Lab 5 : Using Wazuh to add PowerShell script block logging

Introduction

In this exercise, we are going to add **PowerShell Script Block logs** to the Wazuh collection capabilities set. **PowerShell Script Block Logging** is a feature that comes with later versions of PowerShell (**PowerShell v5+**). It allows us to instruct Windows to create an event log for every command run in the PowerShell Terminal. A recent trend sees attackers moving away from traditional means of targeting systems through specialized tools and applications, and favoring a **living-off-the-land** approach, where they use native tools present on the computer systems being attacked instead.

**Submission:** You need to submit a detailed lab report, with screenshots, to describe what you have done and observed. Questions will be defined as you progress through the lab. The lab report will be compiled as a Word document and submitted on Brightspace by **MONTH DAY at TIME AM/PM.**

As an added bonus, stemming from the way logging occurs is that the event log entry is not created until after the command is optionally decoded. So, in case someone is trying to fool the system with an **-EncodedCommand** PowerShell command, which takes a Base64-encoded command to run, the command is first decoded before it is logged. In that way, **powershell -EncodedCommand 'ZABpAHIAIABjADoAXAA='** will be logged as **dir c:\\**, as illustrated here:

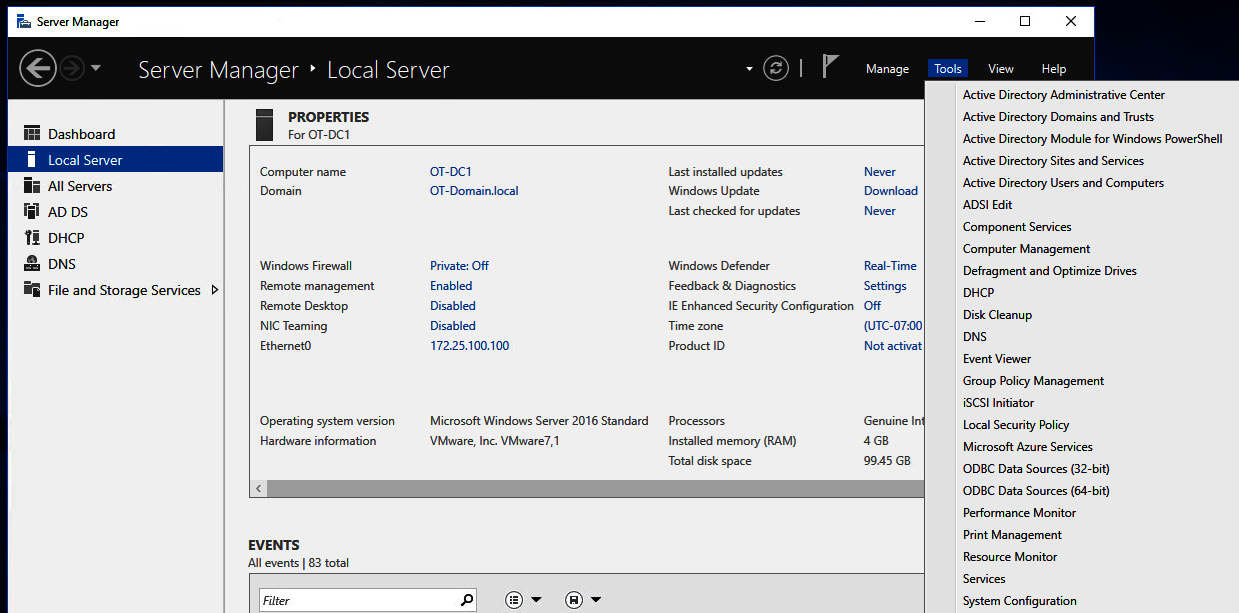


*Figure 9.6 – Exercise 2: Decoded PowerShell command*

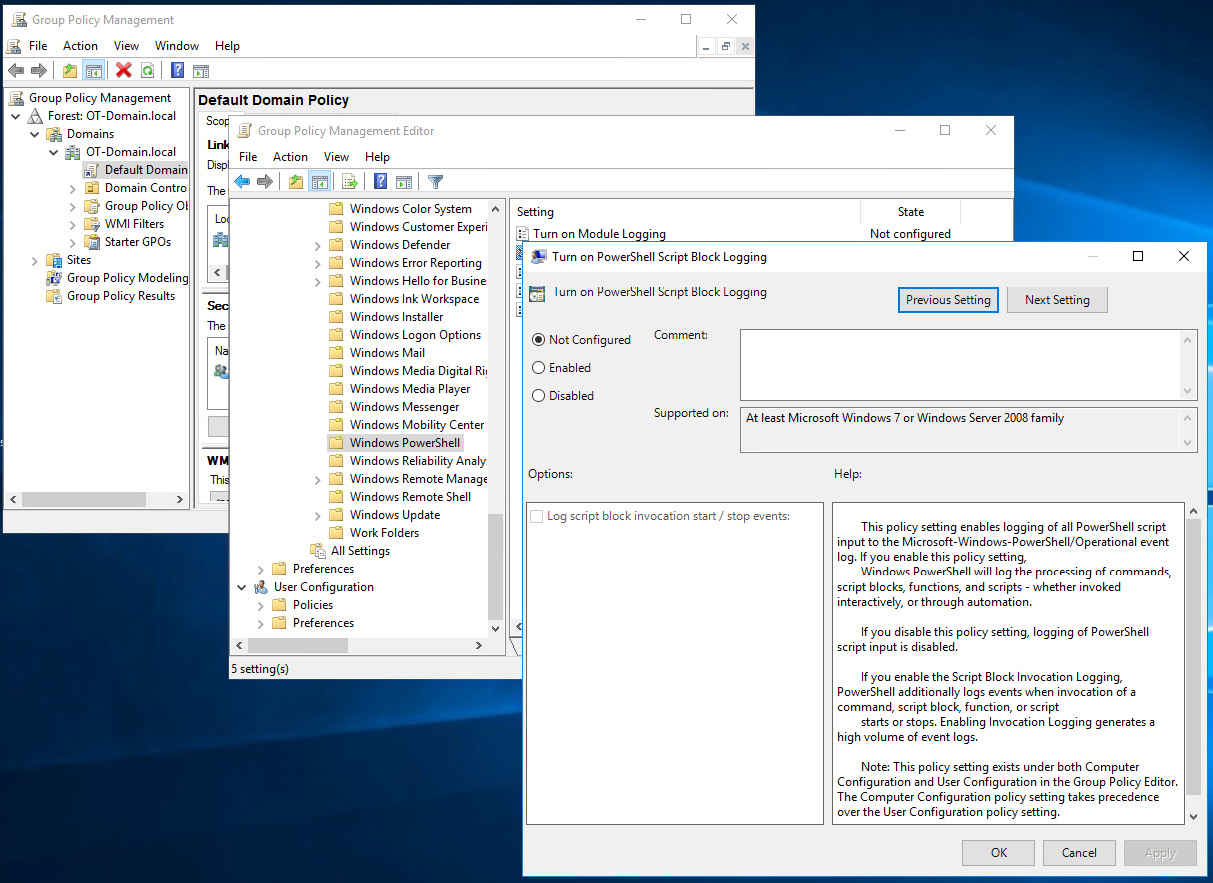
Follow along with these instructions to get **PowerShell Script Block Logging**

configured:

1. **PowerShell Script Block Logging** is enabled through a **Group Policy** setting. We can do this on the local machine (**gpedit.msc**), for Windows computers that are not part of a Windows domain (standalone machines), or it can be done at the domain level by configuring the correct **Group Policy Object** (**GPO**) setting. We will show the GPO option here, but apart from having to use the **gpedit** utility, the standalone method is identical. To start, log in to the domain controller VM (**OT-DC1**).
2. Open the **Server Manager** application and start the **Tools** | **Group Policy Management** tools, as illustrated in the following screenshot:

*Figure 9.7 – Exercise 2: Group Policy Management*

1. Right-mouse click on the **Default Domain Policy** GPO and select **Edit**.
2. In the **Group Policy Management** editor screen that pops up, navigate to **Computer Configuration** | **Policies** | **Administrative Templates** | **Windows Components** | **Windows PowerShell** and double-click on **Turn on PowerShell Script Block Logging**, as illustrated in the following screenshot:

*Figure 9.8 – Exercise 2: Turning on PowerShell Script Block Logging*

1. Check **Enabled** and click on **OK**.
2. That's all for the configuration of **PowerShell Script Block Logging**. The domain replication will take care of configuring all the domain computers.

One detail I need to point out is how Wazuh knows to start looking for PowerShell Script Block logs. Well, if you recall from [*Chapter 6*](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=en-US&rs=en-US&wopisrc=https%3A%2F%2Flivealbany.sharepoint.com%2Fsites%2FGRP-FacetsRootDirectory%2F_vti_bin%2Fwopi.ashx%2Ffiles%2Fab5aa182b4e249318377658da175d6c4&wdorigin=AuthPrompt.OFFICECOM-WEB.START.EDGEWORTH&wdprevioussessionsrc=HarmonyWeb&wdprevioussession=0e2f6a10-a36e-414c-87fa-0afc6e0ae49e&wdenableroaming=1&mscc=1&hid=EB9575A1-9000-7000-6999-8E93F1AAFD04.0&uih=sharepointcom&wdlcid=en-US&jsapi=1&jsapiver=v2&corrid=bbc7af92-3455-cfd6-a86c-4476e8784001&usid=bbc7af92-3455-cfd6-a86c-4476e8784001&newsession=1&sftc=1&uihit=docaspx&muv=1&cac=1&sams=1&mtf=1&sfp=1&sdp=1&hch=1&hwfh=1&dchat=1&sc=%7B%22pmo%22%3A%22https%3A%2F%2Flivealbany.sharepoint.com%22%2C%22pmshare%22%3Atrue%7D&ctp=LeastProtected&rct=Normal&csc=1&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlush#_bookmark58), *Passive Security Monitoring*, we changed the configuration file for the Wazuh agent to include this snippet:

<localfile>

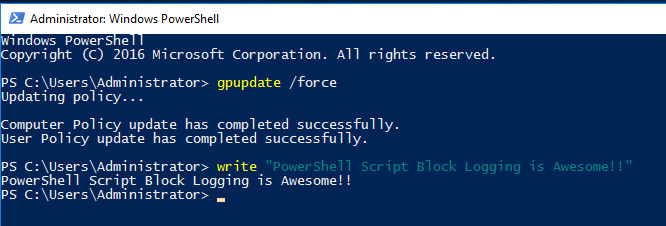
<location>**Microsoft-Windows-PowerShell/Operational**</location>

<log\_format>eventchannel</log\_format>

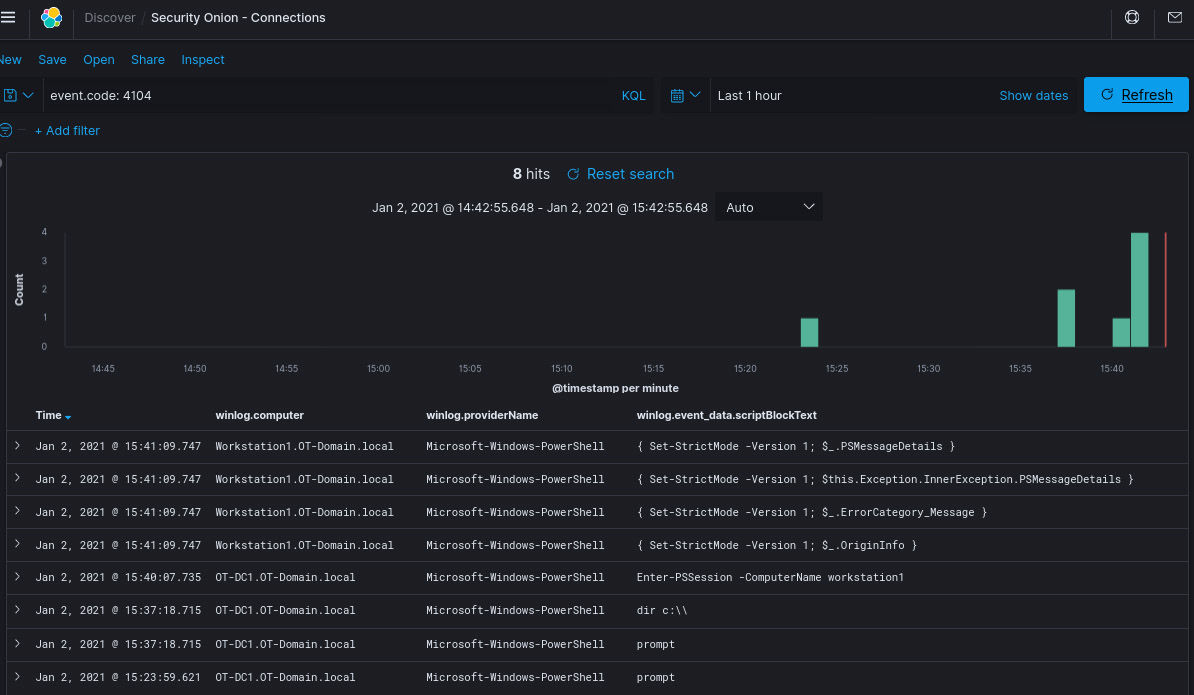
</localfile>

As with the Sysmon logs, the preceding code instructs Wazuh to start monitoring the PowerShell log, sending any new entries to Security Onion to be added into the Elasticsearch database.

Now that this is in place, we enter a command in the PowerShell Terminal of the domain controller (run **gpupdate /force** first, to apply the new group policy), as illustrated in the following screenshot:

*Figure 9.9 – Exercise 2: Generating an example PowerShell Script Block event*

In the following screenshot, we can see this activity being logged in to the Elasticsearch logs via a search for **event.code:4104** (**4104** is the event ID that is attached to **PowerShell Script Block Logging** events):

*Figure 9.10 – Exercise 2: PowerShell Script Block Elasticsearch log entry*

With that, we are now recording PowerShell activity across the domain: a very powerful addition to our security monitoring capabilities.

Next, we are going to look at how to add network intrusion detection capabilities to our pfSense firewall, with the intention of being able to detect attacks on the perimeter.