Enhanced Attack Report

COLDSTEEL

Generated on 2025-04-10

Disclaimer

This report has been generated automatically and should be used for informational purposes only. To generate this report, Large Language Models (LLMs) were used to analyze the provided data. This technology is not perfect and may generate incorrect or misleading results. The results should be reviewed by a human expert before taking any action based on the information provided.

Definitions

Pre-Conditions: Conditions that must be true to execute the attack steps in the milestone.

Post-Conditions: Traces that an attacker leaves behind after executing the attack steps in the milestone.

Attack Steps: Steps that an attacker would take to achieve the goal of the milestone.

MITRE Technique: Techniques from the MITRE ATT&CK; framework that are relevant to the milestone.

STIX

Malware Name: COLDSTEEL
Malware Description: â€¢ COLDSTEEL provides interactive desktop & command line invocation, functionality including the ability to copy files, take screenshots and simulate user input. â€¢ COLDSTEEL persists as a Windows service. â€¢ COLDSTEEL communicates with the C2 server using a raw TCP connection.

Quick Overview

Milestone m1
a1. Supply Chain Compromise: Compromise Software Dependencies and Development Tools as used by the malware (T1195.001)

Milestone m2
a2. Compromise Client Software Binary as used by the malware (T1554)

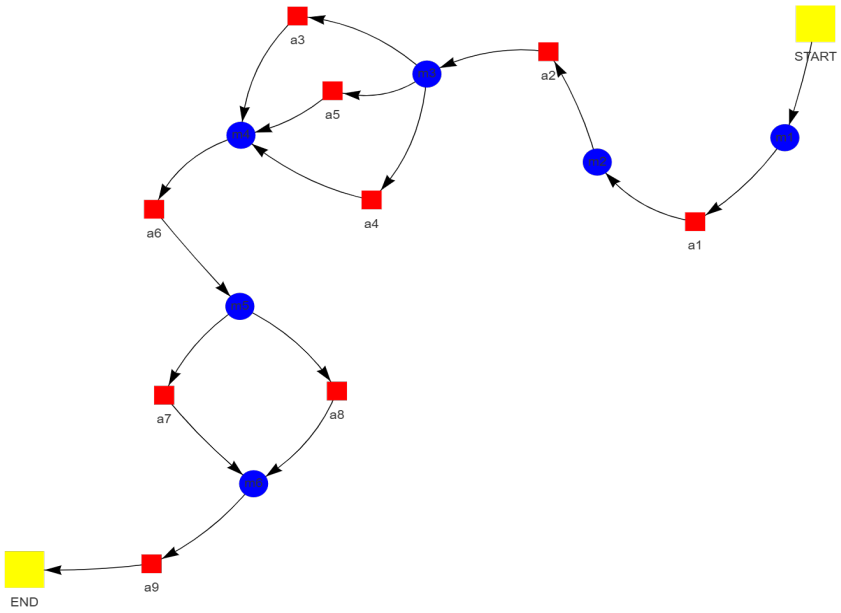
Milestone m3
a3. Deobfuscate/Decode Files or Information as used by the malware (T1140)
a4. Indicator Removal: File Deletion as used by the malware (T1070.004)
a5. Virtualization/Sandbox Evasion: Time Based Evasion as used by the malware (T1497.003)

Milestone m4
a6. Automated Collection as used by the malware (T1119)

Milestone m5
a7. Application Layer Protocol: Web Protocols as used by the malware (T1071.001)
a8. Fallback Channels as used by the malware (T1008)

Milestone m6
a9. Automated Exfiltration as used by the malware (T1020)

Attack Graph



Milestone *m1*

Attack Step *a1*

==========

Name: Supply Chain Compromise: Compromise Software Dependencies and Development Tools as used by the malware

Description: Malicious code is embedded into legitimate 3CX software components via the compromise of software dependencies and development tools. This process involves the insertion of malicious libraries into the 3CXDesktopApp application during the build phase, effectively transforming it into a Trojan horse. The compromised software is subsequently signed by 3CX and undergoes Apple notarization, thereby enhancing its perceived legitimacy and trustworthiness. Distribution occurs through established channels utilized for legitimate 3CX products, resulting in unwitting user installation of malware-infected software.

Pre-Conditions:

- Legitimate software dependencies are used in the 3CX software build.
- Malicious libraries are available.
- Knowledge of software compilation and packaging processes.
- The 3CX software development process is accessible.
- Ability to modify software dependencies during the build process.

Post-Conditions:

- Logs indicating communication with C2 servers.
- Network connections to obfuscated C2 domains.
- Modified 3CX software files (including potentially injected libraries).
- Data exfiltration from infected devices.
- New configuration files created by Smooth Operator.
- Persistence of malware within the system.
- Compromised 3CX software installations.
- Potential for further malicious activity by the threat actor.
- Presence of the ".main_storage" file required for second-stage execution.
- Exfiltrated data files containing sensitive information.

MITRE Technique

ID: T1195.001
Name: Supply chain compromise: compromise software dependencies and development tools
Description: Adversaries may manipulate software dependencies and development tools prior to receipt by a final consumer for the purpose of data or system compromise. Applications often depend on external software to function properly. Popular open source projects that are used as dependencies in many applications may be targeted as a means to add malicious code to users of the dependency.
More info: https://attack.mitre.org/techniques/T1195/001

Indicators

- Domain names may be registered for malicious purposes.
- Files with unusual or generic names may be created.
- URLs may point to malicious websites or servers.

Milestone *m2*

Attack Step *a2*

==========
Name: Compromise Client Software Binary as used by the malware
Description: Compromise Client Software Binary persistence is achieved by Smooth Operator through integration as a component within legitimate 3CX software. Distribution occurs via compromised and signed 3CX software packages, resulting in unwitting user installation of the malicious payload alongside the legitimate application. The malware, specifically the libffmpeg.dylib file (a universal binary for both Intel and ARM macOS), integrates itself into the structure of the larger 3CX software. Due to this integration, Smooth Operator executes with elevated privileges upon execution of the legitimate software. This privileged execution facilitates circumvention of security measures and enables persistent presence on the compromised system.

Pre-Conditions:

- The malicious code has been inserted into the libffmpeg.dylib file.
- The 3CX software is installed on the victim's system.
- A system running the 3CX software.

Post-Conditions:

- Persistence within the system through modified 3CX software
- ".main_storage" file containing victim ID
- Execution of malicious payloads
- Data exfiltration from infected device
- Files written by the "UpdateAgent" binary
- Compromised 3CX software installation
- Obfuscated data files on disk
- Logs indicating execution of malicious code and data transfer
- Modified 3CX configuration files
- Modified libffmpeg.dylib file
- Network connections to C2 servers

MITRE Technique

ID: T1554
Name: Compromise host software binary
Description: Adversaries may modify host software binaries to establish persistent access to systems. Software binaries/executables provide a wide range of system commands or services, programs, and libraries. Common software binaries are SSH clients, FTP clients, email clients, web browsers, and many other user or server applications.
More info: https://attack.mitre.org/techniques/T1554/

Indicators

- Filename: UpdateAgent
- Filename: .main_storage

Milestone *m3*

Attack Step *a3*

==========
Name: Deobfuscate/Decode Files or Information as used by the malware
Description: Deobfuscation/decode operations are executed by Smooth Operator malware to facilitate its functionality. Data intended for transmission to the Command & Control (C2) server is obfuscated using a custom algorithm, rendering it resistant to analysis by security tools. Conversely, data written to files on the infected system and responses received from the C2 server are deobfuscated, enabling the malware to process sensitive information retrieved from the server. A specific technique employed involves the utilization of an XOR key (0x7A) for obfuscating a list of 15 C2 servers and one URL associated with the 3CX website. This capability is essential for concealing malicious activity on the network and granting access to sensitive information through deobfuscated responses, facilitating command execution and data retrieval.

Pre-Conditions:

- Smooth Operator uses a custom algorithm to obfuscate data.
- Smooth Operator malware is present in the system.
- The malware has received data to be deobfuscated.

Post-Conditions:

- Data theft from targeted organizations
- Presence of malware indicators of compromise (IOCs) in system memory and disk
- Potential for further malware infections
- Modified browser history and data files
- Modified 3CX desktop app executable files
- Reputational damage to affected organizations
- Compromised 3CX systems
- Unusual process activity and network traffic patterns
- Network connections to malicious C2 servers
- Logs containing suspicious activity within the 3CX application and system logs
- New files created in user directories (e.g., "UpdateAgent")

MITRE Technique

ID: T1140
Name: Deobfuscate/decode files or information
Description: Adversaries may use Obfuscated Files or Information to hide artifacts of an intrusion from analysis. They may require separate mechanisms to decode or deobfuscate that information depending on how they intend to use it. Methods for doing that include built-in functionality of malware or by using utilities present on the system.
More info: https://attack.mitre.org/techniques/T1140/

Indicators

- Filename: UpdateAgent
- Filename: .main_storage

Attack Step *a4*

==========
Name: Indicator Removal: File Deletion as used by the malware
Description: Indicator removal during the second stage of malware execution is achieved through file deletion. The UpdateAgent binary undergoes self-deletion from the disk immediately following execution. This action is intended to impede forensic analysis and attribution by eliminating evidence of its presence.

Pre-Conditions:

- The malware's code contains the functionality to delete itself from disk.
- The malware is running in an environment with file system access permissions.
- The malware has successfully executed its second stage.

Post-Conditions:

- Potential for further malicious activity.
- Deleted malware binaries from disk.
- Logs of beaconing activity.
- Modified system registry settings.
- Compromised system with malware installed.
- Network connections to command and control (C2) servers.
- Data exfiltration to external servers.
- Presence of malicious code within legitimate processes.
- Obfuscated data files on disk.
- Modified configuration files in the 3CX installation directory.

MITRE Technique

ID: T1070.004
Name: Indicator removal: file deletion
Description: Adversaries may delete files left behind by the actions of their intrusion activity. Malware, tools, or other non-native files dropped or created on a system by an adversary (ex: Ingress Tool Transfer) may leave traces to indicate to what was done within a network and how. Removal of these files can occur during an intrusion, or as part of a post-intrusion process to minimize the adversary's footprint.
More info: https://attack.mitre.org/techniques/T1070/004

Indicators

- Filename: .main_storage

Attack Step *a5*

==========
Name: Virtualization/Sandbox Evasion: Time Based Evasion as used by the malware
Description: Smooth Operator utilizes time-based evasion techniques to circumvent detection within virtualized environments or sandboxed conditions. Execution is intermittently suspended for extended durations, typically exceeding one week, through the implementation of a sleep mechanism. This

prolonged quiescence significantly impedes behavioral analysis within sandboxes, as these environments are frequently terminated or reset prior to the completion of the designated sleep period. Subsequent beaconing activity, involving communication with command-and-control servers, is scheduled to occur following this extended dormant phase. The substantial delay inherent in this approach presents a challenge for sandbox simulations aiming to accurately replicate realistic infection scenarios characterized by prolonged attacker interaction with the malware.

Pre-Conditions:

- Network connectivity for the malware to establish communication with its command and control server.
- Sufficient processing power and memory for the malware to execute its code.
- The malware has access to a system clock.
- A target system with a functioning operating system.
- The malware is running.

Post-Conditions:

- Altered registry settings
- Network traffic to C2 servers
- Deleted files related to malware removal attempts
- Data exfiltration
- New files created (e.g., UpdateAgent)
- Compromised system
- Malware persistence
- Log entries indicating suspicious activity
- System instability
- Modified system files

MITRE Technique

ID: T1497.003
Name: Virtualization/sandbox evasion: time based evasion
Description: Adversaries may employ various time-based methods to detect and avoid virtualization and analysis environments. This may include enumerating time-based properties, such as uptime or the system clock, as well as the use of timers or other triggers to avoid a virtual machine environment (VME) or sandbox, specifically those that are automated or only operate for a limited amount of time.
More info: https://attack.mitre.org/techniques/T1497/003

Indicators

- The filename ".main_storage" is observed.

Milestone *m4*

Attack Step *a6*

==========
Name: Automated Collection as used by the malware
Description: Automated Collection by Smooth Operator is facilitated through the active data gathering performed by its distinct stages from the compromised victim machine. The acquired data is subsequently integrated into either beacons transmitted to the command and control (C2) server or during exfiltration processes. Data collection encompasses a range of categories, including system information such as operating system details, hardware specifications, and installed software lists. User data, encompassing login credentials, browsing history, email contents, and personal files, is also targeted. Network information, comprising IP addresses, network configurations, and connected devices, is gathered. Furthermore, process information, including running applications, system processes, and memory usage, is collected. Prior to transmission, the collected data undergoes obfuscation techniques. These typically involve XOR encryption utilizing a single byte XOR key (0x7A) to scramble the data. Additionally, HTML encoding is employed, replacing special characters with their HTML equivalents to potentially circumvent detection by security systems. The obfuscated data is subsequently incorporated into either beacons, which are regular transmissions sent to the C2 server containing various collected information and awaiting further instructions, or exfiltration processes, where large chunks of stolen data are directly transmitted to the attacker's server for subsequent analysis and exploitation. This automated data collection mechanism enables Smooth Operator to acquire a comprehensive understanding of the victim machine and its environment. This information can be leveraged by attackers for a variety of malicious purposes, including lateral movement, data theft, ransomware deployment, and targeted attacks.

Pre-Conditions:

- The victim machine has an internet connection.
- The malware is running on a victim machine.
- The victim machine has data to be collected.

Post-Conditions:

- Altered registry settings
- New files created (e.g., "UpdateAgent")
- Network traffic to obfuscated C2 servers
- "config.json" file containing stolen system information
- Data exfiltration
- Potential for further malware infection
- Compromised system
- Log entries indicating suspicious activity
- System instability
- Browser data theft
- Modified system files

MITRE Technique

ID: T1119
Name: Automated collection

Description: Once established within a system or network, an adversary may use automated techniques for collecting internal data. Methods for performing this technique could include use of a Command and Scripting Interpreter to search for and copy information fitting set criteria such as file type, location, or name at specific time intervals.
More info: https://attack.mitre.org/techniques/T1119/

Indicators

- A file named "UpdateAgent" is present.
- A file named ".main_storage" is present.

Milestone *m5*

Attack Step *a7*

==========
Name: Application Layer Protocol: Web Protocols as used by the malware
Description: HTTPS is utilized by malware for communication with its command and control (C2) server, facilitating secure data transmission over the internet. Encrypted communication channels are established via HTTPS, rendering intercepted data unintelligible to unauthorized parties. SSL/TLS protocols, employing digital certificates, authenticate the C2 server and ensure secure connection establishment. Malware transmits encrypted requests to the C2 server, seeking instructions or data. These requests may encompass tasks such as information exfiltration, supplementary malware download, or initiation of further attacks. Encrypted responses from the C2 server are received by the malware, subsequently decrypted and executed.

Pre-Conditions:

- A victim machine infected with the Smooth Operator malware.
- The victim machine has an active internet connection.
- The victim machine's operating system allows for HTTPS communication.
- The malware is running on a victim machine.
- A C2 server accessible via HTTPS.

Post-Conditions:

- Network traffic to C2 servers
- Data exfiltration
- System compromise
- Command and control server communication established
- New files created (e.g., malware executables, configuration files)
- Malware persistence
- Logs containing suspicious activity
- Altered registry entries
- Data remnants on victim machine
- Modified system files

MITRE Technique

ID: T1071.001
Name: Application layer protocol: web protocols
Description: Adversaries may communicate using application layer protocols associated with web traffic to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.
More info: https://attack.mitre.org/techniques/T1071/001

Indicators

- The domain "azureonlinestorage.com" is accessed.
- The URL "https://akamaitechcloudservices.com/v2/fileapi" is accessed.

- The URL "https://glcloudservice.com/v1/status" is accessed.
- The URL "https://msedgepackageinfo.com/ms-webview" is accessed.
- The URL "https://pbxsources.com/queue" is accessed.

Attack Step *a8*

==========
Name: Fallback Channels as used by the malware
Description: A list of fifteen distinct C2 server addresses is hardcoded within the malware's code. Prior to initiating a beacon transmission, a server address is randomly selected from this embedded list. Consequently, a single infected device may establish communication with multiple C2 servers over time. This dynamic beaconing strategy enhances the malware's resilience by providing alternative communication channels in the event that a C2 server becomes inaccessible or compromised.

Pre-Conditions:

- The initial C2 connection fails.
- The malware is running on a victim's system.
- A compromised system infected with the Smooth Operator malware.
- The malware has established an initial connection to a C2 server.
- An active internet connection on the victim's system.
- HTTPS protocol support on both the victim's system and the C2 servers.

Post-Conditions:

- Hidden processes running
- Potential for further malware infections
- Exfiltrated data files
- Backdoor connections established
- Loss of sensitive information
- Compromised system integrity
- New user accounts created
- Unusual network traffic logs
- Altered registry entries
- Increased risk of data breaches
- Modified system files
- Disruption of business operations

MITRE Technique

ID: T1008
Name: Fallback channels
Description: Adversaries may use fallback or alternate communication channels if the primary channel is compromised or inaccessible in order to maintain reliable command and control and to avoid data transfer thresholds.
More info: https://attack.mitre.org/techniques/T1008/

Indicators

- A domain name is used.
- A file named "UpdateAgent" is present.

Milestone *m6*

Attack Step *a9*

==========
Name: Automated Exfiltration as used by the malware
Description: The precise mechanism employed for automated data exfiltration by Smooth Operator malware remains to be elucidated. While the provided textual description outlines several salient characteristics of the malware, it does not explicitly detail the specific methods utilized for exfiltration. Deductions regarding potential exfiltration methodologies can be inferred from the available information and prevalent malware tactics. Data collection is likely achieved through a variety of means, including system activity monitoring, user action logging, and sensitive file access. The text alludes to "custom data obfuscation" employed for both beaconing (communication with command-and-control servers) and exfiltration, suggesting that stolen data is encoded to impede detection. The statement that "the extracted domain and account name are concatenated together, separated by a semicolon and exfiltrated" implies that the exfiltration process likely occurs through a communication channel established by Smooth Operator, potentially diverging from the conventional C2 channel for this specific action. Several potential exfiltration mechanisms can be hypothesized: The establishment of a novel, dedicated channel specifically designed for data exfiltration is a possibility. Such a channel might leverage techniques such as DNS tunneling or ICMP tunneling, or utilize HTTP POST requests directed to seemingly innocuous websites. Alternatively, the malware could modify its existing communication with the C2 server to incorporate the exfiltrated data. This modification might involve encapsulating the data within a specific command or message type transmitted to the C2 server, or employing steganography to conceal the data within ostensibly benign files or images sent to the C2 server. To ascertain the precise exfiltration method employed, in-depth analysis of several factors is required: The malware's source code must be scrutinized to identify communication routines and data handling procedures. Network traffic emanating from and directed towards the infected system should be meticulously examined for any anomalous patterns. Logs maintained on the C2 server should be thoroughly reviewed for any indication of unusual data transfers or requests originating from compromised machines.

Pre-Conditions:

- Data has been automatically collected by Smooth Operator.
- Smooth Operator malware is running.
- The custom data obfuscation algorithm is functional.
- A connection to a C2 server is established.

Post-Conditions:

- Compromised system functionality
- Indicator of Compromise (IOC) files on compromised systems.
- Obfuscated communication logs
- Potential for further malware infections
- Lateral movement within the network
- Exfiltrated data files
- Unusual network traffic to command and control servers
- New files created with malicious content
- Altered registry entries
- Logs containing suspicious activity

- Network reconnaissance
- Modified system files
- Data theft

MITRE Technique

ID: T1020
Name: Automated exfiltration
Description: Adversaries may exfiltrate data, such as sensitive documents, through the use of automated processing after being gathered during Collection.
More info: https://attack.mitre.org/techniques/T1020/

Indicators

- A file named "UpdateAgent" is present.
- A file named ".main_storage" is present.