Monitoring Microsoft Defender Like a Boss

Uncategorized | 4 years ago

Microsoft Defender, formerly Windows Defender, is Microsoft's in-built Antivirus solution for Windows. In recent times, Microsoft has significantly improved Defender's capabilities to make it viable as a standalone Antivirus solution.

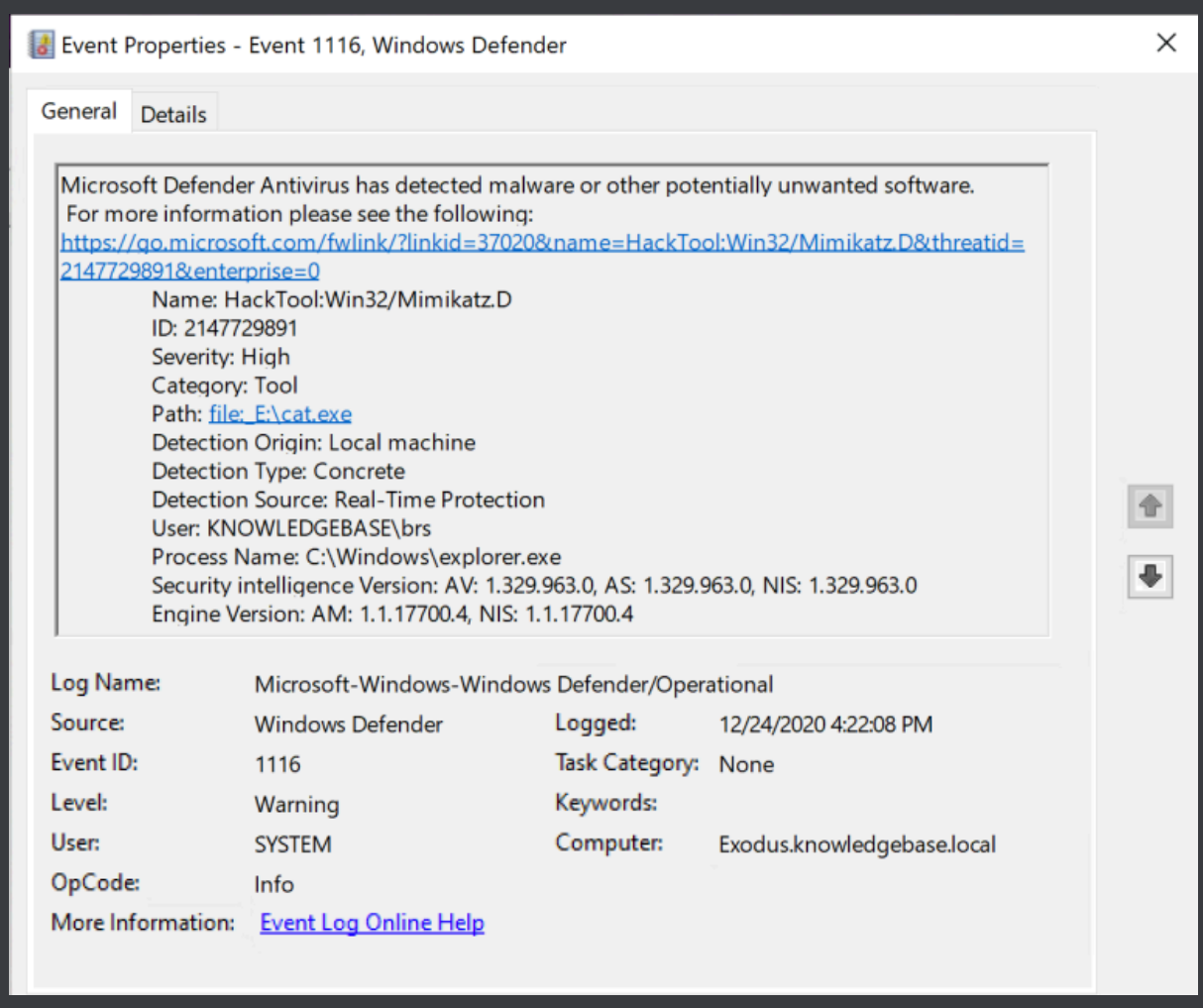
However, we are not here to talk about all of Defender's capabilities but rather a new feature called *Tamper Protection* that is available in all Home and Pro editions of Windows 10 version 1903 and higher and is enabled by default.

TLDR;

If Tamper Protection is enabled then commonly known techniques to disable Defender will not work even if you are **SYSTEM**.
Tamper Protection will not protect **Exclusions** settings.
Defender's logs are housed in *Microsoft-Windows-Windows Defender/Operational* channel. So, make sure to include this channel in your event forwarders like NxLog if you use one for centralized logging.

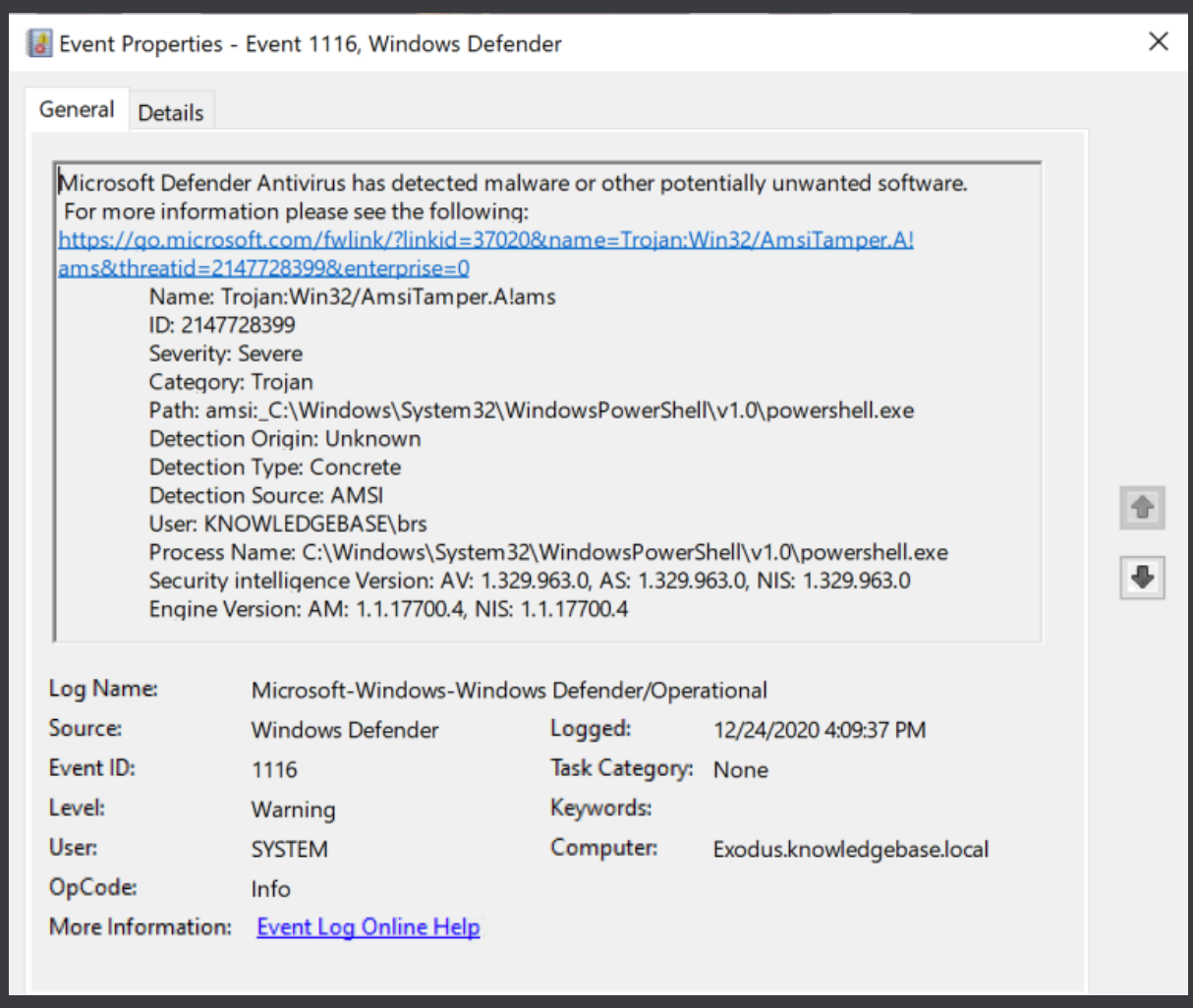
Let's dive into different Defender's Events that may help Blue teams to monitor for any malicious activity in the environment. Example queries for SIEMs are also provided.

Since Microsoft uses the *Computer Antivirus Research Organization (CARO)* malware naming scheme, it is trivial to look for say hacking tools and backdoors detected by Defender via Event ID 1116:



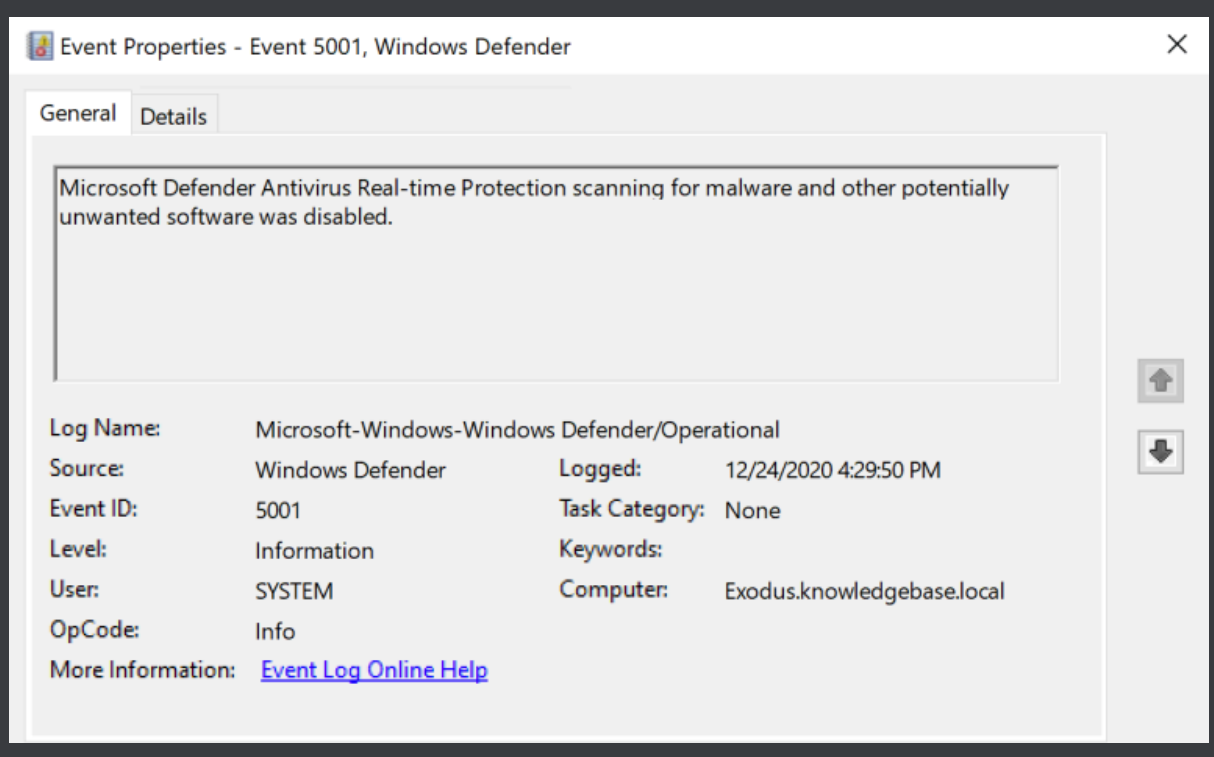

```
EventSource=MicrosoftDefender EventID=1116 DetectionSource="Real-Time Protection" (ThreatName="HackTool:*" OR
ThreatName="Backdoor:*")
```

AMSI trigger alerts by Defender can be searched in the same Event ID by filtering AMSI as Detection Source. Sigma rule also exists for this detection.



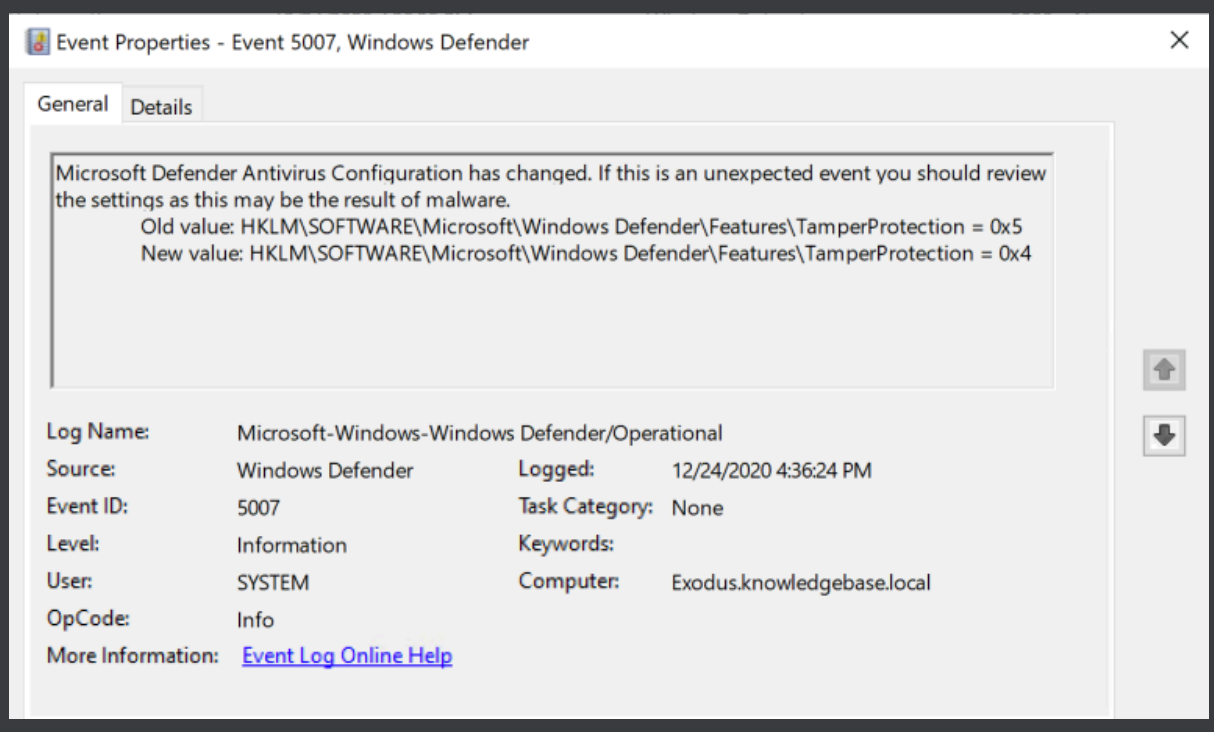

```
EventSource=MicrosoftDefender EventID=1116 DetectionSource=AMSI
```

Event ID 5001 is very important for defenders as it signals the disabling of Defender's Real-Time Protection.




```
EventSource=MicrosoftDefender EventID=5001
```

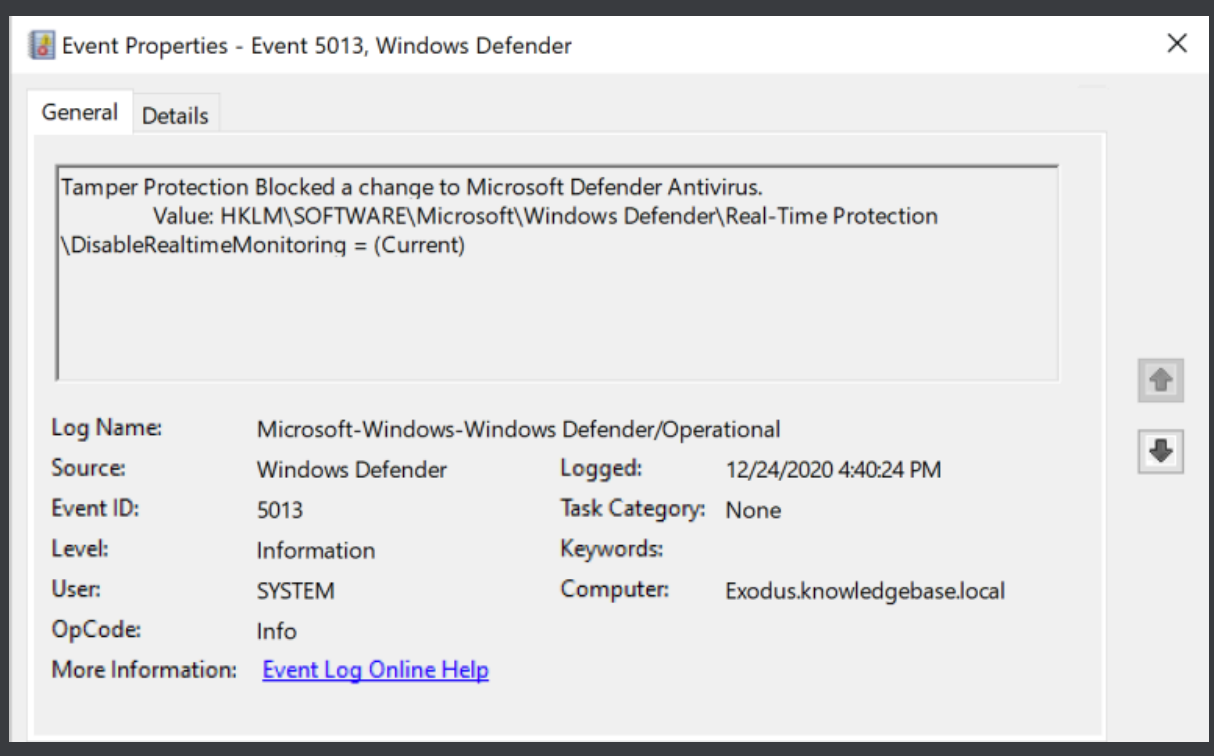
Similarly, Event ID 5007 signals change in Defender's settings, the most critical change being disabling of Tamper Protection.




```
EventSource=MicrosoftDefender EventID=5007 OldValue="*TamperProtection = 0x5" NewValue="*TamperProtection = 0x4"
```

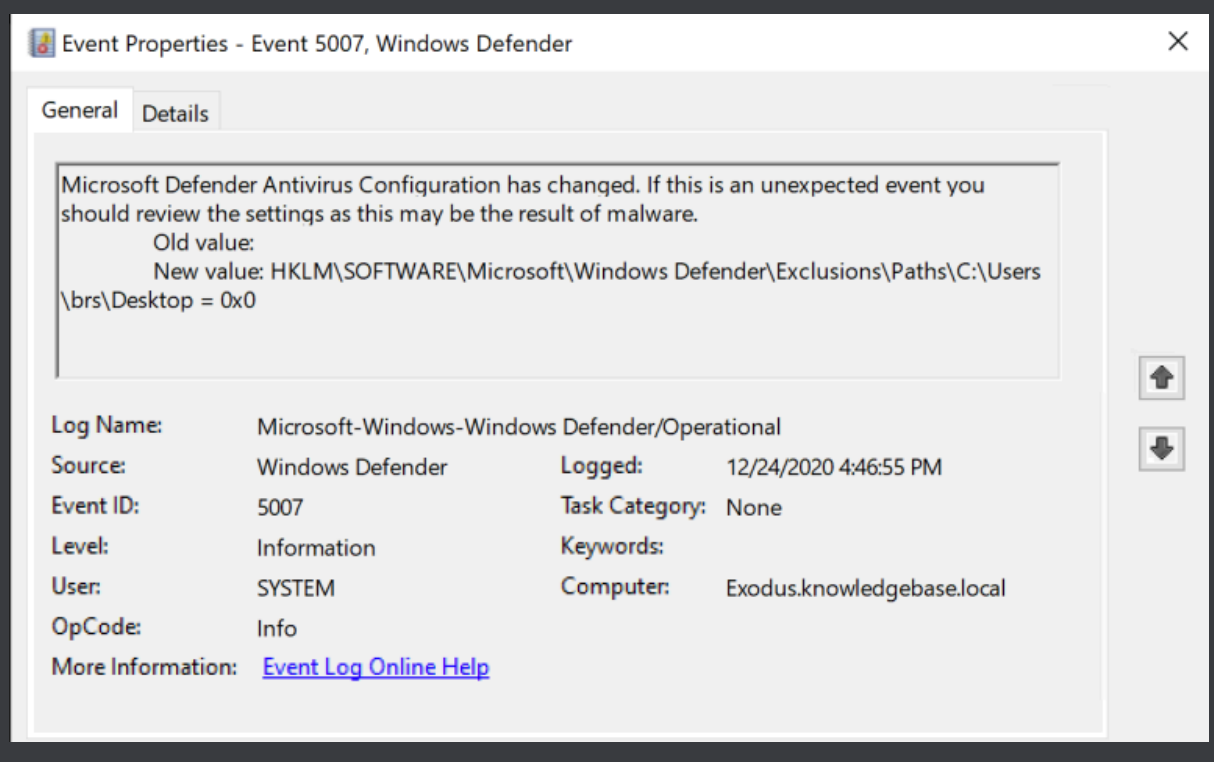
If Tamper Protection is enabled then, any attempt to change any of Defender's settings is blocked and Event ID 5013 is generated that states which setting changes were blocked. For example, the following log is generated if an adversary tries to disable Real-Time Protection via PowerShell cmdlet.

```
Set-MpPreference -DisableRealTimeProtection $true
```




```
EventSource=MicrosoftDefender EventID=5013 Value="*DisableRealtimeMonitoring*"
```

However, it should be kept in mind that, Tamper Protection does NOT protect alterations to Exclusion items. So, for any changes to Exclusions, we have to rely on Event ID 5007.



```
EventSource=MicrosoftDefender EventID=5007 NewValue="*\Exclusions\Paths\*"
```

PS: I do not know why Defender generates two logs (containing OldValue and NewValue) instead of one.

If Tamper Protection is enabled then, as far as I know, the only way to disable it is via restoring the registry config that has the Tamper Protection feature disabled via the **SYSTEM** user. Big shout-out to *@freefirex* from TrustedSec that discovered this hack.

In this hack, First, disable Tamper Protection in your machine then export the registry hive of Defender.

```
reg export "HKLM\Software\Microsoft\Windows\Windows Defender\Features" defender.hiv
```

Transfer this exported hive to the target machine then perform restore operation.

```
reg restore "HKLM\Software\Microsoft\Windows\Windows Defender\Features" defender.hiv
```

A restart is required for the new registry configuration to take effect which is not a problem if you already have achieved persistence on that windows box.

In conclusion, as enterprises update their Windows systems to newer builds, adversaries may find other methods to disable Defender if found to have been used as the standalone AV. Detecting such events is critical for Enterprise Defenders.

About Bhabesh