Silencing Microsoft Defender for Endpoint using firewall rules

medium.com/csis-techblog/silencing-microsoft-defender-for-endpoint-using-firewall-rules-3839a8bf8d18

Søren Fritzbøger January 21, 2021

Søren Fritzbøger

Jan 21, 2021

.

6 min read

Windows Defender for Endpoint (formerly Windows Defender ATP) is a so-called “cloud powered” EDR product[1], i.e. alerts and events are pushed to the cloud where defenders can respond to them.

When doing Red Team assignments one of the biggest hurdles usually lie in evading EDR products and ensuring that our actions are not detected. While a lot of work and research has been put into evading and bypassing Windows Defender for Endpoint, little research explores the possibility of simply silencing MD for Endpoint such that no data is sent to the cloud.

This article will explore one possible way of silencing MD for Endpoint by utilizing firewall rules to ensure that no events are sent to Microsoft Defender Security Center (i.e. https://securitycenter.windows.com), and how to detect this kind of activity.

Known methods of tampering with MD for Endpoint was detailed in Chris Thompson’s (@retBandit) talk “Red Team Techniques for Evading, Bypassing, and Disabling MS Advanced Threat Protection and Advanced Threat[4] Analytics” from BlackHat Europe 2017. In this talk the usage of firewall rules to block traffic to known MD for Endpoint URLs is discussed. While this technique still works, it is cumbersome to create firewall rules for all the hosts detailed in the service URLs for MD for Endpoint, which can be found here[2][3]. Furthermore, the list of URLs also contain entries such as

- *.blob.core.windows.net
- *.azure-automation.net

which might break certain Windows services. In the case of a Red Team or penetration test, you don’t want to negatively affect the client more than necessary or alert defenders by obstructing the usual workflow.

Therefore, instead of blocking communication to certain URLs, we wanted to asses if it was possible to block communication for specific services and processes in order to silence MD for Endpoint. To do this, we needed to figure out which processes communicate with known MD for Endpoint URLs, and block these specific processes. There are many ways to examine which domains processes contact, including Microsoft Network Monitor[5], Wireshark[6] and probably a ton of others. However, all the information we need is already collected by MD for Endpoint and available to us through Microsoft Defender Security Center!

MD for Endpoint gathers network connections from all running processes, and can therefore be used to figure out what processes communicate with known MD for Defender URLs. The following Kusto query can be run to see these processes.

Kusto query to find processes making connections to known MD for Endpoint URLs

InitiatingProcessFileName	Count	RemoteUrl	InitiatingProcessCommandLine
svchost.exe	457	["eu-v20.events.data.microsoft.com","winatp-gw-neu.microsoft.com"]	["svchost.exe -k utcsvc -p","svchost.e
Teams.exe	27	["eu-v20.events.data.microsoft.com","winatp-gw-neu.microsoft.com"]	["\"Teams.exe\" --type=renderer --au
SenseIR.exe	48	["automatedirstrprdneu.blob.core.windows.net","winatp-gw-neu.microsoft.com","wseu1westprod.blob.core.windows.net","wseu1northprod.blob.core.windows.net"]	["\"SenseIR.exe\" \"OnlineSenseIR\" \
SenseCncProxy.exe	1484	["winatp-gw-neu.microsoft.com"]	["3868","3352","3952","2028","2656",
MsSense.exe	15592	["eu-v20.events.data.microsoft.com","winatp-gw-neu.microsoft.com"]	["\"MsSense.exe\""]
MsMpEng.exe	4289	["europe.cp.wd.microsoft.com","https://europe.cp.wd.microsoft.com","winatp-gw-neu.microsoft.com"]	["\"MsMpEng.exe\""]
MpCmdRun.exe	211	["europe.cp.wd.microsoft.com"]	["\"MpCmdRun.exe\" SpyNetService
MonitoringHost.exe	803	["winatp-gw-neu.microsoft.com","winatp-gw-cus.microsoft.com","winatp-gw-eus.microsoft.com"]	["\"MonitoringHost.exe\" -Embeddin
HealthService.exe	526	["df69e070-cc8d-4b7a-b116-fd5cb3d9e918.ods.opinsights.azure.com","df69e070-cc8d-4b7a-b116-fd5cb3d9e918.oms.opinsights.azure.com"]	["\"HealthService.exe\""]
	16	["winatp-gw-neu.microsoft.com"]	[""]

Processes making connections to known MD for Endpoint URLs

Going through the list, we have a few processes that stick out as processes that do not have anything to do with MD for Endpoint. For example, Teams.exe, which is the official Teams client for Microsoft, apparently sends events to the known URLs. Furthermore, we have a process with no name, which we did not look further into.

The rest of the processes should be very familiar to anyone who has experience with Defender for Endpoint. However, a thorough understanding of these processes will aid in blocking them effectively using Windows Firewall.

- Executable that spawns as a child process of MsSense.exe when initiating a “Live Response Session” through Microsoft Defender Security Center
- Executable that spawns as a child process of MsSense.exe. The description calls it “Communication Module”, so presumably handles communication to the aforementioned URLs and also proxying if you applied this
- Main service executable for Defender for Endpoint. This runs as a service with the name “Sense” (On Windows 10)
- Main service executable for Microsoft Defender Antivirus. This runs as a service with the name “WinDefend” (On Windows 10)
- Command-line tool to perform various Microsoft Defender Antivirus functions

- Executable that spawns as a child process of the HealthService service. Part of Microsoft Monitoring Agent, which is installed when onboarding devices with previous versions of Windows (Server 2016 and below + Windows 8.1 Enterprise and below)[6] [7]
- Main service executable for Microsoft Monitoring Agent, which is installed when onboarding devices with previous versions of Windows (Server 2016 and below + Windows 8.1 Enterprise and below)[6][7]

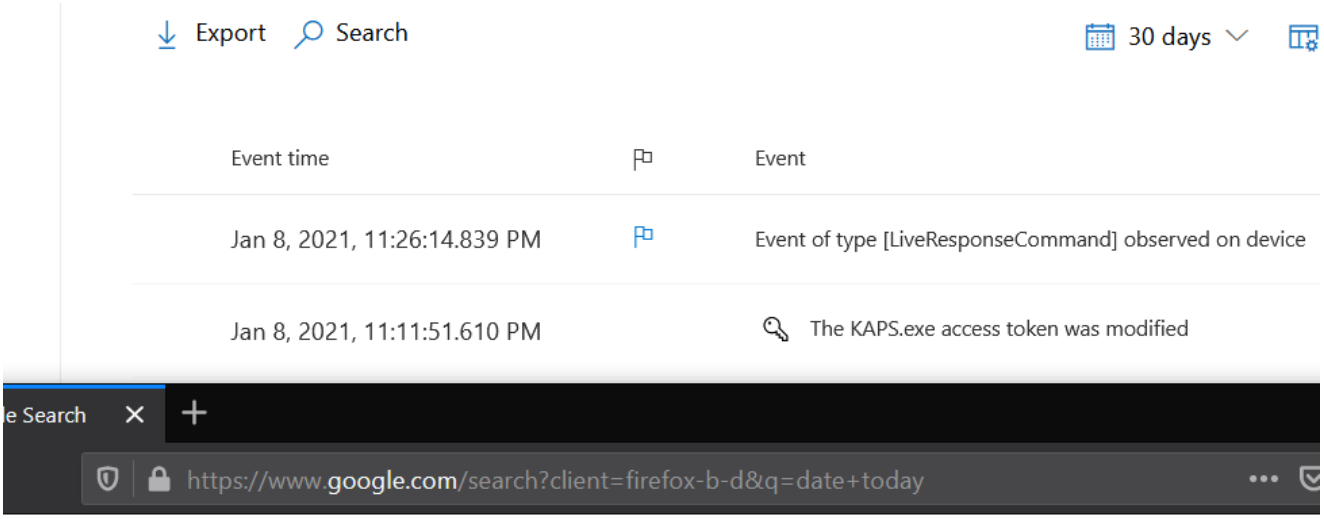
Furthermore, there is a bunch of native windows services such as utcsvc and diagtrack that also communicate with the known URLs, but in our testing they were not sending any event or alert data related to MD for Endpoint.

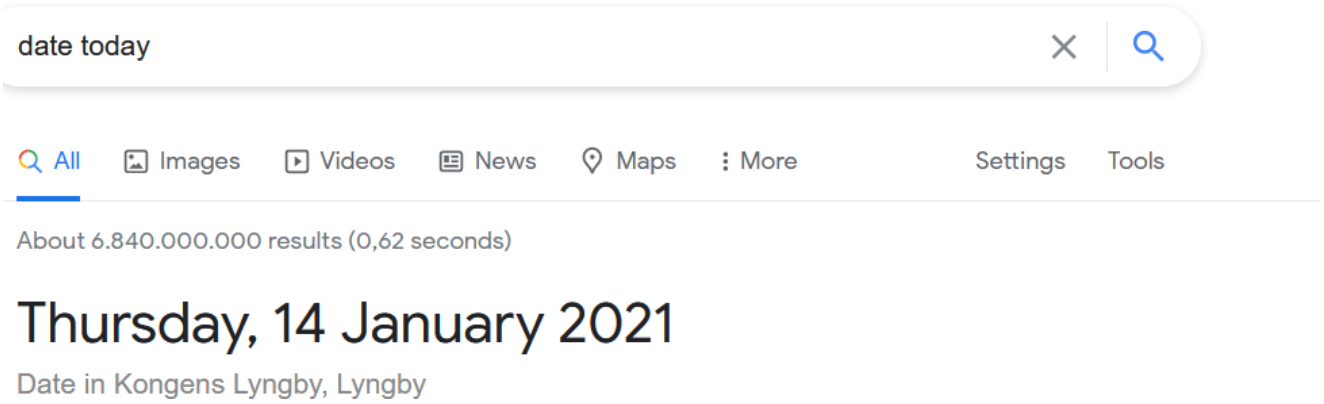
PoC — Firewall rules

Windows 10 and Windows Server 2019

Silencing MD for Endpoint on Windows 10 and Windows Server 2019 is fairly easy, as only the three processes, MsMpEng, SenseCncProxy and MsSense need to be blocked from sending traffic outbound on port 443. Running the following PowerShell script will effectively silence Windows Defender without triggering any alerts at the time of writing:

Firewall rules to silence MD for Endpoint on Windows 10 and Windows Server 2019
Adding firewall rules in Windows requires local admin privileges, so the attack cannot be done from a low privileged user account. As you can see from the following screenshot taken from Microsoft Defender Security Center, no events are received once the firewall rules are applied.





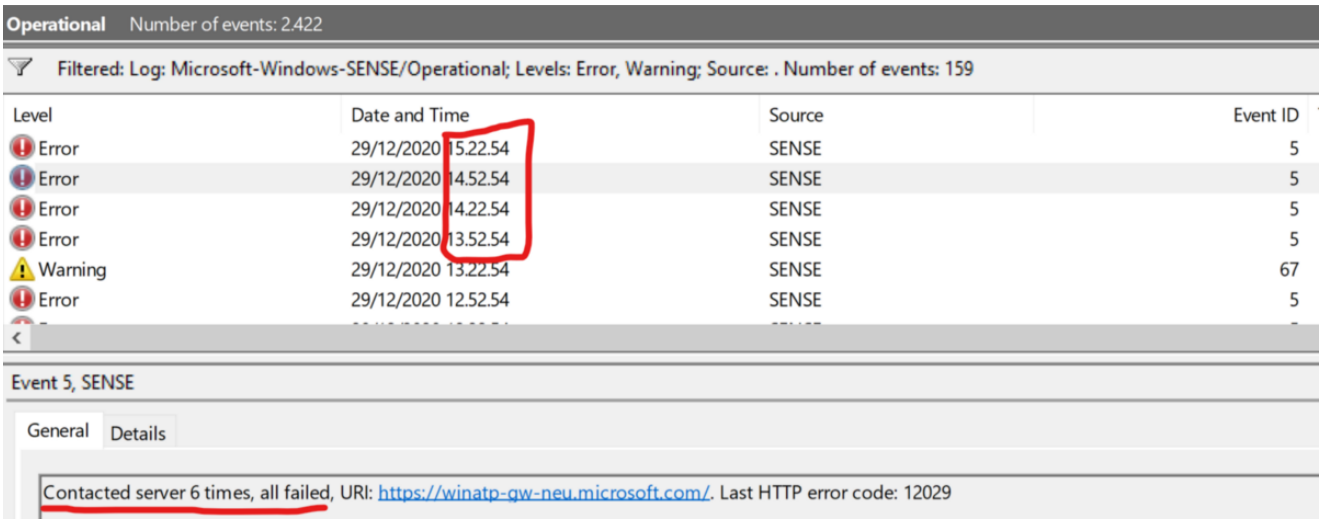
As you can see, no events are shown in Security Center after the firewall rules have been applied
It seems that all events are cached locally, so once the firewall rules are removed, the events and alerts will start showing up in Microsoft Defender Security Center.

Previous versions of Windows?

While we could give the answer to this question, we will leave this as an exercise to the reader. All the information should already be given, so it is just a matter of doing a little bit of research and trying stuff out.

Detection

If you rely solely on MD for Endpoint for host-based intrusion detection, you are at loss here. As no event or alert is sent to the cloud solution, it is, at the time of writing, impossible to detect using MD for Endpoint in any useful way. Luckily for us, MD for Endpoint logs connection errors in the Microsoft-Windows-SENSE/Operational event log with event id 5. An example can be seen below.



Event logs for detecting when the MD for endpoint client is unable to contact the cloud server
While you could also detect the attack based on process execution or PowerShell logs, it is not a robust solution as many ways of adding firewall rules exist in Windows, for example using the GUI or netsh.exe[9]. I have not found a proper way of detecting the creation of firewall rules, as most resources point to event id 4947 (A change has been made to Windows Firewall exception list. A rule was modified) which only shows the RuleId and RuleName, but not the rule content.

Addition: When MD for Endpoint has been blocked for 7 days, the health state will change to inactive[11]. This means that for the first 7 days, the device is not marked as inactive even though it is not sending any alerts or events.

Addition 2: In the Device Inventory overview you can filter devices that have “Imparied communications” or “No sensor data”[11]. According to @SiliconShecky this list will show the devices within 24 hours.

Addition 3: @bh4b3sh has published a post describing how creation of firewall rules can be detected by using registry auditing. See more information in his post: https://bhabeshraj.com/post/detect-addition-of-new-firewall-rules-in-defender-firewall/

Prevention

While it is possible to create custom detection rules in MD for Endpoint[10], these are only evaluated on the server and does not block any actions on the endpoints. It merely alerts on them for an operator to take action. Therefore, to prevent this specific attack we have to wait until Microsoft decides to block the action itself in Microsoft Defender through AMSI or similar methods.

I hope you enjoyed the article. If you have any questions don’t hesitate to contact me on twitter @fritzboger.

References

[1] https://www.microsoft.com/en/microsoft-365/security/endpoint-defender

[2] https://docs.microsoft.com/en-us/windows/security/threat-protection/microsoft-defender-atp/configure-proxy-internet#enable-access-to-microsoft-defender-for-endpoint-service-urls-in-the-proxy-server

[3] https://github.com/MicrosoftDocs/windows-itpro-docs/raw/public/windows/security/threat-protection/microsoft-defender-atp/downloads/mdatp-urls.xlsx

[4] https://www.blackhat.com/docs/eu-17/materials/eu-17-Thompson-Red-Team-Techniques-For-Evading-Bypassing-And-Disabling-MS-Advanced-Threat-Protection-And-Advanced-Threat-Analytics.pdf

[5] https://docs.microsoft.com/en-us/troubleshoot/windows-server/networking/network-monitor-3

[6] https://docs.microsoft.com/en-us/windows/security/threat-protection/microsoft-defender-atp/onboard-downlevel

[7] https://docs.microsoft.com/en-us/windows/security/threat-protection/microsoft-defender-atp/configure-server-endpoints

[8] https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon

[9] https://docs.microsoft.com/en-us/troubleshoot/windows-server/networking/netsh-advfirewall-firewall-control-firewall-behavior

[10] https://docs.microsoft.com/en-us/windows/security/threat-protection/microsoft-defender-atp/custom-detection-rules

[11] https://docs.microsoft.com/en-us/windows/security/threat-protection/microsoft-defender-atp/fix-unhealthy-sensors#inactive-devices