



SEE EVERYTHING, FEAR NOTHING

Threat Solution Series

SPEAR PHISHING



WHAT IS SPEAR PHISHING?

Spear phishing is an attempt to entice a specifically targeted victim to open a malicious attachment or visit a malicious website with the intent of gaining insight into confidential data and/or acting on nefarious objectives against the victim's organization. A common tactic used by an attacker is a spoofed email address designed to look like it's coming from a source that is trusted by the victim. Reconnaissance and social engineering tactics may also help produce content and wording that makes the delivery email more believable to the victim.

A Typical Attack Scenario

A common tactic used in spear phishing campaigns is delivery of a malicious file as an email attachment. The attachment is often a common file format (zip, rtf, doc, xls) with an embedded executable or exploit that serves to provide the attacker a foothold in the environment. One common delivery mechanism is by way of an executable file embedded within an obfuscated zip:

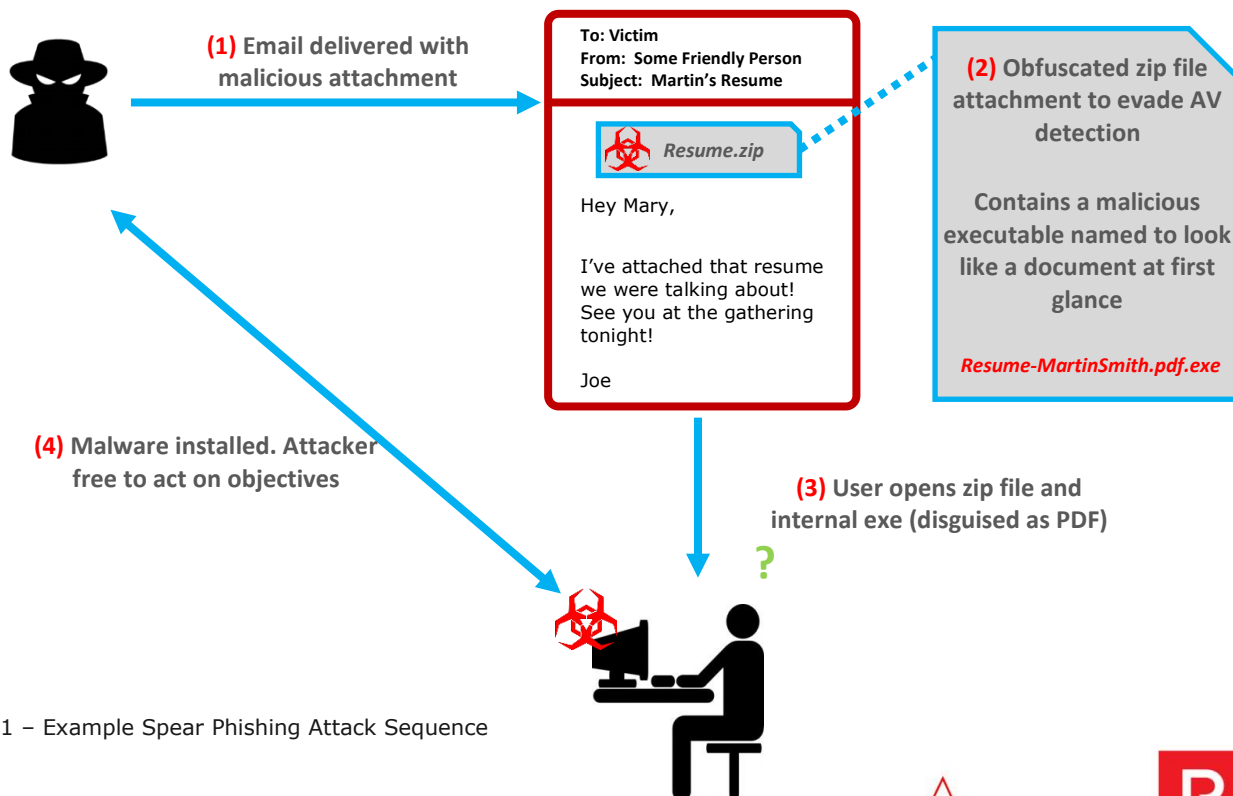


Figure 1 – Example Spear Phishing Attack Sequence

Detection and Response

A motivated attacker can get a weaponized file through traditional signature-based email security solutions. Traditional tools must rely on signatures and are easily left blind by intentional obfuscation of attachments and embedding of unique malicious code. In order to effectively respond to spear phishing attacks, defenders must maximize visibility into each stage of the attack lifecycle in order to understand the delivery mechanism, the infection (i.e. did the user fall for it), and the impact to the business by having full visibility into network, endpoint, and user activity. The following chart contrasts the visibility by attack stage into an attacker’s tools, tactics, and procedures (TTPs) provided by traditional tools with the RSA Advanced SOC Portfolio:

	<u>Delivery</u>	<u>Exploit/Installation</u>	<u>C2</u>	<u>Action</u>
	Targeted Email Attachment Embedded Links	Opening of targeted malware on the endpoint Installation and hooking into the system	Malware Beaconing	Data Exfiltration Lateral Movement Disruption
AV/FW/IDS/IPS:				
Traditional SIEM:				
RSA ASOC:				

No Visibility	Partial Visibility/Signature	Full Visibility
---------------	------------------------------	-----------------

The ability to reconstruct the entire email session (analysts are great at confirming whether an email is truly phishing) as well as extract and perform analysis on all attachments is crucial to understanding the delivery mechanism. The ability to extract the initial payload is an invaluable way for investigators to perform deep analysis on potentially malicious files. Furthermore, the only way to truly determine whether or not an end user fell victim to the attack is to have deep visibility into the endpoint without relying on signature-based anti-virus solutions (a motivated attacker can easily evade AV).

SPEAR PHISHING VISIBILITY WITH RSA ASOC – DETAILS

Key solutions: RSA Security Analytics for Packets, RSA ECAT

In our example, RSA Security Analytics for Packets detects a suspicious zip file as an attachment to an email being sent to an employee:

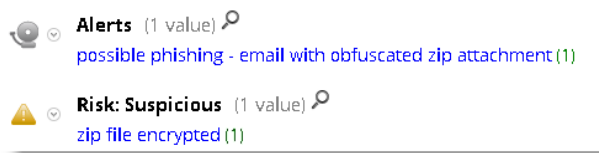


Figure 2 – RSA Security Analytics Alert

The analyst then drills through to view the entire email message to get a clearer picture:

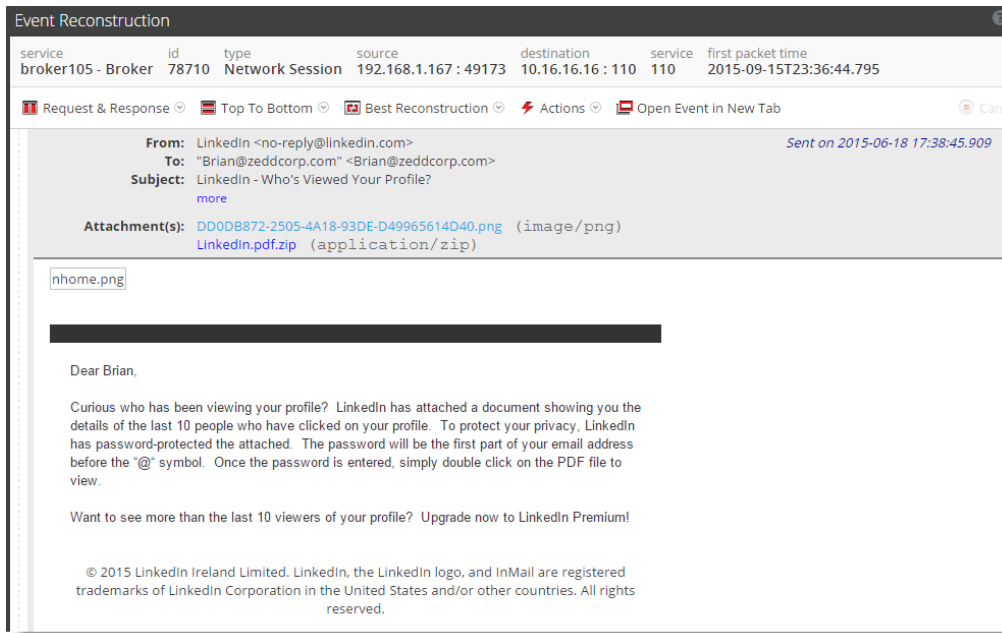


Figure 3 – A Closer Look at the Email Message

While this may look normal to an unsuspecting end user, the analyst can quickly see the suspicious nature of this email by examining the headers, paying attention to the wording of the email message, and examining the atypical attachment. The analyst may later choose to extract the attachments from the email for further analysis. Having confirmation that this is most likely a spear phishing attack, the next logical question is "did the user click on the attachment?" To answer that, the first thing the analyst needs to determine is which machine the user was working from when the email was received. Looking at the metadata of the reconstructed email, the analyst can see the username "bcline" has been associated with this email address through a context feed deployed in RSA Security Analytics:

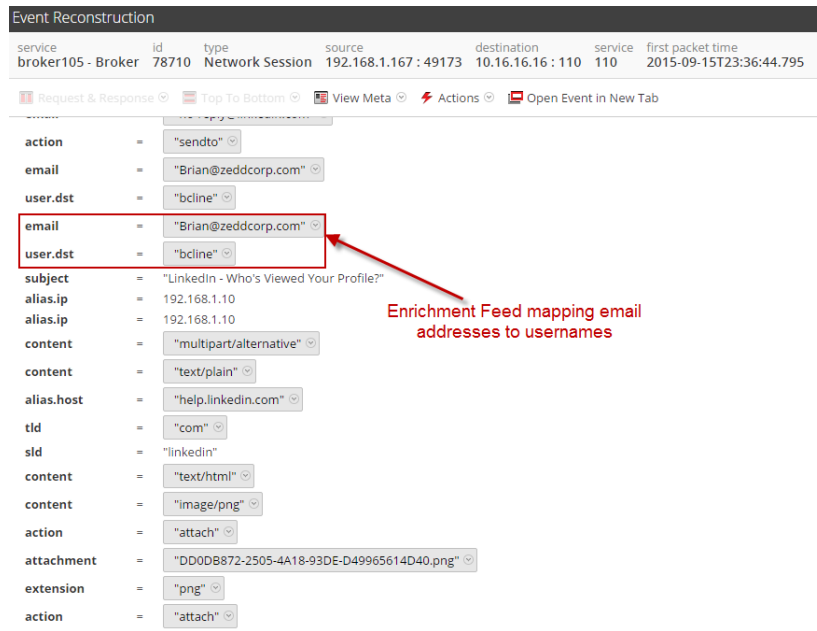


Figure 4 – Metadata from the Email

With the username determined, any related logon history can be retrieved from the domain controller logs in order to focus the investigation on the proper endpoint:

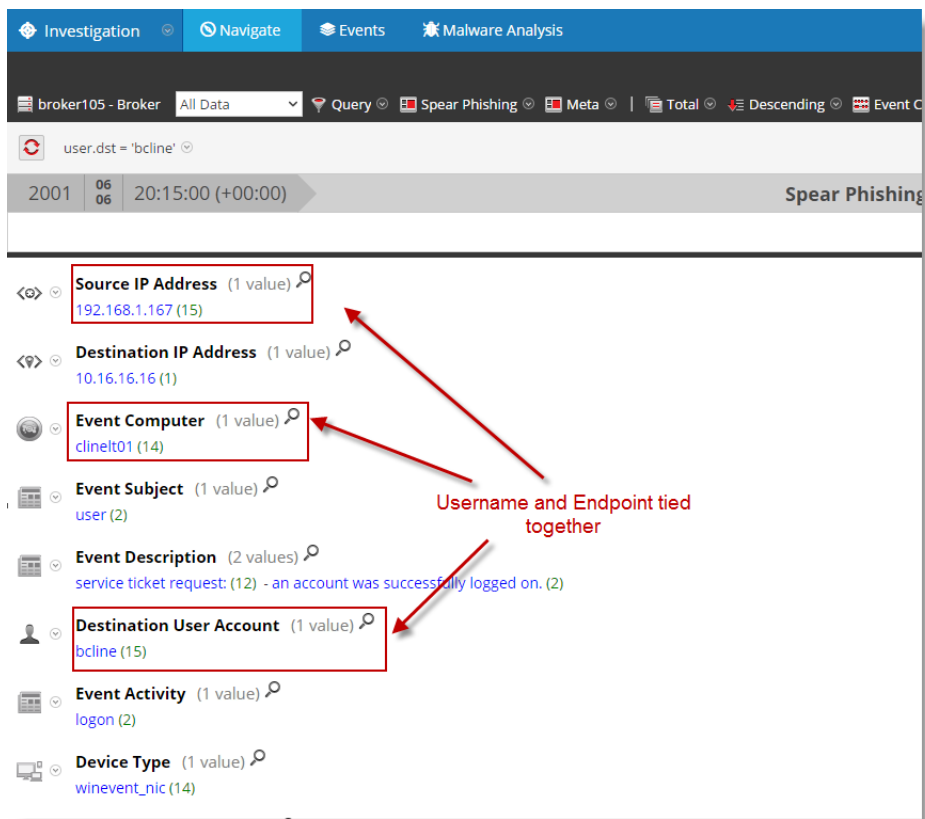


Figure 5 – Using Logs to Determine Where the User was Logged in at the Time of the Email

Next, knowing the machine name, the analyst can pivot directly into RSA ECAT see whether or not any malicious behavior has been detected:

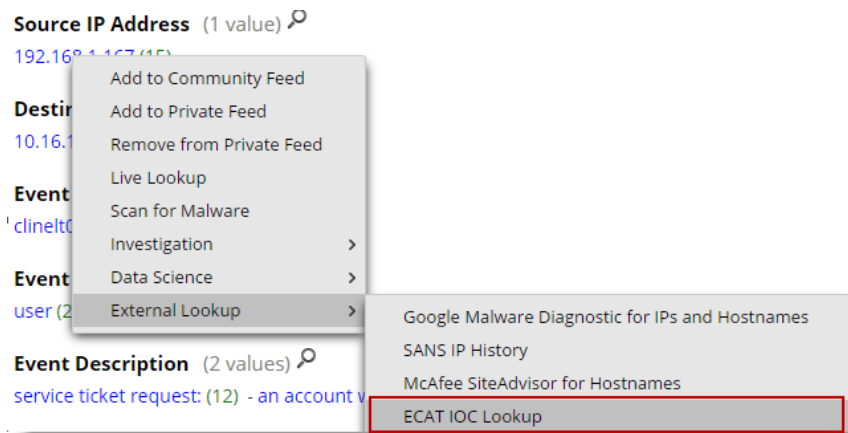


Figure 6 – Direct Pivot into ECAT

The pivot brings up the machine summary page for CLINELT01:

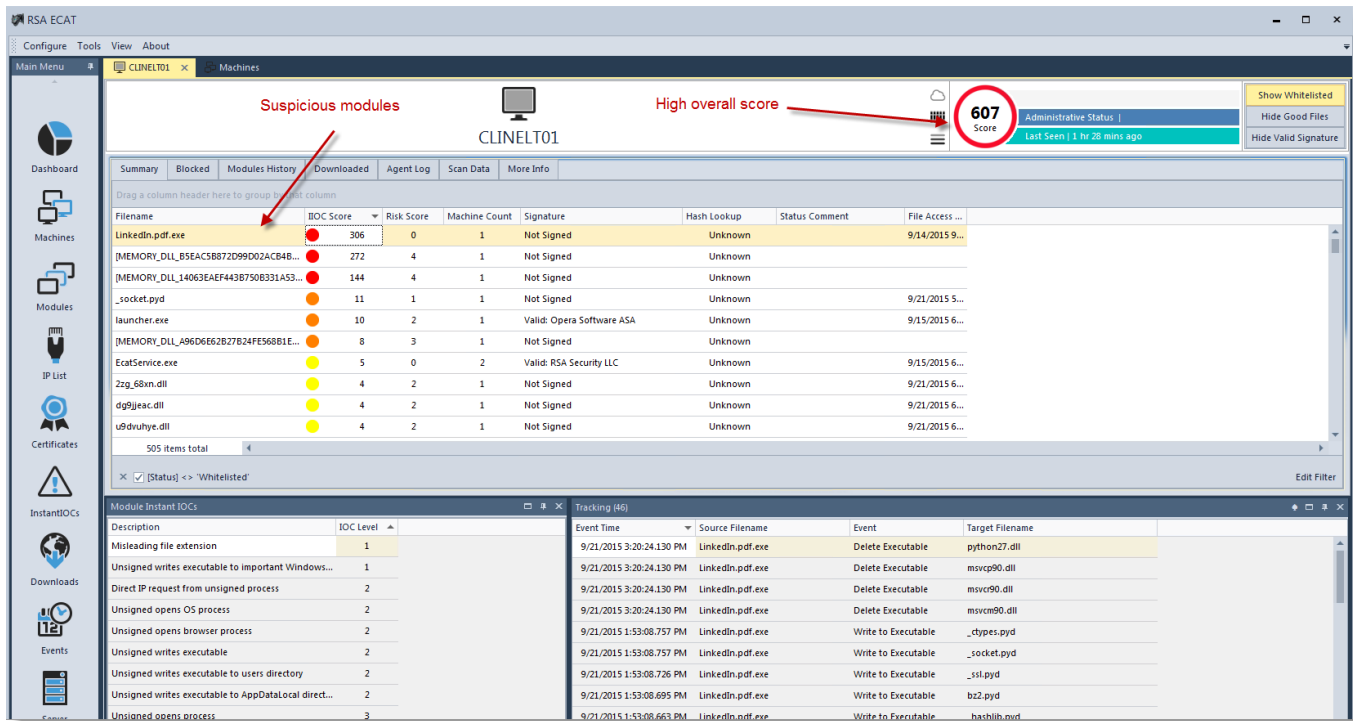
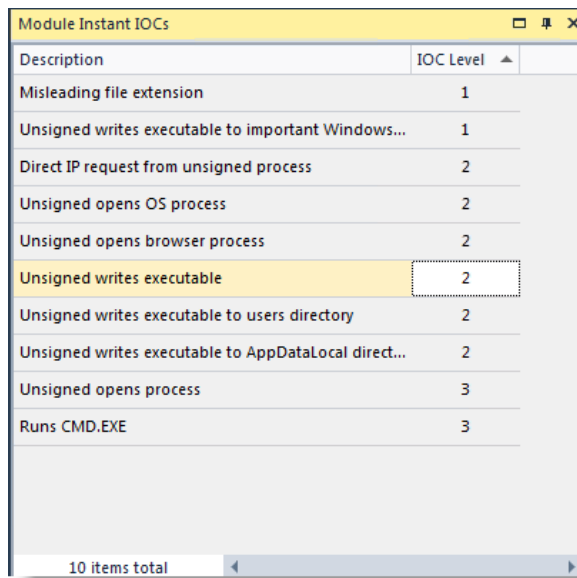


Figure 7 – Summary Information of What RSA ECAT Sees on CLINELT01

The analyst immediately notices that RSA ECAT is reporting a highly suspicious score. The summary table makes reference to a number of malicious processes (with high scores in red) including injected DLLs (top) that are all reporting bad behavior. Most notable is the attachment from the original email, *LinkedIn.pdf.exe*. Drilling into *LinkedIn.pdf.exe*, the analyst can see a large number of suspicious properties, behaviors, and network connections:



Figures 8 – 11 Suspicious Instant IOCs Tied to *LinkedIn.pdf.exe*'s Behavior

The Module Instant IOCs show all things impacting the score of the executable and call out many suspicious behaviors such as opening a command shell, spawning browser processes, and making direct connections to an IP address on the internet. The analyst can get a timeline of many of these behaviors by opening up the "Tracking" and "Network" tabs:

Event Time	Source Filename	Event	Target Filename
9/21/2015 1:53:08.429 PM	LinkedIn.pdf.exe	Write to Executable	msvc90.dll
9/21/2015 12:17:40.595 PM	LinkedIn.pdf.exe	Self Delete Executable	cmd.exe
9/21/2015 12:17:40.595 PM	explorer.exe	Delete Executable	LinkedIn.pdf.exe
9/21/2015 12:17:39.799 PM	LinkedIn.pdf.exe	Delete Executable	python27.dll
9/21/2015 12:17:39.799 PM	LinkedIn.pdf.exe	Delete Executable	msvc90.dll
9/21/2015 12:17:39.799 PM	LinkedIn.pdf.exe	Delete Executable	msvcr90.dll
9/21/2015 12:17:39.799 PM	LinkedIn.pdf.exe	Delete Executable	msvc90.dll
9/21/2015 11:54:05.463 AM	WmiPrivSE.exe	Open Process	LinkedIn.pdf.exe
9/21/2015 11:52:15.932 AM	LinkedIn.pdf.exe	Create Process	cmd.exe
9/21/2015 11:51:59.489 AM	LinkedIn.pdf.exe	Open Process	LinkedIn.pdf.exe
9/21/2015 11:51:59.488 AM	LinkedIn.pdf.exe	Open Browser Process	opera.exe
9/21/2015 11:51:59.488 AM	LinkedIn.pdf.exe	Open Process	audiodg.exe

46 items total

Tracking (46) Network (6) Paths (1) Machines (1) Autoruns Diagram

Figure 9 – Full Behavior Tracking Timeline

Process	Module	IP	Domain	Port	Listen
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...	68.146.213.32	68.146.213.32:8080	8080	<input type="checkbox"/>
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...	68.146.213.32	68.146.213.32:8080	8080	<input type="checkbox"/>
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...	68.146.213.32	68.146.213.32:8080	8080	<input type="checkbox"/>
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...	68.146.213.32	68.146.213.32:8080	8080	<input type="checkbox"/>
LinkedIn.pdf.exe		68.146.213.32	68.146.213.32:8080	8080	<input type="checkbox"/>
LinkedIn.pdf.exe		68.146.213.32	68.146.213.32:8080	8080	<input type="checkbox"/>

6 items total

Tracking (46) Network (6) Paths (1) Machines (1) Autoruns Diagram

Figure 10 – Network Tracking for LinkedIn.pdf.exe

The analyst notices a lot of activity generated by the suspicious executable, using an injected DLL to connect to 68.146.213.32 over port 8080 and will pivot back into RSA Security Analytics to see if anything else can be gathered. Before doing that, however, the analyst quickly browses to the “Machines” tab to see if this executable has been seen on any other endpoints in the environment:

Machi...	Machine Name	Admin Status
	CLINELT01	

1 items total

Tracking (46) Network (6) Paths (1) Machines (1) Autoruns Diagram

Figure 11 – Validating that CLINELT01 is the Only Endpoint with this Executable

Focusing back on the network data, the analyst can pivot back into RSA Security Analytics to investigate:

Process	Module	IP	Domain	Port	Listen
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...]	68.146.213.32	68.146.213.32:8080	8080	<input type="checkbox"/>
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...]	68.146.213.32		8080	<input type="checkbox"/>
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...]	68.146.213.32		8080	<input type="checkbox"/>
LinkedIn.pdf.exe	[MEMORY_DLL_B5EAC5B872D99D0...]	68.146.213.32		8080	<input type="checkbox"/>
LinkedIn.pdf.exe		68.146.213.32		8080	<input type="checkbox"/>
LinkedIn.pdf.exe		68.146.213.32		8080	<input type="checkbox"/>

Figure 12 – Pivot into RSA Security Analytics to Investigate Network Traffic to 68.146.213.32

This drill will bring the analyst directly to a view of all network traffic destined to 68.146.213.32 in order to determine if anything else happened:

Investigation | Navigate | Events | Malware Analysis

broker105 - Broker | Custom | 2015-09-01 06:48:00 | 2015-09-22 00:09:45 | Go | Query | Spear Phishing | Meta | Total | Descending

ip.dst=68.146.213.32

2015 09 01 06:48:00 (+00:00) | Spear Phishing : Custom

Alerts (4 values)

- outbound (2) - rfc1918 src (2) - known service over non-standard port (1) - outbound exe (1)

Risk: Informational (19 values)

- exe abnormal_e_cblp (1) - exe abnormal_e_cparhdr (1) - exe abnormal_e_crc (1) - exe abnormal_e_cs (1) - exe abnormal_e_csum (1) - exe abnormal_e_ip (1) - exe abnormal_e_oeminfo (1) - exe abnormal_e_res1_1 (1) - exe abnormal_e_res1_2 (1) - exe abnormal_e_res1_3 (1) - exe abnormal_e_res1_4 (1) - exe abnormal_e_res1_5 (1) - ssl over non-standard port (1)

Risk: Suspicious (4 values)

- escalation multiple informational (1) - exe suspicious (1) - plaintext ftp password (1) - ssl 3.0 (1)

Risk: Warning (1 value)

- exe many dos header anomalies (1)

Service Type (2 values)

- FTP (1) - SSL (1)

TCP Destination Port (2 values)

- 21 (ftp) (1) - 8080 (1)

Source IP Address (1 value)

- 192.168.1.167 (2)

Destination IP Address (1 value)

- 68.146.213.32 (2)

Extension (4 values)

- dll (1) - doc (1) - pdf (1) - xls (1)

Network evidence to add to the investigation

Figure 13 – Continuing the Network Investigation with New Information Gleaned from RSA ECAT

The analyst can now drill into the connection on port 8080 that was seen in RSA ECAT, or pivot to other notable activity, specifically the FTP connection outbound to the same implicated IP address. With a couple clicks, they open up the FTP session and reconstruct it to a human-friendly view:

service	id	type	source	destination	service	first packet time
broker105 - Broker	78713	Network Session	192.168.1.167 : 1070	68.146.213.32 : 21	21	2015-09-15T23:37:55.414

Request	Response
PASS GOV10MIL	230>User mao2 has group access to: 925 230 OK. Current directory is /
TYPE I	200 TYPE is now 8-bit binary
MKD ./yzZTtbhcHR1KKdJf	257 "yzZTtbhcHR1KKdJf" : The directory was successfully created
ALLO 39904	200 Zzz...
PASV	227 Entering Passive Mode (86,57,246,177,32,15)
STOR ./yzZTtbhcHR1KKdJf/1051a1IowaDems.pdf	150 Accepted data connection 226-File successfully transferred 226 4.381 seconds (measured here), 8.90 Kbytes per second
ALLO 65307	200 Zzz...

Uploaded Data

↓

processed 100 of 223 packets; 1 new event(s) Show Reconstruction Log

Figure 14 – Evidence of Data Exfiltration via an FTP Session to our Attacker at 68.146.213.32

Using RSA Security Analytics and RSA ECAT, the analyst was able to gain visibility into the Delivery, Exploitation, Installation, Command-and-Control, and Action phases of this attack.

REFERENCES

Trend Micro Spear Phishing Analysis: <http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-spear-phishing-email-most-favored-apt-attack-bait.pdf>
 Cyber Kill Chain: <http://www.lockheedmartin.ca/us/what-we-do/information-technology/cyber-security/cyber-kill-chain.html>