov. 3600    IN      A       198.81.129.146
relay.ucia.gov.    3600    IN      CNAME   relay1.ucia.gov.
relay-int.ucia.gov. 3600    IN      CNAME   ain-relay1-int.ucia.gov.
relay1.ucia.gov.   3600    IN      A       198.81.129.193
relay1-ext.ucia.gov. 3600    IN      CNAME   relay1.ucia.gov.
relay1-int.ucia.gov. 3600    IN      CNAME   ain-relay1-int.ucia.gov.
relay2.ucia.gov.   3600    IN      A       198.81.129.194
relay2-ext.ucia.gov. 3600    IN      CNAME   relay2.ucia.gov.
relay2-int.ucia.gov. 3600    IN      CNAME   ain-relay2-int.ucia.gov.
relay2a.ucia.gov.  3600    IN      A       198.81.129.200
relay2y.ucia.gov.  3600    IN      A       198.81.129.68
relay2z.ucia.gov.  3600    IN      A       198.81.129.69
relay7.ucia.gov.   3600    IN      A       198.81.129.186
relay7-ext.ucia.gov. 3600    IN      CNAME   relay7.ucia.gov.
relay7a.ucia.gov.  3600    IN      A       198.81.129.197
relay7b.ucia.gov.  3600    IN      A       198.81.129.198
res.ucia.gov.      3600    IN      A       198.81.129.116
wais.ucia.gov.     3600    IN      CNAME   relay2.ucia.gov.
```

Once the network hosts have been enumerated, they can then be scanned to assess information about the network. McNab itself explains that “Network scanners are used to perform bulk automated scanning of IP ranges to identify vulnerable network service components. The two most popular open source network scanners are Nmap and Nessus.” EX1005, 13. Nessus, for example, “is a vulnerability assessment package that can perform many automated tests against a target network, including ICMP, TCP, and UDP scanning, testing of specific network services (such as Apache, MySQL, Oracle, Microsoft IIS, and many others), and rich reporting of vulnerabilities identified.” *Id.* By identifying vulnerabilities, the network can be given a risk assessment score.

This process of enumerating network assets and performing vulnerability scans to generate risk assessment scores was described by other sources as well. For example, a 2001 paper titled “An Introduction to Internet Attack & Penetration” (EX1015) describes an “Anatomy of a Hack” that lists the following steps (at 6):

1. Identification of Internet-based points of presence
2. Network scanning & reconnaissance
3. Accessing the target system or network
4. Carrying out objectives
5. Manipulation of logs and system files

The paper then goes on to describe attacks such as the exploitation of vulnerabilities, or various forms of malware. EX1015, 9-14.

In 2002, the National Institute of Standards and Technology (“NIST”) (an organization part of the U.S. Department of Commerce) published a paper called “Risk Management Guide for Information Technology Systems” (EX1016), which provides guidelines and methodologies for conducting risk assessments, mitigating risks, and ensuring the security of IT systems within the federal government. Section 3.1.2 of the paper outlines various information-gathering techniques, such as “Automated Scanning Tool[s]” such as “network mapping tool [to] identify the services that run on a large group of hosts and provide a quick way of building individual profiles of the target IT system(s).” EX1016, 11-12. This results in a “picture of the IT system environment, and delineation of system boundary.” *Id.* The paper then goes on in Chapter 3.3 to discuss scanning these services to identify vulnerabilities, noting that “[t]he analysis of the threat to an IT system must include an analysis of the vulnerabilities associated with the system environment.” *Id.*, 15. Based on this analysis, the paper recommends producing a risk rating, which is derived “by multiplying the ratings assigned for threat likelihood (e.g., probability) and threat impact,” resulting in various “risk levels of High, Medium, and Low” within a risk-level matrix. *Id.*, 24.

Table 3-6. Risk-Level Matrix

Threat Likelihood	Impact		
			High (100)
High (1.0)	Low 10 X 1.0 = 10	Medium 50 X 1.0 = 50	High 100 X 1.0 = 100
Medium (0.5)	Low 10 X 0.5 = 5	Medium 50 X 0.5 = 25	Medium 100 X 0.5 = 50
Low (0.1)	Low 10 X 0.1 = 1	Low 50 X 0.1 = 5	Low 100 X 0.1 = 10

*Risk Scale: High (>50 to 100); Medium (>10 to 50); Low (1 to 10)*⁸

Id., 25.

A 2007 presentation given at the Blackhat conference provides an overview of “gathering and discovery techniques, with a concentration on third-party services and new tools” and then using such tools to “gain[] access to a target network.” EX1017, 3. Thus, in Chapter 3, the paper states “[t]he first step to any security engagement is the initial discovery process,” and describes “discovery techniques, starting from the outside.” *Id.*, 6. The chapter goes on to provide resources for discovering employees of an organization, such as through Google searches, or using a tool called Evolution that “is able to cross-reference information from a large set of public data sources.” *Id.*, 7. The chapter then explains how to use tools to perform “network discovery” using “DNS zone transfers, Whois requests, and reverse DNS lookups.” *Id.*, 8. The chapter then provides descriptions of more advanced tools such as a “‘Domain Dossier’ service which combines the various DNS and Whois

requests into [a] single report, with the option to perform a quick port scan as well.”

Id. There is also the “DomainTools.com web site” which “accepts an IP address or host name as an input and provides a list of all domains that reverse back to that IP.”

Id.

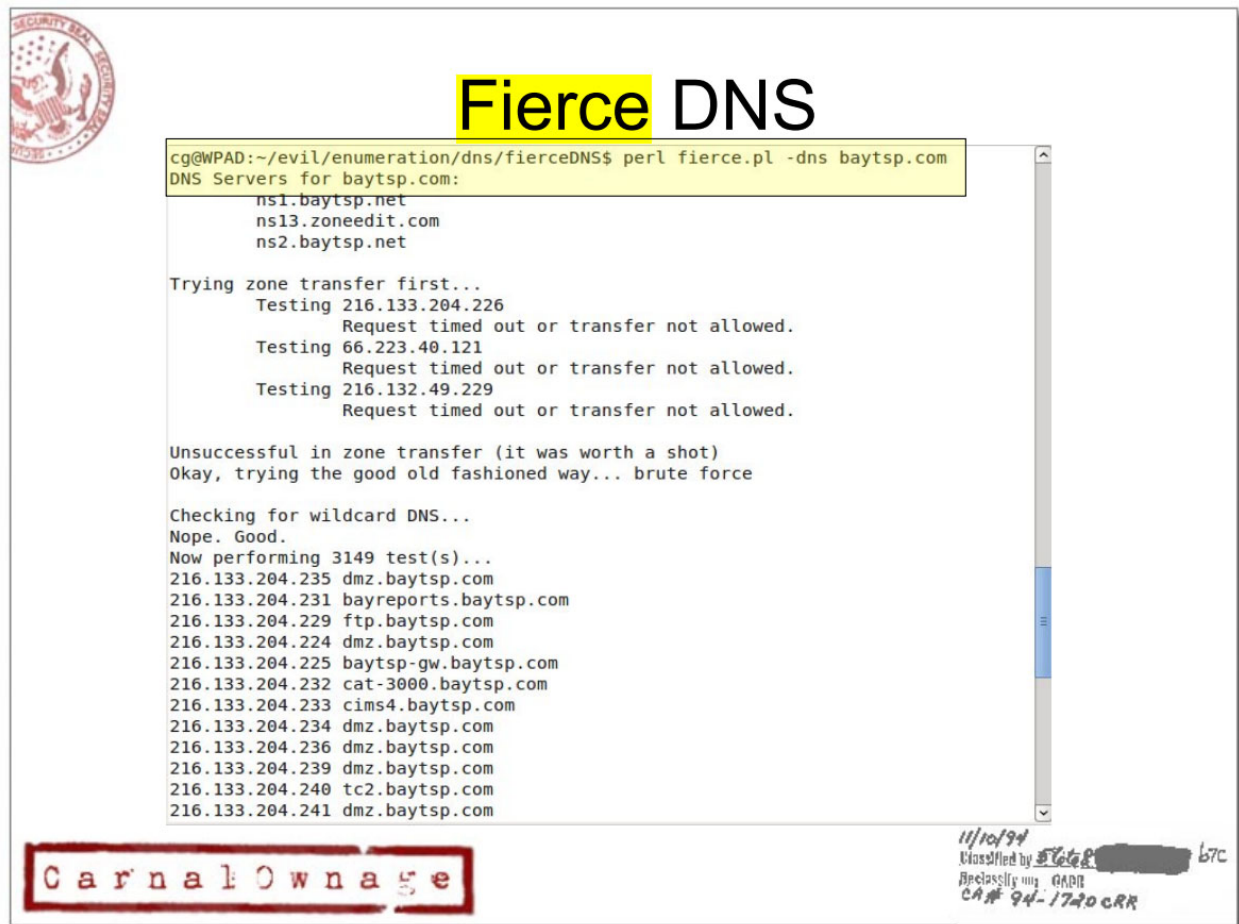
The paper also describes using “the Metasploit Framework” which “includes a number of application discovery modules ... to detect services that are difficult to find otherwise.” *Id.*, 13. For example, the below output searched on an IP range and returned DNS and SNMP services running on hosts within that IP range:

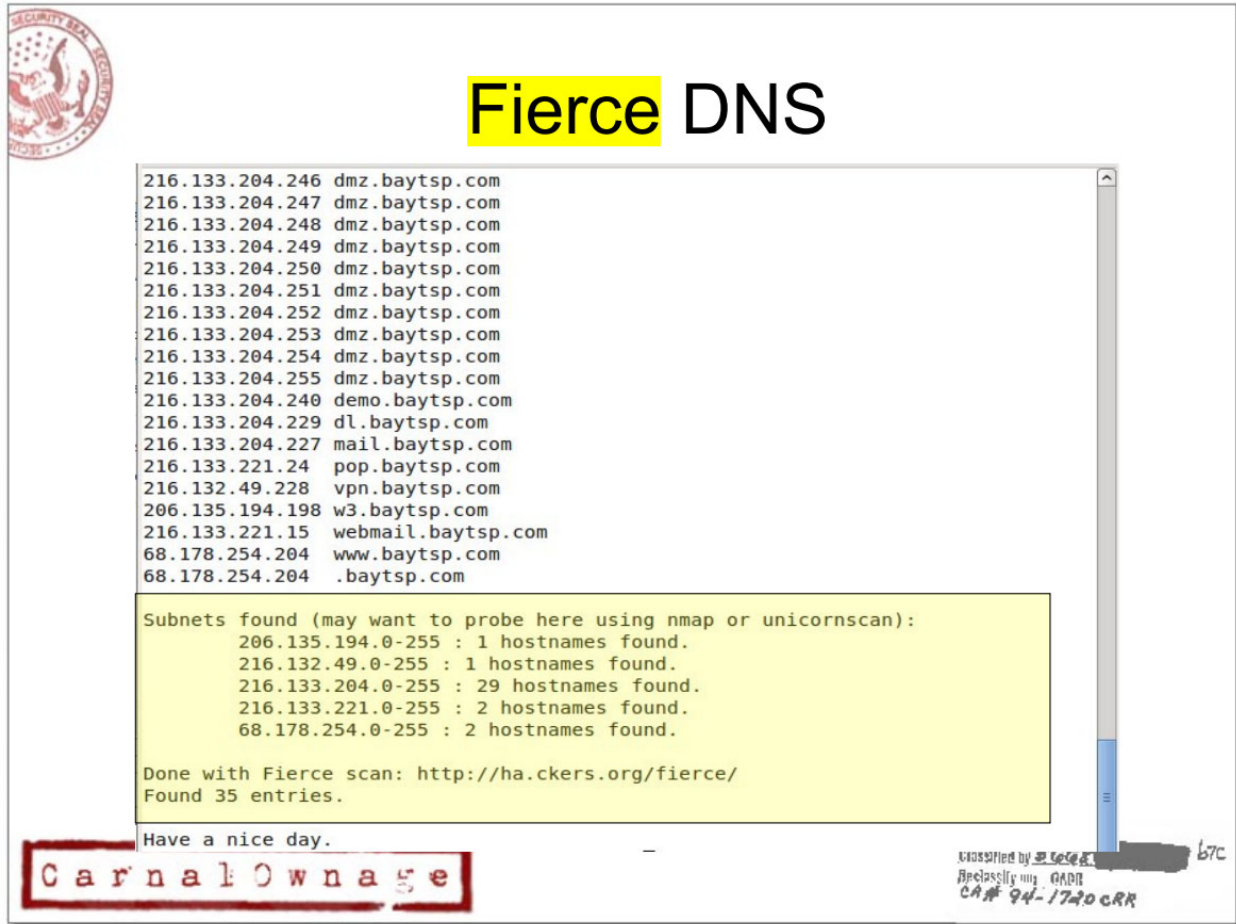
```
[*] Sending 6 probes to xxx.xxx.xxx.0->xxx.xxx.xxx.255 (256 hosts)
[*] Discovered DNS on xxx.xxx.xxx.19 (TinyDNS)
[*] Discovered DNS on xxx.xxx.xxx.249 (BIND 8.4.6-REL-NOESW)
[*] Discovered DNS on xxx.xxx.xxx.250 (Microsoft)
[*] Discovered SNMP on xxx.xxx.xxx.170 (Ethernet Routing Switch)
[*] Discovered SNMP on xxx.xxx.xxx.171 (ProCurve J8692A)
```

Id. The paper goes on to describe various ways to attack these services. *Id.*, 19-33.

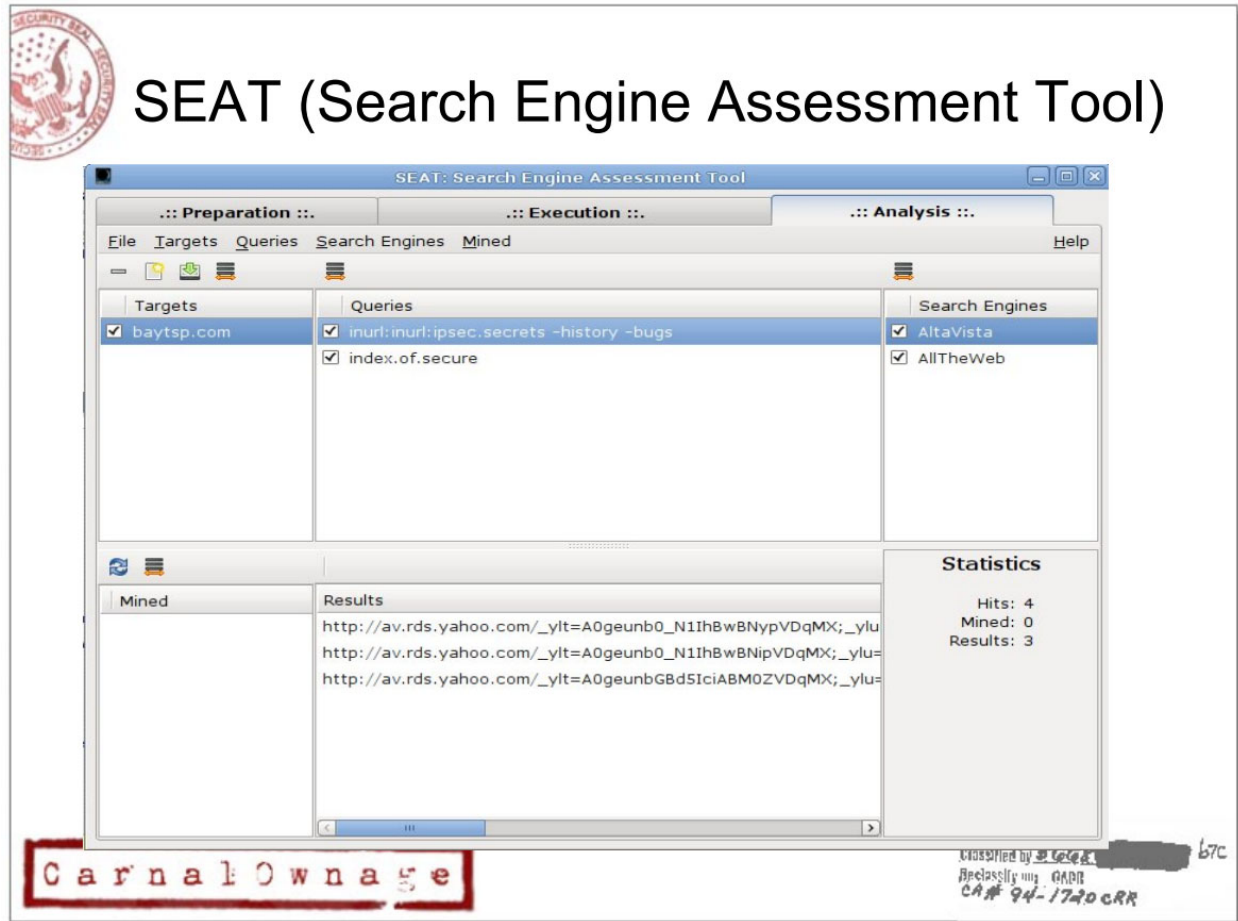
In 2008, a well-known security researcher named Chris Gates delivered a presentation called “New School Information Gathering” (EX1018). The presentation describes a “[b]lack box approach to info gathering” where “the customer wants you to find everything about them you can via ‘open source’ methods.” EX1018, 5. OSINT information gathering involves “no direct contact with victim’s servers OR no non-standard traffic directed toward victim.” *Id.*, 9. The presentation then describes several tools for gathering such information. For

example, the presentation describes one tool called Fierce DNS which “locate[s] likely targets both inside and outside a corporate network.” *Id.*, 18. Fierce DNS uses a number of DNS queries to find hosts on the same domain and IP blocks used by the domain. *Id.*, 20. The below output for “bytsp.com” shows the various hosts for that domain name. *Id.*, 21-22.





Another tool called “SEAT (Search Engine Assessment Tool)” “uses information stored in search engine databases, cache repositories, and other public resources to scan a site *for potential vulnerabilities*.” *Id.*, 24 (emphasis added). The tool involves providing a domain name as input (such as “baytsp.com” as used in the slides) and returning a set of results about that domain. *Id.*, 26.

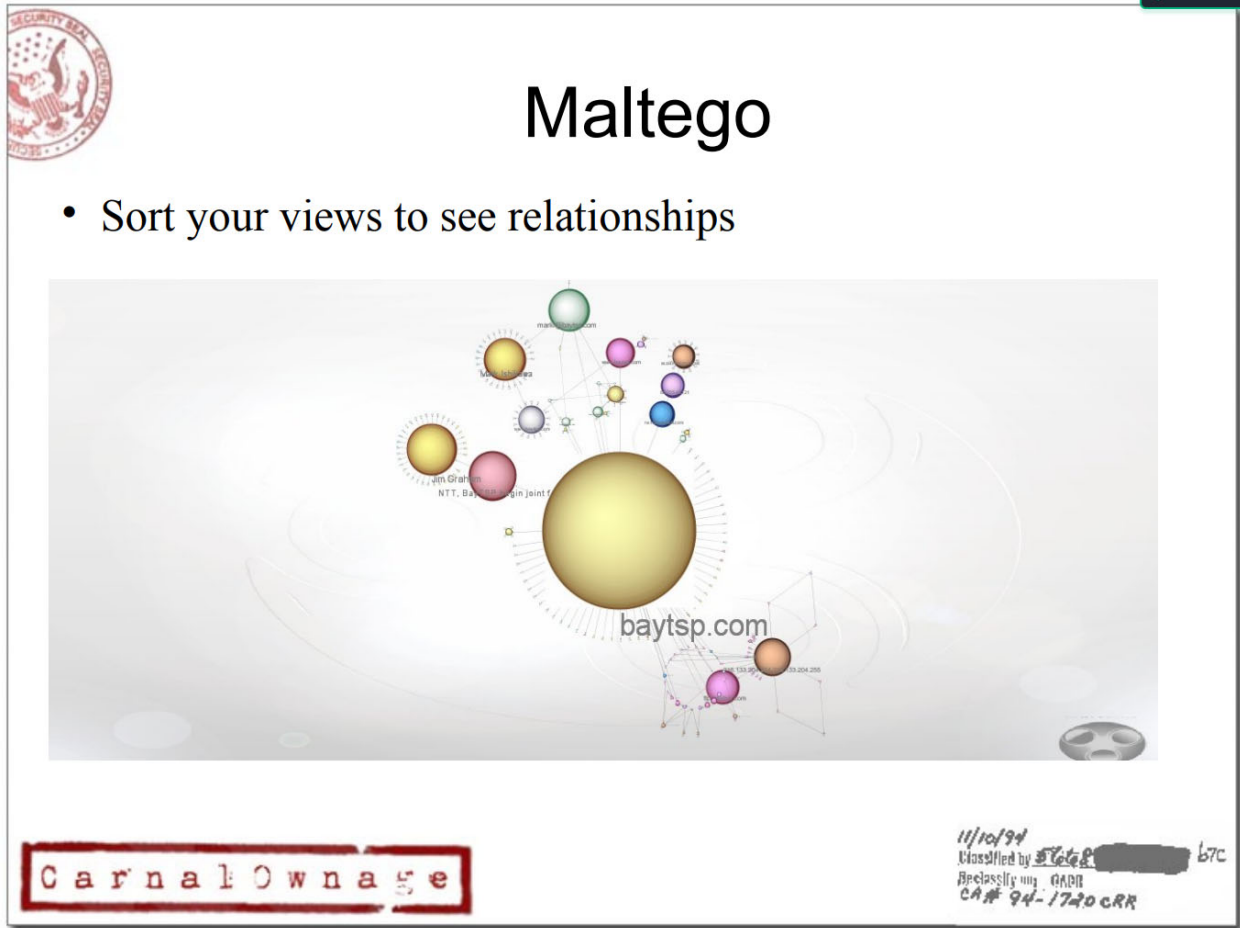


The presentation also describes methods for harvesting email addresses from the provided domain name. *Id.*, 28-32.

Another tool called Metagoofil searches for PDF, Word, and other documents found on the provided domain name and extracts metadata from those documents to, for example, “[r]eveal the creator of a document,” “[r]eveal the application that created the document,” “[r]eveal creation date,” all in order to “deliver a directed client side attack for something installed in the enterprise.” *Id.*, 35-40.

Finally, the presentation describes a tool called Maltego, which can perform many of the above functions automatically in one piece of software. Maltego “is a

program that can be used to determine the relationships and real world links between: People ... Companies ... Web sites ... Internet infrastructure such as: Domains, DNS names, Netblocks, IP addresses” and it does this “using open source intelligent (OSINT).” *Id.*, 67. Maltego can enumerate network assets given a domain name, such as by finding “MX and NS servers” for the domain name and determining “what hostnames are shared by the DNS servers,” and “brute forc[ing] our domain names and resolve them to IPs.” *Id.*, 67-72. From there, email addresses and documents can be harvested. *Id.*, 73-77. The presentation explains that Maltego can also search blogs, social networks and incoming links. *Id.*, 82. Maltego also provides graphical representations of the mapped network assets, showing them being associated with various entities. *Id.*, 80.



Maltego is further described and shown at Exhibit 1021, which is a 2009 video entitled “Social Engineering Using Maltego.” The video shows how Maltego can be used to discover network assets for a particular domain, as well as additional domains that share a DNS name server.

Thus, as shown above, ordinary artisans had long had experience seeking information from various sources to enumerate network assets and collect information for providing a risk assessment score.

B. Public Availability and Authenticity

McNab and other exhibits discussed in this Petition were publicly available before September 24, 2010, as discussed below. EX1003, ¶¶75-88.

1. McNab

McNab is entitled “Network Security Assessment” and is a book published by O’Reilly Media in 2008. EX1005, 5. According to Dr. Hall-Ellis and her review of library records, McNab was publicly available at the Penrose Library at the University of Denver by November 10, 2008. EX1014, ¶44. She also notes that the Library of Congress created a catalog entry for the book by June 8, 2009. *Id.*, ¶45. Thus, she concludes that McNab was available by 2008, and further concludes that Exhibit 1005 is a true and correct copy as it was published in 2008. *Id.*, ¶¶42, 48.

Dr. Almeroth also agrees that McNab was indexed and made available to those of skill in the art by 2008 based on his personal knowledge and as an expert in the field, because O’Reilly Media is “a reputable publisher such that copyright dates on their publications are a reliable and trustworthy indicator of when the publication was published.” EX1003, ¶¶76-79. Thus, he also concludes that McNab was publicly available by 2008 and that Exhibit 1005 is a true and correct copy as it was published in 2008. *Id.*

2. Internet Archive Pages

Exhibits 1008-1012, 1015, 1017, 1020, and 1022 are webpages and documents obtained from the Internet Archive’s Wayback Machine. According to

Mr. Frank-White, the Wayback Machine archives webpages “by use of web archiving software that automatically stores copies of files available via the Internet, each file preserved as it existed at a particular point in time.” EX1013, ¶4. Each Internet Archive URL contains the date and time that the webpage was archived. *Id.*, ¶5. All of the Internet Archive exhibits are dated before September 24, 2010, and thus indicate that they were publicly available on the Internet before that date, and that the copies attached at Exhibits 1008-1012, 1015, 1017, 1020, and 1022 are true and correct copies of the webpage on the date they were archived.¹

Dr. Almeroth, who is familiar with the Wayback Machine, agrees with Mr. Frank-White and agrees that the Internet Archive exhibits are archives of publicly available webpages at the time they were archived as indicated in the exhibits’ URL, and that those exhibits are true and correct copies. EX1003, ¶¶80-81; *see Valve*

¹ While Mr. Frank-White’s description of the Internet Archive applies to any Internet Archive URL, his declaration at Exhibit 1013 only provides “browser screenshots of the Internet Archive’s records of the archived files for the URLs” for Exhibits 1008-1012 and 1022. EX1013, ¶¶6-7. Petitioner is seeking an additional declaration from the Internet Archive for Exhibits 1015, 1017, and 1020, but had not yet received it at the time of this Petition’s filing.

Corp. v. Ironburg Inventions, Ltd., 8 F.4th 1364, 1374-75 (Fed. Cir. 2021) (Wayback

Machine webpages are “highly persuasive evidence of public accessibility”).

3. Public Videos

Exhibits 1019 and 1021 are printed webpages from Vimeo and YouTube, respectively. Vimeo and YouTube are video-hosting sites, and the URLs for Exhibits 1019 and 1021 link to a video hosted on those sites. Exhibit 1019 is a video of Chris Page presenting the slideshow at Exhibit 1018. Exhibit 1021 is a video providing a demonstration of the Maltego product. Exhibit 1031 are screenshots of relevant portions of the YouTube video at Exhibit 1021.

Dr. Almeroth opines that Vimeo’s indication that Exhibit 1019’s video was uploaded on January 6, 2009 is “generally a reliable indicator of when a video was publicly available.” EX1003, ¶¶82-85. In addition, Exhibit 1020 provides corroboration that the video was available in 2009, since it is a March 2, 2009 archived link of the Vimeo webpage. Thus, Dr. Almeroth opines that the video linked at Exhibit 1019 was publicly available by early 2009 and is a true and correct copy. In addition, Ms. Melissa Nezhnik of Vimeo’s Trust and Safety group confirmed that the video at Exhibit 1019 was “uploaded on January 6, 2009” by the user “carnal0wnage.” Ex. 1029.

Dr. Almeroth also opines that YouTube’s date of September 14, 2009 for Exhibit 1021 is also generally reliable. EX1003, ¶¶86-88. Because the video has

thousands of views, he also opines that it was publicly available by 2009, and that it was indexed by YouTube, like all uploaded videos are by default. *Id.* He also notes that the video itself shows dates from early 2009, providing corroborating evidence of the date of its public availability, and that it is a true and correct copy. *Id.* Metadata about the YouTube video indicates that it was “published[]At” September 14, 2009 and has a “view[]Count” of 60,168 views, thereby further indicating its public availability by 2009. EX1030.

Analogous to the Federal Circuit’s recognition that Wayback Machine webpages are “highly persuasive evidence of public accessibility,” pages from YouTube and Vimeo are likewise persuasive evidence of public accessibility because they are “prominent or well-known among persons of ordinary skill in the art.” *M & K Holdings, Inc. v. Samsung Electronics Co.*, 985 F.3d 1376, 1381 (Fed. Cir. 2021). In addition, the Board has held that the date provided by the YouTube website along with testimonial support from an expert is enough evidentiary support to establish the video’s public availability. *Mission Integrated Technologies, LLC v. Clemente*, IPR2023-01285, Paper 16 at 22-23 (PTAB Feb. 23, 2024).

C. ’524 patent

The ’524 patent to Boyer *et al.* was filed November 18, 2015 and claims priority to a provisional application filed September 24, 2010.

The '524 patent is directed to conventional techniques for measuring the security vulnerability of a network from external observances. Specifically, it can “creat[e] a composite security rating” from data that is “derived from externally observable characteristics.” EX1001, Abstract. The claims of the '524 patent further require displaying such ratings on a map of the assets within the network. *See, e.g., id.*, cl. 1; EX1003, ¶¶68-69.

1. Prosecution History

During prosecution, the applicant amended the claims several times, though failed to overcome the examiner’s rejections. Finally, following an interview with the examiner, applicant added the following limitations: (1) that the entity map is based on determining IP information specifically for “a mail server, an SMTP server, [or] an FTP server,” (2) and the step of “associating the observable cyber-security characterizations with a respective entity based at least in part on the derived server name,” and (3) that the characterizations comprise “a number of technical assets” or “duration of detected malicious activity.” Applicant noted that while one of the references, Wanser, “appears to describe domain name naming conventions in general,” it did not describe it specifically for “a mail server, an SMTP server [or] an FTP server,” nor disclose attributing events to a particular entity.

The examiner initiated an interview and proposed an amendment where the characterizations would comprise both “a number of technical assets” and “duration

of detected malicious activity.” Following these amendments, the examiner allowed the claims. EX1003, ¶¶70-72.

VI. CLAIM CONSTRUCTION

The claim terms should be given their plain and ordinary meaning under *Phillips*. EX1003, ¶74.

VII. GROUNDS

A. Ground 1: McNab and McGovern

McNab discloses the creation of a map of assets for an organization’s networks/entities, and furthermore discloses various methods for assessing the security of those network assets, including the performance of vulnerability scans from outside the networks.

To the extent PO argues that McNab does not disclose determining the duration of malicious activity, McGovern discloses making such a determination. EX1003, ¶¶89-99.

1. Motivation to Combine

(a) McGovern

A POSITA would have found it obvious to combine McNab’s network enumeration and security scanning with McGovern’s teachings of determining the duration of a malicious activity as well as its teachings of quantifying cyber-security observations into a numerical score to be displayed in a web-based user interface. EX1003, ¶¶91-99.

A POSITA would be motivated to incorporate such teachings because they were known to increase the security of a network as well as increase user-friendliness of the system and provide the user an easier assessment of the network. Simplifying the display of security information would be of paramount interest to a POSITA as it would aid in identifying and addressing security breaches and other security issues within the user's network.

McGovern discloses the capability of determining the “average response or patch time after security breach detection.” EX1007, ¶38. A POSITA would be motivated to modify McNab to include this functionality in McNab's security assessment because the duration of a security breach is an important factor in determining overall network security. *Id.* A security breach is a known, highly dangerous event that puts all of the organization's data and resources at risk of damage or theft or liability, such as theft of confidential information or customer/employee information. *Id.* Thus, a POSITA would be motivated to minimize the duration of any security breach—and it would be beneficial for a network administrator to know whether the network and the organization's IT policies are robust and proactive to such security breaches (resulting in a minimal duration) or if they are particularly vulnerable (because of a lengthy duration). EX1003, ¶93.

McGovern also discloses the capability of assigning “a numerical score” to “each risk level or degrees of compliance (associated with each security parameter)” following a network assessment. EX1007, ¶37. McGovern likewise teaches that the score is “incorporated into a software program with a user interface,” which can be “designed to run as a Web-based service or application” wherein users can “access the scoring and related functionalities remotely via standard browsers or similar user interfaces.” EX1007, ¶¶42, 45.

A POSITA would be motivated to incorporate these teachings into McNab because it provides an easier and more efficient method of evaluating a network’s security assessment, improving usability and security. EX1003, ¶95. McGovern expressly teaches that the score is an important factor in providing an “overall risk assessment.” EX1007, ¶37. McGovern also teaches that the numerical scores that partly make up the ISA score reflects “degrees of compliance with security standards” and “goodwill and negative publicity” of the entity to which the network belongs. *Id.* These teachings would motivate a POSITA to use McGovern’s ISA scores to provide beneficial information about compliance and publicity. A POSITA would understand that displaying a score in a UI increases ease of identifying network risks, as it takes less effort and time to evaluate compared to a report. EX1003, ¶95. As a network’s assets increase, an network administrator will need access to quick and efficient evaluations. *Id.* A POSITA would also be further

motivated by McGovern’s teachings of a remote user interface because this allows network administrators to access the scores from anywhere, increasing ease-of-use and accessibility. Thus, a POSITA would be motivated to incorporate such teachings in order to increase security and usability. EX1003, ¶95.

A POSITA would have combined McNab and McGovern with a reasonable expectation of success. EX1003, ¶96. The majority of McNab is dedicated to explaining various methods for performing network intrusion tests. *See* EX1005, Chapters 5-16. Adding the duration of a security breach would simply be another test that an administrator could perform. McNab does not state any limit on the number of tests that could be performed and adding an additional test would not pose any hurdle in computing power or evaluation time. *Id.*

A POSITA would have also expected success in combining McNab with McGovern’s numerical score. EX1003, ¶97. McGovern provides an algorithm for calculating the numerical score that could be applied to any of the tests described by McNab, including the Nessus vulnerability scanning software. Indeed, McNab discloses that Nessus already “categorizes findings into three severity levels: security hole, security warning, and security info.” EX1005, 390. Quantifying such levels (such as assigning each level a particular number like 1, 2, and 3) would have been a trivial task well-within the capabilities of a POSITA.

In addition, a POSITA would understand that implementing McGovern's measurement of "response or patch time after security breach detection" in McNab's system would have been well-within the capabilities of a POSITA. In particular, as discussed in fuller detail below, McNab describes a Nessus vulnerability scanner which can detect whether assets were infected by malicious viruses. McNab also discloses that Nessus supports "the ability ... to compare two reports for differences." EX1005, 391. Thus, given these teachings, a POSITA would have understood that McGovern's "response or patch time" could be measured by evaluating reports by the Nessus vulnerability scanner until the asset is no longer infected and determining the duration it was infected. EX1003, ¶98.

Finally, McNab and McGovern are analogous art to the '524 patent because all three references are directed to the field of network security. EX1005, xvi; EX1007, Abstract; EX1001, Abstract; EX1003, ¶99.

(b) Nessus

As discussed herein, McNab discloses in detail a third-party vulnerability scanner called Nessus. McNab discloses that documentation about Nessus can further be found on the website "<http://www.nessus.org/documentation>". EX1005, 379. McNab further describes that Nessus supports plug-ins which are provided on the website. *Id.*, 377. Thus, documentation and plug-in information about Nessus

as described in Exhibits 1008-1012 are incorporated into McNab by reference. EX1003, ¶100.

To the extent PO argues this was not an incorporation by reference, it would have been obvious to combine Exhibits 1008-1012 with McNab. Exhibits 1008-1012 are archived webpages (and all but Exhibit 1011 are from the Nessus website) that provide additional detail about the Nessus product described in McNab, and are expressly referenced by McNab. Thus, a POSITA would be motivated to include such additional functionality, such as the plug-ins which detect malicious software such as MSBlaster and So.Big (Exs. 1008-1009) or documentation that describes user interface functionality for Nessus (Exs. 1010-1011), because they describe improved security capabilities and displays and based on McNab's express teaching to do so. EX1003, ¶101. A POSITA would have a reasonable expectation of success because these documents describe the Nessus product and therefore do not require any modifications to the product itself. *Id.*

2. Claim 1

(a) 1[pre], 1[a]

McNab discloses several methods for performing external security tests on networks of entities, beginning by maintaining a database of an entity map that maps technical assets to the entities of a particular organization (or multiple organizations). EX1003, ¶¶102-19. One such test is called "Forward DNS

Grinding” which determines IP information by using mail server and SMTP server naming conventions.

“entity map that maps technical assets to respective companies or other entities”: The ’524 patent describes an “entity map” as “map[ping] data back to an entity,” such as mapping an IP address to an entity. EX1001, 6:15-17.

McNab discloses generating an **entity map** by discovering and mapping technical assets for a respective entity. EX1003, ¶¶104-07. For example, McNab generally “tackles one single area of information security in detail: that of undertaking IP-based network security assessment in a structured and logical way.” EX1005, xvi. One method of security assessment is described in Chapter 3, called “Internet Host and Network Enumeration,” which explains how to “**map** your network, from open web searches to DNS sweeping and querying of authoritative name servers.” EX1005, xx. This network mapping is performed externally, as if done by an “attacker.” *Id.*, 17; EX1003, ¶105.

McNab discloses enumeration by “querying open sources for information relating to the target organization and its networks.” EX1005, 17 These include: “Domain and IP WHOIS registrars,” “Border Gateway Protocol (BGP) looking glass sites and route servers,” “Public DNS servers,” and “DNS querying and grinding against specific name servers.” *Id.* Such information is used to “build a useful **map**

of your networks and understand where potential weaknesses may lie.” *Id.*; EX1003, ¶106.

McNab explains that these queries results in obtaining *assets*. EX1005, 17. For example, McNab discloses that “Internet Host and Network Enumeration” techniques discussed in Chapter 3 involve “obtain[ing] useful data about *the structure of the target* network” by gathering “details of Internet-based network blocks, internal IP addresses gathered from DNS servers, insight into the target organization’s DNS structure (including *domain names, subdomains, and hostnames*), and details of relationships between physical locations.” *Id.*, 5; *see also id.*, 9 (“[T]he network enumeration process ... identif[ies] further network components.”). For example, by using “DNS querying,” one can “enumerate hostnames and sub-domains” within a network. *Id.*, 40. One method McNab describes is “Forward DNS Grinding,” which is a “forward brute-force grinding attack ... to enumerate valid DNS address records and aliases relating to the domain and its hosts,” such as SMTP mail servers of a domain. *Id.*, 35-36. A POSITA would understand that network components, hostnames, and sub-domains such as SMTP mail servers are *assets associated with the entity*. As the ’524 patent explains, “IT assets” include “IP addresses controlled by the entity and the associated activity of those IP addresses.” EX1001, 9:65-10:1; EX1003, ¶107.

“companies or other entities with which the assets are associated”: McNab

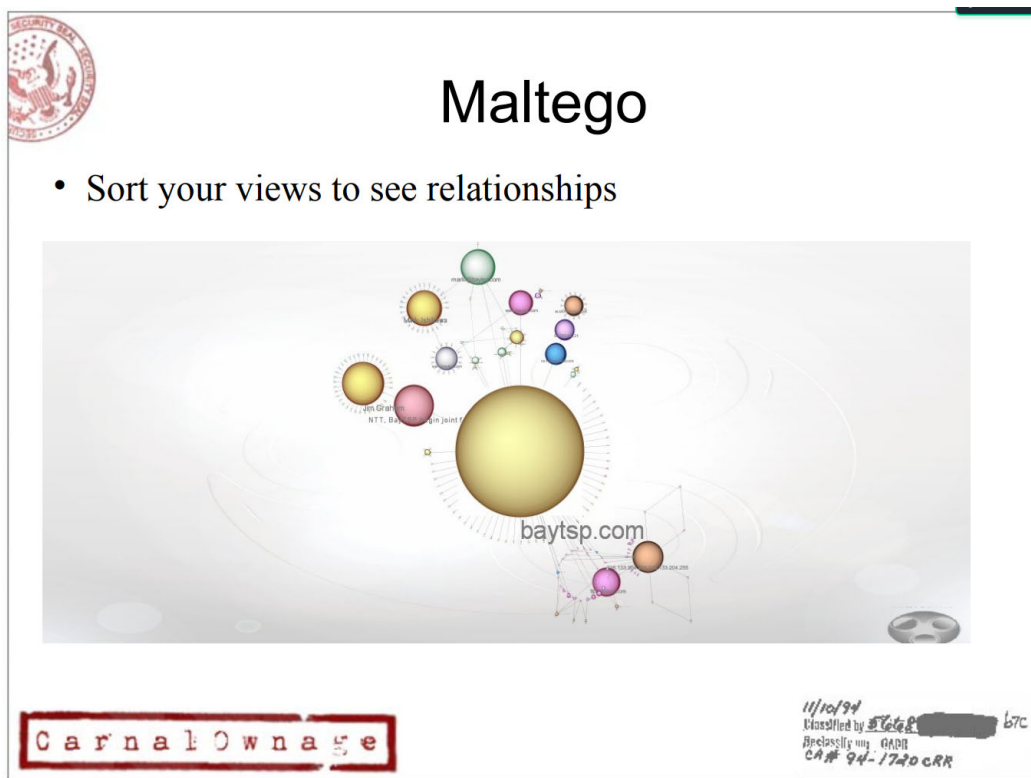
discloses that its techniques for building an ***entity map*** that maps assets to an organization is done with respect to the assets of multiple organizations. EX1003, ¶¶108-111.

McNab discloses that its techniques involve gathering “information relating to the target organization and its networks.” EX1005, 17. McNab does not limit its techniques to a single organization and a POSITA would understand they could be used on any number of organizations. For example, McNab provides examples of mapping technical assets for different organizations: Nintendo of America (EX1005, 24-25), the CIA (*id.*, 34-35), and the Bank of England (*id.*, 35-36). McNab also discloses that its techniques can be used on “companies and organizations.” *Id.*, 5. Any domain name associated with any organization could be used to enumerate assets. EX1003, ¶109. For example, Forward DNS Grinding using the “TXDNS” tool will “enumerate valid DNS address records and aliases relating to the domain and its hosts,” including ***any*** domain. Ex. 1005, 35-36; EX1003, ¶109.

McNab also describes that as “New domain names and IP addresses” are discovered, network enumeration is performed again. EX1005, 8. Consistent with the specification of the ’524 patent, a POSITA would understand that the plain meaning of “entity” includes a computer system associated with a “domain name” (such as bankofengland.co.uk). For example, the ’524 patent describes that

“[m]apping may also include *a domain name* associated with the third party computer system,” the third party computer system being an “entity.” EX1001, 2:35-36, 5:57-58 (“an entity (e.g., a third party computer system)”); *see also id.* 9:4-9 (“It is also possible to query the [RIR] database by an *entity name* and get a list of IP addresses allocated to that *entity*.”). Thus, McNab discloses that as new domain names are discovered (*i.e.* new *entities*), it will discover and map assets to those new entities. EX1003, ¶110.

In addition, as discussed above in the Technical Background, numerous tools were available that generated mappings for multiple entities. EX1003, ¶111. Maltego is one such exemplary tool that would discover and map assets to new entities as they were discovered. EX1018, 67-80.



“maintaining in a database an entity map”: McNab discloses network enumerations are “gathered” and stored, because such network enumeration information is “used to perform structured bulk network scanning and probing exercises to further assess the target network space and investigate potential vulnerabilities.” EX1005, 5. McNab also describes a “Cyclic Assessment Approach” where the mapping is updated as “New domain names and IP addresses” are discovered. *Id.*, 8 (“[A]n analyst may identify previously unknown IP address blocks, which can then be fed back into the network enumeration process to identify further network components.”). A POSITA would therefore understand that the entity map must be stored. EX1003, ¶112.

A POSITA would have considered it obvious to store the entity map in a database. EX1003, ¶113. It was well-known that databases were designed to efficiently store and query data, and by 2010, it was common for enterprises to store all manner of data in databases—even the smallest organizations had databases to store information about IT assets such as network users. *Id.* A POSITA would have been motivated by a database’s speed and space-efficiency. A POSITA would have also been familiar with storing and retrieving data in a database, and would have had a reasonable expectation of success in storing McNab’s entity map into a database. For example, it would have been well-within the capabilities of a POSITA to store list of entity assets, including their hostnames and IP addresses, in a database table,

as well as a list of entities to be mapped to such assets. A POSITA would have been familiar with the necessary database operations to store such data. *Id.*

“wherein mapping technical assets to respective companies or other entities comprises determining IP information by annotating a domain name with naming conventions for at least one of a mail server, an SMTP server, and an FTP server, thereby deriving a likely server name having assigned IP addresses”: McNab discloses that generating an entity map includes a method called “Forward DNS Grinding” where a domain name is annotated with a “dictionary of common mail server names” to determine the IP address of a mail server or SMTP server. EX1005, 35; EX1003, ¶¶114-19.

Specifically, McNab discloses that the “map of your networks” includes information from “DNS querying and grinding against specific name servers.” EX1005, 17. “Forward DNS Grinding” refers to a “forward brute-force grinding attack ... to enumerate valid DNS address records and aliases relating to the domain and its hosts.” *Id.* “Forward DNS Grinding” involves appending conventional prefixes to a chosen domain name, and attempting DNS resolution for that name. If the asset exists, the DNS resolution will resolve its IP address. This is precisely what has been claimed in this limitation. EX1003, ¶115.

McNab discloses the use of a “very effective forward DNS grinding tool” known as TXDNS. McNab provides an example of a grinding attack against the

“bankofengland.co.uk domain, using a small dictionary of common mail server names.” EX1005, 17. This results in the discovery of three “SMTP mail servers”: “mail.bankofengland.co.uk,” “mail2.bankofengland.co.uk,” and “mailhost.bankofengland” and their IP addresses. *Id.*, 36. 95 other attempted domain names failed. EX1003, ¶116.

Later in McNab, it provides another example of using TXDNS against the “apple.com” domain using a different dictionary file “smalllist.txt” that reveals the IP address of an FTP server “ftp.apple.com,” in addition to other servers. EX1005, 85-86.

Example 5-6. Forward DNS grinding using txdns.exe under Windows

```
C:\tools> txdns -f smalllist.txt -s 17.254.0.50 apple.com
```

```
-----  
TXDNS (http://www.txdns.net) 2.0.0 running STAND-ALONE Mode  
-----
```

```
> ftp.apple.com           - 17.254.16.10  
> guide.apple.com        - 17.254.12.37  
> help.apple.com         - 17.254.3.26  
> mercury.apple.com      - 17.250.248.40  
> research.apple.com     - 17.255.4.30  
> search.apple.com       - 17.254.0.160  
> vpn.apple.com          - 17.252.68.41  
> webmail.apple.com      - 17.254.13.52  
-----
```

```
Resolved names: 8  
Failed queries: 348  
Total queries: 356  
-----
```

Thus, McNab discloses discovering *assets* by *annotating a domain name* (such as bankofengland.co.uk or apple.com) *with naming conventions for at least*

one of a mail server, an SMTP server, and an FTP server (such as the mail or ftp prefix) to *derive a likely server name* which can be used to *determine its IP address*.

EX1003, ¶119.

(b) 1[b]

McNab discloses a vulnerability scanner called Nessus that runs on the Internet and uses multiple sensors to collect externally observable cyber-security characterizations of each entity's technical assets. EX1003, ¶¶120-30.

“automatically using [sic]”: McNab discloses that external observations are automatically performed after entity mapping. Specifically, McNab discloses a “Network Security Assessment Methodology” where after the entity mapping described above (described as “Network reconnaissance to identify IP networks and hosts of interests” or “Internet Host and Network Enumeration”), the methodology performs “[b]ulk network scanning and probing to identify potentially vulnerable hosts” and “[i]nvestigation of vulnerabilities.” EX1005, 4-5. A POSITA would understand that such steps are performed automatically. EX1003, ¶121.

One of the tools McNab discloses for vulnerability scanning is called Nessus. See EX1005, 377-392. McNab describes Nessus as an *“automated* system[]” for “vulnerability scanning” that requires “minimal hands-on qualification and assessment of vulnerabilities.” *Id.*, 3. Nessus was “a free vulnerability scanner that can be used to perform a number of network-wide bulk security check.” *Id.*, 377.

Although Nessus is described in Chapter 15 of McNab, a POSITA would have understood that its disclosures are part of the same “Network Security Assessment Methodology” described in Chapter 1, which also includes the entity mapping described above. *See id.*, 4-5. In addition, McNab discloses that Nessus is part of the “Network Security Assessment Platform” and is one of the “components and tools that make up a professional security consultant’s toolkit for performing tasks including reconnaissance, network scanning, and exploitation of vulnerable software components.” *Id.*, 10, 12-13. In the alternative, a POSITA would have considered it obvious to combine the teachings in these chapters, not least because the McNab reference teaches that Nessus is one of the network assessment tools to be performed on assets discovered from entity mapping. *Id.*, 4-5, 10, 12-13; EX1003, ¶122. A POSITA would have also been motivated to use Nessus after entity mapping, since McNab instructs that “network scanning and probing to identify potentially vulnerable hosts” and “[i]nvestigation of vulnerabilities” should be performed after entity mapping, and McNab teaches that Nessus is an effective tool for performing such scans. EX1005, 4-5, 377. A POSITA would have had a reasonable expectation of success in combining these chapters in McNab because they would simply involve using off-the-shelf tools in their known, conventional manner. EX1003, ¶122.

McNab discloses that Nessus performs a vulnerability scan *automatically* because it “significantly [reduces] the amount of time spent during a penetration test

performing manual checks.” EX1005, 377. McNab also discloses that Nessus can be “start[ed]” automatically “by using an operating system-appropriate startup file.” *Id.*, 382. Finally, McNab also discloses a “command-line client for scanning,” and POSITA would have understood that a command line can be automated by using any number of well-known techniques, such as the “cron” command in UNIX or “Task Scheduler” in Windows. A POSITA would have considered it obvious to use the “cron” or “Task Scheduler” utility to automate performing scans using Nessus after network enumeration, as that would provide the well-known benefit of allowing for regularly scheduled vulnerability scans to ensure network assets are not vulnerable. EX1003, ¶123. Indeed, it was well-known that vulnerability scanning was often performed automatically after network enumeration, such as described in the NIST paper wherein “Automated Scanning Tool[s]” are used for “network mapping” and then performing vulnerability scanning. EX1016, 11-12, 15. A POSITA would have had a reasonable expectation of success because “cron” and “Task Scheduler” were utilities a POSITA would have been familiar with. EX1003, ¶123.

“sensors on the Internet to collect externally observable cyber-security characterizations of the technical assets that have been mapped to each of the entities”: McNab discloses that Nessus uses multiple sensors on a server on the

Internet to collect externally observable cyber-security characterizations of target assets mapped to entities. EX1003, ¶¶124-128.

McNab discloses that Nessus is “a free *vulnerability scanner* that can be used to perform *a number of network-wide* [*i.e. multiple sensors*] bulk security check.” EX1005, 377. The Nessus client “configures the various target, scanning, and plug-in options, and it reports the findings from the scan to the user.” *Id.* These configurations and plug-ins are each a *sensor* that collects the *cyber-security characterizations*, such as those disclosed in the figures below. *Id.*, 388. EX1003, ¶125.

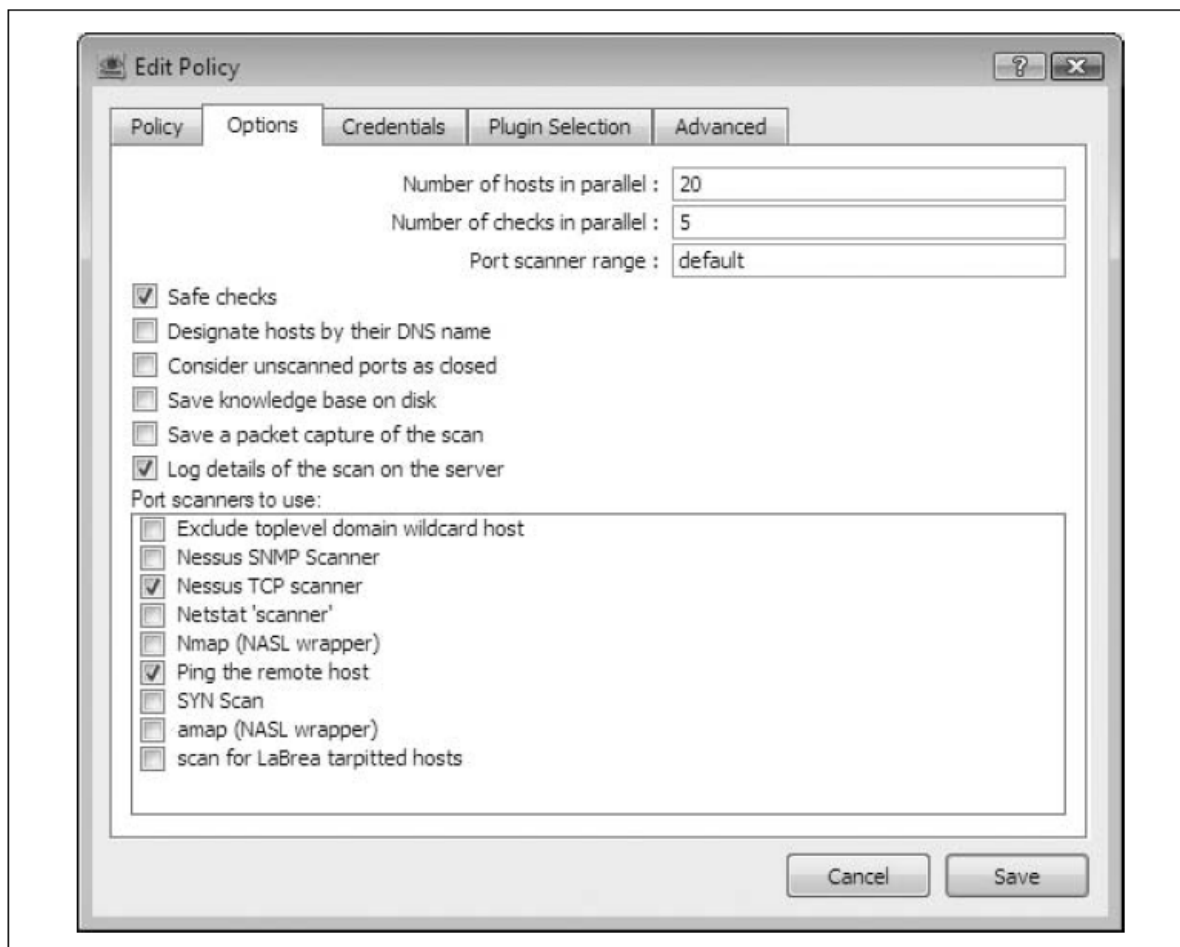


Figure 15-2. NessusClient 3 default scan policy options

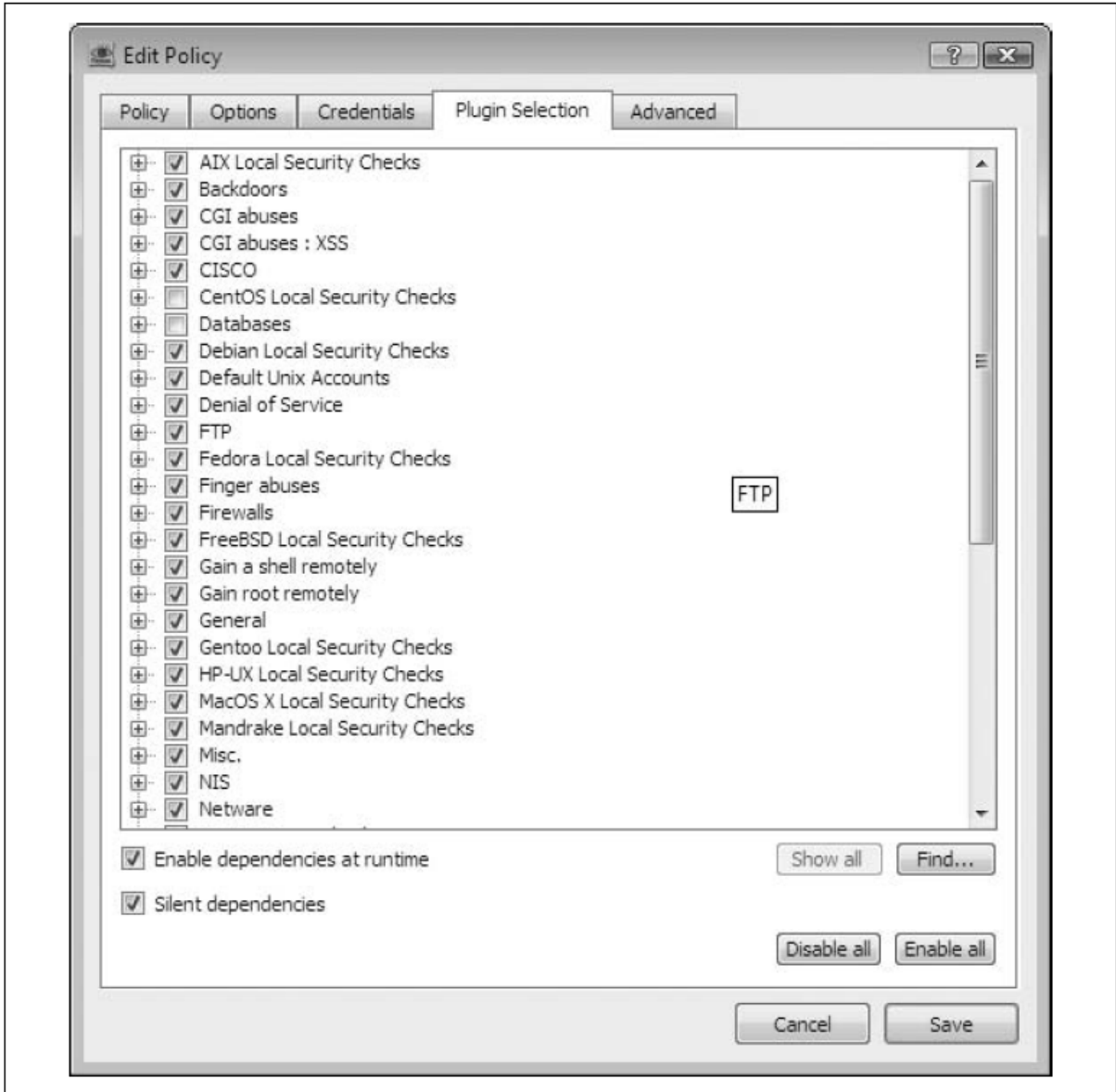


Figure 15-3. NessusClient 3 plug-in selection window

McNab discloses that Nessus collects *externally observable* cyber-security characterizations because it “can be run in two different modes: with or without authentication credentials” and “[w]hen run *without credentials*, Nessus will perform *remote network-based security checks*, testing how the target host responds to specific network probes.” EX1005, 388. McNab further discloses that Nessus

performs its scans using a server on *the Internet*. Specifically, the “Nessus server performs all of the scanning and security checks” and is “Internet-connected.” *Id.*, 377, 383; EX1003, ¶126.

Some of the characterizations that can be performed by Nessus via plug-ins include detecting whether “[t]he remote host [is] infected by the MS Blaster worm” (EX1008) or “check[ing] for the presence of different virii on the remote host” (EX1009). These exhibits are webpages from Nessus’s website archived in 2006 describing certain plug-ins supported by Nessus. EX1003, ¶127.

Finally, McNab discloses that Nessus collects the *cyber-security characterizations* for the *technical assets that have been mapped to each of the entities*. Specifically, McNab discloses that Nessus performs its scans on “target network addresses or hosts” using “specific hostnames or IP addresses, ranges of network addresses, or through supplying a file with a target host on each line.” EX1005, 385. As discussed above, McNab discloses (or renders obvious) that Nessus is part of the Network Security Assessment Methodology, where Nessus is performed on hosts identified through entity mapping, *i.e. technical assets that have been mapped to each of the entities*. *Id.*, 4 (“Bulk network scanning and probing to identify potentially vulnerable hosts”), 3 (“Vulnerability scanning uses automated systems (such as Nessus ...)”); EX1003, ¶128.

“and associating the observable cyber-security characterizations with a respective entity based at least in part on the derived server name”: McNab discloses that the Nessus vulnerability scanner associates its cyber-security characterizations with a particular entity based on the derived server name. EX1003, ¶¶129-30. The term ***derived server name*** has an antecedent basis to the server name that was derived by “annotating a domain name with naming conventions for ... a mail server, an SMTP server [or] an FTP server,” as recited in limitation 1[a].

Specifically, McNab discloses that Nessus requires “target network addresses or hosts to be scanned,” which can be specified using “specific hostnames.” EX1005, 385. As discussed above, McNab discloses (or renders obvious) that Nessus is part of the Network Security Assessment Methodology, where Nessus is performed on hosts identified through entity mapping, including the derived hostnames identified with “Forward DNS Grinding.” The Nessus scan is then performed on an entity based on the chosen hostname, resulting in the collection of ***observable cyber-security characterizations*** associated with that hostname belonging to an entity (*e.g.* the hostname mail1.bankofengland.co.uk belonging to the entity Bank of England). *Id.*, 389-90. McNab then generates a report, indicating that the characterizations were associated with the entity. *Id.*, 390-91.

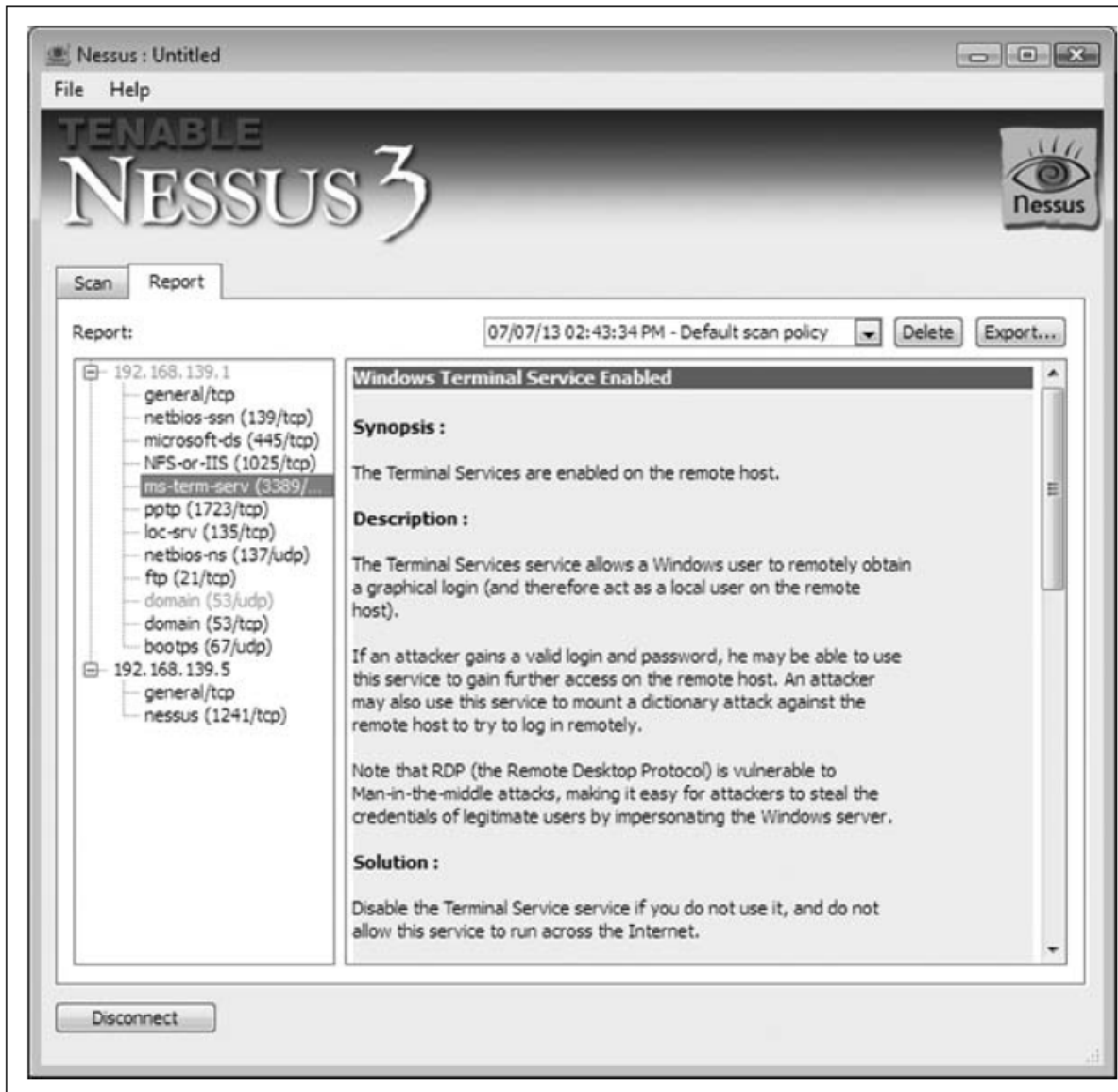


Figure 15-6. NessusClient 3 completed scan

(c) 1[c]

“automatically deriving observations about the technical assets from the collected cyber-security characterizations: McNab discloses deriving observations from the cyber-security characterizations. EX1003, ¶¶131-35. These observations include technical assets that have been reported to be malicious. In addition,

McGovern discloses deriving observations such as “average response or patch time” following a security breach detection, which indicates duration of detected malicious activity. As discussed above, it would have been obvious to combine McNab with McGovern.

McNab itself discloses that the Nessus vulnerability scan results in reports (*i.e. derived observations*) from the *cyber-security characterizations*, such as characterizations about whether a host is “infected by the MS Blaster worm” or some other virus. EX1008-1009. From these characterizations, Nessus generates a report that provides information about those characterizations, such as the one displayed in the below figure. EX1005, 390-91.

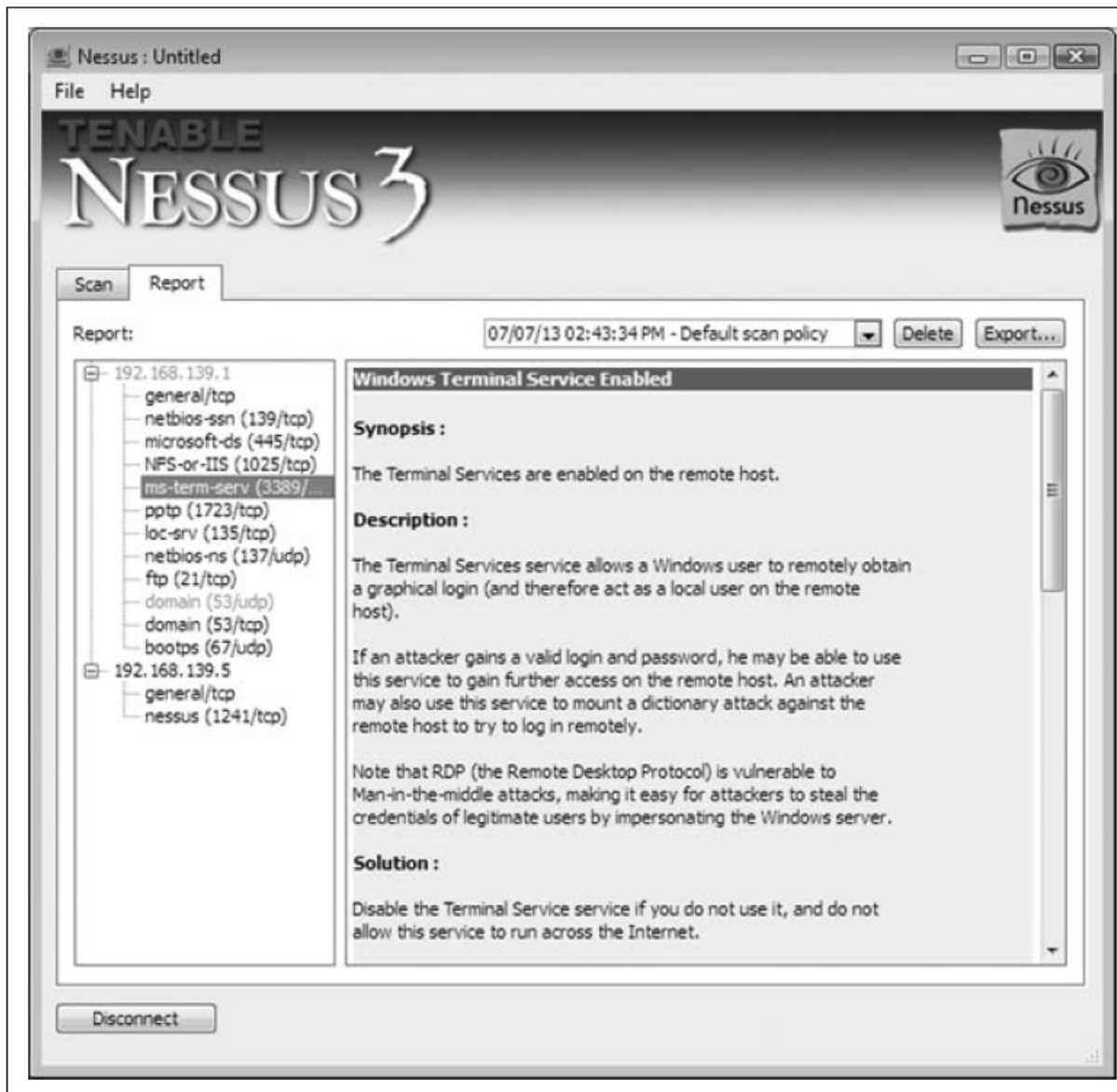
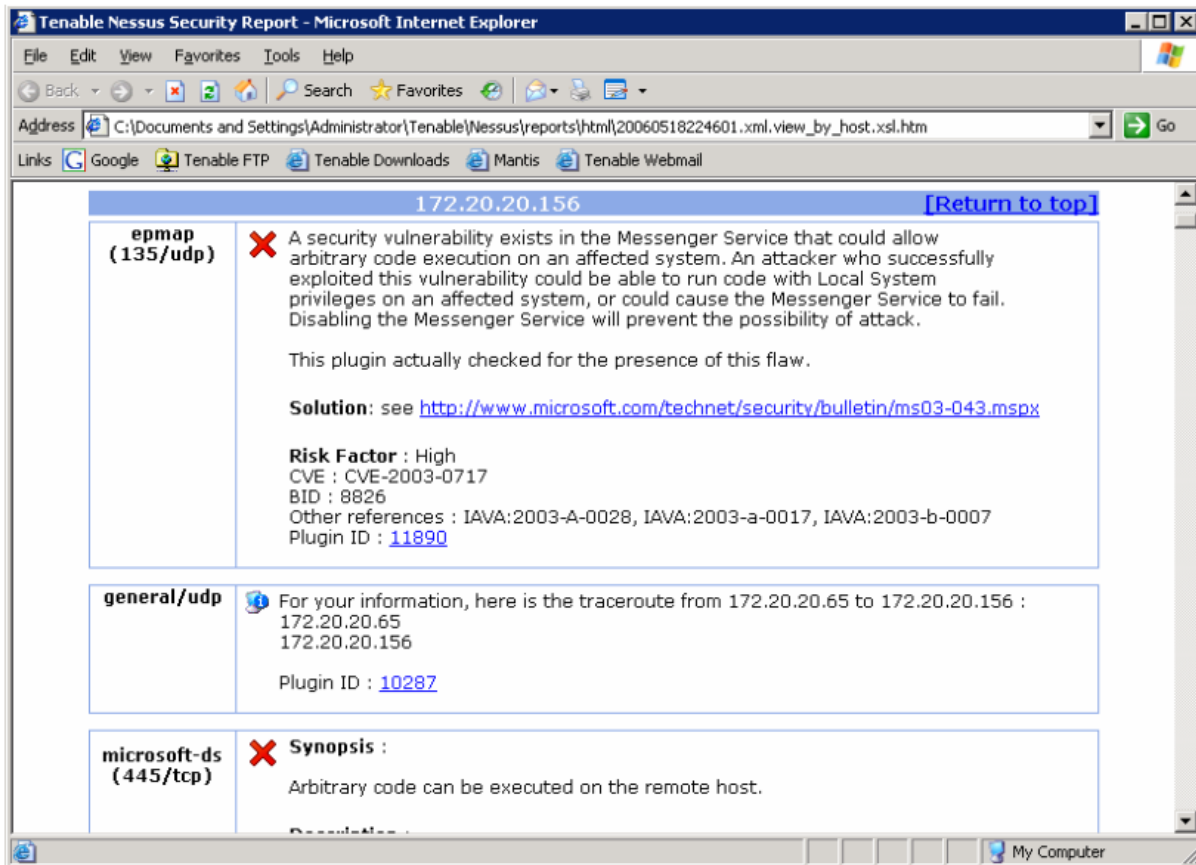


Figure 15-6. NessusClient 3 completed scan

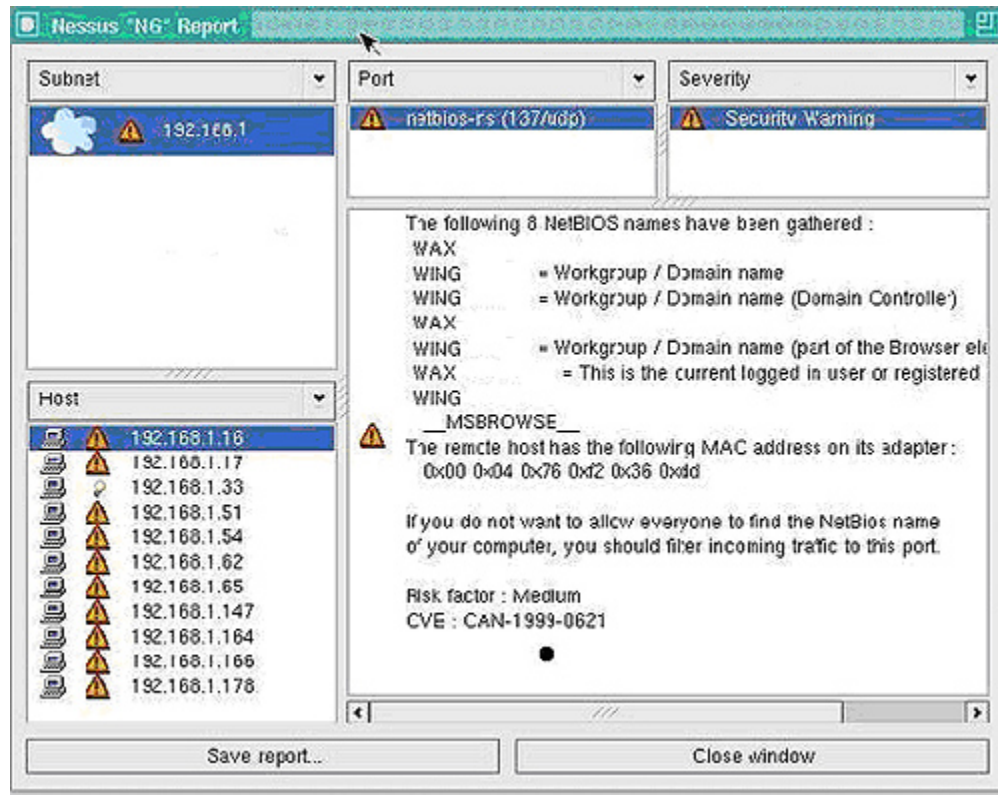
While McNab describes that Nessus “categorizes findings into three security levels,” it does not provide an example of how Nessus depicts these security levels. However, webpages that were archived in 2006 and 2007 do provide such examples. EX1003, ¶¶134-35. For example, Exhibit 1010, which is the Nessus-published document “Nessus 3.0 Client Guide,” archived on March 12, 2007 from Nessus’s

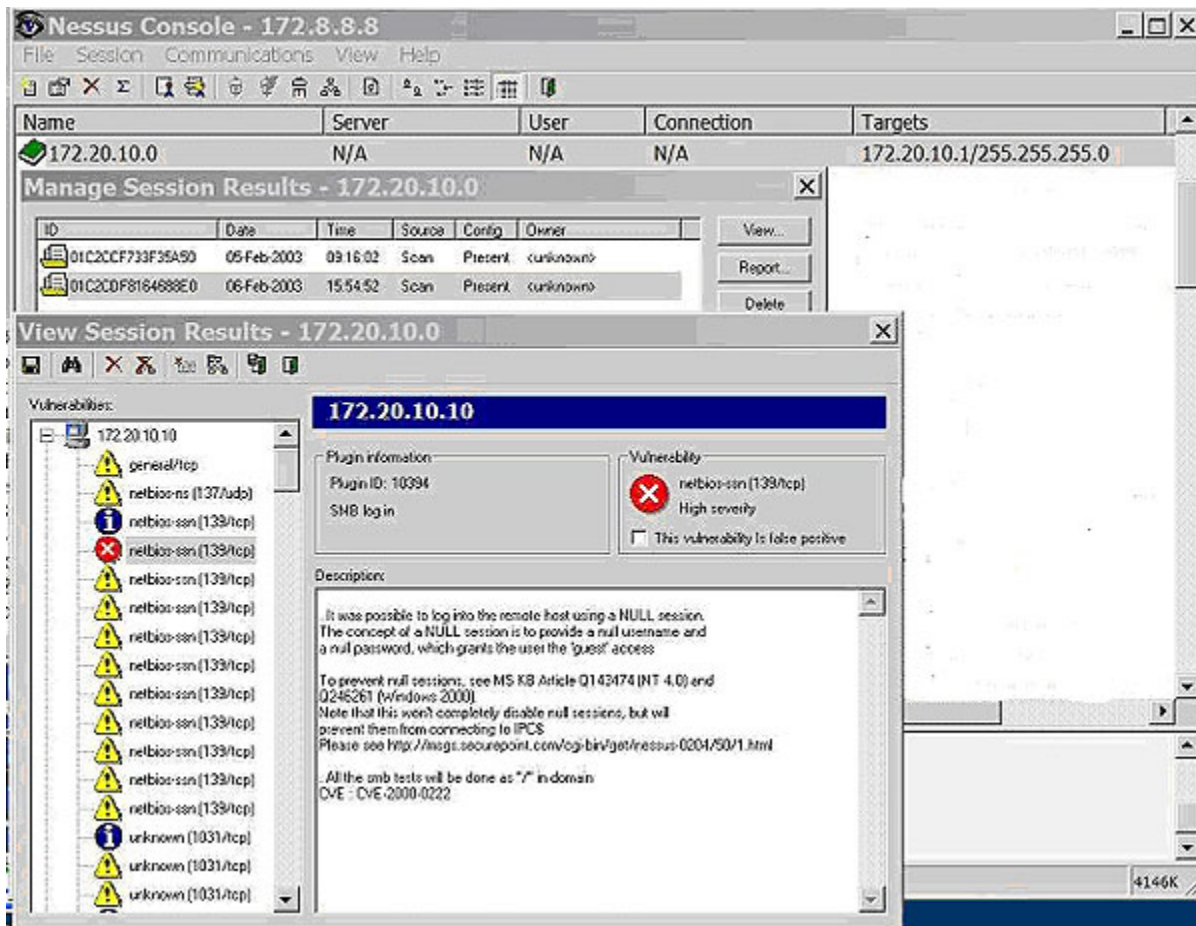
website, shows an example of a report being viewed a web browser for a specific host (*i.e.* 172.20.20.156). EX1010, 8. As can be seen, the report provides icons for two of the “security levels,” *i.e.* **derived observations**. *Id.* One of them is a red “X” because the “epmap” scan returned a high risk factor, the other is an informational icon from the “general/udp” scan merely returning “information.” *Id.*



Similarly, Exhibit 1011 is an article on securityfocus.org that was archived in 2006 and linked from the Nessus website (*see* EX1012), titled “Nessus, Part 3: Analysing Reports.” The article provides several examples of how the “three security levels” are depicted as icons. In Figure 1, each host is associated with an

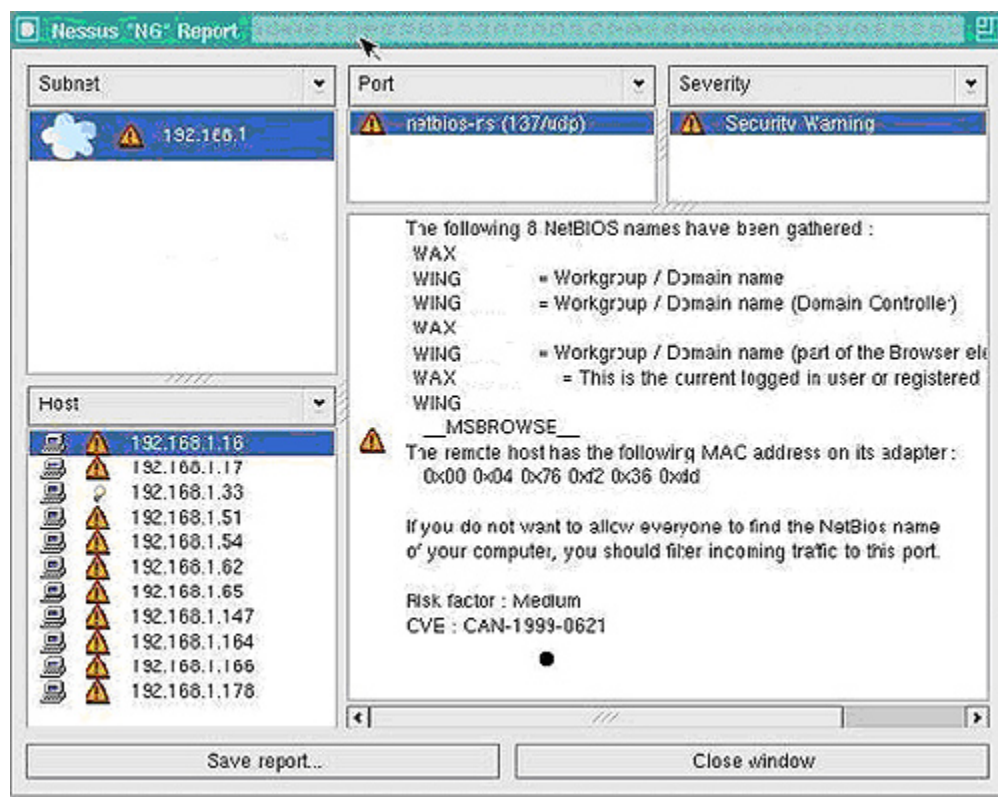
icon indicating the “Severity” of the security risk, for example, a severity of “Security Warning” has an icon of a yellow triangle with an exclamation mark. In Figure 2, there is a red “X” icon for “High severity.”





“wherein the derived observations comprise (i) a number of technical assets that have been reported to be malicious”: As discussed above for limitation 1[b], Nessus provided plug-ins that could detect the presence of certain viruses on the target host. *See* EX1008-1009. If detected, these plug-ins will report a “Risk factor” of “High” or “Critical.” *Id.* These viruses cause the infected host to be malicious. *See, e.g.,* EX1022 (“[T]he worm will cause compromised systems to initiate a denial of service attack against windowsupdate.com.”). Thus, these plug-ins provide *derived observations* indicating that a technical asset is *reported to be malicious*. EX1003, ¶¶136-139.

While Nessus does not display the *number of malicious assets*, since the Nessus client user interface provides readily-observable icons indicating high severity for each host that is reported to be malicious, the presence of those icons indicate the *number of malicious assets*. EX1003, ¶137. For example, the below screenshot indicates that 10 hosts (*i.e. assets*) have a risk level of “Security Warning.”



If those hosts were detected to have viruses (*i.e. reported to be malicious*), then a POSITA would understand they would have red “X” icons indicating that they are malicious. This is especially the case if the administrator decides to only

run virus scans on the hosts (and not other scans that also may result in risk factors of high or critical). EX1003, ¶138.

In the alternative, it would have been obvious to derive the ***number of assets reported to be malicious***. Since Nessus already reports which assets are malicious, a POSITA would have found it obvious to collect that information and derive the number of said malicious assets. A POSITA would be motivated to do this because deriving such a number would allow the number to be displayed to the administrator, thereby giving the administrator an easily-accessible number to indicate the health of the network. Deriving the number would also allow the system to alert the administrator if the number exceeds zero. A POSITA would have a reasonable expectation of success in doing this, because since Nessus discloses that its reports can be exported to XML or a database (EX1005, 391), it would be well-within the skill of a POSITA to write a script that takes the XML or database report information and derives the number of assets reported to be malicious by checking the status of the virus detecting plug-ins for each host. The script could further email said number to the administrator. EX1003, ¶139.

“and (ii) a duration of detected malicious activity associated with the technical assets”: McGovern discloses identifying and collecting “security parameters,” *i.e.* ***cyber-security characterizations***, and evaluating “risk levels or degrees of compliance may be established for each security parameter” (***deriving***

observations from the collected cyber-security characterizations). EX1007, ¶35.

McGovern further discloses that the risk levels can be qualitatively, quantitatively, binary, or even have more than two values. *Id.*; EX1003, ¶¶140-142.

McGovern discloses that one security parameter is the “average response or patch time after security breach detection,” which a POSITA would understand to be the *duration of detected malicious activity associated with the technical assets* because it indicates how long on average for the technical assets the malicious activity occurred before it was addressed. EX1007, ¶38. An average response time includes one or more durations of measured response times. EX1003, ¶141.

As discussed above, a POSITA would have found it obvious to combine McNab with McGovern, and in particular, by using Nessus’s ability to generate reports as to whether an asset has been infected by malicious viruses and “compare two reports for differences.” EX1005, 391; EX1003, ¶¶140-42.

(d) 1[d]

“automatically generating a cyber-security rating for each of the entities using the entity map and the derived observations”: McGovern discloses generating a score based on the observations, and it would have been obvious to combine McNab and McGovern. EX1003, ¶¶143-46.

McGovern discloses that “for each risk level or degrees of compliance (associated with each security parameter)” (*the derived observations*), “a numerical

score may be assigned” where such numerical score is used to “quantify the contribution of each security parameter to ultimately reach an overall risk assessment” (*cyber-security ratings*). EX1007, ¶37. Specifically, “a higher score reflects a greater risk exposure” and is “determined by the evaluation of underlying assets,” such as “degrees of compliance with security standards.” *Id.* This numerical score along with “the security parameters” and “risk levels” can further be “organized into one or more data structures” from which “to calculate information security assessment (ISA) scores ... concerning *an information system*,” *i.e.* an *entity*. *Id.*, ¶42; EX1003, ¶144.

Because McGovern discloses that a security assessment score is generated for the entire entity based on the security assessments of the assets, a POSITA would understand that, in view of McNab’s disclosure of entity maps that map assets to different entities, each entity would be given an numerical score based on the assets mapped to it in the McGovern-McNab combination. EX1003, ¶145. A POSITA would be motivated to do this in order to provide to the administrator security assessment scores based on each entity’s network, since each entity would likely have different networks wherein evaluation of the particular entity’s network would provide useful information. *Id.*

As discussed above, a POSITA would have been motivated to combine McNab and McGovern such that the results of Nessus’s security reports (*i.e. derived*

observations) would be used in McGovern’s security assessment score. EX1003, ¶146. While McGovern does not expressly describe how to generate such a score, since Nessus describes its security reports as being categorized into “three severity levels,” each severity level would be given different weights within the security assessment score. *Id.*

(e) 1[e]

“through a communication network, exposing to users in a user interface of a portal, information about the cyber-security ratings of the entities that is useful to decisions about entities with which to do business”: McGovern discloses that the security assessment scores, *i.e. cyber-security ratings of the entities*, “are recorded and organized into one or more [scoring] data structures.” EX1005, ¶41. Such data structures “may be incorporated in a software program ***with a user interface***” and the software can be either a “stand-alone, self-contained package” or “designed to run as a Web-based service or application” wherein users can ***“access the scoring*** and related functionalities ***remotely via standard browsers*** or similar ***user interfaces***.” EX1007, ¶¶42, 45.

A POSITA would understand that the security assessment score is ***information useful to decisions about entities with which to do business***. EX1003, ¶¶147-48. As McGovern explains, the ISA score is “a composite score indicative of an overall security assessment of the information system.” EX1007, ¶42. These

composite scores are generated from “numerical scores” that indicate “degrees of compliance with security standards.” *Id.*, ¶37. The numerical scores also are indicative of “goodwill and negative publicity.” *Id.* All of these are information ***useful to decisions about entities with which to do business*** because it would be more desirable to do business with an entity with a better security score and less desirable to do business with a worse security score. This is because the security score is indicative of the security of the ***entity’s network***, which is important if, for example, doing business with that entity involves transferring confidential or private information. EX1003, ¶149. These disclosures are also commensurate with how the ’524 describes useful business information, where it states that “a cyber security assessment of the other company to determine if it is following good security practices,” and that “[i]f the auditing company is satisfied with the assessment, it may choose to continue doing business with the other company.” EX1001, 1:35-40.

3. Claim 2: The method of claim 1 in which the technical assets comprise an IP space.

As discussed above in limitation 1[a], McNab discloses discovering the IP addresses of assets such as mail, SMTP or FTP servers using “Forward DNS Grinding.” A POSITA would understand that such network assets have IP addresses within an IP space (*i.e.* the space of valid IP addresses for the network) defined by a subnet mask. EX1003, ¶150. Indeed, McNab discloses “the subnet mask used by the target host.” EX1005, 43. As explained by Dr. Almeroth, a subnet mask defines

the IP space by indicating which bits of the IP address define the sub-network.

EX1003, ¶150.

In addition, McNab discloses that Nessus can target “ranges of network addresses” for its vulnerability scans, indicating that the *assets comprise an IP space*. *Id.*, 385. EX1003, ¶151.

4. Claim 3: The method of claim 1 in which maintaining an entity map comprises querying a regional Internet registry.

McNab discloses in Chapter 3 that “*Regional Internet Registries* (RIRs) store useful information (primarily as *network*, *route*, and *person* objects) relating to IP network blocks.” EX1005, 23-24. Thus, “*RIRs can be queried.*” *Id.*; EX1003, ¶152.

5. Claim 4: The method of claim 3 in which maintaining an entity map comprises submitting a name of one of the entities to the regional Internet registry.

McNab discloses that “*American Registry for Internet Numbers (ARIN)*” and “*Asia Pacific Network Information Centre (APNIC)*” are two of many such Regional Internet Registries that can be queried. *Id.* McNab then provides an example of submitting a name, specifically “nintendo” to multiple RIRs. *Id.*, 25-26; EX1003, ¶153.

Example 3-2. Enumerating the Nintendo objects in ARIN

The diagram illustrates the process of enumerating Nintendo objects in the ARIN registry. A box labeled "regional Internet registry" has an arrow pointing to "ARIN". Another arrow points from "ARIN" to a box labeled "submitting an entity name". A third arrow points from this box to the command "\$ whois nintendo -h whois.arin.net" in the terminal output. The terminal output lists various Nintendo entities and their associated IP ranges.

```
$ whois nintendo -h whois.arin.net
Nintendo North America (NNA-21)
Nintendo of America (TEND)
Nintendo of America (NINTEN-1)
Nintendo Of America inc. (NINTEN)
Nintendo of America, Inc. (NINTE-1)
Nintendo of America, Inc. (NINTE-2)
Nintendo Network Administration (NNA12-ARIN) netadmin@noa.nintendo.com +1-425-882-2040
Nintendo Of America inc. (AS11278) NINTENDO 11278
Nintendo North America SAVV-S233299-1 (NET-207-149-2-192-1) 207.149.2.192 - 207.149.2.199
Nintendo North America SAVV-S263732-2 (NET-209-67-111-168-1) 209.67.111.168 - 209.67.111.175
Nintendo North America SAVV-S233299-2 (NET-216-74-145-64-1) 216.74.145.64 - 216.74.145.127
Nintendo of America NET-NOA (NET-206-19-110-0-1) 206.19.110.0 - 206.19.110.255
Nintendo of America NINTENDO-COM (NET-205-166-76-0-1) 205.166.76.0 - 205.166.76.255
Nintendo Of America inc. NOA (NET-192-195-204-0-1) 192.195.204.0 - 192.195.204.255
Nintendo of America, Inc. SAVV-S263732-3 (NET-216-32-20-248-1) 216.32.20.248 - 216.32.20.255
Nintendo of America, Inc. SAVV-S263732-3 (NET-209-67-106-128-1) 209.67.106.128 - 209.67.106.255
NINTENDO ABOV-T461-209-133-66-88-29 (NET-209-133-66-88-1) 209.133.66.88 - 209.133.66.95
NINTENDO ABOV-T461-209-133-66-72-29 (NET-209-133-66-72-1) 209.133.66.72 - 209.133.66.79
Nintendo MFN-N389-64-124-44-48-29 (NET-64-124-44-48-1) 64.124.44.48 - 64.124.44.55
```

6. Claim 5: The method of claim 4 in which submitting the entity name includes submitting spelling variations of the entity name.

McNab renders this limitation obvious. EX1003, ¶154. McNab discloses that “[e]ach respective regional registrar’s WHOIS database contains information relevant to that particular region.” *Id.*, 24-25. For example, RIPE “doesn’t contain information about ... objects that are found in the Americas” and ARIN “only contains details of North American objects.” *Id.* A POSITA would have found it obvious that, for example, spelling variants exist between different languages or different dialects of the same language. EX1003, ¶154. Thus, a POSITA would have found it obvious different spelling variations of the entity name may need to be submitted to different regional Internet registries. *Id.*

7. **Claim 6: The method of claim 1 in which maintaining an entity map comprises using a domain name associated with the entity.**

As discussed in limitation 1[a], McNab discloses maintaining an entity map by using a domain name associated with the entity, such as nintendo.com, cia.gov or bankofengland.co.uk or 54ymantec.com. *Id.*, 22-25, 34-36; EX1003, ¶155.

8. **Claim 7: The method of claim 6 in which maintaining an entity map comprises using a dig tool on the domain name to determine a published IP address.**

McNab discloses using utilities “such as *nslookup*, *host*, and *dig* ... to issue DNS requests *relating to domains and IP address blocks* identified” and that the “*dig* client” is capable of querying DNS servers directly. EX1005, 30. McNab discloses an example “dig” command that uses the domain name “ucia.gov” to determine an IP address. *Id.*; EX1003, ¶156.

Example 3-8. Using dig to perform a DNS zone transfer

```

dig tool → $ dig @relay2.ucia.gov ucia.gov axfr
; <<>> DiG 9.2.4 <<>> ucia.gov @relay2.ucia.gov axfr
;; global options: printcmd
ucia.gov. 3600 IN SOA relay1.ucia.gov. root.ucia.gov.
511210023 7200 900 604800 900
ucia.gov. 3600 IN NS relay1.ucia.gov.
ucia.gov. 3600 IN NS relay7.ucia.gov.
ucia.gov. 3600 IN NS auth100.ns.uu.net.
ucia.gov. 3600 IN MX 5 mail2.ucia.gov.
ain.ucia.gov. 3600 IN A 198.81.129.68
ain-relay.ucia.gov. 3600 IN CNAME relay1.ucia.gov.
ain-relay-int.ucia.gov. 3600 IN CNAME ain-relay1-int.ucia.gov.
ain-relay1.ucia.gov. 3600 IN CNAME relay1.ucia.gov.
ain-relay1-ext.ucia.gov. 3600 IN CNAME relay1.ucia.gov.
ain-relay1-int.ucia.gov. 3600 IN A 192.168.64.2
ain-relay2.ucia.gov. 3600 IN CNAME relay2.ucia.gov.
ain-relay2-ext.ucia.gov. 3600 IN CNAME relay2.ucia.gov.
ain-relay2-int.ucia.gov. 3600 IN A 192.168.64.3
ain-relay7.ucia.gov. 3600 IN CNAME relay7.ucia.gov.
ain-relay7-ext.ucia.gov. 3600 IN CNAME relay7.ucia.gov.
ain-relay7-int.ucia.gov. 3600 IN A 192.168.64.67
ex-rtr.ucia.gov. 3600 IN CNAME ex-rtr-129.ucia.gov.
ex-rtr-129.ucia.gov. 3600 IN A 198.81.129.222
ex-rtr-129.ucia.gov. 3600 IN HINFO "Cisco 4000 Router" "NP-1E Board"
ex-rtr-191-a.ucia.gov. 3600 IN A 192.103.66.58
ex-rtr-191-b.ucia.gov. 3600 IN A 192.103.66.62
foia.ucia.gov. 3600 IN NS relay1.ucia.gov.
foia.ucia.gov. 3600 IN NS auth100.ns.uu.net.
mail1.ucia.gov. 3600 IN A 198.81.129.68
mail1out.ucia.gov. 3600 IN A 198.81.129.71
mail2.ucia.gov. 3600 IN A 198.81.129.148
mail2out.ucia.gov. 3600 IN A 198.81.129.146
relay.ucia.gov. 3600 IN CNAME relay1.ucia.gov.
relay-int.ucia.gov. 3600 IN CNAME ain-relay1-int.ucia.gov.
relay1.ucia.gov. 3600 IN A 198.81.129.193
relay1-ext.ucia.gov. 3600 IN CNAME relay1.ucia.gov.
relay1-int.ucia.gov. 3600 IN CNAME ain-relay1-int.ucia.gov.
relay2.ucia.gov. 3600 IN A 198.81.129.194
relay2-ext.ucia.gov. 3600 IN CNAME relay2.ucia.gov.
relay2-int.ucia.gov. 3600 IN CNAME ain-relay2-int.ucia.gov.
relay2a.ucia.gov. 3600 IN A 198.81.129.200
relay2y.ucia.gov. 3600 IN A 198.81.129.68
relay2z.ucia.gov. 3600 IN A 198.81.129.69
relay7.ucia.gov. 3600 IN A 198.81.129.186
relay7-ext.ucia.gov. 3600 IN CNAME relay7.ucia.gov.
relay7a.ucia.gov. 3600 IN A 198.81.129.197
relay7b.ucia.gov. 3600 IN A 198.81.129.198
res.ucia.gov. 3600 IN A 198.81.129.116
wais.ucia.gov. 3600 IN CNAME relay2.ucia.gov.

```

Annotations in the image:

- A blue box labeled "dig tool" points to the command line.
- A yellow box labeled "domain names" points to the domain names in the output.
- A red box labeled "IP addresses" points to the IP addresses in the output.

9. Claim 8: The method of claim 1 in which maintaining an entity map comprises probing IP addresses around a published IP address.

McNab discloses that an IP address block can be identified from a published IP address of a domain name. For example, in *Example 3-8*, several hosts have IP addresses that begin with 198.81.129.xxx, such as “mail1.ucia.gov” (192.81.129.68)—from these hosts, an IP address block “198.81.129.0” is “identified as being used.” EX1005, 33-34. Then, McNab discloses performing

reverse DNS sweeping of IP addresses in the IP address block “198.81.129.0”—*i.e.* IP addresses around the, for example, published IP address 198.81.129.68 of “mail1.ucia.gov.” *Id.*, 36-37; EX1003, ¶157.

10. Claim 10: The method of claim 1 in which maintaining an entity map comprises using a name server lookup to verify the derived server name.

McNab discloses that using nslookup (“*name server lookup*”) to verify a derived server name. EX1005, 30; EX1003, ¶158. McNab discloses “[u]sing nslookup to enumerate basic domain details” about a domain name. EX1005, 30-31. It would have been obvious that TXDNS uses nslookups to determine which server names are verified to be real. EX1003, ¶¶158-59. This is because, as explained above, TXDNS uses known prefixes to generate server name guesses and some methodology to determine whether they exist or not—nslookup would have been an obvious design choice to confirm, since it is the standard method for confirming whether a domain name has an IP address. *Id.*

11. Claim 11: The method of claim 2 comprising using the IP space for collecting the cyber-security characterizations.

McNab discloses that the Nessus Security Scanner targets assets with IP addresses, which means when Nessus collects cyber-security characterizations, it used the IP space (*e.g.*, an IP address within the IP space) to collect the characterizations. EX1003, ¶160.

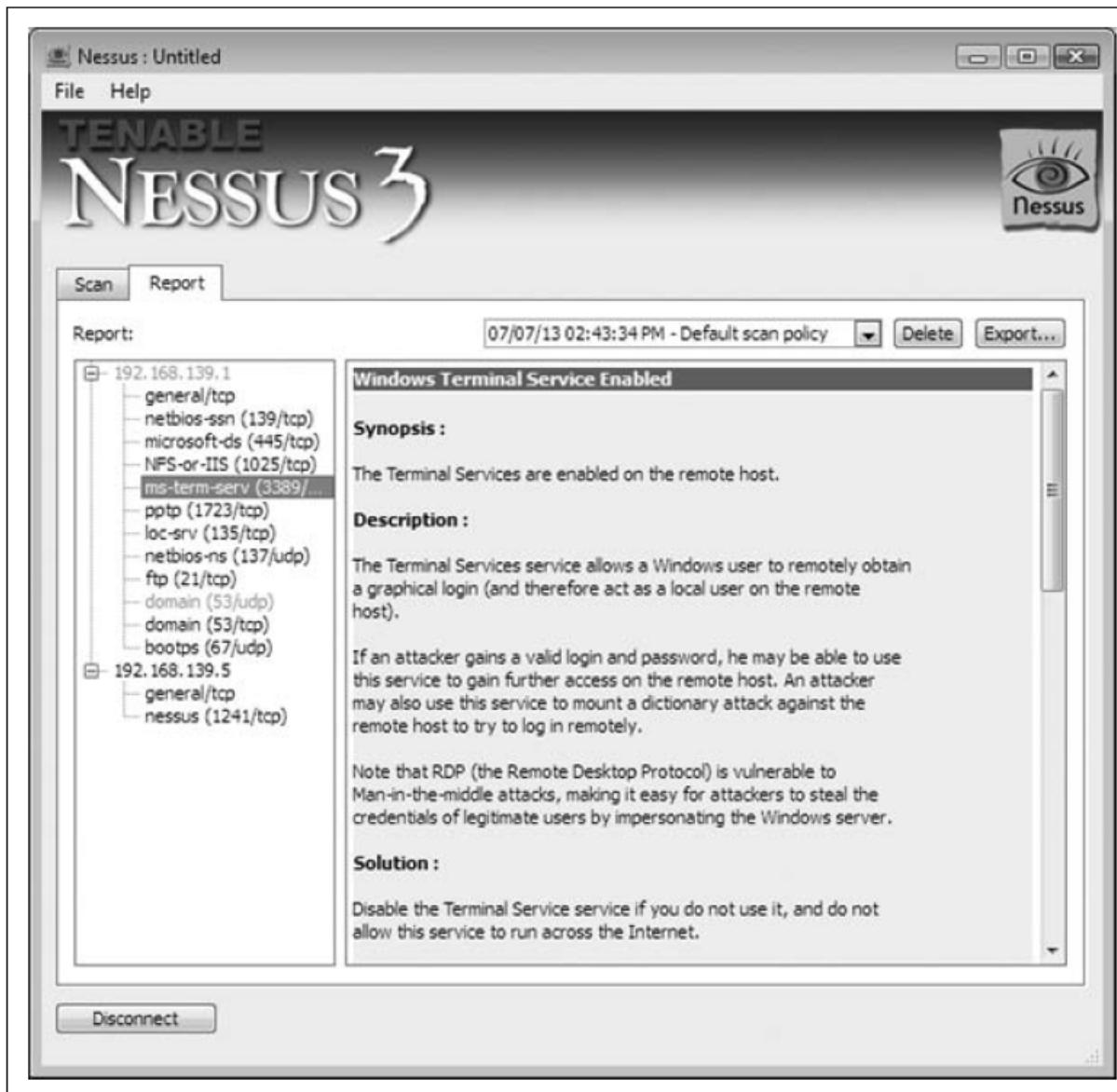


Figure 15-6. NessusClient 3 completed scan

- 12. Claim 12: The method of claim 1 in which the technical assets that are associated with cyber-security characterizations of the respective entities comprise technical assets that the entity owns, controls, uses, or is affiliated with.**

As discussed above for limitation 1[a], McNab discloses generating an entity map for an organization's "networks." EX1005, 17. The assets are therefore

affiliated with entities (*i.e.* **entities**) because they belong to the entity’s network.

EX1003, ¶161.

13. Claim 13: The method of claim 1 in which the technical assets comprise ranges of IP addresses.

As discussed for Claim 2, McNab discloses that the technical assets comprise ranges of IP addresses. EX1003, ¶162.

14. Claim 14: The method of claim 1 in which the technical assets comprise servers.

As discussed for limitation 1[a], McNab discloses “SMTP gateways” or “mail relay servers” as assets. EX1003, ¶163.

15. Claim 15: The method of claim 1 in which maintaining an entity map comprises using information received from Internet Service Providers.

McNab discloses that network enumeration involves performing a “BGP query[]” where a domain name’s “Autonomous System (AS)” number is determined from WHOIS information and then queried to determine an IP address block. EX1005, 28-29. As McNab explains, the IP address block for a particular AS number is allocated to an “ISP.” *Id.* Thus, McNab discloses using information received from an ISP to maintain the entity map. EX1003, ¶164.

16. Claim 16: The method of claim 1 in which maintaining an entity map comprises using heuristics.

The ’524 patent explains that “heuristics” include at least: “Using the ‘dig’ ... tool to determine any IP information published by an entity” or “us[ing] the IP

addresses and domain names published to find ranges of IP addresses actually used” or attempting to guess “servers whose names may be guessed” such as adding the “mail,” “smtp” or “ftp” prefix to a domain name. EX1001, 9:16-38.

McNab discloses adding the mail prefix to a domain name to guess server names. McNab also discloses using “dig.” Thus, McNab discloses *maintaining an entity map by using heuristics*. EX1003, ¶¶165-66.

17. Claim 18: The method of claim 1, wherein annotating comprises prepending to the domain name at least one of “mail,” “smtp,” and “ftp”.

As discussed in limitation 1[a], McNab discloses using TXDNS to prepend “mail” or “ftp” to a domain name. EX1005, 35-36, 85-86; EX1003, ¶¶167-68.

18. Claims 19 and 20

Claims 19 and 20 mirror the above claims as set forth in the following chart. McNab-McGovern render these claims obvious for the same reasons as above. EX1003, ¶¶169-91.

Claim Limitation	Corresponding Claim Limitation(s)
19[pre]	1[pre]
19[a]	1[a], 2
19[b]	1[a], 1[b], 4, 6, 8, 10, 15, 18
19[c]	1[c]
19[d]	1[d]
19[e]	1[e]

Claim Limitation	Corresponding Claim Limitation(s)
20[pre]	1[pre]
20[a]	1[a]
20[b]	1[a]
20[c]	1[b]
20[d]	1[c]
20[e]	1[d]
20[f]	1[e]

Claim 20 does ***not*** require that each of the claimed operations be performed automatically. Thus, while Petitioner relies on its analysis from claim 1, it is not necessary to prove that the combination performs the various functions automatically.

B. Ground 2: Cole, McNab, and McGovern

Cole discloses a system for “automated testing of vulnerabilities to intrusion on a target network” including, among other things, “identification of target network topology and target computers” and “active assessment of vulnerabilities based on information acquired from target computers.” EX1006, Abstract. It would have been obvious to combine Cole with teachings in McNab regarding network enumeration (such as “Forward DNS Grinding”) and McGovern regarding duration of malicious activity. EX1003, ¶¶192-207.

1. Motivation to Combine

(a) McNab

A POSITA would be motivated to incorporate McNab’s teachings of “Forward DNS Grinding” and other methods for enumerating network assets because such teachings were known to provide another additional reliable methods for mapping additional network assets—which as McNab teaches was useful and necessary for performing network assessments. EX1003, ¶¶195-201.

A POSITA would have been motivated to incorporate McNab’s teachings of “Forward DNS Grinding” (described in Ground 1 and also herein) from McNab to provide a well-known and reliable method of determining the IP addresses of assets. EX1003, ¶197. Cole already discloses that as “a first step” it “creates a scan list of IP addresses” for a target network (*i.e.* domain name). EX1006, 15:1-3. However, Cole does not describe how to generate the list of IP addresses. A POSITA would have understood that one well-known and reliable method of generating IP addresses of potential network assets is McNab’s “Forward DNS Grinding” method using conventional tools such as TXDNS. EX1003, ¶197. A POSITA would have been motivated to give effect to Cole’s instruction to generate a list of IP addresses. *Id.*

In addition, a POSITA would have been motivated to use McNab’s teachings because generating an accurate list of IP addresses is important for properly performing vulnerability assessments on the network. EX1003, ¶198. As Cole

teaches, the list of IP addresses is used to determine the target computers for vulnerability testing. Moreover, as discussed in the Technical Background, numerous papers and resources were known to provide network enumeration prior to vulnerability scanning. *See, e.g.*, EX1015 (“Anatomy of a Hack” begins with “Identification of Internet-based points of presence”), EX1016, 11-12 (“network mapping tool [to] identify the services that run on a large group of hosts and provide a quick way of building individual profiles of the target IT system(s)”), EX1017, 7 (“The first step to any security engagement is the initial discovery process”). Thus, a POSITA would find it desirable to have a list of IP addresses that includes *all* computers within the network, or at least all computers that are accessible via the Internet. Thus, a POSITA be motivated to use any and all mechanisms that may result in determining assets within the network, as that would increase the reliability of Cole’s vulnerability assessment and increase the overall security of the network. EX1003, ¶198. Indeed, if Cole’s system failed to test an asset within an entity’s network, that could prove dangerous if the asset became vulnerable—Cole’s system would be unable to make that determination and the administrators would not be able to timely mitigate the issue. *Id.*

For similar reasons, a POSITA would have been motivated to also incorporate other network enumeration techniques described in McNab. For example, McNab describes querying “Regional Internet Registries” which “store useful information

... relating to IP network blocks.” EX1005, 23-24. McNab also describes the use of the “dig” and “nslookup” tools which provide further information for network enumeration. *Id.*, 30. McNab further describes BGP querying to determine IP address blocks. *Id.*, 28-29. All of these methods would have been obvious to use with Cole for similar reasons described above, namely that they were well-known network enumeration techniques that a POSITA would be motivated to include any technique that would increase the likelihood of enumerating all assets. EX1003, ¶199.

A POSITA would have reasonably expected success in combining Cole with McNab’s “Forward DNS Grinding.” McNab teaches that “Forward DNS Grinding” can be performed using a simple command-line tool called TXDNS, and it would have been well-within the capabilities of a POSITA to incorporate that command-line tool in the automated system of Cole to generate a list of IP addresses for performing vulnerability assessments. EX1003, ¶200.

Finally, Cole and McNab are analogous art to the ’524 patent because all three references are directed to the field of network security. EX1006, Abstract; EX1005, xvi; EX1001, Abstract; EX1003, ¶201.

(b) McGovern

A POSITA would be motivated to incorporate McGovern’s teachings of calculating a duration of malicious activity as well as a remote web-based user

interface because such teachings were known to increase the security of a network as well as increase user-friendliness of the system and provide the user an easier assessment of the network, including by remotely accessing such assessments. EX1003, ¶¶202-07.

McGovern discloses the capability of determining the “average response or patch time after security breach detection.” EX1007, ¶38. A POSITA would be motivated to incorporate this teaching into Cole, which discloses various methods and measurements for evaluating the security of a network, because of the known benefits. A POSITA would understand that it would be beneficial to include the duration of a security breach in the network vulnerability assessments of Cole, because the duration of a security breach is an important factor in determining overall network security. A POSITA would understand that a security breach is an incredibly dangerous event that puts all of the organization’s data and resources at risk of damage or theft. A POSITA would further understand that a security breach can cause damage to the company via various ways, including the theft of confidential information, putting the business at a competitive disadvantage, or the theft of customer or employee information, therefore harming said customers or employees, harming the company’s reputation, and increasing the liability risk to the company. A POSITA would therefore understand that it would be highly desirable to minimize the duration of any security breach—thus, it would be beneficial for a

network administrator to know whether the network and the organization's IT policies are robust and proactive to such security breaches (resulting in a minimal duration) or if they are particularly vulnerable (because of a lengthy duration). EX1003, ¶203.

McGovern also discloses the capability of displaying a network assessment score in “a software program with a user interface,” which can be “designed to run as a Web-based service or application” wherein users can “access the scoring and related functionalities remotely via standard browsers or similar user interfaces.” EX1007, ¶¶42, 45. A POSITA would be motivated to incorporate these teachings into Cole because it allows network administrators access to the assessment reports from anywhere an Internet browser is available—at the time of the '524 patent, that would have included smartphones. A POSITA would therefore be motivated to make this combination in order to increase ease-of-use and accessibility. EX1003, ¶204.

A POSITA would have combined Cole and McGovern with a reasonable expectation of success. Cole is dedicated to performing vulnerability assessments of target networks—including customizable tests using scripts. EX1006, Abstract. Adding McGovern's teaching of determining the duration of a security breach would simply be another test that an administrator could perform that was well-within the capability of a POSITA at the time of the '524 patent. Cole does not state any limit

on the number of vulnerability tests that could be performed, and a POSITA would understand that adding an additional test would not pose any significant hurdle in computing power or evaluation time, especially in view of the benefits of said test described above. Indeed, because Cole already discloses determining the presence of malicious software, and discloses tracking of fixes for those vulnerabilities, a POSITA would understand that McGovern's teachings could be readily implemented by determining the duration of when the malicious activity was first detected to when it was verified that the vulnerability was fixed. EX1006, 70:54-58 ("components for tracking detected vulnerabilities, and for performing targeted vulnerability fix verification tests of specific host computers"); EX1003, ¶205.

A POSITA would have also reasonably expected success in combining Cole with McGovern's remote user interface. Indeed, Cole already describes a testing system that "may be a web-based interface that is accessible to network users via an ordinary web browser" (EX1006, 71:50-59) and it would have been well-within the skills of a POSITA to incorporate McGovern's teaching of providing reports through a web-based interface. EX1003, ¶206.

Finally, as discussed above, Cole and McGovern are analogous art to the '524 patent because all three references are directed to the field of network security. EX1003, ¶207.

2. Claim 1

(a) 1[pre], 1[a]

“A computer implemented method comprising: on one or more computers, maintaining in a database an entity map that maps technical assets to respective companies or other entities with which the assets are associated”: Cole discloses targeting a network to build lists of assets and their IP addresses and storing them in a database, thus generating an ***entity map***. EX1003, ¶¶208-10. Cole further discloses targeting multiple networks. Indeed, each network belongs to “an ***entity*** seeking to ‘harden’ its network.” EX1006, 1:55-56.

Cole discloses a “highly automated” system. EX1006, Abstract. Specifically, Cole discloses that “a first step” is “defin[ing] the target network and creat[ing] a scan list of IP addresses.” EX1006, 15:1-3. Cole determines which IP addresses are “‘live’ or ‘probably live,’ and the associated IP addresses are added to a respective live ***database*** 324 or probably live ***database*** 326 accordingly.” *Id.*, 15:8-17. Thus, this process generates a mapping of ***technical assets to an entity which the assets are associated*** which is ***maintained in a database***. EX1003, ¶209.

Cole discloses that the discovery of assets can be performed on ***multiple entities***. Specifically, Cole discloses that its method “operates on multiple target networks concurrently.” *Id.*, 15:1-3, 18:19-21. Cole describes that every entity has “its network” it seeks to “harden.” *Id.*, 1:55-56. Thus, a POSITA would understand

that each target network belongs to a different entity. These mappings are maintained in databases, thereby forming an *entity map*. EX1003, ¶210.

“wherein mapping technical assets to respective companies or other entities comprises determining IP information by annotating a domain name with naming conventions for at least one of a mail server, an SMTP server, and an FTP server, thereby deriving a likely server name having assigned IP addresses”: Cole does not expressly describe how to create a network’s initial list of IP addresses. However, McNab discloses a method called “Forward DNS Grinding” which as discussed in Ground 1 meets this limitation. It would have been obvious to incorporate this teaching into Cole to generate some of the IP addresses in the scan from which *assets are discovered and mapped to entities*. EX1003, ¶¶211-12.

(b) 1[b]

Cole discloses that its “highly automated” system performs “automatic detection, monitoring and reporting of network vulnerabilities.” EX1006, Abstract.

“automatically useing [sic] sensors on the Internet”: Cole discloses that its methods are performed by a “network security system 100” is “outside the target network 102” but connected to it via ***“the Internet.”*** EX1006, 11:34-37. After the network assets are mapped, Cole performs an “active assessment routine” and a “scoring routine” which “establishes a vulnerability score ... for the network based on the results of the active assessment and based on the vulnerability information in

the known vulnerability database 366.” *Id.*, 17:59-18:5. Cole further discloses that it performs this vulnerability testing using “automatic script[s]” for each vulnerability.” *Id.*, 41:11-15. Thus, Cole discloses using “[m]ultiple FASL scripts ... run in parallel,” each of which a POSITA would understand to be one of many *sensors on the Internet*. EX1003, ¶¶213-14.

“to collect externally observable cyber-security characterizations of the technical assets that have been mapped to each of the entities”: Cole discloses that it performs vulnerability testing using “FASL scripts,” which collect *externally observable cyber-security characterizations*. Cole discloses that such scripts are run on “target system[s],” *i.e. technical assets that have been mapped to an entity*. EX1006, 41:11-15; 53:11-16 (“the m_nIPAddress member of the FASL variable has been set to the IP address of the target machine upon which the FASL script is to run. All network activity that takes place in a FASL script uses this IP address as its destination”). Because the security system is external to the network and targets a specific IP address, a POSITA would understand that the FASL scripts collect *externally observable* information. EX1006, 60:53-54 (“Quantitative Scoring of Target Network *External Vulnerability*”); 63:53-59 (“[T]he preceding calculations provide a measure of the security of ports visible *to the Internet* and other *external networks*, the ease with which an unauthorized user could breach the network firewall, and the ease with which an unauthorized user could access data inside the

network from a computer *outside of the network*, among other things.”); EX1003, ¶215.

A FASL script “permits one to perform active assessment of potential vulnerabilities on a known target computer with known open ports.” EX1006, 58:55-58. These scripts collect information needed to perform the vulnerability assessment, *i.e. externally observable cyber-security characterizations*. *Id.*, 59:28-31 (“[I]nformation collected from a successful vulnerability test ... is used to advantageously further test the vulnerability of the target network and individual target computers.”). Cole generally describes these assessments as related to “the security of ports visible to the Internet and other external networks, the ease with which an unauthorized user could breach the network firewall, and the ease with which an unauthorized user could access data inside the network from a computer outside of the network, among other things,” all of which a POSITA would consider to be *cyber-security characterizations*. EX1006, 63:53-59; EX1003, ¶216.

For example, a script can collect information about open ports found on a target computer whether they are vulnerable. EX1006, 59:37-57; EX1003, ¶217.

“and associating the observable cyber-security characterizations with a respective entity based at least in part on the derived server name”: Cole discloses that “the information collected by the FASL script from the target computer,” *i.e.*

cyber-security characterizations, are “stored in a database related to the target computer for later reporting.” EX1006, 41:26-31; EX1003, ¶¶218-20.

Because this information is later used to “provide[] objective, quantitative indicia of the overall security of the target *network*,” *i.e. the network of the respective entity*, a POSITA would understand that the characterizations are associated with the *respective entity*, especially since, as discussed for limitation 1[a], the system already associates the target computer with the target network. EX1003, ¶219. A POSITA would have also considered it obvious to associate the characterizations with the respective entity (and its network) because they would be motivated to give effect to the security assessment score that is later generated for the overall network. *Id.* A POSITA would have also been motivated by the benefits of well-organized data in that subsequent review of the security characterizations would be made much easier if they are associated with the entity for which the assessment was made. *Id.* A POSITA would have had a reasonable expectation of success because Cole already discloses that the characterizations are stored in a database and it would be well-within the capabilities of a POSITA to reference that data to information about the entity already stored in the database as discussed in limitation 1[a]. *Id.*

Although Cole does not describe a *derived server name*, as discussed above, McNab describes using “Forward DNS Grinding” to generate a derived server name

(such as ftp.apple.com). EX1003, ¶220. A POSITA would have understood that the characterizations are associated with the entity *based on a derived server name* because the characterizations are associated with a target computer that was identified by an IP address that, in accordance with McNab’s “Forward DNS Grinding,” was determined by generating a derived name. *Id.* In addition, a POSITA would have considered it obvious that the FASL scripts could identify the target computer using the *derived server name* as that would allow the benefit of more easily identifying the target computer and would simply require a single DNS request to resolve the server name to an IP address. *Id.*

(c) 1[c]

“automatically deriving observations about the technical assets from the collected cyber-security characterizations,”: Cole discloses that each FASL script will automatically use the information it collects (*i.e. collected cyber-security characterizations*) to “respond with a success or failure result noting whether the target computer was vulnerable or not to the given vulnerability identification code,” *i.e. deriving observations about the technical asset*. EX1006, 41:23-26; EX1003, ¶221.

“wherein the derived observations comprise (i) a number of technical assets that have been reported to be malicious”: Cole describes that FASL scripts can be programmed to determine the presence of malicious software on an asset. EX1003,

¶¶222-24. Specifically, Cole notes that one type of vulnerability for which a FASL script will assess is a “trojan.” EX1006, 60:20-52 (“TABLE: Vulns”). As Cole explains it, a “trojan” allows an attacker to maliciously retrieve “usernames and passwords, enumeration of other network information resources, retrieval of host data, or launching an attack against other systems/networks.” EX1006, 67:1-17. Thus, a FASL script that indicates it has detected a trojan in a target computer has *derived an observation reporting the asset to be malicious*. EX1003, ¶222.

Cole also describes *deriving an observation* where “all vulnerabilities found in the target network have been counted” using a “vulnerability counter.” EX1006, 63:3-4, 10. Cole describes that the trojan detected by a FASL script is part of a set of “Vulns” that are scanned for. *Id.*, 60:20-52. Thus, the “vulnerability counter” would be understood to be a *derived observation of the number of technical assets that have been reported to be malicious*. EX1003, ¶223.

In the alternative, it would have been obvious to derive the *number of assets reported to be malicious*. Since Cole already reports which assets are malicious, a POSITA would have found it obvious to count that number. EX1003, ¶224. A POSITA would be motivated to do this because it would allow the number to be displayed to the administrator, thereby giving the administrator an easily-accessible number to indicate the health of the network. *Id.* Deriving the number would also allow the system to alert the administrator if the number exceeds zero. *Id.* A

POSITA would have a reasonable expectation of success in doing this, because it would be well-within the skill of a POSITA to include within the FASL script a count of malicious assets. *Id.* The script could further email said number to the administrator. *Id.*

“and (ii) a duration of detected malicious activity associated with the technical assets”: As discussed above in Ground 1, McGovern discloses deriving an observation that includes ***a duration of detected malicious activity associated with the technical assets***. EX1007, ¶38 (“average response or patch time after security breach detection”). As discussed above, a POSITA would have found it obvious to combine Cole with McGovern, and in particular, by using Cole’s disclosed tracking of vulnerabilities to determine the duration of when the malicious activity was first detected to when it was verified that the vulnerability was fixed. EX1006, 70:54-58; EX1003, ¶225.

(d) 1[d]

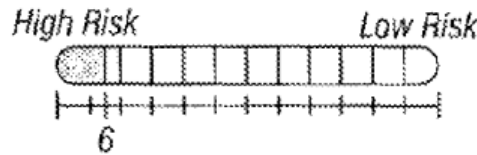
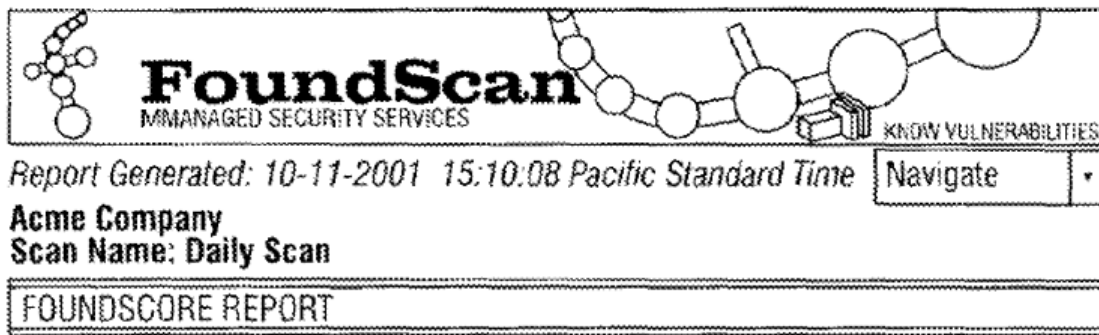
“automatically generating a cyber-security rating for each of the entities using the entity map and the derived observations”: Cole discloses that its testing system features “an application of a quantitative score to objectively and comparatively rank the security of the target network.” EX1006, 2:35-37. For example, Cole derives an objective security score (***cyber-security rating***) for a network “that depends at least on the vulnerability risk levels of the vulnerabilities,

the vulnerability risk levels of the nodes, and the number of nodes on the network.” EX1006, 10:33-41. Cole’s system further discloses a scoring routine that “establishes a vulnerability score for each target computer and for the network based on the results of the active assessment and based on the vulnerability information in the known vulnerability database.” *Id.*, 18:1-5; EX1003, ¶226.

Cole in particular describes a “FoundScore F [that] provides a quantitative score between 0 and 100 representing an objective indicia of the relative security of the target network” where “the higher the Foundscore ..., the greater the relative security of the target network.” *Id.*, 62:54-58. Cole describes that this Foundscore is determined using the vulnerabilities identified by the FASL scripts. *Id.*, 41:26-31; EX1003, ¶227.

(e) 1[e]

“through a communication network, exposing to users in a user interface of a portal, information about the cyber-security ratings of the entities”: Cole’s testing system discloses “compilation of a dynamic, graphical report representing the network topology, network computers, and network vulnerabilities in a hierarchical report including both overview and detail documents” and demonstrated several examples of such report in Figures 10-34. EX1003, ¶¶228-29. In particular, Cole discloses that an “exemplary FoundScore Report is set forth below in FIGS. 27-28.”

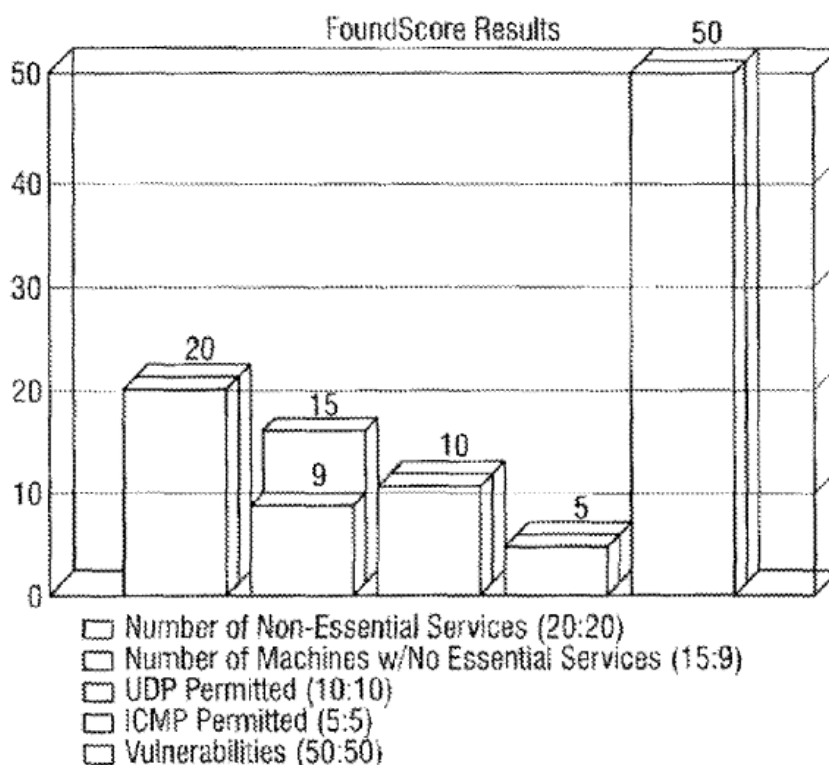


FoundScore Risk Rating

You scored a total of 6 out of a possible 100 points. Your FoundScore places you in the Poor category. Thus, relative to other organizations, your risk rating is poor. Please review the FoundScore Results graph and FoundScore Categories table below for details.

A [Full Explanation](#) of the FoundScore scoring methodology is also available.

FIG. 27



FoundScore: Vulnerabilities	Your Results	Deductions	Running Score
High Risk Vulnerabilities	<u>13</u>	50	50
Medium Risk Vulnerabilities	<u>7</u>	0	50
Low Risk Vulnerabilities	<u>3</u>	0	50
Score after Vulnerability Deductions:			50

FoundScore: Exposure	Your Results	Deductions	Running Score
Number of Non-Essential Services	<u>150</u>	20	30
Number of Machines with No Essential Services	<u>9</u>	9	21
UDP Permitted	<u>Yes</u>	10	11
ICMP Permitted	<u>Yes</u>	5	6
Total Score:			6
			Poor

To understand what systems and ports affected your FoundScore, click on the specific item of interest in the "Your Results" column in the above table.

FIG. 28

Cole’s testing system further discloses that, for example, Figure 10 “illustrates an embodiment of a hierarchical security report 1000 *presented on a display for viewing* by a user” and a user “may select any of the reports shown in FIG. 10 (or other reports that are not shown) and request additional details.” *Id.*, 69:29-41; EX1003, ¶229.

Alternatively, a POSITA would have found it obvious that such report can also be accessed through a web-based user interface, because Cole’s testing system also discloses a user interface which “may be a web-based interface that is accessible to network users via an ordinary web browser” for the functionality of the ticket management component. *Id.*, 71:50-59. A POSITA would have been motivated to use that web-based interface to allow viewing of the vulnerability report by any location. EX1003, ¶230.

In addition, as discussed above in Ground 1, McGovern describes displaying an ISA score through a remote client, and it would have been obvious to combine Cole and McGovern. EX1003, ¶231.

“information about the cyber-security ratings that is useful to decisions about entities with which to do business”: Cole describes that the vulnerability reports provide “(1) a dynamic graphical display representing the network topology, the target computers found, and the vulnerabilities found throughout the target network; (2) a comprehensive list of target computers, vulnerabilities found, and

vulnerability explanations; (3) an objective scoring report describing the approximate security rating of the target network; (4) an exposure report describing the ports, services, and (5) detailed information that sets forth testing results on a per-machine, per-port, or per-vulnerability basis.” EX1006, 68:63-69:4. As shown in, for example, Figure 28, each report is generated for a particular *entity*. EX1003, ¶232.

A POSITA would understand that all of this information in the report is *useful to decisions about entities with which to do business* because it would be more desirable to do business with an entity with a better security score and less desirable to do business with a worse security score. EX1003, ¶233. This is because the report’s scores are indicative of the security of the *entity’s network*, which is important when, for example, doing business with that entity involves transferring confidential or private information. *Id.*

3. Claim 2

Cole discloses that “target computers” have an IP address that belongs to various classes of sub-networks. EX1006, 3:41-46. Further, Cole discloses that while “creating a topological representation of a network” the system can obtain a “*plurality of sequences of IP addresses.*” *Id.*, 9:46-50. A POSITA would understand that a sequence of IP addresses is part of an IP space for reasons explained above in Ground 1. EX1003, ¶234.

4. Claim 3

See Ground 1. It was obvious to incorporate such teachings with Cole.
EX1003, ¶235.

5. Claim 4

See Ground 1. It was obvious to incorporate such teachings with Cole.
EX1003, ¶236.

6. Claim 5

See Ground 1. It was obvious to incorporate such teachings with Cole.
EX1003, ¶237.

7. Claim 6

See Ground 1. It was obvious to incorporate such teachings with Cole.
EX1003, ¶238.

8. Claim 7

See Ground 1. It was obvious to incorporate such teachings with Cole.
EX1003, ¶239.

9. Claim 8

See Ground 1. It was obvious to incorporate such teachings with Cole.
EX1003, ¶240.

10. Claim 10

See Ground 1. It was obvious to incorporate such teachings with Cole.
EX1003, ¶241.

11. Claim 11

See Ground 1. It was obvious to incorporate such teachings with Cole. EX1003, ¶242.

12. Claim 12

Cole discloses that the target network is “defined by one or more *ranges of IP addresses controlled by the target network.*” EX1006, 2:54-59, 3:46-51; EX1003, ¶243.

13. Claim 13

As discussed in Claim 2, Cole discloses that the assets comprise a range of IP addresses. EX1006, 9:46-50; EX1003, ¶244.

14. Claim 14

Cole discloses that the target network’s assets “includes ... a *server.*” EX1006, 2:44-50; EX1003, ¶245.

15. Claim 15

See Ground 1. It was obvious to incorporate such teachings with Cole. EX1003, ¶246.

16. Claim 16

As discussed for limitation 1[a], McNab discloses using a “mail” prefix to generate possible server names, which the ’524 patent describes as using a heuristic. EX1001, 9:16-38; EX1003, ¶¶247-48.

17. Claim 18

See Ground 1. It was obvious to incorporate such teachings with Cole.

EX1003, ¶249.

18. Claims 19 and 20

Claim 19 and 20 mirror the above claims as set forth in the following chart.

Cole render these claims obvious for the same reasons as above. EX1003, ¶¶250-72.

Claim Limitation	Corresponding Claim Limitation(s)
19[pre]	1[pre]
19[a]	1[a], 2
19[b]	1[a], 1[b], 4, 6, 8, 10, 15, 18
19[c]	1[c]
19[d]	1[d]
19[e]	1[e]
20[pre]	1[pre]
20[a]	1[a]
20[b]	1[a]
20[c]	1[b]
20[d]	1[c]
20[e]	1[d]
20[f]	1[e]

Claim 20 does *not* require that each of the claimed operations be performed automatically.

VIII. *FINTIV*

Fintiv weighs against denial. A stay has not been requested, trial has not been scheduled, discovery has not yet started, and the Court has not even set forth any case schedule. In addition, the Petition challenges more claims than Patent Owner asserted in its complaint. Moreover, the Petition presents compelling grounds of unpatentability. Thus, all of the *Fintiv* factors weigh against denial.

IX. §325(D)

Neither Cole nor McGovern were considered during prosecution and therefore §325(d) provides no basis to deny this Petition. Moreover, even though McNab is listed as a cited (but not considered) reference (EX1001, 3), the entry indicates that it is the 2004 edition which was cited and more importantly, that only “13 pages” of McNab were cited. *Id.* The McNab reference here is the 2008 edition and is 506 pages. EX1005. Moreover, this Petition relies on far more than 13 pages of McNab, therefore the examiner could not have possibly cited or considered all of the relied-upon portions of McNab.

X. MANDATORY NOTICES AND FEES

A. Real Party-In-Interest

The real party-in-interest is Normshield Inc. d/b/a Black Kite Inc.

B. Related Matters

The '524 patent is asserted in the district court case *BitSight Technologies, Inc. v. NormShield Inc.*, 1:23-cv-12055 (D. Mass.).

C. Counsel and Service Information

Petitioner provides the following counsel and service information.

Petitioner consents to electronic service the email addresses listed in the table below. Pursuant to 37 C.F.R. § 42.10(b), Power of Attorney accompany this Petition.

LEAD COUNSEL	BACK-UP COUNSEL
James M. Glass (Reg. No. 46,729) jimglass@quinnemanuel.com QUINN EMANUEL URQUHART & SULLIVAN, LLP 51 Madison Avenue, 22 nd Floor New York, NY 10010 Tel: (212) 849-7000 Fax: (212) 849-7100	David Elihu (Reg. No. 65,717) davideliu@quinnemanuel.com QUINN EMANUEL URQUHART & SULLIVAN, LLP 865 S. Figueroa St., 10th Floor Los Angeles, California 90017 Tel : (213) 443-3000 Fax : (213) 443-3000 Quincy Lu (Reg. No. 76,954) quincylu@quinnemanuel.com QUINN EMANUEL URQUHART & SULLIVAN, LLP 1109 First Avenue, Suite 210 Seattle, WA 98101 Tel: (206) 905-7000 Fax: (206) 905-7100 Brett Watkins (Reg. No. 60,458) brettwatkins@quinnemanuel.com QUINN EMANUEL URQUHART & SULLIVAN, LLP 700 Louisiana St., Suite 3900 Houston, TX 77002 Tel: (713) 221-7000 Fax: (713) 221-7100

	Andrew Bramhall (<i>pro hac vice</i> forthcoming) andrewbramhall@quinnemanuel.com QUINN EMANUEL URQUHART & SULLIVAN, LLP 555 Twin Dolphin Dr., 5th Floor Redwood Shores, California 94065 Tel: (650) 801-5000 Fax: (650) 801-5100
--	---

D. Payment of Fees

The undersigned authorizes the Office to charge the fee required for this Petition for *Inter Partes* Review to Deposit Account No. 50-5708. Any additional fees that might be due are also authorized.

Date: September 6, 2024

Respectfully submitted,

By: /s/ James M. Glass
James M. Glass

Counsel for Petitioner

**CERTIFICATE OF COMPLIANCE WITH
TYPE-VOLUME LIMITATION, TYPEFACE REQUIREMENTS,
AND TYPE STYLE REQUIREMENTS**

1. This Petition complies with the type-volume limitation of 14,000 words, comprising 13,940 words, as counted using the Microsoft Word software that was used to prepare this paper, excluding the parts exempted by 37 C.F.R. § 42.24(a).

2. This Petition complies with the general format requirements of 37 C.F.R. § 42.6(a) and has been prepared using Microsoft® Word 2016 in 14-point Times New Roman.

Date: September 6, 2024

By: /s/ James M. Glass
James M. Glass
QUINN EMANUEL URQUHART
& SULLIVAN, LLP
51 Madison Avenue, 22nd Floor
New York, NY 10010
Tel: (212) 849-7000
Fax: (212) 849-7100

Counsel for Petitioner

CERTIFICATE OF SERVICE

The undersigned hereby certifies that on September 6, 2024, true and correct copies of the foregoing document and supporting materials were served in its entirety on the Patent Owner at the following address of record as listed on PAIR via Priority Mail Express® or Express Mail:

GOODWIN PROCTER LLP
Patent Administrator
100 Northern Avenue
Boston, MA 02210
United States

Official Correspondence Address

The undersigned hereby certifies that on September 6, 2024, true and correct copies of the foregoing document and supporting materials were served in its entirety on the Patent Owner at the following address of record via Priority Mail Express® or Express Mail:

Hunter Malboeuf
Goodwin Procter, LLP
100 Northern Avenue
Boston, MA 02210
United States

Courtesy copies were also sent via electronic mail to Patent Owner's counsel of record in the related district court proceeding:

Douglas J. Kline
Goodwin Procter, LLP
100 Northern Avenue
Boston, MA 02210
617-570-1000
Email: dkline@goodwinlaw.com
ATTORNEY TO BE NOTICED

Kelly Grosshuesch
Goodwin Procter LLP
1900 N Street, N.W.

Washington, DC 20036

202-346-4000

Fax: 202-346-4444

Email: KGrosshuesch@goodwinlaw.com

ATTORNEY TO BE NOTICED

Naomi Birbach

Goodwin Procter LLP

The New York Times Building

620 Eighth Avenue

New York City, NY 10018

212-459-7374

Email: nbirbach@goodwinlaw.com

ATTORNEY TO BE NOTICED

Robert D. Carroll

Goodwin Procter, LLP

100 Northern Avenue

Boston, MA 02210

617-570-1000

Fax: 617-523-1231

Email: rcarroll@goodwinlaw.com

ATTORNEY TO BE NOTICED

Timothy Keegan

Goodwin Procter LLP

620 Eighth Avenue

New York, NY 10018

212-459-7022

Email: tkeegan@goodwinlaw.com

ATTORNEY TO BE NOTICED

Date: September 6, 2024

Petition for *Inter Partes* Review

U.S. Patent No. 9,973,524

By: /s/ *James M. Glass*

James M. Glass

QUINN EMANUEL URQUHART

& SULLIVAN, LLP

51 Madison Avenue, 22nd Floor

New York, NY 10010

Tel: (212) 849-7000

Fax: (212) 849-7100

Counsel for Petitioner