v2

Threat Hunting Professional

Hunting with PowerShell

Section 03 | Module 05

Table of Contents

5.1

Introduction

5.1 Introduction

In this module, we'll look at some tools, built with PowerShell, that are designed to gather and scan data at a large scale for incident response and threat hunting purposes.

5.1 Introduction

We will also look at some new tools created by Microsoft that can aid us in hunting for and catching malicious actions and/or attacks against machines in our environment.

Lastly, we will look at some additional techniques on how to minimize and defend against the misuse of PowerShell in our environments, aside from just for hunting for malicious actions.

PowerShell Hunting Tools

5.2.1 Kansa

Kansa is a PowerShell incident response framework.

This framework can be used in the enterprise to collect data for use during an incident response, breach hunts, or for building an environment baseline.

You can download Kansa from GitHub here.

https://github.com/davehull/Kansa

5.2.1 Kansa

The primary use of Kansa is to collect data from many hosts.

It takes advantage of Windows Remote Management and PowerShell's ability to run jobs across multiple machines in parallel.

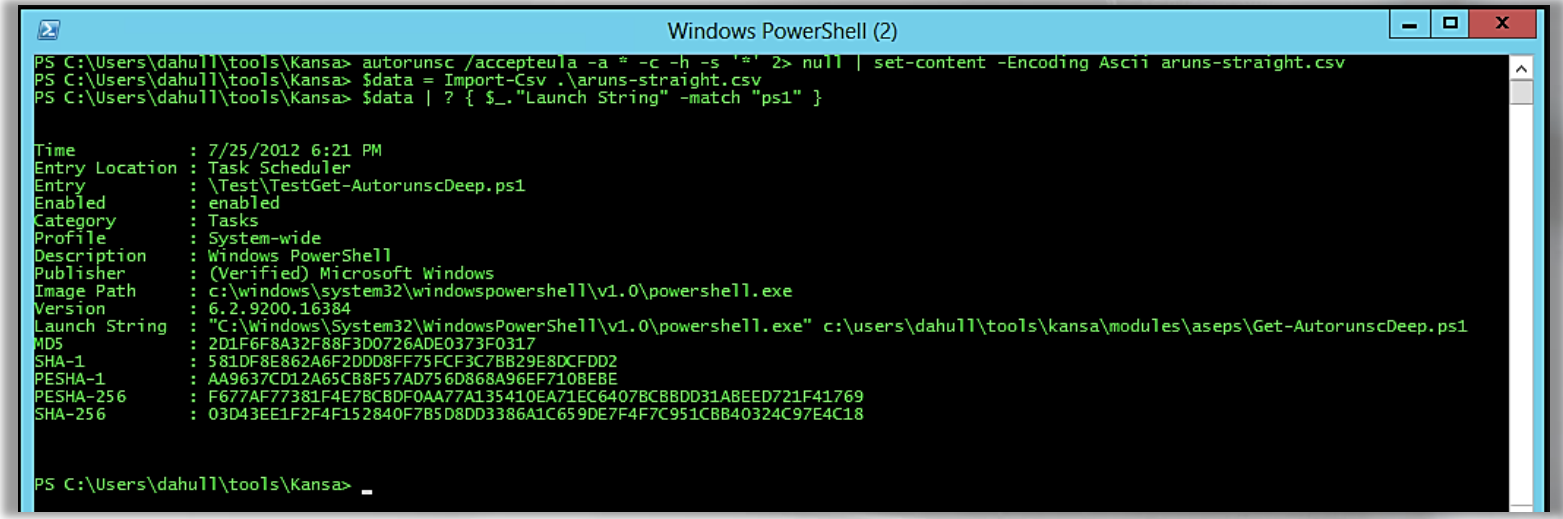
5.2.1 Kansa

Kansa was designed to be modular. It features a core script, collector modules, and analysis scripts. These analysis scripts can perform frequency analysis of specific fields in a given data set.

To enable these capabilities, Kansa requires **LogParser**, a tool we already discussed while hunting for web shells.

5.2.1 Kansa

Here is a screenshot of Kansa extracting autoruns and filtering those that contain “ps1”:



https://trustedsignal.blogspot.com/search/label/Kansa

5.2.1 Kansa

Its suggested that you download Kansa and get familiar with this tool.

You can read more about the functionality of Kansa here and here.

http://www.powershellmagazine.com/2014/07/18/kansa-a-powershell-based-incident-response-framework/

5.2.2 PSHunt

PSHunt is a PowerShell Threat Hunting Module designed to scan remote endpoints for indicators of compromise, or survey them for more comprehensive information related to the state of those systems (active processes, autostarts, configurations, and/or logs).

PSHunt is divided into several modules, functions, and folders. Below are the modules of PSHunt:

- Scanners
- Survey
- Discovery
- Utilities
- Analysis

5.2.2 PSHunt

You can download PSHunt from GitHub here.

You can also view a presentation on PSHunt from BSidesLV 2016 here.

https://github.com/Jean-Francois-T/PSHunt
https://www.youtube.com/watch?v=2MrrOxsJk_M

5.2.3 NOAH

NOAH is an agentless open source Incident Response framework based on PowerShell, called "**No Agent Hunting**" (NOAH), to help security investigation responders to gather a vast number of key artifacts without installing any agent on the endpoints, saving precious time.

NOAH was revealed at Black Hat USA 2017 in a presentation called “NOAH: UNCOVER THE EVIL WITHIN! RESPOND IMMEDIATELY BY COLLECTING ALL THE ARTIFACTS AGENTLESSLY”. You can download the tool from GitHub, here.

https://www.blackhat.com/us-17/arsenal/schedule/#noah-uncover-the-evil-within-respond-immediately-by-collecting-all-the-artifacts-agentlessly-7965
https://github.com/giMini/NOAH

5.2 PowerShell Hunting Tools

You should become familiar with some, if not all of these tools, as a threat hunter in the enterprise.

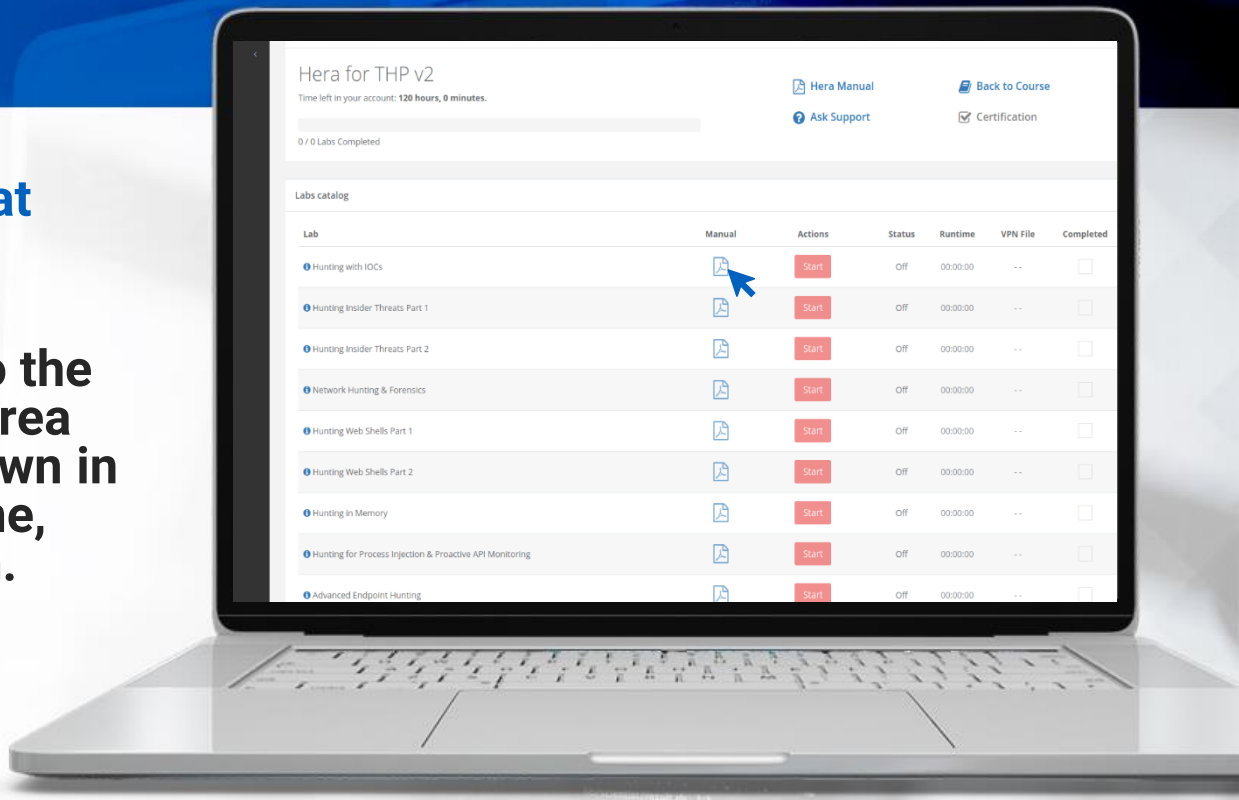
You want to hunt efficiently at scale, and PowerShell is a great tool to aid us with hunting.

5.2.4 Hera Lab

Put what you've learned to practice with the **Hunting at Scale with Osquery** lab!

To ACCESS your lab, go to the course in your members area and click the labs drop-down in the appropriate module line, then click the manual icon.

All labs are only available in Full or Elite Editions of the course. To upgrade, click **LINK**.



***NOTE: some courses contain several labs and manuals, please make sure to click the file icon as it may be a zip that contains multiple lab manuals.**

Windows Advanced Threat Protection

5.3 Windows Advanced Threat Protection

Windows Defender Advanced Threat Protection (ATP) provides preventative protection, detects attacks and zero-day exploits, and gives you centralized management for your end-to-end security lifecycle.

You can review more information about the product on its official page here.

5.3 Windows Advanced Threat Protection

Windows Defender ATP is agentless and built into the operating system.

ATP can adapt to changing threats, deploy new defenses, and orchestrate remediation.

5.3 Windows Advanced Threat Protection

ATP uses the following to protect you from advanced threats:

- Windows Defender System Guard
- Windows Defender Application Guard
- Windows Defender Exploit Guard
- Windows Defender Antivirus
- Windows Defender Application Control

5.3 Windows Advanced Threat Protection

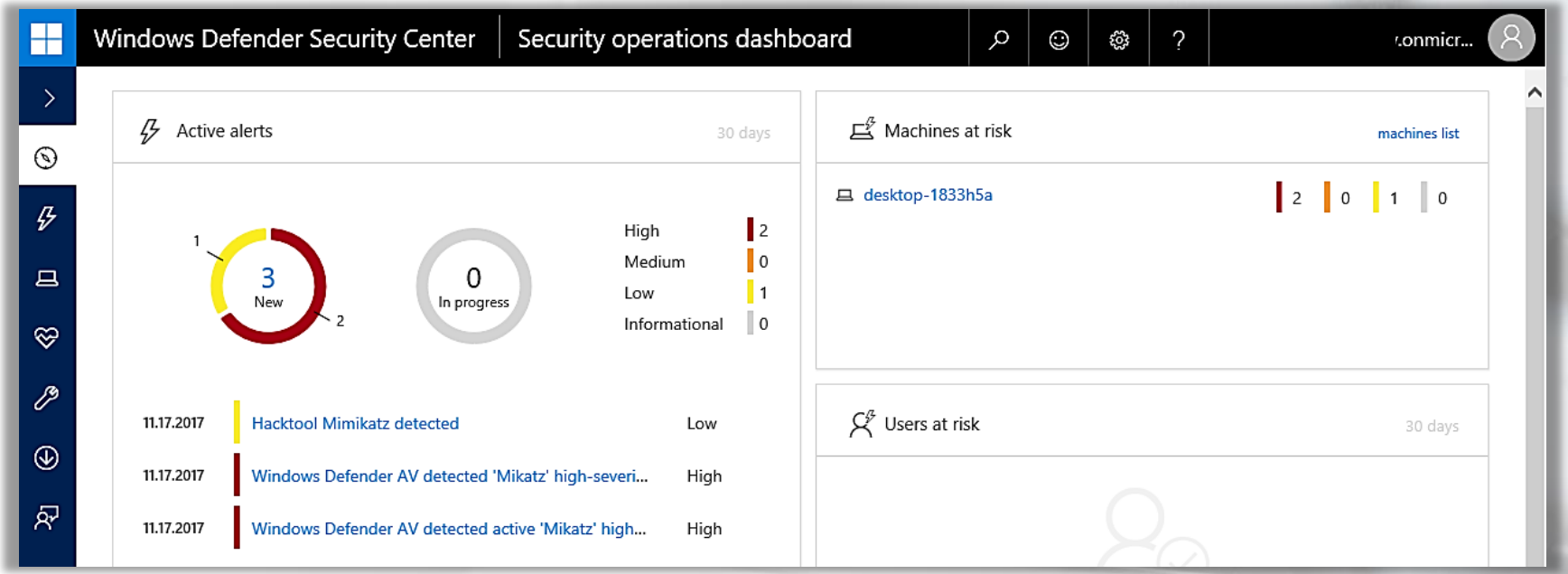
Microsoft also advertises it as a threat hunting tool: “Instantaneously search and explore 6 months of historical data across your endpoints.”

To obtain a trial copy of Windows Defender ATP, you must agree to the Trial Online Service Terms and register for the product. If you’re approved, then you will be given a 90-day trial to test-drive ATP. Visit this link here to begin the process.

https://winatpregistration-prd.trafficmanager.net/UserAgreement

5.3 Windows Advanced Threat Protection

ATP Dashboard

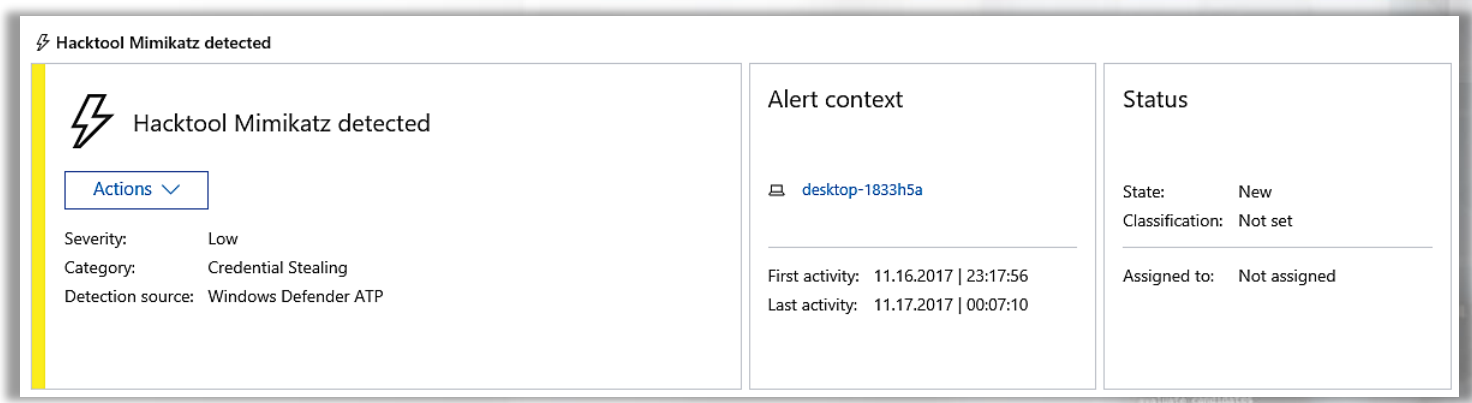


5.3 Windows Advanced Threat Protection

In the previous screenshot, ATP successfully detected Mimikatz simply being dropped onto the machine, without being executed.

When we click on the alert, we're presented with another window which provides more information.

5.3 Windows Advanced Threat Protection

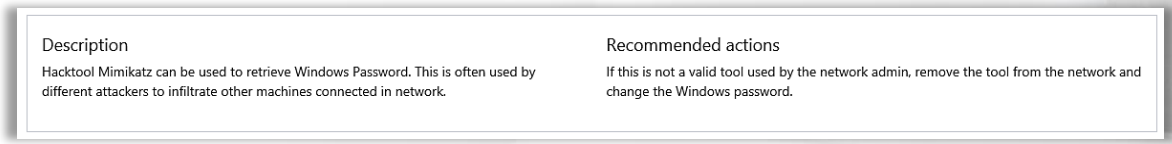


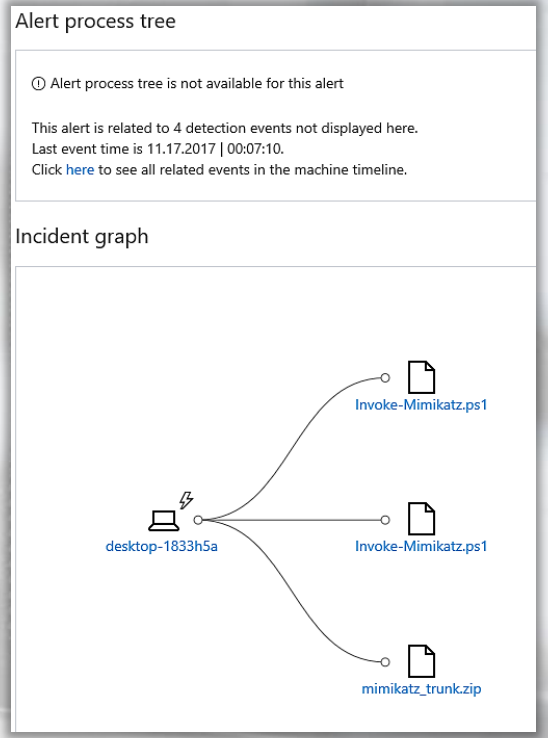
From the above screenshot, we see that it gives us a severity level, in this case it's low. It tells us the type of malware this would fall under; in this case, it would be credential stealing. We also see the machine that was affected along with date & timestamps.

5.3 Windows Advanced Threat Protection

The alert also gives us a brief description of the malware, recommended actions, an alert process tree, and an incident graph.

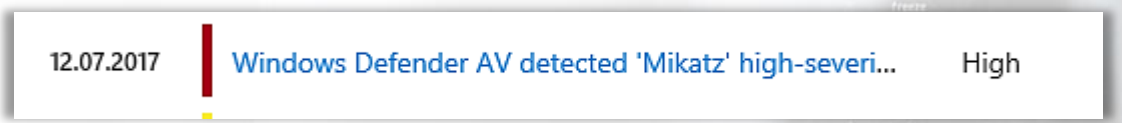
From the incident graph, we can see what was dropped onto the machine. In this case, a zip file containing Mimikatz and a PowerShell-based Mimikatz script.



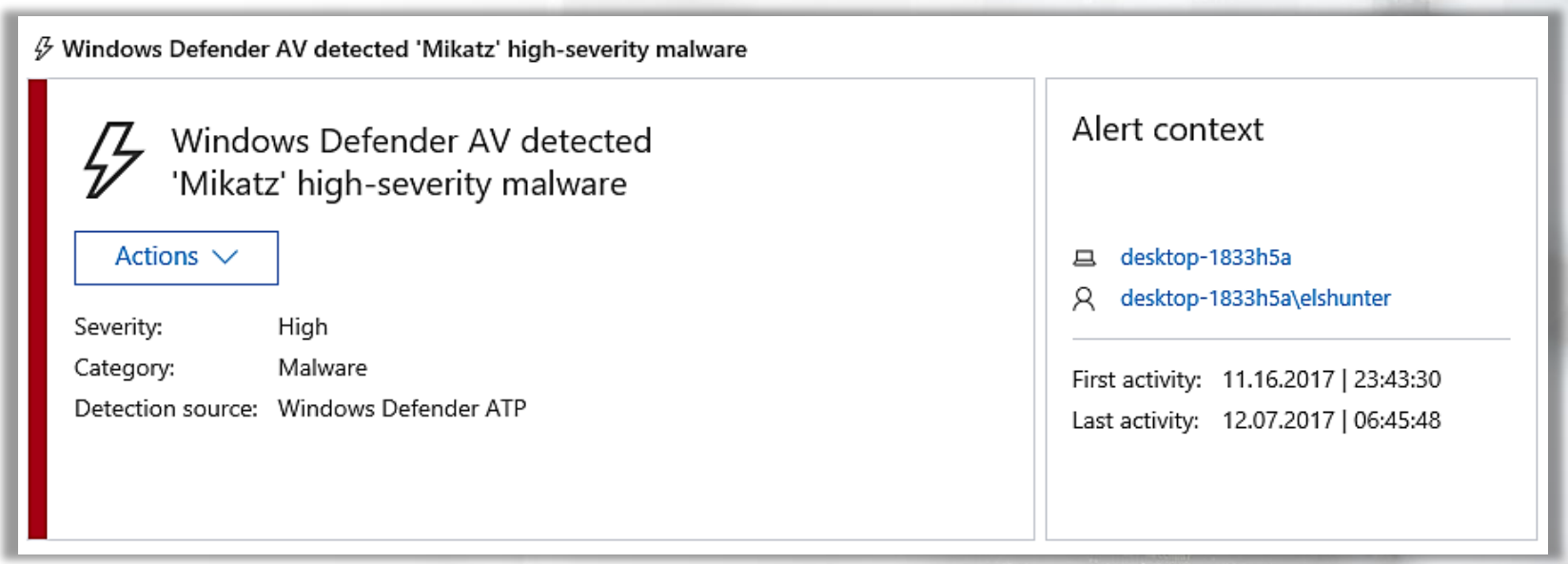


5.3 Windows Advanced Threat Protection

Now, let's look at one of the high alerts on the dashboard, which seems to also be related to Mimikatz.



5.3 Windows Advanced Threat Protection



5.3 Windows Advanced Threat Protection

Description

High-severity malware refers tools used by advanced Threat Activity Groups to target victims. Such Activity Groups may target individuals or institutions. They typically engage on industrial, military, diplomatic, and political espionage rather than more mundane activities such as identity theft or denial of service attacks. Some groups engage in acts of deliberate sabotage and destruction in order to cause real-world effects, such as disruptions to the victim's operations.

This category of malware includes tools such as:
- Exploits used to gain access to targeted computers or escalate privileges on infected computers;
- Backdoors used to maintain persistent command and control over infected computers in a stealthy manner;
- Lateral movement tools that permit attackers to scan the local network, locate targets of interest, and access additional computers;
- Counter-forensics tools used to delay and disrupt incident response activities, including destructive malware that can render computers inoperable;
- "Weaponized" tools that enable acts of deliberate sabotage or destruction or denial of service.

Recommended actions

A. Validate the alert.
1. Check for other suspicious activities in the machine timeline.
2. Locate unfamiliar processes in the process tree. Check files for prevalence, their locations, and digital signatures.
3. Submit relevant files for deep analysis and review file behaviors.
4. Identify unusual system activity with system owners.

B. Scope the incident. Find related machines, network addresses, and files in the incident graph.

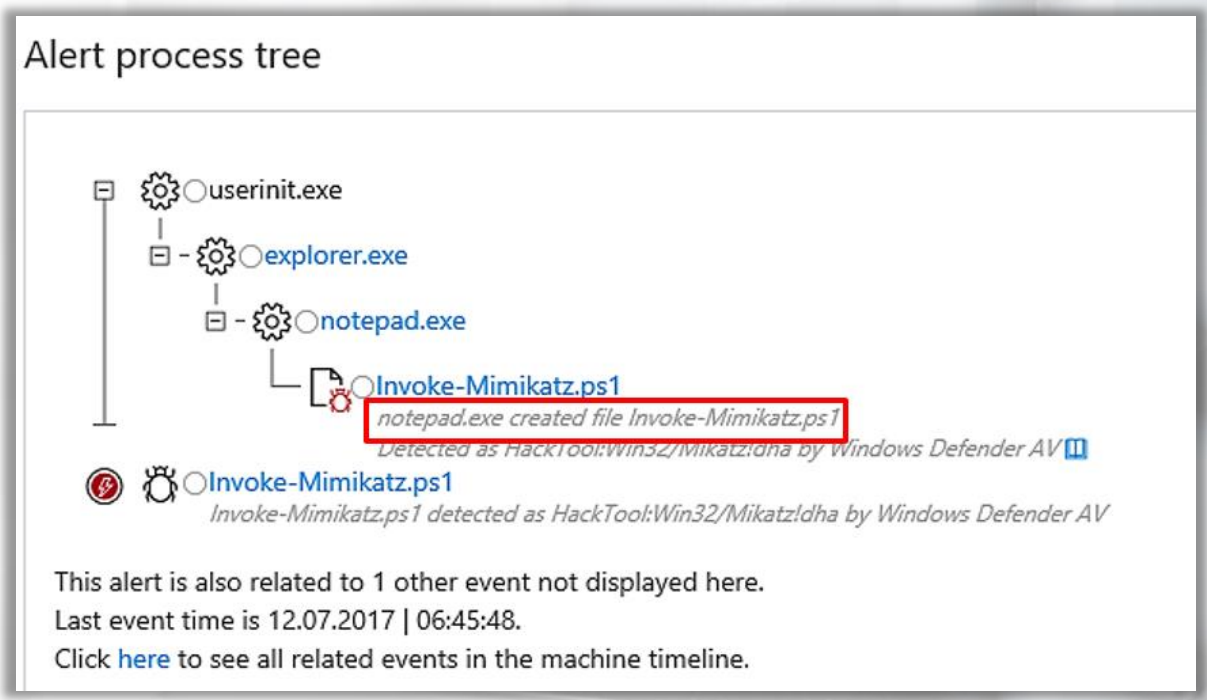
C. Contain and mitigate the breach. Stop suspicious processes, isolate affected machines, decommission compromised accounts or reset passwords, block IP addresses and URLs, and install security updates.

D. Contact your incident response team, or contact Microsoft support for forensic analysis and remediation services.

Disclaimer: These guidelines are for reference only. They do not guarantee successful threat removal.

5.3 Windows Advanced Threat Protection

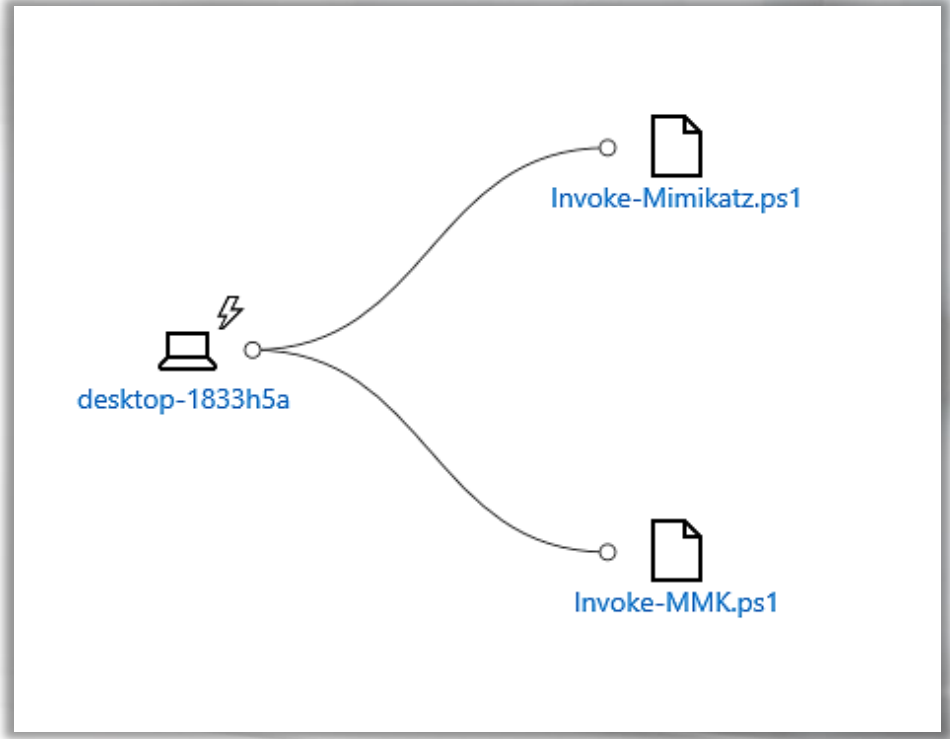
This alerts us that this file was created or copied/pasted into notepad and called Invoke-Mimikatz.ps1.



5.3 Windows Advanced Threat Protection

The PS1 file on the image is called Invoke-Mimikatz.ps1,

and then it was renamed to Invoke-MMK.ps1



5.3 Windows Advanced Threat Protection

ATP gives us the file location on disk as well as its hash.

5.3 Windows Advanced Threat Protection

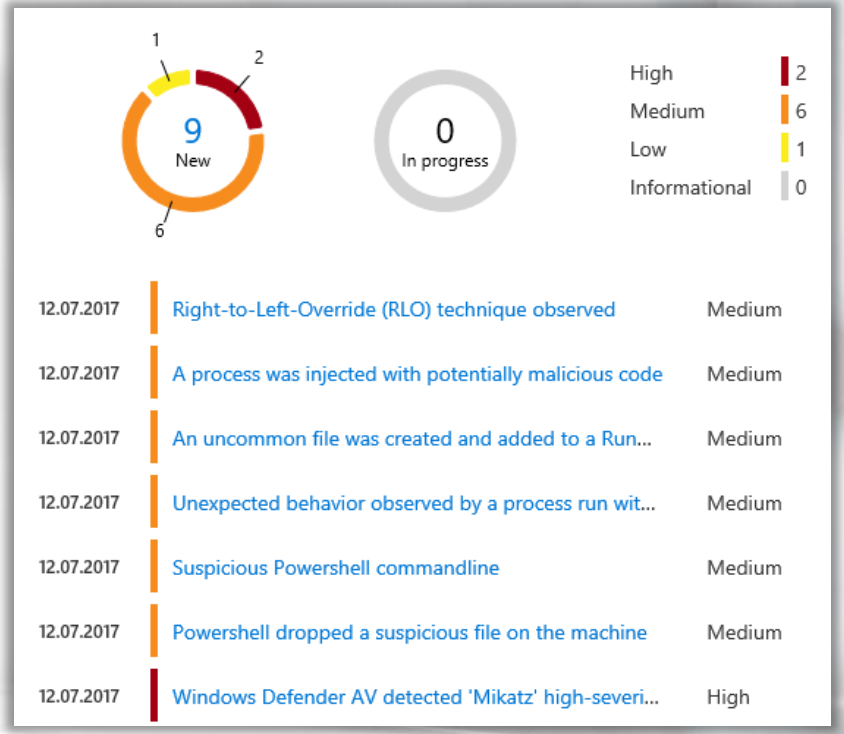
As part of the demo, Microsoft provides a benign Microsoft Word document that will simulate an attack on your test machine.

We will get an idea as to what this simulated attack will do and how ATP detected the ‘attack’.

5.3 Windows Advanced Threat Protection

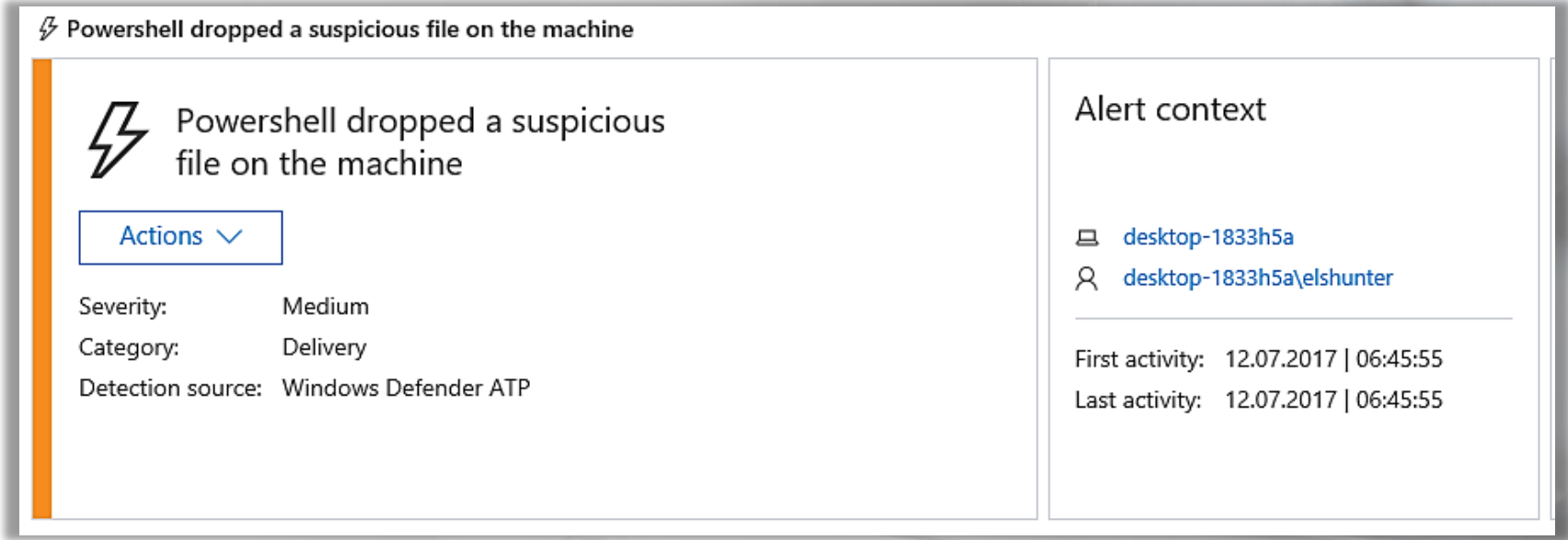
An example of multiple threats detected is shown on the image to the right. Note that a risk rating is also provided.

Let's further investigate one of the alerts.



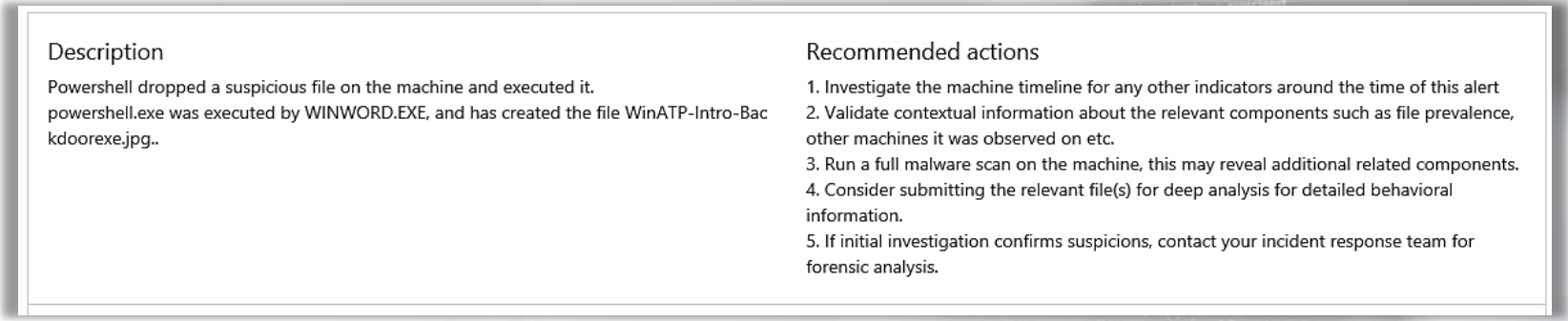
5.3 Windows Advanced Threat Protection

Again, we can look into additional details by expanding the reported suspicious activity:



5.3 Windows Advanced Threat Protection

The description provided includes details on parent process and the suspicious execution event:



Description

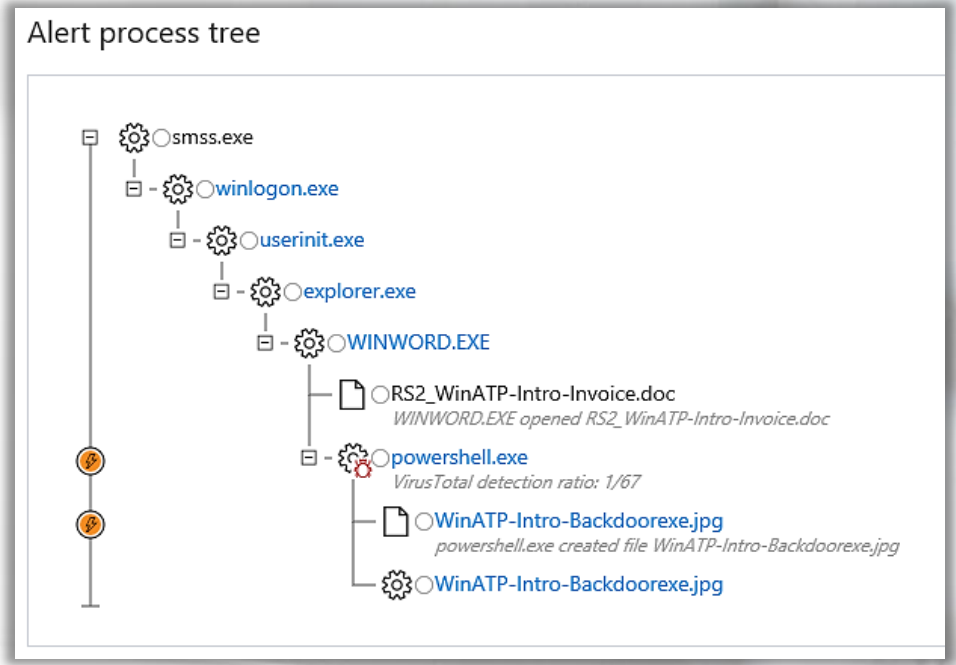
Powershell dropped a suspicious file on the machine and executed it.
powershell.exe was executed by WINWORD.EXE, and has created the file WinATP-Intro-Backdoorexe.jpg..

Recommended actions

1. Investigate the machine timeline for any other indicators around the time of this alert
2. Validate contextual information about the relevant components such as file prevalence, other machines it was observed on etc.
3. Run a full malware scan on the machine, this may reveal additional related components.
4. Consider submitting the relevant file(s) for deep analysis for detailed behavioral information.
5. If initial investigation confirms suspicions, contact your incident response team for forensic analysis.

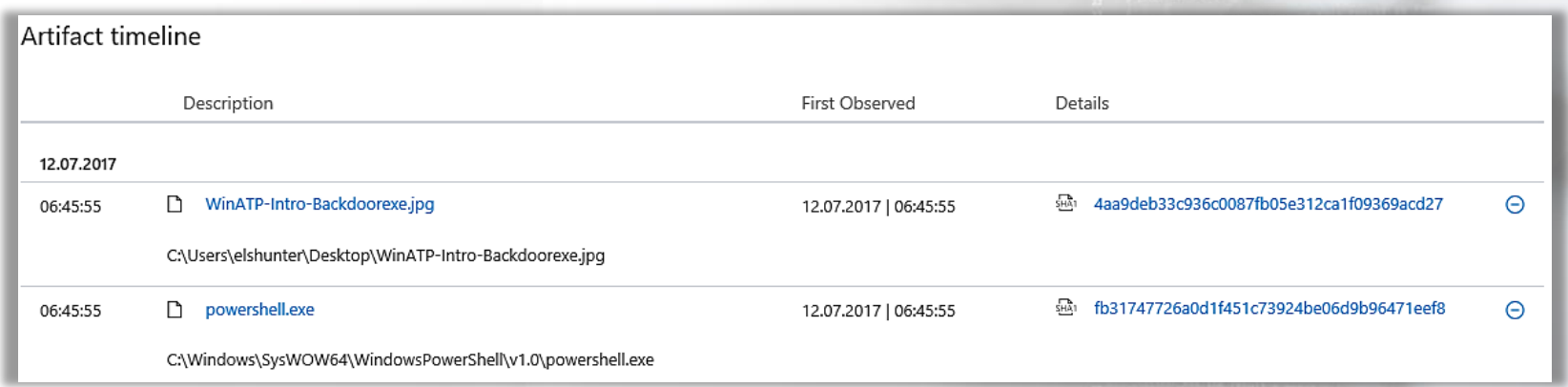
5.3 Windows Advanced Threat Protection

On the image to the right, we can see the full process tree that lead to this activity being flagged as suspicious.



5.3 Windows Advanced Threat Protection

Finally, we are also provided with the hashes of the files.



5.3 Windows Advanced Threat Protection

Basically, Microsoft Word launched PowerShell and PowerShell created a file that seems like a JPG file.

Let’s look at a few more alerts to see what ATP detected regarding this JPG file.

5.3 Windows Advanced Threat Protection

Under Description, we see a snippet of the PowerShell command which was executed from the Word document.

Description

A suspicious Powershell commandline was found on the machine. This commandline might be used during installation, exploration, or in some cases with lateral movement activities which are used by attackers to invoke modules, download external payloads, and get more information about the system. Attackers usually use Powershell to bypass security protection mechanisms by executing their payload in memory without touching the disk and leaving any trace.
The process powershell.exe was executing suspicious commandline

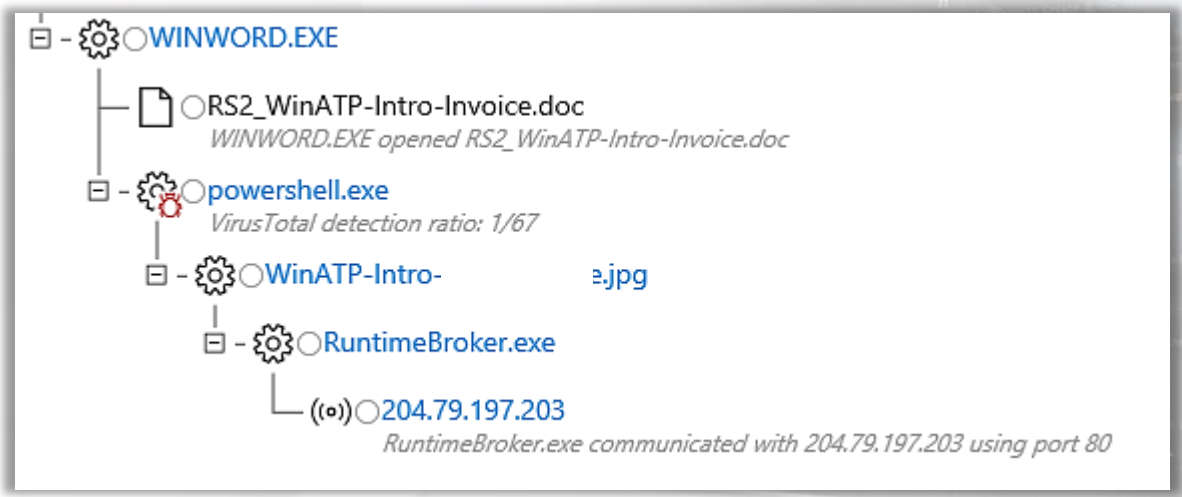
powershell.exe -W Hidden -Exec Bypass -Command cd /;$file= [Convert]::FromBase64Strin
g([string]'TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAgAAAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIG
Nhbm5vdCBiZSBydW4gaW4gRE9TIG1vZGUuDQ0KJAAAAAAAAABQRQAAZIYCAOAsm1gA

Recommended actions

1. Examine the PowerShell commandline to understand what commands were executed. Note: the script may need to be decoded if it is base64-encoded
2. Search the script for more indicators to investigate - for example IP addresses (potential C&C servers), target computers etc.
3. Explore the timeline of this and other related machines for additional suspect activities around the time of the alert.
4. Look for the process that invoked this PowerShell run and their origin. Consider submitting any suspect files in the chain for deep analysis for detailed behavior information.

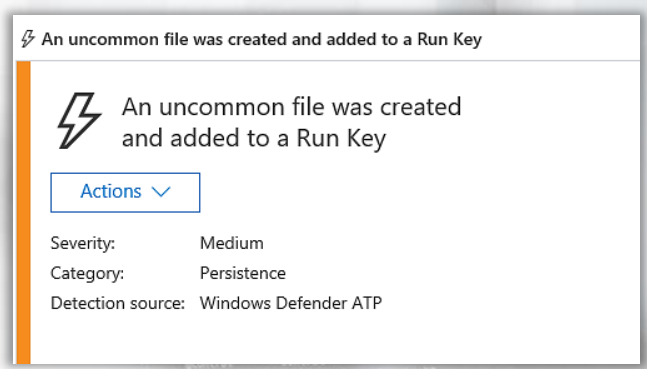
5.3 Windows Advanced Threat Protection

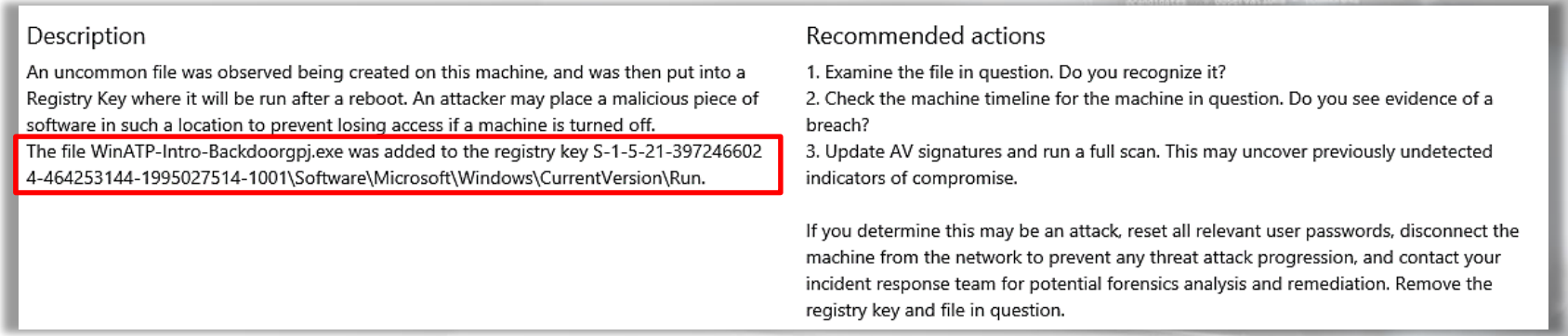
We can then see RuntimeBroker.exe started by the “JPG” file making an outbound connection on port 80.



5.3 Windows Advanced Threat Protection

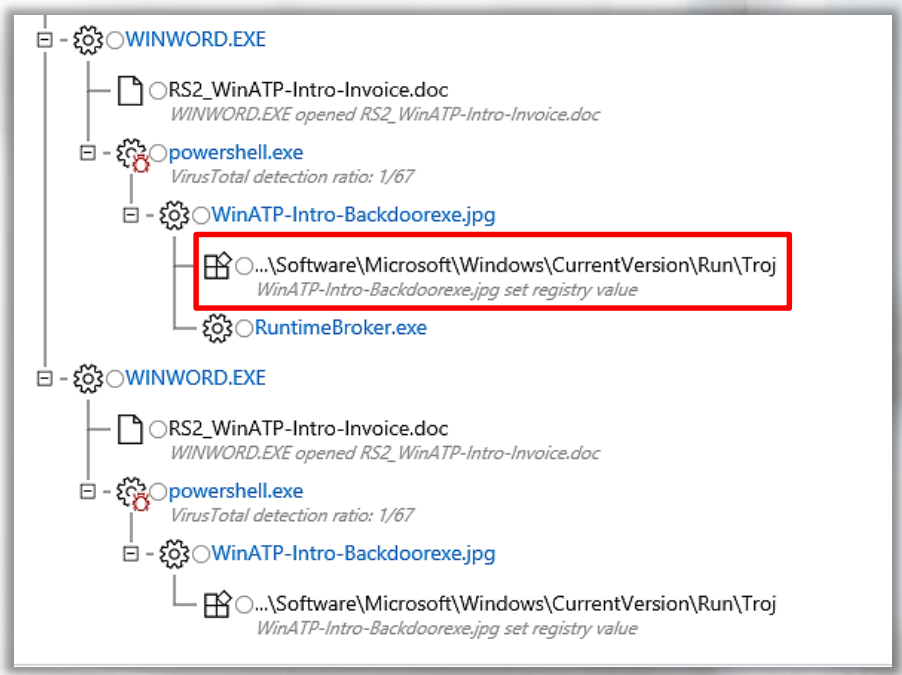
Persistence was discovered by abusing a Registry Run key.





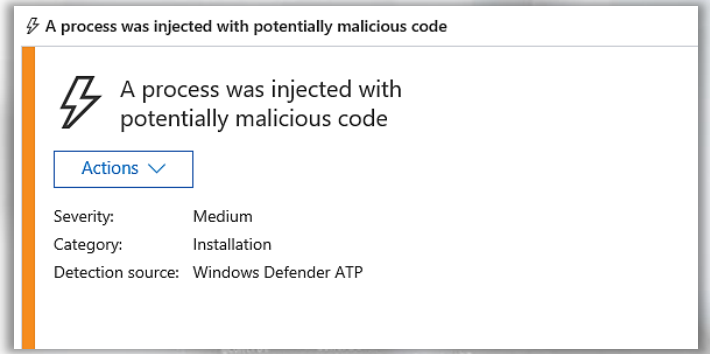
5.3 Windows Advanced Threat Protection

The addition of the registry key is also displayed in a tree visual diagram.



5.3 Windows Advanced Threat Protection

Another alert associated with the file is the detection of process injection:



Description

A process has injected code into another process, indicating suspicious code being run in the target process memory. Injection is often used to hide malicious code execution within a trusted process.
As a result, the target process may exhibit abnormal behaviors such as opening a listening port or connecting to a command and control server.

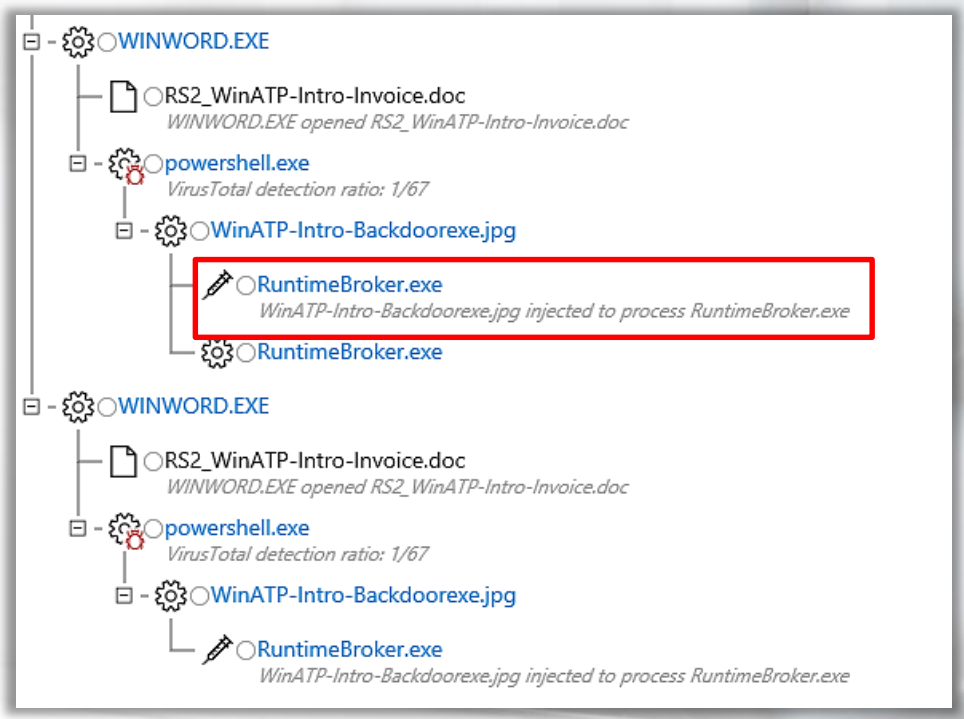
The process WinATP-Intro-Backdoorgpj.exe has injected into the process RuntimeBroker.exe.

Recommended actions

1. Investigate the machine's timeline for any other indicators around the time of this alert
2. Validate contextual information about the relevant components such as file prevalence, other machines it was observed on etc.
3. Contact the machine's user to verify whether they received an email with a suspicious attachment or link around the time of the alert.
4. Run a full malware scan on the machine, this may reveal additional related components.
5. Consider submitting the relevant file(s) for deep analysis for detailed behavioral information.
6. If initial investigation confirms suspicions, contact your incident response team for forensic analysis.

5.3 Windows Advanced Threat Protection

The activity is again displayed in a visual diagram:

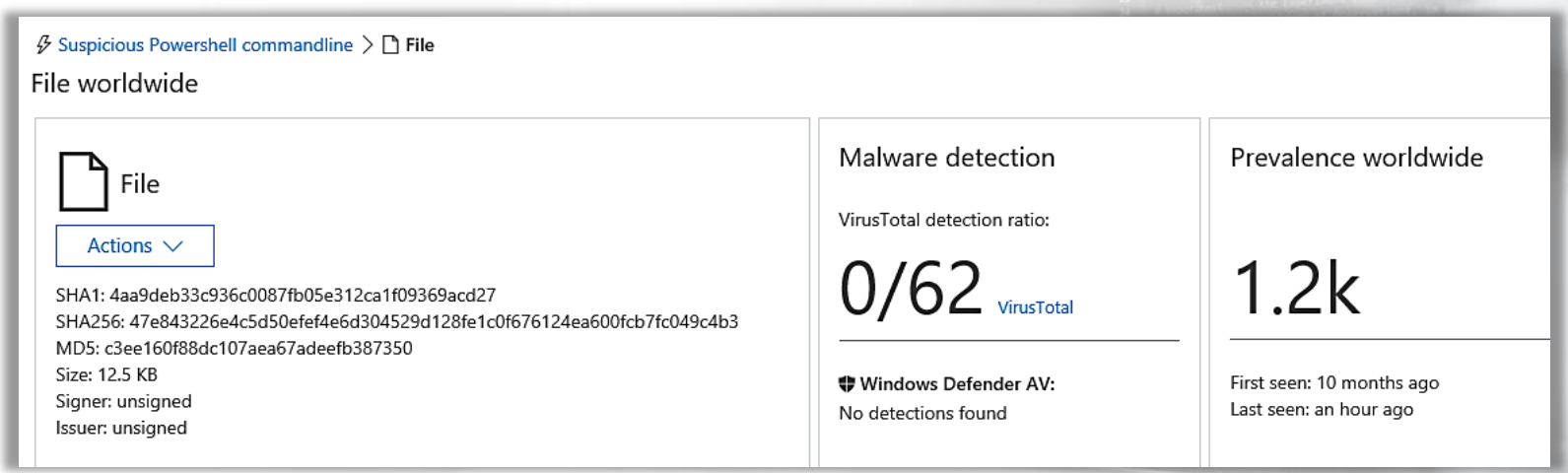


5.3 Windows Advanced Threat Protection

When we click on the JPG file, we're presented with another window showing various information specific to the analysis of the file.

5.3 Windows Advanced Threat Protection

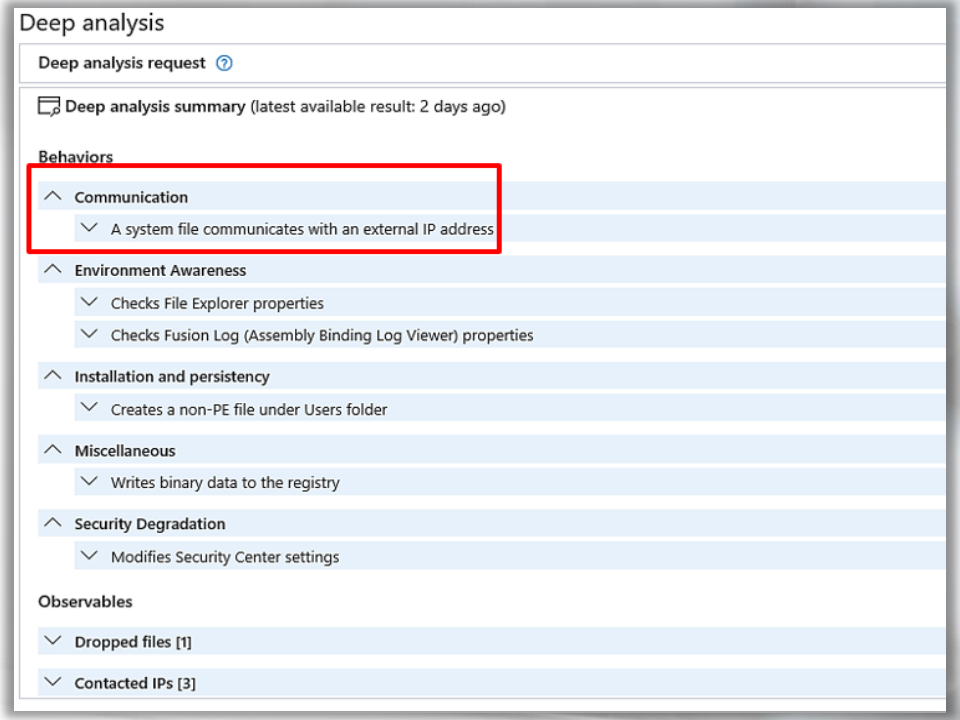
We see ATP gives us the file's hashes. It also submits the file to VirusTotal for analysis. ATP tells us how many other endpoints globally reported being infected with this file.



5.3 Windows Advanced Threat Protection

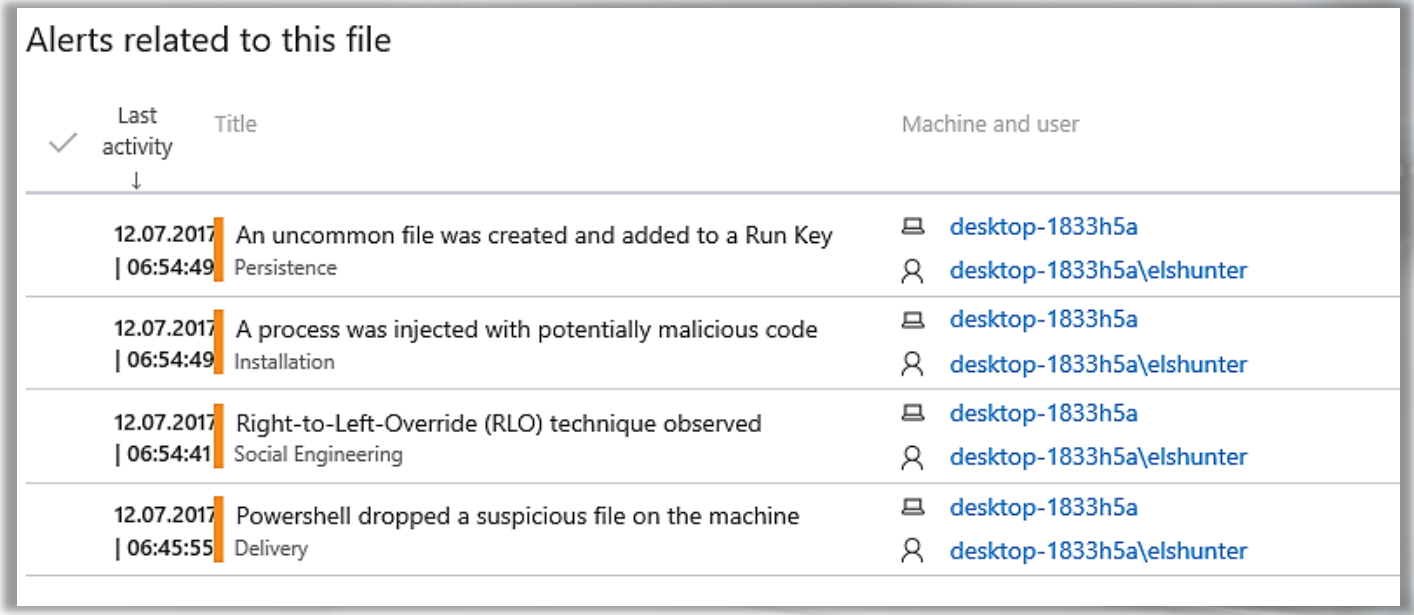
Here we see what malicious actions the file took while on the machine.

What quickly stands out is that this file communicated outbound to an external IP address, which we already saw in another alert.



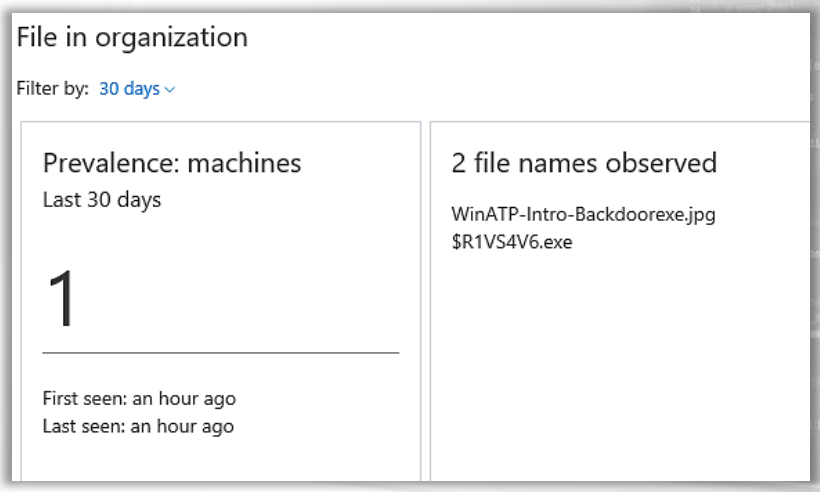
5.3 Windows Advanced Threat Protection

Here we can view an overview of detected alerts:



5.3 Windows Advanced Threat Protection

Statistics view that tell us that the file was discovered on only 1 host machine:



5.3 Windows Advanced Threat Protection

Windows Advanced Threat Protection is definitely a useful tool, and navigating through the tool is pretty straightforward.

The information is simple and useful.

5.4

Microsoft Advanced Threat Analytics

5.4.1 Microsoft Advanced Threat Analytics

Another useful tool provided by Microsoft is called **Microsoft Advanced Threat Analytics** (ATA).

As described on the website: Reduce your risk of costly damage and get all the information you need in a succinct, real-time view of the attack timeline with Advanced Threat Analytics. All the intelligence to learn, analyze, and identify normal and suspicious user or device behavior is built-in.

5.4.1 Microsoft Advanced Threat Analytics

ATA boasts that there is no need to create rules, fine-tune, or monitor a flood of security reports.

It is advanced, self-learning, ready-to-analyze events and threat intelligence.

5.4.1 Microsoft Advanced Threat Analytics

By description:

“ATA works by combining analysis of network traffic, events, and pulling contextual data about the entities from the directory, such as group memberships, titles, and manager information. Once ATA is deployed it begins monitoring the activity of all the entities in the organization, learning the normal behavior of entities, and detecting abnormal behavior and known techniques used by advanced attackers and insiders.”

5.4.1 Microsoft Advanced Threat Analytics

It’s also worth noting that ATA can integrate with your existing SIEM, and will automatically receive updates, including new behavioral detections.

Unlike Windows Defender ATP, you don’t have to go through an approval process to test out ATA. You can read more information and see example detections here, or even try it out here.

https://docs.microsoft.com/en-us/advanced-threat-analytics/what-is-ata
https://www.microsoft.com/en-us/cloud-platform/advanced-threat-analytics

5.4.2 Azure Advanced Threat Protection

Azure ATP is a variation of Microsoft ATA, which can be deployed in (Azure) hybrid environments. Configuration and alerts are available directly in Azure.

The full documentation including videos, investigative examples, and playbooks are available here.

https://docs.microsoft.com/en-us/azure-advanced-threat-protection/

5.4.2 Azure Advanced Threat Protection

For example, the playbook on Lateral movement, here, displays detection of techniques such as Pass-the-hash and Pass-the-ticket.

We recommend reading all of them to get yourself more familiar with this tool and its capabilities.

https://docs.microsoft.com/en-us/azure/azure-advanced-threat-protection/atp-playbook-lateral-movement

5.5

PowerShell Defenses

5.5 PowerShell Defenses

In the previous module, we discussed PowerShell logging and what Event IDs to look for when hunting for PowerShell usage in the environment.

In the upcoming slides, we're going to look at some techniques to defend PowerShell in the enterprise.

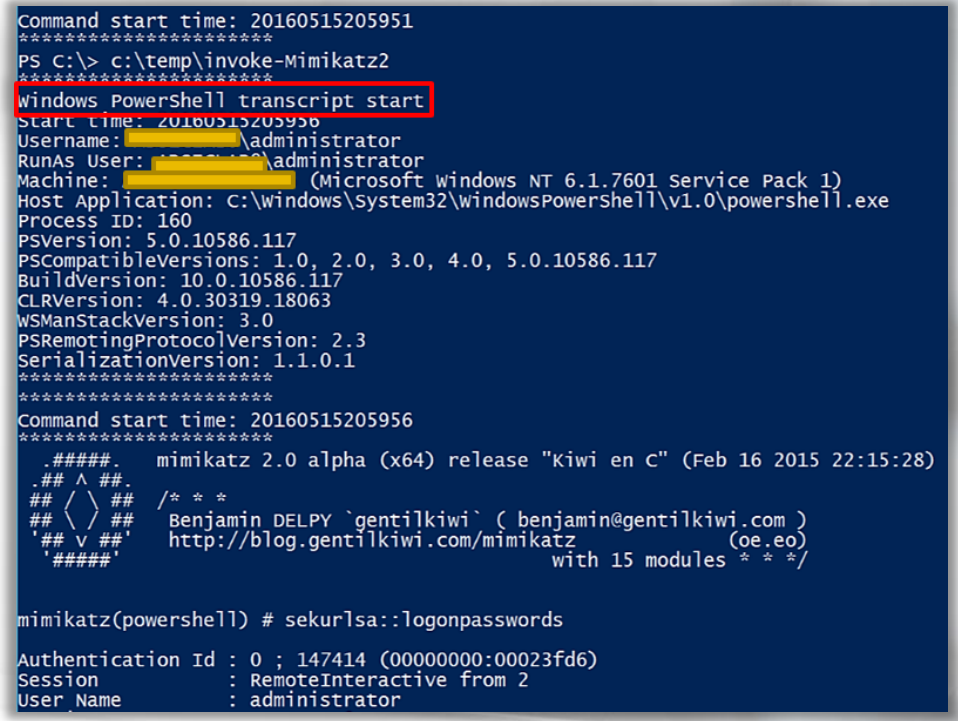
5.5.1 System-Wide Transcript File

If the environment has “system-wide transcript file” enabled, a share on the network will exist where everything typed in PowerShell (transcript file) will be sent to that network share.

This means that the environment’s Blue Team will have an over-the-shoulder transcript of everything that was typed, for every computer/user.

5.5.1 System-Wide Transcript File

In the image to the right, you can see a system-wide transcript file in action, displaying the executed command and its output.



5.5.2 Constrained Language Mode

Constrained language mode is a feature mode that limits the capability of PowerShell to base functionality.

.NET or COM access and Win32 API calls through PowerShell are not possible when constrained language mode is enforced.

5.5.2 Constrained Language Mode

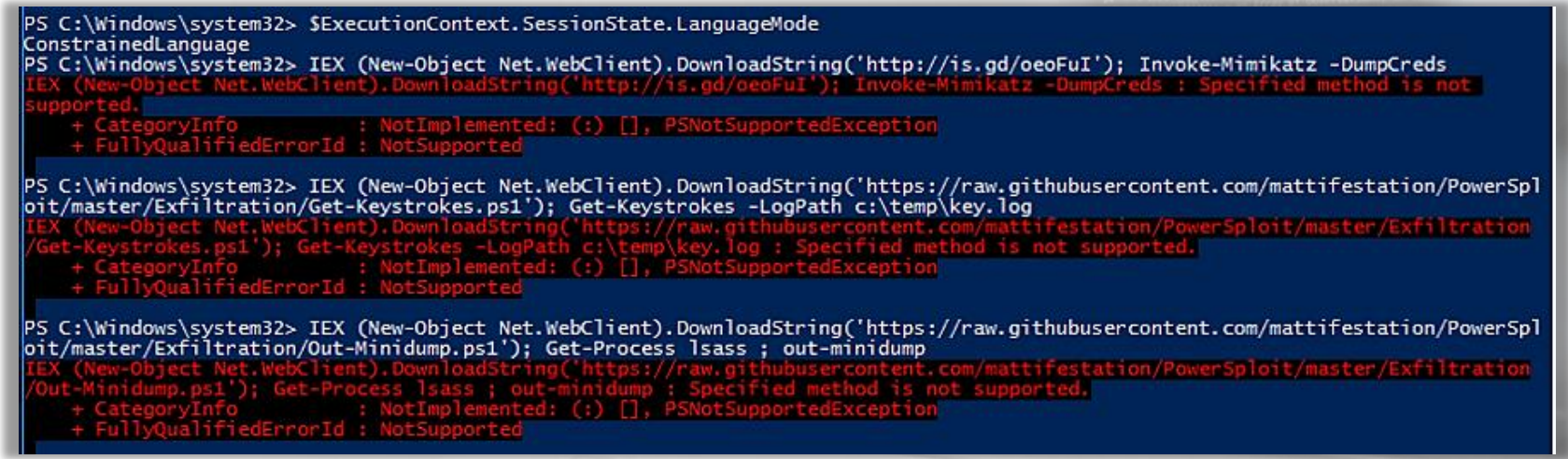
If an environment has PowerShell version 5 and AppLocker Script rules in enforce mode, PowerShell locks down to constrained language mode automatically.

The same will happen if Device Guard with UMCI is deployed.

https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/applocker-overview
https://docs.microsoft.com/en-us/windows/device-security/device-guard/introduction-to-device-guard-virtualization-based-security-and-code-integrity-policies

5.5.2 Constrained Language Mode

You can see constrained language mode in action on the image below (cannot download files).



5.5.3 Anti-Malware Scan Interface

In Windows 10, it gets even more interesting due to the introduction of the AMSI (Anti-Malware Scan Interface).

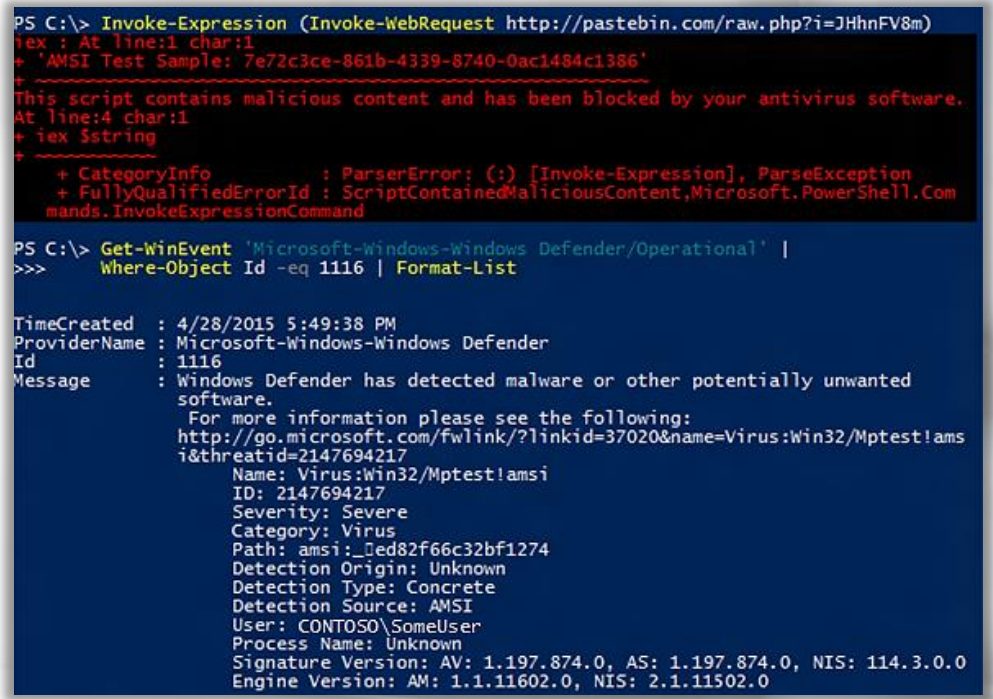
On AMSI powered systems, AMSI picks up any PowerShell or VBScript code before it's executed by the PowerShell engine. The AMSI, in turn, sends it over to the anti-malware solution. The anti-malware solution will give a thumbs up or a thumbs down based on its signature database.

5.5.3 Anti-Malware Scan Interface

If it's a thumbs down, PowerShell will not execute that code, whether it is downloaded from the internet and run in memory or run from a script.

There are some vendors that support AMSI, and these are Microsoft, ESET, and AVG.

Here, you can see AMSI in action.



Module Conclusion

This concludes the module on Hunting with/for PowerShell. We have covered:

- Various PowerShell tools to aid us in hunting in the enterprise.
- Windows Defender Advanced Threat Protection.
- Windows ATA and Azure ATP.
- Additional techniques to defend the malicious use of PowerShell.

References

References

Kansa

https://github.com/davehull/Kansa

Kansa blog

http://trustedsignal.blogspot.com/search/label/Kansa

Kansa: A PowerShell-based incident response framework

http://www.powershellmagazine.com/2014/07/18/kansa-a-powershell-based-incident-response-framework/

PSHunt

https://github.com/Infocyte/PSHunt

References

BSidesLV 2016 PSHunt

https://www.youtube.com/watch?v=2MrrOxsJk_M

NOAH at BH2017

https://www.blackhat.com/us-17/arsenal/schedule/#noah-uncover-the-evil-within-respond-immediately-by-collecting-all-the-artifacts-agentlessly-7965

NOAH 2

https://github.com/giMini/NOAH

Windows ATP

https://www.microsoft.com/en-us/windowsforbusiness/windows-atp

References

Windows ATP 2

https://winatpregistration-prd.trafficmanager.net/UserAgreement

Windows ATA

https://cloudblogs.microsoft.com/enterprisemobility/2017/01/19/introducing-microsoft-advanced-threat-analytics-for-your-datacenter/

App Locker

https://technet.microsoft.com/en-us/library/dd723678(v=ws.10).aspx

Device Guard with UMCI

https://docs.microsoft.com/en-us/windows/device-security/device-guard/introduction-to-device-guard-virtualization-based-security-and-code-integrity-policies

References

What is Advanced Threat Analytics?

https://docs.microsoft.com/en-us/advanced-threat-analytics/what-is-ata

Advanced Threat Analytics

https://www.microsoft.com/en-us/cloud-platform/advanced-threat-analytics

Azure Advanced Threat Protection documentation

https://docs.microsoft.com/en-us/azure-advanced-threat-protection/

Tutorial: Lateral movement playbook

https://docs.microsoft.com/en-us/azure-advanced-threat-protection/atp-playbook-lateral-movement

References

AppLocker

https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/applocker-overview

Labs

Hunting at Scale with Osquery

In this lab you will learn how to hunt for process injection on Linux endpoints with the help of Osquery. You will also be shown how to execute pre-baked Osquery queries through Kollide fleet.

**Labs are only available in Full or Elite Editions of the course. To ACCESS your labs, go to the course in your members area and click the labs drop-down in the appropriate module line. To UPGRADE to gain access, click LINK.*