

Petitioner's Demonstratives

Keysight Technologies, Inc. and Palo Alto Networks, Inc.

v.

Centripetal Networks, LLC

IPR2023-00446

(IPR2023-01353 has been joined with this proceeding)

U.S. Patent No. 10,567,343

Oral Argument

May 6, 2024

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

‘343 patent claim 1

1. A method comprising:

receiving, by a computing system comprising memory and at least one processor, a plurality of packets, wherein the plurality of packets comprises a first portion of packets and a second portion of packets;

determining, based on a packet header field value, whether each packet of the plurality of packets comprises data corresponding to first criterion specified by one or more packet-filtering rules;

responsive to a determination by the computing system that a packet header field value of the first portion of packets comprises data corresponding to the first criterion specified by at least one matching packet-filtering rule, applying, by the computing system and to each packet in the first portion of packets, one or more operators specified by the at least one matching packet-filtering rule;

determining, based on an application header field value, the second portion of packets based on whether the first portion of packets comprises data corresponding to second criterion specified by one or more operators specified by the at least one matching packet-filtering rule; and

responsive to determining the second portion of packets that comprises data corresponding to the second criterion specified by one or more operators specified by the at least one matching packet-filtering rule, applying, by the computing system and to each packet in the second portion of packets that match the second criterion, at least one packet transformation function configured to prevent an exfiltration operation, wherein the at least one packet transformation function indicates whether each packet in the second portion of packets is allowed to continue toward its destination.

EX-1001, Claim 1

'343 patent's methods to prevent exfiltration operations

218

Five-tuple

Rule #	IP Protocol	Source IP Address	Source Port	Destination IP Address	Destination Port	Operator
1 (302)	TCP	140.210.*	*	140.212.*	22	ALLOW
2 (304)	TCP	140.210.*	*	140.212.*	25	ALLOW
3 (306)	TCP	140.210.*	*	140.212.*	110	ALLOW
4 (308)	TCP	140.210.*	*	140.212.*	143	ALLOW
5 (310)	TCP	140.210.*	*	140.212.*	443	REQUIRE-TLS-1.1-1.2
6 (312)	TCP	140.210.*	*	214.*	80	HTTP-EXFIL
7 (314)	*	*	*	*	*	BLOCK

Pet. (Paper 2) 9; EX-1001, Fig. 3

GET). Because attackers may often use HTTP PUT or POST methods to exfiltrate sensitive data, operators such as HTTP-EXFIL may be used to stop such exfiltrations.

Reply (Paper 20) 3; EX-1001 7:21-23

rule 5 310 may allow web browsers attached to network 104 to conduct HTTPS sessions (e.g., secure web sessions) with any web servers attached to network 106, but may, as described in greater detail below, utilize the REQUIRE-TLS-1.1-1.2 operator to ensure that only HTTPS secure web sessions using version 1.1 or 1.2 of the Transport Layer Security (TLS) protocol to secure such HTTPS sessions are allowed (e.g., because the popular TLS version 1.0 protocol has a known security vulnerability that attackers may exploit to decrypt HTTPS sessions).

Reply (Paper 20) 3; EX-1001 6:61-7:3

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

3

Sourcefire teaches importing rules from text files

2. Click **Import Rules**.

The Import SEU page appears.

One-Time SEU/Rules Import

Source	☐ SEU or text rule file to upload and install Browse...
Policy Reapply	☐ Download new SEU from the support site
	☐ Reapply intrusion policies after the SEU import completes
Import	

Recurring SEU Imports
The scheduled SEU import feature is not enabled.

Enable Recurring SEU Imports	☐
Save Cancel	

TIP! You can also click **Import SEU** on the SEU page, which you access by selecting **Policy & Response > IPS > SEU**.

3. Select **SEU or text rule file to upload and install** and click **Browse** to navigate to the rule file. Note that all rules uploaded in this manner are saved in the local rule category.

4. Click **Import**.

The rule file is imported. Make sure you enable the appropriate rules in your intrusion policies. The rules are not activated until the next time you apply the affected policies.

IMPORTANT! 3D Sensors do **not** use the new rule set for inspection until after you apply their intrusion policies. See [Applying an Intrusion Policy](#) on page 347 for procedures.

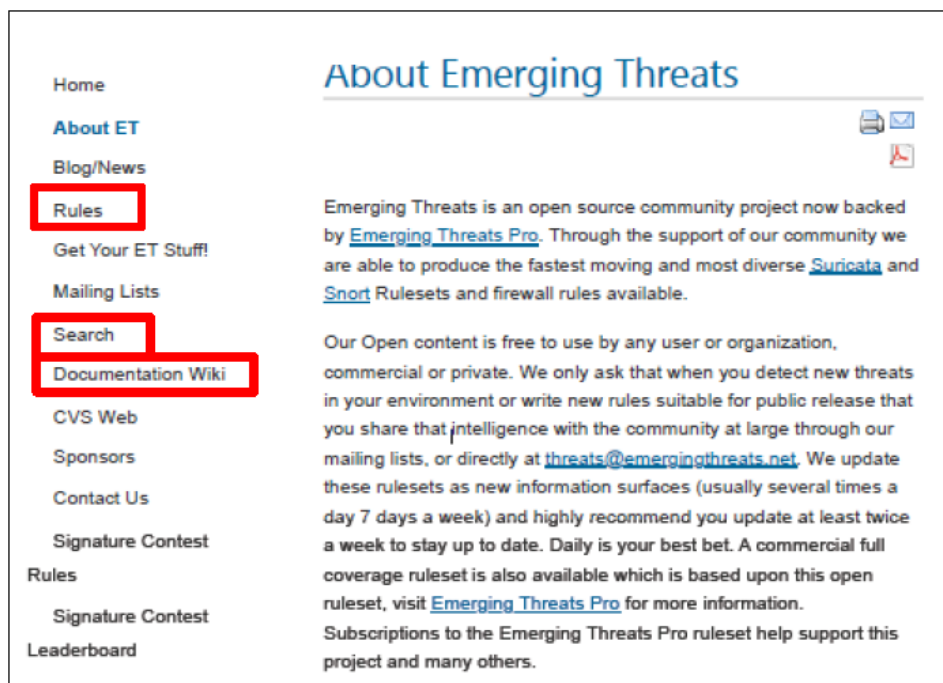
- IPS always sets local rules that you import to the disabled rule state; you must manually set the state of local rules before you can use them in your intrusion policy. See [Setting Rule States](#) on page 435 for more information.

Pet. (Paper 2) 33; EX-1004 1858

Pet. (Paper 2) 32-33; EX-1004 1859

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Emerging Threats was a leading source of Snort rules



Reply (Paper 20) 18; EX-1102 1 (annotated)



Reply (Paper 20) 20; EX-2016 1 (annotated)

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Emerging Threats was a leading source of Snort rules

164. As I will describe in more detail below, Emerging Threats was a well-known source of such custom rules. I used Snort rules, including custom Snort rules, extensively in the late 1990s and early 2000s and was familiar with Bleeding Snort, which later became Emerging Threats. A POSA by 2010 would have been familiar with Emerging Threats as a source of custom rules, and would have downloaded text (ASCII) files of rules from Emerging Threats to load into Snort devices, such as Sourcefire. As addressed in the prior two paragraphs, Sourcefire would disable the rules by default upon their import, and a user would then select whether to set the rules to alert or drop. A user would often initially set imported rules to have an “alert” action to evaluate their performance in the user’s specific

Pet. (Paper 2) 35; EX-1003 ¶164

206. As I explained above, I personally used Snort rules extensively before 2010 and was made aware of the existence of the Bleeding Snort project on multiple occasions, which later became Emerging Threats. A POSA by 2010 would have been familiar with Emerging Threats as a source where custom rules could be downloaded for loading onto devices such as Sourcefire. Centripetal’s own press release in 2014 confirms that Emerging Threats was founded in 2003 and by 2014 was a world-leading provider of threat intelligence rulesets:

Pet. (Paper 2) 35; EX-1003 ¶206

Emerging Threats was a leading source of Snort rules

Q. And how -- well, did you ever download the Emerging Threats ruleset from the Emerging Threats website?

A. Yes, I did.

Q. And when did you first do that?

A. I believe it was when I was in the internship before the year of 2010.

Reply (Paper 20) 20; EX-1101 (Marinho Tr.) 19:3-9

Q. Did you have any trouble downloading the rules from the Emerging Threats ruleset?

A. No, never had a trouble downloading it. Just as easy as clicking on a link and download, and yes.

Reply (Paper 20) 20; EX-1101 (Marinho Tr.) 20:5-8

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Q. And when you said prior to your work at Emerging Threats you were able to go to the website and download the all rules, those would have been the open rules; is that correct?

A. That's correct, yes.

Q. This next paragraph, Paragraph 5, refers to the Emerging Threats reference, and then you have the archived URL. It shows "emerging-all.rules." Is that the all rules ruleset that you were referring to before that you would download from the Emerging Threats website?

A. That is right, sir.

Reply (Paper 20) 21; EX-1101 (Marinho Tr.) 45:11-22

Q. Was Emerging Threats the most well known third-party provider of SNORT rules at the time?

A. I mean yes. I believe so because they were the only one. So I believe so. Had another one? I simply don't know.

Reply (Paper 20) 24; EX-1101 (Marinho Tr.) 20:19-23

Emerging Threats rules detected exfiltration operations

```
#alert tcp $HOME_NET any -> $EXTERNAL_NET
1024:65535 (msg:"ET POLICY TLS/SSL Encrypted
Application Data on Unusual Port SSLv3";
flowbits:isset,BS.SSL.Established;
flow:established,to_server; content:"|17 03 00|";
depth:4; threshold:type limit, count 1, seconds
120, track by_src; classtype:unusual-client-port-
connection;
reference:url,doc.emergingthreats.net/2003021;
reference:url,www.emergingthreats.net/cgi-
bin/cvswb.cgi/sigs/POLICY/POLICY_SSL_TLS_on_High_P
ort; sid:2003021; rev:8;)
```

Pet. (Paper 2) 54; EX-1020 (Emerging Threats) 109

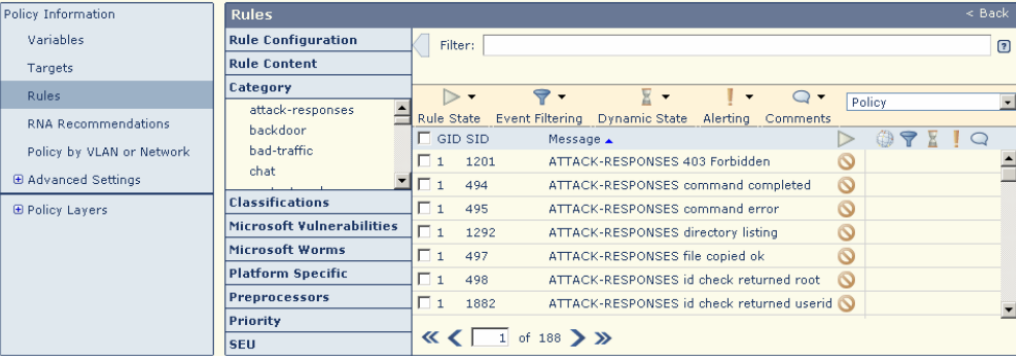
267. Thus this rule was examining all traffic outbound from the protected network and going towards a high-numbered server port somewhere on the Internet, and indicating the transmission of application data via SSL v3.0. Thus this rule is hoping to find exfiltration of some kind of encrypted data, or prevent loss of data via the use of outdated insecure versions of SSL/TLS. See Section IX C above.

Pet (Paper 2) 54; EX-1003 (Staniford Decl.) ¶267

229. A POSA would recognize that the operator in rule 2003021 is very similar to the ‘Require-TLS’ operator of the ‘343 patent. EX-1001 7:41-49, 8:7-

Pet (Paper 2) 43; EX-1003 (Staniford Decl.) ¶229

Sourcefire teaches changing rules to alert or drop



Pet. (Paper 2) 33-34; EX-1004 437

5. Select the rule or rules where you want to set the rule state. You have the following options:
 - To select a specific rule, select the check box next to the rule.
 - To select all the rules in the current list, select the check box at the top of the column.
6. You have the following options:
 - To generate events when traffic matches the selected rules, select **Rule State > Generate Events**.
 - To generate events and drop the traffic in inline deployments when traffic matches the selected rules, select **Rule State > Drop and Generate Events**.

Note that in passive deployments, including when an inline interface is in tap mode, setting the rule state to **Drop and Generate Events** causes rules to generate events without dropping packets.
 - To not inspect traffic matching the selected rules, select **Rule State > Disable**.

IMPORTANT! Sourcefire **strongly** recommends that you **do not** enable all the intrusion rules in an intrusion policy. The performance of your sensor is likely to degrade if all the rules are enabled. Instead, tune your rule set to match your network environment as closely as possible.

Pet. (Paper 2) 34; EX-1004 438

Sourcefire teaches changing rules to alert or drop

268. A POSA would understand that while Emerging Threats ships all rules as “alert” format for initial evaluation on a user network, the POSA would be motivated to change one or more rules to “drop” when tuning the rules to their network environment. As described above in Section IX A, a user of a Sourcefire system would import text rules and have them initially disabled, and would then tune their rule set to match the specific network environment, including selecting rules to activate and selecting whether to set each rule to block or alert. EX-1004 at 1857-1859, 438. For example, when using the Emerging Threats rules with Sourcefire, a user would enable select rules with an “alert” action to evaluate their performance in the user’s specific network environment, and then set the rules to “drop” once the user observes that they operate as desired. This would have the effect of blocking suspected exfiltration operations of this particular kind. Meanwhile, packets in connections that match the same 5-tuple, but that did not show this pattern (of SSL v3 application field headers), would be allowed through.

Pet. (Paper 2) 34; EX-1003 (Staniford Decl.) ¶268

289. As I discussed above in Section IX A, a POSA would understand that part of importing the Emerging Threats rules into Sourcefire and tuning their packet-filtering device was deciding which rules to change to a “drop.” A POSA would therefore understand that this rule could be set to drop which would have the obvious desirable effect of blocking transfers via SSL v3 on high numbered ports that might well be exfiltrations. See also Claim 1e above, as well as Section IX D.

Pet. (Paper 2) 67; EX-1003 (Staniford Decl.) ¶289

PETITIONER’S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

PO argues claims do not cover rules that are “overbroad” or that fail to preserve the flow of legitimate traffic without disruption

Claim 1[e] requires:

“...applying ... at least one packet transformation function
configured to prevent an exfiltration operation.”

Reply (Paper 20) 2; quoting EX-1001, Claim 1, 11:54-58

The point was (and remains) that it was known that overbroad rules could not be used to prevent exfiltrations at the time of the invention, and merely enabling an overbroad rule to drop traffic would render the network unusable. *See* Ex. 1085 at

PO Sur-Reply (Paper 22) 8

PO argues claims do not cover rules that are “overbroad” or that fail to preserve the flow of legitimate traffic without disruption

The underlying problem with the Emerging Threats rules, and all previous rule-based attempts to solve the exfiltration problem, were that they were overbroad, and such rules to actively drop traffic in a network environment would have made the network unusable. ’343 Patent at 7:24-40; Ex. 1085 at 112. That is why Emerging Threats wrote alert rules for use on an IDS device rather drop rules for use on an IPS device like the Sourcefire 3D Appliance: while useful for understanding and responding to intrusions, they would not have “preserv[ed] the flow of normal legitimate network traffic without disruption” if enabled to drop traffic in a network environment. *See* POR at 10-12; Ex. 1085 at 111, ¶ 9 and 112, ¶ 11.

PO Sur-Reply (Paper 22) 4

PO's claim construction argument would improperly exclude the HTTP-EXFIL and REQUIRE-TLS-1.1-1.2 embodiments

218

Five-tuple

Rule #	IP Protocol	Source IP Address	Source Port	Destination IP Address	Destination Port	Operator
1 (302)	TCP	140.210.*	*	140.212.*	22	ALLOW
2 (304)	TCP	140.210.*	*	140.212.*	25	ALLOW
3 (306)	TCP	140.210.*	*	140.212.*	110	ALLOW
4 (308)	TCP	140.210.*	*	140.212.*	143	ALLOW
5 (310)	TCP	140.210.*	*	140.212.*	443	REQUIRE-TLS-1.1-1.2
6 (312)	TCP	140.210.*	*	214.*	80	HTTP-EXFIL
7 (314)	*	*	*	*	*	BLOCK

Pet. (Paper 2) 9; EX-1001, Fig. 3

GET). Because attackers may often use HTTP PUT or POST methods to exfiltrate sensitive data, operators such as HTTP-EXFIL may be used to stop such exfiltrations.

Reply (Paper 20) 3; EX-1001 7:21-23

rule 5 310 may allow web browsers attached to network 104 to conduct HTTPS sessions (e.g., secure web sessions) with any web servers attached to network 106, but may, as described in greater detail below, utilize the REQUIRE-TLS-1.1-1.2 operator to ensure that only HTTPS secure web sessions using version 1.1 or 1.2 of the Transport Layer Security (TLS) protocol to secure such HTTPS sessions are allowed (e.g., because the popular TLS version 1.0 protocol has a known security vulnerability that attackers may exploit to decrypt HTTPS sessions).

Reply (Paper 20) 3; EX-1001 6:61-7:3

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Exfiltration operations trigger Emerging Threats rule 2003021

```
#alert tcp $HOME_NET any -> $EXTERNAL_NET
1024:65535 (msg:"ET POLICY TLS/SSL Encrypted
Application Data on Unusual Port SSLv3";
flowbits:isset,BS.SSL.Established;
flow:established,to_server; content:"|17 03 00|";
depth:4; threshold:type limit, count 1, seconds
120, track by_src; classtype:unusual-client-port-
connection;
reference:url,doc.emergingthreats.net/2003021;
reference:url,www.emergingthreats.net/cgi-
bin/cvswb.cgi/sigs/POLICY/POLICY_SSL_TLS_on_High_P
ort; sid:2003021; rev:8;)
```

Pet. (Paper 2) 54; EX-1020 (Emerging Threats) 109

An example of programmatic logic, written in pseudo-code, for a REQUIRE-TLS-1.1-1.2 operator is provided below. The REQUIRE-TLS-1.1-1.2 operator may be associated with filter rules for HTTPS sessions, such as rule 5310 of dynamic security policy 218. HTTPS may be used to encrypt HTTP sessions. HTTPS is not a protocol per se, but rather the result of layering the HTTP protocol on top of the TLS protocol. For an HTTPS session composed of IP packets, the application packets contained in the IP packets may be TLS Record Protocol packets. The header fields of TLS Record Protocol packets may not be encrypted. One of the header fields may contain a value indicating the TLS version.

Exemplary programmatic logic for a REQUIRE-TLS-1.1-1.2 operator, written in pseudocode, is as follows:

```
Operator REQUIRE-TLS-1.1-1.2(ip-packet):
Inspect app-pkt(ip-packet) match 1.0 return BLOCK;
Inspect app-pkt(ip-packet) match 1.1 return ALLOW;
Inspect app-pkt(ip-packet) match 1.2 return ALLOW;
Return BLOCK;
End Operator REQUIRE-TLS-1.1-1.2;
```

Pet. (Paper 2) 9-10, 43; Reply (Paper 20) 3; EX-1001 8::7-27

Exfiltration operations trigger Emerging Threats rule 2003021

Q. If the message parameter of rule 2003021 doesn't mention exfiltration and the comment doesn't mention exfiltration, why do you believe this rule is configured to prevent an exfiltration operation?

A. Well, you know, it goes back to what is -- you know, what else -- I mean, it's certainly not detecting any kind of intrusion, right. It's using an old version of SSL and a high-numbered port. It's not got any signature for -- you know, for detecting any particular kind of exploit technique, right. So it's basically -- so data is going out in a weird way, you know, by 2010, when this rule set was -- SSL version 3 was very old.

So you know, and my experience is we were using essentially this for exfiltration detection at FireEye. So that's why I think that.

Reply (Paper 20) 7; EX-2011 (Staniford Tr.) 102:5-23

Exfiltration operations trigger Emerging Threats rule 2003021

Q. And does that mean if -- well, if malware was exfiltrating data encrypted can SSL V3 from a network to a port between 1024 and 65535, would that trigger this rule?

A. It could trigger the rule.

Q. But there's also other traffic that would not be exfiltrating data that could trigger this rule; is that right?

A. Exactly. And these are the false positives that I'm trying to explain.

Q. So is it correct that this rule would both be triggered by malware exfiltrating data and other traffic that was not exfiltrating data?

A. That is correct, sir.

Reply (Paper 20) 5; EX-1101 (Marinho Tr.) 77:23-78:11

Q. While there might be some packets that are on those ports, if exfiltration was being performed using SSLv3 on a high port, would it trigger this rule?

A. So if you're asking -- generally, if a packet is using SSL on a high port that is not excluded, then it will alert. Again, this rule doesn't do anything, and then if you take away the pound sign, it would simply alert but not block any packets that match the rule.

Reply (Paper 20) 5; EX-1100 (Goodrich Tr.) 62:18-25

Exfiltration operations trigger Emerging Threats rule 2003021

Have a working string of sigs to identify a good ssl or tls session. They're targeted now at high ports, over 8081. The impetus to create this was a new bot that's using standard ssl for a command and control session on a high port. This catches that.

Reply (Paper 20) 5-6; EX-2018

Q. And is it correct that one of the reasons that malware would communicate with the command and control server is to exfiltrate data?

A. That's correct, sir, yes. One of the reasons. Yes, sir.

Reply (Paper 20) 6; EX-1101 (Marinho Tr.) 118:5-9

So you know, popular -- this is a very popular trojan thing to do, to basically every time you use it, fills in a form in the browser, swipe the data, and upload it to a malware command and control, and the theory that's often log-in credentials.

I mean, that requires a little more sophistication, but the message is, like, trojan leaking user data. It's pretty clear that that's the trojan exfiltration using data, so you don't want to --

Reply (Paper 20) 6; EX-2011 (Staniford Tr.) 139:12-22

Many Emerging Threats rules target exfiltration operations by matching application header field values and satisfy PO's construction

```
#from spyware LP data, by matt jonkman
#
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET
MALWARE Bizconcept.info Spyware Checkin";
flow:to_server,established; content:"POST "; depth:5; nocase;
uricontent:"/zuzu.php?&r="; nocase; classtype:trojan-activity;
reference:url,doc.emergingthreats.net/bin/view/Main/2005319;
reference:url,www.emergingthreats.net/cgi-
bin/cvsweb.cgi/sigs/MALWARE/MALWARE_Bizconcept.info; sid:2005319;
rev:3;)
```

*Pet. (Paper 2) 61-62; Reply (Paper 20) 8;
EX-1020 (Emerging Threats) 45*

It's looking at TCP traffic going from anywhere in the local network to remote HTTP servers, and it's saying if the content is POST -- so we have, again, the HTTP-EXFILL technique matching of -- you know, that it's a POST versus some other kind of HTTP method. Depth:5 is making sure that it's at the beginning of the packeted connection.

And then it's got some additional -- it's got basically one additional match, it has to be URL, has to have the /zuzu.php column and r=, right.

Reply (Paper 20) 8; EX-2011 (Staniford Tr.) 85:11-22

So the comment certainly suggests that this software is capable of exfiltrating data. I mean, I don't know what else, you know -- I don't know what else a POSA would read into spyware.

Reply (Paper 20) 8; EX-2011 (Staniford Tr.) 87:22-88:1

Many Emerging Threats rules target exfiltration operations by matching application header field values and satisfy PO's construction

```
#from spyware LP data, by matt jonkman
#
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET
MALWARE Bizconcept.info Spyware Checkin";
flow:to_server,established; content:"POST "; depth:5; nocase;
uricontent: "/zuzu.php?&r="; nocase; classtype:trojan-activity;
reference:url,doc.emergingthreats.net/bin/view/Main/2005319;
reference:url,www.emergingthreats.net/cgi-
bin/cvsweb.cgi/sigs/MALWARE/MALWARE_Bizconcept.info; sid:2005319;
rev:3;)
```

*Pet. (Paper 2) 61-62; Reply (Paper 20) 8;
EX-1020 (Emerging Threats) 45*

Q. And this rule, where it says, "content: 'POST'; depth:5," what does that mean?

A. It is attempting to match on the HTTP a method post that on the five first -- five initial bytes of the packet's payload.

Q. And that's an application header field value and an HTTP packet; right?

A. That is right.

Reply (Paper 20) 8; EX-1101 (Marinho Tr.) 106:2-9

Q. What are the -- what are the variety of things that spyware is used for?

A. Spyware, it can -- it can like take a screenshot of our screen and send it out over a common control to a server. It can capture what you're typing on a text file and send it out to a common control server. So spywares were designed to actually spy, collect information on the system and exfiltrate it.

Reply (Paper 20) 8; EX-1101 (Marinho Tr.) 110:5-13

Many Emerging Threats rules target exfiltration operations by matching application header field values and satisfy PO's construction

```
#by anon 4
#
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET
CURRENT_EVENTS FakeAlert/FraudPack/FakeAV/Guzz/Dload/Vobfus/ZPack
HTTP Post 1"; flow:established,to_server; content:"POST ";
depth:5; uricontent:"/senm.php?data="; nocase; content:"data=";
nocase; classtype:trojan-activity;
reference:url,threatinfo.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=TROJ_AGENT.GUZZ&VSect=T;
reference:url,www.threatexpert.com/report.aspx?
md5=7ca709f154e6abc678fbc4df8a3256b6;
reference:url,www.threatexpert.com/threats/trojan-fraudpack-
sd6.html; reference:url,doc.emergingthreats.net/2010234;
reference:url,www.emergingthreats.net/cgi-
bin/cvswb.cgi/sigs/CURRENT_EVENTS/CURRENT_fakealerts;
sid:2010234; rev:4;)
```

*Pet. (Paper 2) 64-65; Reply (Paper 20) 8-9;
EX-1020 (Emerging Threats) 18*

THE WITNESS: Well, certainly matching the content:"POST" again like the -- you know, like the patent says, you know. The patents disclose technique of exfiltration detection, right, in HTTP-EXFIL --

COURT REPORTER: (Requests clarification.)

THE WITNESS: Of exfiltration detection, you know, in HTTP-EXFIL, which is the name of an operator that the patent uses as an example. You know, the idea there is it would allow the GETs through and block other HTTP methods.

This is doing the -- now, do I think that's actually a very good exfiltration technique? No. But that's what the patent is saying. And this is certainly matching on contents, right, with the addition that it's narrowed to a specific URL, right. So it's testing a uricontent:"/senm.php?data=" and then it's looking for another data=, right. So these

Reply (Paper 20) 9; EX-2011 (Stanford Tr.) 81:4-24

Many Emerging Threats rules target exfiltration operations by matching application header field values and satisfy PO's construction

Q. And in order for this rule to alert, it would have to both match in the application packet header -- well, the application packet header values would have to include the post method, and they would also have to include the URI content/SEMM.php?data equals; right?

A. That is correct, sir.

Q. And so if a packet was leaving the network that matched the packet header field values set forth in this rule, and also included the application header field values post and the URI content specified here, that would then trigger the rule; right?

A. That's correct, sir, yes.

Reply (Paper 20) 9; EX-1101 (Marinho Tr.) 115:6-18

Q. But this rule -- if I understand right, this rule was designed for sending an alert when it detects that there is outbound traffic from malware to a command and control center; right?

A. Yes. That's correct.

Q. And is it correct that one of the reasons that malware would communicate with the command and control server is to exfiltrate data?

A. That's correct, sir, yes. One of the reasons. Yes, sir.

Reply (Paper 20) 9; EX-1101 (Marinho Tr.) 117:25-118:9

Sourcefire and Emerging Threats Renders Obvious Claims 5 and 12

The Petition and Mr. Marinho's testimony rebut PO's Response argument. *Reply (Paper 20) at 13*

Petitioner argues that claims 5 and 12 are taught by Emerging Threats rule

SID: 2010234. Pet. at 64-66. However, Petitioner fails to identify any analysis of application header field values.

PO Response (Paper 15) 26

function, such as alert or drop) if the method is "POST." The operator is also checking whether the request is to a specific resource on the target server, specifically it checks that 'uricontent: "/senm.php?data="', such that the second criterion will only be matched if the request is going to the url "/senm.php?data=".
Pet. (Paper 2) 64

("Guzz/Dload/Vobfus/ZPack"). EX-1003 ¶284. Therefore, when the "application header field value indicates a request for specified resource data," for example because the request method was a GET, the rule would not trigger and the packets would be allowed through to continue toward their respective destinations.

Pet. (Paper 2) 65

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Q. And so if a packet was leaving the network that matched the packet header field values set forth in this rule, and also included the application header field values post and the URI content specified here, that would then trigger the rule; right?

A. That's correct, sir, yes.

Reply (Paper 20) 13; EX-1101 (Marinho Tr.) 115:13-18

Sourcefire and Emerging Threats Renders Obvious Claims 5 and 12

PO's Sur-Reply argues that URI content is not “specified resource data”

C. Claims 5 and 12

Claims 5 and 12 are patentable because Emerging Threats Rule 2010234 does not include an application layer header field that identifies specified resource data, and because a POSITA would not have been motivated to modify the rule to drop packets. POR at 26-27.

First, Petitioner has never attempted to explain with specificity why the URI content specified in the rule is “specified resource data,” and it made no effort to cure that basic deficiency in the Reply. Pet. at 64-65; Reply at 13. The '343 Patent uses the term “resource” to refer the parties to a communication session, which are identified by 5-tuple data (i.e. IP addresses and port numbers). See '343 Patent at 8:45-52 (“Conceptually, the first stage may determine if the network policy allows any communications between the resources identified in the 5-tuple rule; if so, the second stage may determine if the policy allows the specific method or type of communication (e.g., file read, file write, encrypted communication, etc.) between the resources.”) (emphasis added).

PO Sur-Reply (Paper 22) 14-15

PETITIONER'S DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

allowing, based on a determination that the application header field value indicates a request for **specified resource data**, each packet of the second portion of packets indicating a request for **specified resource data** to continue toward its respective destination;

EX-1001 ('343 Patent) Claim 5