Detecting Zerologon attacks

Zerologon CVE-2020-11472 is a technique used by attackers to target a Microsoft Windows Domain Controller to reset its computer account password. Attackers can then provide themselves with high privileges and take over the Domain Controller.

These searches are designed to identify activities that indicate the Domain Controller Computer Account has been targeted via remote code exploit or via the use of the Mimikatz tool as the payload carrier.

Required data

- Microsoft: Sysmon
- Microsoft: Windows event logs

Some commands, parameters, and field names in the searches below may need to be adjusted to match your environment. In addition, to optimize the searches shown below, you should specify an index and a time range when appropriate.

How to use Splunk software for this use case

Detect computer changed with anonymous account

To complete this process, your deployment needs to ingest Windows event logs. This process requires audit computer account management to be enabled on the system in order to generate event code 4742. You should specify your environment-specific configurations (index, source, sourcetype, etc.) for Windows event logs.

This search looks for event code 4742 (computer change) or event code 4624 (an account was successfully logged on) with an anonymous account.

```
| search (LogonType=3 TargetUserName="ANONYMOUS LOGON" eventtype=wineventlog_security (EventCode=4624 OR EventCode=4742))
| stats count values(host) AS host, values(TargetDomainName) AS Domain, values(user) AS user
```

Detect credential dumping through LSASS

To complete this process, your deployment needs to ingest Sysmon data and a Sysmon configuration, which includes event code 10 with lsass.exe. You should specify your environment-specific configurations (index, source, sourcetype, etc.) for Sysmon logs.

This search looks for Local Security Authority Subsystem Service (LSASS) memory reads consistent with credential
dumping.

False positives from this search might occur since other tools can access LSASS for legitimate reasons, and it's possible this event could be generated in those cases. In these cases, false positives should be fairly obvious and you may need to tweak the search to eliminate noise.

```
| search (EventCode=10 TargetImage=*lsass.exe (GrantedAccess=0x1010 OR GrantedAccess=0x1410) (source=Syslog:Linux-Sysmon/Operational OR source=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational OR sourcetype=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational)) | stats count min(_time) AS firstTime max(_time) AS lastTime BY Computer, SourceImage, SourceProcessId, TargetImage, TargetProcessId, EventCode, GrantedAccess |
| rename Computer AS dest |
| convert timeformat="%Y-%m-%dT%H:%M:%S" ctime(firstTime) |
| convert timeformat="%Y-%m-%dT%H:%M:%S" ctime(lastTime) |
```

Next steps

The content in this article comes from Splunk Enterprise Security (ES). As a Splunk premium security solution, ES solves a wide range of security analytics and operations use cases including continuous security monitoring, advanced threat detection, compliance, incident investigation, forensics and incident response. Splunk ES delivers an end-to-end view of an organization's security posture with flexible investigations, unmatched performance, and the most flexible deployment options offered in the cloud, on-premises, or hybrid deployment models. If you have questions about this use case, see the Security Research team's support options on GitHub.

In addition, these Splunk resources might help you understand and implement this use case:

- E-book: Ransomware, malware and cyberthreats
- Blog: Hunting with Splunk: The basics