New Form of Ransomware impacting healthcare providers

There was a fundamental shift in tradecraft used by malicious threat actors that recently attacked a healthcare provider in California and several other companies outside of the healthcare industry. The shift has significant implications to health providers since the delivery approach was not through conventional methods: a phishing email or malware infecting a personal device. Instead this approach used a known vulnerability in how web applications are configured on web servers. In other words, the methods to prevent this type of attack apply to software security practices specifically replacing outdated Jboss applications (version 6.x, 5.x, 4.x or 3.x).

Please initiate a review of the web applications in your respective environment and upgrade outdated Jboss versions to 7.x to avoid successful attacks of this type of ransomware going forward. The specific attack on Hollywood Presbyterian Hospital appears to have been opportunistic given other attacks using the same techniques outside of healthcare. The attackers used an open source exploit tool (JEX-BOSS) to attack a Jboss server version 6.1. The same attack methods were deployed in at least two other enterprises outside of healthcare according to Dell SecureWorks, who provide services to the security operations center for the NH-ISAC.

The victims were running JBOSS Application Server 6.1.0 Final "Neo" and the following vulnerabilities are associated with this version:

- https://www.cvedetails.com/vulnerability-list/vendor_id-25/product_id-14972/version_id-150006/Redhat-Jboss-Enterprise-Application-Platform-6.1.0.html
  - The attackers used open source exploit toolkit called “JEX-BOSS” which will exploit JBOSS Application Server versions 3, 4, 5, and 6.
- https://github.com/joaomatosf/jexboss
  - The attackers also used Re-George
- https://github.com/sensepost/reGeorg
  - Attackers uploaded a .war file which contains a webshell
  - Attackers create a new user account which is added to the Administrators group. The account is logged in via RDP from localhost (127.0.0.1). 127.0.0.1 is logged as source IP when RDP is tunneled over SSH
  - Attackers used the account to install the Hyena network scanner
  - Attackers also used legit Microsoft tool “csvdev.exe” (internal name ldifldap.lib) from 2010 to map the entire network and identify accessible systems. Output of the tool is saved to a file named “com.csv”
  - They also deployed Microsoft Sysinternals tool PSExec.exe to remotely execute their malware
  - They also deployed Microsoft Sysinternals tool SDelete.exe to delete traces of their malware.
  - No network call-outs or C2 servers were involved in any of the attacks
  - The account name used on the Jboss server by the attacker was “jboss”

This information is provided by the Dell SecureWorks Counter Threat Unit(TM) (CTU) who provide core security operations services to the National Health Information Sharing & Analysis Center (NH-ISAC).

Dell SecureWorks Counter Threat Unit(TM) (CTU) researchers observed **a new ransomware threat during incident response engagements** and named it SamsamCrypt based on its use of the filename samsam.exe. It is also known as Ransom:MSIL/Samas.A and CRYPSAM. SamsamCrypt does not appear to use typical delivery mechanisms such as email phishing campaigns, malvertising campaigns, malicious OLE documents, or exploit kits. Instead, CTU researchers have observed threat actors deploying SamsamCrypt after they compromise a network and establish persistent or brief backdoor access. In analyzed incidents, CTU researchers observed initial intrusion occurring via outdated JBOSS applications. The threat actors then scanned the network and mapped accessible hosts, often using freely available or open source tools, before deploying the SamsamCrypt ransomware to accessible hosts and attempting to locate and delete network backups using a separate custom tool. CTU researchers previously identified other post-intrusion ransomware deployments.

The SamsamCrypt ransomware binary (samsam.exe), the custom tool to locate and delete network backups (delfiletype.exe), and an RSA public key (*<ComputerName>*_PublicKey.xml) used in the encryption process were manually copied from a remote server onto a system within the attacker-accessible network. The ransomware is executed on the system, and the RSA public key in XML format is passed as an argument to the malware. SamsamCrypt is also copied to all other network-accessible systems, including servers and workstations, and is remotely executed using the Microsoft Sysinternals PSExec.exe utility.

The SamsamCrypt ransomware appears to have been active since at least mid-December 2015. It was discovered running on servers as a background process and also affected Microsoft Exchange databases by stopping related processes and encrypting database files. It rendered endpoint security software useless if the software failed to detect the threat and the application files got encrypted.

SamsamCrypt enumerates all files on the local hard drive and any connected network shares to find files associated with a hard-coded list of file extensions (see Figure 1). Identified files are encrypted using the Windows Cryptography API, with a symmetric-encryption algorithm (Rijndael) key that is randomly generated on the compromised system.

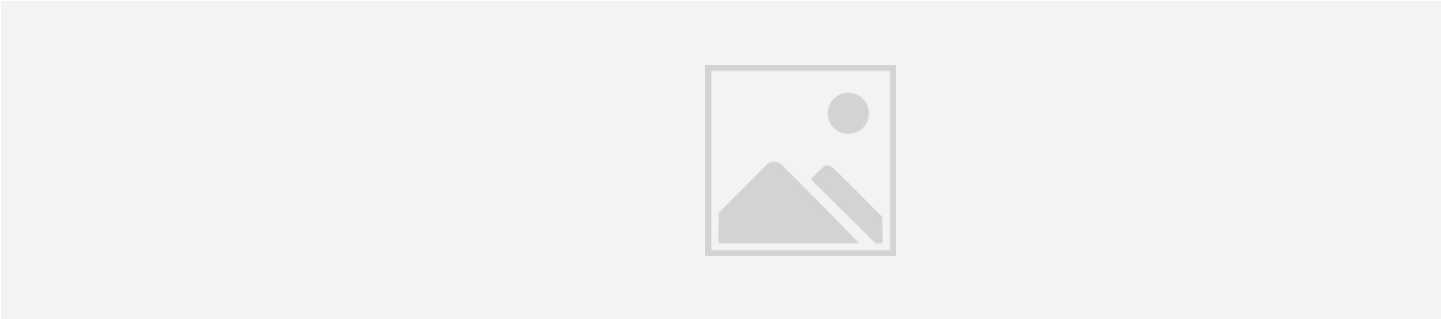

Figure 1. Hard-coded list of targeted file extensions in samsam.exe. (Source: Dell SecureWorks)

The Rijndael encryption algorithm is slightly different from the Advanced Encryption Standard (AES) algorithm. The Rijndael 128-bit key and its initialization vector (IV) are encrypted with an RSA-2048 public key (asymmetric encryption) and placed within the header of each encrypted file. The RSA public and private key pair are generated on the attacker's server. The RSA private key, controlled by the threat actor, is required to decrypt the files. The attacker deploys the RSA public key and passes it as an argument to samsam.exe. The key is stored as an XML file named *<ComputerName>*_PublicKey.xml within the %TEMP% folder of each affected system. The presence of this file indicates ransomware activity and can be used to identify infected hosts. In some samples, the RSA public key is embedded in the main binary. Encrypted files have the .encryptedRSA file extension. SamsamCrypt also creates the HTML file HELPDECYPRT_YOUR_FILES.html in all affected folders and in the %Desktop% folder, and displays it to victims. This file contains instructions for how victims can decrypt their files (see Figure 2).

Please feel free to reach out to the NH-ISAC (www.nh-isac.org) for more information on software security practices to prevent these types of attacks.

Figure 2. Ransom note displayed by SamsamCrypt ransomware. (Source: Dell SecureWorks)

The ransom note demands one or more bitcoins for each affected system, gives a deadline of seven days from the date of infection to retrieve the decryption tool, and provides a Bitcoin address to send the payment. The victim is advised to visit a certain blog (for example, union83939k . wordpress . com) and leave a comment under the post with the compromised computer's name and Bitcoin transaction reference (see Figure 3). The attacker then provides a download link to an executable file that supposedly decrypts the encrypted files. The tool contains an executable file and the corresponding RSA private key in XML format, and it is unique to each victim.

Figure 3. Attacker-created blogs for SamsamCrypt ransomware. (Source: Dell SecureWorks)

CTU researchers observed the main SamsamCrypt ransomware binary using the filenames samsam.exe or VID282117004.exe. It is a .NET compiled binary that has two PE executable files embedded within its resource section:

- del.exe (MD5: e189b5ce11618bb7880e9b09d53a588f) — the legitimate SDelete.exe tool from Microsoft Sysinternals that is used to securely wipe free disk space

- selfdel.exe (MD5: 710a45e007502b8f42a27ee05dcd2fba) — a .NET compiled binary used to delete samsam.exe and del.exe

After encrypting files of interest on a system, SamsamCrypt launches the SDelete program (del.exe -c C: /accepteula) to wipe the free space on the disk and prevent recovery efforts. It uses selfdel.exe, invoking del.exe, to securely delete the samsam.exe file (del.exe -p 16 samsam.exe) and then deletes del.exe. The threat actor deploys another .NET compiled binary, delfiletype.exe (MD5: 8ef662e043bb8e75a0caa45cc5db358d), to delete all backup and restore files from the local system and any network accessible drives. CTU researchers have also observed this file using the filename sqlsrvtmg1.exe. The binary searches for the hard-coded file extensions shown in Figure 4. If the binary cannot delete identified files because their corresponding process is running in memory, it lists the process ID based on filename (tasklist /v /fo csv) and then kills that process (taskkill /f /pid). After the process is killed, the corresponding file on disk is deleted. An early December 2015 sample analyzed by CTU researchers also included the command to delete shadow volume copies (vssadmin delete shadows /all /quiet).

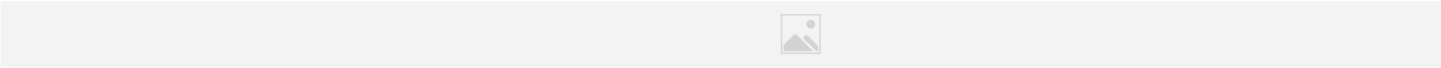

Figure 4. Hard-coded list of targeted file extensions in delfiletype.exe. (Source: Dell SecureWorks)

SamsamCrypt has unique characteristics not commonly observed in other ransomware:

- There is no registry persistence.
- The malware binary is deleted from the file system.
- There are no scheduled tasks to start or stop encryption activity.
- It is spread to all accessible systems on the network via attacker-deployed tools.
- There is no malicious network activity (e.g., command and control (C2) server communication, use of Tor, data exfiltration).

Unlike ransomware that infects endpoints via phishing or exploit kits and just impacts files on the local system and on network-mapped drives, SamsamCrypt can affect files located on internal endpoints within a network. In incidents analyzed by CTU researchers, threat actors either gained access to the network or exploited a web application vulnerability to remotely deploy and execute the malware. CTU researchers have observed SamsamCrypt deployed on as many as 30% of a compromised organization's workstations and servers, with as many as 143 servers infected with SamsamCrypt in a single environment. Use and employment of SamsamCrypt in compromised environments appears to be opportunistic across industry verticals. As of this publication, this threat has impacted the business services, transportation and hospitality, education, and technology provider verticals.

CTU researchers recommend that clients use a cloud-based backup solution that is set up to perform scheduled backups. Clients should also frequently test that the backup and restore is working. Local backups such as network-attached storage (NAS) are not sufficient because the attackers delete local and network accessible backup files.

The CTU research team has developed the Red Cloak rule listed in Table 1 to detect activity associated with this threat. Dell SecureWorks iSensor countermeasures are not feasible because SamsamCrypt does not generate network activity. Third-party devices receive updated protection as it is released from the respective vendors and deployed by Dell SecureWorks device management security teams.

Name	GUID
Volume Shadow Copy Deletion - All	5f126953-bfcc-4112-80d6-68feaccdd0f6

Table 1. Dell SecureWorks Red Cloak rule covering this threat.

To mitigate exposure to this threat, CTU researchers recommend that clients use available controls to restrict access using the indicators in Table 2. The URLs may contain malicious content, so consider the risks before opening them in a browser.