Spoolsv suspicious loaded modules

You might need to search for spoolsv.exe suspicious loaded modules when doing the following:

- Detecting print spooler attacks

Prerequisites

In order to execute this procedure in your environment, the following data, services, or apps are required:

- Technologies: Splunk Enterprise or Splunk Cloud Platform
- Data:
  - Windows event logs
  - Microsoft Sysmon

Example

Some attacks such as PrintNightmare perform module loads into Spoolsv.exe based on dll loading from a specific path. This search detects the loaded modules made by spoolsv.exe after the exploitation.

To optimize the search shown below, you should specify an index and a time range.

1. Ensure you have at least version 6.0.4 of the Sysmon TA.
2. Run the following search:

```
sourcetype=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational OR source=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational EventCode=7 Image ="*\spoolsv.exe" ImageLoaded="*\Windows\System32\spool\drivers\x64\*\" ImageLoaded = "*.dll"
| stats dc(ImageLoaded) AS countImgloaded values(ImageLoaded) AS ImgLoaded count min(_time) AS firstTime max(_time) AS lastTime BY Image Computer EventCode
| where countImgloaded >= 3
```

Search explanation

The table provides an explanation of what each part of this search achieves. You can adjust this query based on the specifics of your environment.
<table>
<thead>
<tr>
<th>Splunk Search</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>sourcetype=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational OR source=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational</td>
<td>Search Sysmon operational data.</td>
</tr>
<tr>
<td>EventCode=7</td>
<td>Search for when a module is loaded in a specific process.</td>
</tr>
<tr>
<td>Image = &quot;<em>\spoolsv.exe&quot; ImageLoaded = &quot;</em>\Windows\System32\spool\drivers\x64*&quot; ImageLoaded = &quot;*.dll&quot;</td>
<td>Search for modules related to spoolsv.exe.</td>
</tr>
<tr>
<td>\</td>
<td>Return the number of times each distinct image was loaded, the name of image loaded, and the first and last time each image was loaded, renaming the fields as shown. Then sort the results by the computer and the event code.</td>
</tr>
<tr>
<td></td>
<td>Return results only when 3 or more modules are loaded by the exploit (unidrv.dll, kernelbase.dll, and the suspect dll) at the same time in the specific target folder.</td>
</tr>
</tbody>
</table>

**Result**

Ensure you filter for false positives on this search.

During triage, isolate the endpoint and review for source of exploitation. Capture any additional file modification events.

If your results indicate an attack has occurred, the host or computer where the vulnerability is detected needs to be further investigated and remediated according to your response plan. This involves a final step of re-imaging the system with a known good system build after investigation.