Unusual Child Process of dns.exe

Identifies an unexpected process spawning from dns.exe, the process responsible for Windows DNS server services, which may indicate activity related to remote code execution or other forms of exploitation.

Rule type: eql

Rule indices:

- winlogbeat-*
- logs-endpoint.events.process-*
- logs-windows.forwarded*
- logs-windows.sysmon_operational-*
- endgame-*
- logs-system.security*
- logs-m365_defender.event-*
- logs-sentinel_one_cloud_funnel.*

Severity: high

Risk score: 73

Runs every: 5m

Searches indices from: now-9m (Date Math format, see also `Additional look-back time`)

Maximum alerts per execution: 100

References:

- https://research.checkpoint.com/2020/resolving-your-way-into-domain-admin-exploiting-a-17-year-old-bug-in-windows-dns-servers/
- https://msrc-blog.microsoft.com/2020/07/14/july-2020-security-update-cve-2020-1350-vulnerability-in-windows-domain-name-system-dns-server/
- https://github.com/maxpl0it/CVE-2020-1350-DoS
- https://www.elastic.co/security-labs/detection-rules-for-sigred-vulnerability

Tags:

- Domain: Endpoint
- OS: Windows
- Use Case: Threat Detection
- Tactic: Lateral Movement
- Resources: Investigation Guide
- Data Source: Elastic Endgame

Rule authors:

- Elastic

Rule license: Elastic License v2

Investigation guide

edit

Triage and analysis

Investigating Unusual Child Process of dns.exe

SIGRed (CVE-2020-1350) is a wormable, critical vulnerability in the Windows DNS server that affects Windows Server versions 2003 to 2019 and can be triggered by a malicious DNS response. Because the service is running in elevated privileges (SYSTEM), an attacker that successfully exploits it is granted Domain Administrator rights. This can effectively compromise the entire corporate infrastructure.

This rule looks for unusual children of the `dns.exe` process, which can indicate the exploitation of the SIGRed or a similar remote code execution vulnerability in the DNS server.

Possible investigation steps

- Investigate the process execution chain (parent process tree) for unknown processes.
- Any suspicious or abnormal child process spawned from dns.exe should be carefully reviewed and investigated. It’s impossible to predict what an adversary may deploy as the follow-on process after the exploit, but built-in discovery/enumeration utilities should be top of mind (`whoami.exe`, `netstat.exe`, `systeminfo.exe`, `tasklist.exe`).
- Built-in Windows programs that contain capabilities used to download and execute additional payloads should also be considered. This is not an exhaustive list, but ideal candidates to start out would be: `mshta.exe`, `powershell.exe`, `regsvr32.exe`, `rundll32.exe`, `wscript.exe`, `wmic.exe`.
- If a denial-of-service (DoS) exploit is successful and DNS Server service crashes, be mindful of potential child processes related to `werfault.exe` occurring.
- Investigate any abnormal behavior by the subject process such as network connections, registry or file modifications, and any spawned child processes.
- Investigate other alerts associated with the host during the past 48 hours.
- Check whether the server is vulnerable to CVE-2020-1350.
- Assess whether this behavior is prevalent in the environment by looking for similar occurrences across hosts.

False positive analysis

- This activity is unlikely to happen legitimately. Benign true positives (B-TPs) can be added as exceptions if necessary.

- Reimage the host operating system or restore the compromised server to a clean state.
- Install the latest patches on systems that run Microsoft DNS Server.
- Consider the implementation of a patch management system, such as the Windows Server Update Services (WSUS).
- Run a full antimalware scan. This may reveal additional artifacts left in the system, persistence mechanisms, and malware components.
- Determine the initial vector abused by the attacker and take action to prevent reinfection through the same vector.
- Review the privileges assigned to the user to ensure that the least privilege principle is being followed.
- Using the incident response data, update logging and audit policies to improve the mean time to detect (MTTD) and the mean time to respond (MTTR).

Rule query

edit

```
process where host.os.type == "windows" and event.type == "start
  not process.name : "conhost.exe"
```

Framework: MITRE ATT&CK™

- Tactic:
  - Name: Lateral Movement
  - ID: TA0008
  - Reference URL: https://attack.mitre.org/tactics/TA0008/
- Technique:
  - Name: Exploitation of Remote Services
  - ID: T1210
  - Reference URL: https://attack.mitre.org/techniques/T1210/

« Unusual Child Process from a System Virtual Process

Unusual Child Processes of RunDLL32 »