# Enhanced Attack Report

## COLDSTEEL

*Generated on 2025-04-07*

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

## Quick Overview

**Milestone 1**
1. Exploit Public-Facing Application as used by the malware (T1190)

**Milestone 2**
1. Command and Scripting Interpreter: Windows Command Shell as used by the malware (T1059.003)
2. System Services: Service Execution as used by the malware (T1569.002)

**Milestone 3**
1. Create or Modify System Process: Windows Service as used by the malware (T1543.003)

**Milestone 4**
1. Obfuscated Files or Information: Software Packing as used by the malware (T1027.002)
2. Modify Registry as used by the malware (T1112)
3. Indicator Removal: File Deletion as used by the malware (T1070.004)
4. Access Token Manipulation: Create Process with Token as used by the malware (T1134.002)

**Milestone 5**
1. System Information Discovery as used by the malware (T1082)
2. File and Directory Discovery as used by the malware (T1083)
3. Process Discovery as used by the malware (T1057)

**Milestone 6**
1. Non-Application Layer Protocol as used by the malware (T1095)

# Milestone 1

## Attack Step 1.1

```
==================================================
```
**Name:** Exploit Public-Facing Application as used by the malware
**Description:** The COLDSTEEL malware exhibits techniques for handle manipulation within Windows systems. Processes and thread handle creation/duplication events are intercepted via ObRegisterCallbacks. Based on image path or PID, user-mode processes listed in the malware's internal global process list have their "PROCESS_TERMINATE" permission selectively removed. This action suggests an attempt to impede the termination of compromised targets by legitimate system processes. Variations of COLDSTEEL demonstrate functional differences, including support for Windows 10 absent in older versions. Deployment of COLDSTEEL is strongly implied to occur following exploitation of a public-facing application. Exploitation likely leverages vulnerabilities such as Log4j to achieve initial access. Subsequent takeover of the system by COLDSTEEL involves manipulation of system processes and potentially further malicious actions. Vulnerabilities within publicly accessible applications, such as Log4j, are exploited to gain control of vulnerable systems, enabling arbitrary code execution. This compromised state facilitates the deployment of the COLDSTEEL malware onto the affected system. Deployment methods may include downloading and executing the malware or utilizing existing backdoors/tunnels established during initial access.

## Pre-Conditions:

- The attacker possesses knowledge of the specific vulnerabilities exploited.
- A vulnerable public-facing application exists.
- The attacker has network access to the target system.

## Post-Conditions:

- What commands are being executed?
- Malware delivery: The URLs might lead to sites hosting malware that could be downloaded and executed.
- Data exfiltration: The attacker might be trying to steal data from the targeted systems.
- 223.34.198, 103.224.80.76
- Web server exploitation: Attempting to access or exploit vulnerabilities on web servers at the given IP addresses.
- What files are being accessed or modified?
- Domain names: bd82563c72e6f72adff76bd8c6940c6037516a2a89c5fd0c23b8af622f0e91939b486 e9db7faef192.95.36[.]61vpn2.smi1egate[.]comsvn1.smi1egate[.]comgiga.gnisoft[.]com
- URLs: hxxp://104.223.34.[.]198/111.php, hxxp://104.223.34.[.]198/1dll.php, hxxp://104.223.34.[.]198/syn.php
- What network connections are being established?
- To determine the consequences and traces, I need more context about the specific actions being performed.

### *MITRE Technique*

**ID:** T1190
**Name:** Exploit public-facing application

**Description:** Adversaries may attempt to exploit a weakness in an Internet-facing host or system to initially access a network. The weakness in the system can be a software bug, a temporary glitch, or a misconfiguration.
**More info:** https://attack.mitre.org/techniques/T1190/

## *Indicators*

- The file name is "newdev.dll".
- The path includes "AppData\Roaming\newdev.dll".

# Milestone 2

## Attack Step 2.1

==================================================
**Name:** Command and Scripting Interpreter: Windows Command Shell as used by the malware
**Description:** Command execution functionality is exhibited by various COLDSTEEL malware variants through distinct mechanisms. Certain variants directly utilize cmd.exe for command execution, potentially mimicking legitimate system processes. To circumvent security measures focused on cmd.exe executions, other variants employ obfuscation techniques. These techniques involve copying cmd.exe to the dllhost.exe process, leveraging the reduced scrutiny associated with dllhost.exe processes to facilitate covert command execution. A unique custom command, not present in the original Gh0st RAT source code, has been incorporated into the backdoor functionality. This custom command transmits information pertaining to active system sessions to the attacker's server, demonstrating the malware's capabilities beyond basic command execution. The precise technical implementation of command transmission to cmd.exe (e.g., through shellcode injection or API calls) remains subject to further analysis.

## Pre-Conditions:

- A functioning Windows Command Shell is available.
- The malware has successfully infected the target machine.
- The malware has access to the Windows Command Shell.
- A Windows operating system is present.

## Post-Conditions:

- Potential Data Exfiltration
- Modified Command Execution Logs
- System Information Compromise
- Screenshots and Cursor Position Data
- DLL Injection Traces in Process Memory
- Modified Registry Keys
- System Event Logs (service installation/modification, process creation)
- File Access Logs (modified or accessed files)
- File and Directory Access Changes
- Network Traffic Logs (TCP connections with C2 server)
- New Files Created (e.g., malware binaries, configuration files)
- Process Information Exposure
- Service Installation

### *MITRE Technique*

**ID:** T1059.003
**Name:** Command and scripting interpreter: windows command shell
**Description:** Adversaries may abuse the Windows command shell for execution. The Windows command shell (cmd) is the primary command prompt on Windows systems. The Windows command prompt can be used to control almost any aspect of a system, with various permission levels required for different subsets of commands. The command prompt can be invoked remotely via Remote Services such as SSH.

**More info:** https://attack.mitre.org/techniques/T1059/003

### *Indicators*

- Path: C:\Users\\AppData\Roaming\newdev.dll: [ file : name = ' newdev . dll ' A N D file : parent _ directory _ ref . path = ' C : \ \ Users \ \ < user > \ \ AppData \ \ Roaming ' ]

# Attack Step 2.2

=================================================
**Name:** System Services: Service Execution as used by the malware
**Description:** The COLDSTEEL malware family exhibits continuous development, with variations such as FBI20111024, MileStone 2016, and MileStone 2017, each potentially possessing distinct capabilities and targeting specific entities. Persistence within infected systems is achieved through the exploitation of Windows services, enabling the malware to execute even after system restarts and ensuring prolonged access for malicious actors. System information gathering is conducted by COLDSTEEL to acquire details regarding the infected machine, including operating system version, installed software, and network configurations. Communication with a Command & Control (C2) server is established via raw TCP connections, facilitating the transmission of instructions to COLDSTEEL and the receipt of gathered information from the compromised system. Exploitation of Windows services for malicious code execution is a key characteristic of COLDSTEEL. This process involves the creation of a new Windows service, often disguised with a legitimate-sounding name, followed by the injection of pre-loaded malicious code within its configuration files or as separate modules. Upon service initiation, the malicious code is executed, potentially evading detection by security software and users due to the elevated privileges associated with services. Potential impacts of COLDSTEEL infection include data theft, facilitated by the gathered system information which can identify valuable assets on the compromised machine, and remote control, enabled by C2 commands that may grant attackers full control over the infected system, allowing for the installation of additional malware, surveillance of user activity, or launching further attacks against other systems within the network. Mitigation strategies encompass regular security updates to address known vulnerabilities exploited by COLDSTEEL, the utilization of reputable anti-malware software with up-to-date definitions for detection and removal of malware, and the implementation of network monitoring tools to detect suspicious communication patterns indicative of C2 connections.

## Pre-Conditions:

- System privileges sufficient to create and execute services.
- The system is running a supported version of Windows.
- COLDSTEEL malware code is present on the system.
- A vulnerable system running a supported version of Windows.

## Post-Conditions:

- System instability
- Unusual file modifications or creations
- Modified system registry entries
- Potential for further malware infections
- Presence of malicious code in memory
- Loss of sensitive information
- Network traffic to C2 server IPs
- Data exfiltration

- Altered firewall rules
- Windows service installation
- Hidden processes running in the background
- New log files with suspicious activity
- Compromised system functionality

## *MITRE Technique*

**ID:** T1569.002
**Name:** System services: service execution
**Description:** Adversaries may abuse the Windows service control manager to execute malicious commands or payloads. The Windows service control manager (services.exe) is an interface to manage and manipulate services. The service control manager is accessible to users via GUI components as well as system utilities such as sc.exe and Net.
**More info:** https://attack.mitre.org/techniques/T1569/002

## *Indicators*

- Path: C:\Users\\AppData\Roaming\newdev.dll: [ file : name = ' newdev . dll ' A N D file : parent _ directory _ ref . path = ' C : \ \ Users \ \ < user > \ \ AppData \ \ Roaming ' ]
- IPv4: 192.95.36.61: [ ip v 4 - a d d r : v a l u e = ' 1 9 2 . 9 5 . 3 6 . 6 1 ' ]

# Milestone 3

## Attack Step 3.1

==================================================
**Name:** Create or Modify System Process: Windows Service as used by the malware
**Description:** A new Windows service is created by COLDSTEEL malware utilizing the Create or Modify System Process technique (T1543.003). This newly created service is configured to automatically initiate upon system reboots. The persistence of COLDSTEAL on the compromised system is ensured through the automatic startup configuration of the service, enabling continuous background operation even after system reboots.

## Pre-Conditions:

- An attacker possesses sufficient privileges to create or modify system processes.
- A Windows system with administrative privileges is available.
- The system has a valid Windows operating system installed.
- Network connectivity is required to communicate with the C2 server.
- The malware code for COLDSTEEL is accessible.

## Post-Conditions:

- Unusual process activity
- System instability
- Log files containing malicious activity indicators
- Modified system registry entries
- Potential for further malware infections
- Indicators of compromise (IOCs) in network traffic analysis
- Data exfiltration
- New files created in various directories
- Windows service installation
- Altered system configurations
- Network traffic logs showing communication with C2 server
- Compromised system functionality

### MITRE Technique

**ID:** T1543.003
**Name:** Create or modify system process: windows service
**Description:** Adversaries may create or modify Windows services to repeatedly execute malicious payloads as part of persistence. When Windows boots up, it starts programs or applications called services that perform background system functions. Windows service configuration information, including the file path to the service's executable or recovery programs/commands, is stored in the Windows Registry.
**More info:** https://attack.mitre.org/techniques/T1543/003

### Indicators

- file : name = 'newdev.dll'
- file : parent_directory_ref.path = 'C:\Users\\AppData\Roaming'

# Milestone 4

## Attack Step 4.1

===================================================
**Name:** Obfuscated Files or Information: Software Packing as used by the malware
**Description:** The MileStone2017 malware variants are packaged using Themida, a commercial packer renowned for its robust anti-disassembly and obfuscation capabilities. This packaging technique falls under the Defense Evasion tactic as it is designed to impede the analysis and reverse engineering of the malicious code. Themida achieves this through several mechanisms: The original source code undergoes rewriting and rearrangement, resulting in significant obfuscation that hinders comprehension of its functionality by analysts. Anti-debugging techniques are implemented within Themida to detect the attachment of debuggers to the process, thereby preventing analysis tools from effectively inspecting the malware's behavior. Resources embedded within the packed executable are encrypted, further obscuring their contents and hindering static analysis. The utilization of Themida by the MileStone2017 developers aims to significantly increase the difficulty for security researchers to analyze and comprehend the malware's operational mechanisms, ultimately enhancing its potential to evade detection and execute malicious activities undetected.

## Pre-Conditions:

- The malware utilizes software packing techniques.
- Access to the infected system.
- The malware is present in the system.
- Tools capable of analyzing packed files (e.g., debuggers, disassemblers).

## Post-Conditions:

- System instability
- Data corruption
- Log entries indicating unauthorized access or modifications
- Hidden files and folders
- Unusual network traffic to unknown IP addresses
- Loss of system control
- Modified system files
- New processes running with suspicious names and behavior
- Performance degradation
- Backdoors and malware payloads
- Increased security vulnerabilities
- Altered registry entries

### *MITRE Technique*

**ID:** T1027.002
**Name:** Obfuscated files or information: software packing
**Description:** Adversaries may perform software packing or virtual machine software protection to conceal their code. Software packing is a method of compressing or encrypting an executable. Packing an executable changes the file signature in an attempt to avoid signature-based detection. Most decompression techniques decompress the executable code in memory. Virtual machine software protection translates an executable's original code into a special format that only a special virtual

machine can run. A virtual machine is then called to run this code.
**More info:** https://attack.mitre.org/techniques/T1027/002

### *Indicators*

- file : name = ' newdev . dll'
- file : hashes.sha256 = '...' (Replace with actual SHA256 hash)
- file : hashes.md5 = '...' (Replace with actual MD5 hash)
- file : parent_directory_ref.path = 'C : \ \ Users \ \ < user > \ \ AppData \ \ Roaming'

## Attack Step 4.2

==================================================
**Name:** Modify Registry as used by the malware
**Description:** Registry modification is executed by COLDSTEEL malware through direct manipulation of registry keys. A description is appended to its service entry within the registry. This action is undertaken to obfuscate its malicious intent and potentially circumvent detection by security mechanisms reliant on conventional service descriptions.

## Pre-Conditions:

- The malware has successfully infected the target system.
- The malware has gained sufficient privileges to modify registry keys.
- A Windows operating system is present.

## Post-Conditions:

- System instability
- Service disruption
- Data loss or corruption
- New .dll file in %APPDATA%\newdev
- Presence of Themida packer remnants
- Performance degradation
- Modified registry entries for service creation and hiding
- Remote access by attacker
- Log files containing suspicious activity
- Altered system configurations
- Network connections to command and control servers

### *MITRE Technique*

**ID:** T1112
**Name:** Modify registry
**Description:** Adversaries may interact with the Windows Registry to hide configuration information within Registry keys, remove information as part of cleaning up, or as part of other techniques to aid in persistence and execution.
**More info:** https://attack.mitre.org/techniques/T1112/

### *Indicators*

- Path: C:\Users\\AppData\Roaming\newdev.dll: [ file : name = ' newdev . dll ' A N D file : parent _ directory _ ref . path = ' C : \ \ Users \ \ < user > \ \ AppData \ \ Roaming ' ]

# Attack Step 4.3

==================================================
**Name:** Indicator Removal: File Deletion as used by the malware
**Description:** Indicator removal through file deletion is executed by COLDSTEEL variants, such as MileStone2017, via the following processes: A malicious code injection into the Windows operating system is facilitated by the creation of a new service often designated with a legitimate-sounding name to circumvent detection. The core malware functionality resides within a DLL file, which is subsequently loaded into the newly created service process. Upon service initiation, the ServiceMain function within the injected DLL is executed. Within the ServiceMain function, COLDSTEEL likely possesses code designed to scan for specific indicators of compromise (IOCs). These IOCs may encompass log files documenting suspicious activity, temporary files generated by security software or antivirus programs, and registry entries associated with security tools. Identified IOC files are subsequently deleted from the infected machine utilizing the Windows API (Application Programming Interface). This action aims to conceal the malware's presence from security analysts and investigators while removing evidence of past malicious activities. Variations in the specific IOCs targeted and deletion methods may exist across different COLDSTEEL variants.

## Pre-Conditions:

- The malware has successfully executed.
- Operating System
- Execution permissions for the malware process
- Access to the file system
- The files to be deleted exist on the system.


## Post-Conditions:

- Deleted files (1.bat, syn.exe, 1.dll)
- Network traffic logs showing communication with the specified IP addresses and domains
- PowerShell execution logs
- Backdoors or other malicious tools installed on the system
- bat, syn.exe, 1.dll)
- System event logs indicating process creation and deletion
- Modified system configuration files

### 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

- file : name = 'newdev.dll'
- file : parent_directory_ref.path = 'C:\Users\\AppData\Roaming'

# Attack Step 4.4

====================================================
**Name:** Access Token Manipulation: Create Process with Token as used by the malware
**Description:** Malware samples exhibiting shared characteristics are analyzed for their potential threat level. These characteristics include digital signatures obtained through certificate theft to circumvent security filters and utilization of command and control (C2) infrastructure comprising designated domains and IP addresses. The implementation of the ObRegisterCallbacks technique within a Windows driver framework enables malware to intercept system calls pertaining to process creation. This suggests a focus on manipulating process creation mechanisms, potentially for stealth or privilege escalation purposes. The "Create Process with Token" technique allows malware to generate new processes inheriting a specific access token. This function deviates from the standard inheritance of a parent process's access token, granting malware the capability to execute as different users, including elevated privileges such as SYSTEM. This manipulation can facilitate bypassing security measures reliant on user permissions. The "Milestone2016" variant demonstrates this technique by creating processes under the identity of the "ANONYMOUS" user. Access token manipulation techniques pose a significant threat due to their ability to execute code with elevated privileges, enabling complete system control for attackers. Furthermore, these techniques facilitate lateral movement within networks and evasion of detection by operating under trusted user or service identities. Fortinet's FortiEDR solution demonstrates effectiveness in detecting and mitigating these threats through behavioral analysis identifying suspicious process creation patterns and API utilization such as CreateProcessWithTokenW. Threat intelligence plays a crucial role in recognizing malicious domains, IP addresses, and malware signatures associated with these campaigns. Real-time response capabilities enable swift blocking of identified threats through process and network connection interruptions.

## Pre-Conditions:

- Network connectivity may be required for communication with a command and control server.
- The malware has obtained elevated privileges.
- A vulnerable operating system.
- A process is running on the target system.
- The malware code is present and loaded onto the target system.

## Post-Conditions:

- System compromise
- Network traffic to C2 server
- Registry modifications
- New files created with malicious content
- Modified system files
- Log entries indicating suspicious activity
- Firewall bypass attempts documented in logs
- Service persistence
- Data exfiltration
- Unusual network connections established
- Windows service installation
- Network communication redirection

### *MITRE Technique*

**ID:** T1134.002
**Name:** Access token manipulation: create process with token
**Description:** Adversaries may create a new process with an existing token to escalate privileges and bypass access controls. Processes can be created with the token and resulting security context of another user using features such as CreateProcessWithTokenW and runas.
**More info:** https://attack.mitre.org/techniques/T1134/002

## *Indicators*

- Path: C:\Users\\AppData\Roaming\newdev.dll: [ file : name = ' newdev . dll ' A N D file : parent _ directory _ ref . path = ' C : \ \ Users \ \ < user > \ \ App Data \ \ Roaming ' ]
- IPv4: 192.95.36.61: [ ip v 4 - a d d r : v a l u e = ' 1 9 2 . 9 5 . 3 6 . 6 1 ' ]

# Milestone 5

## Attack Step 5.1

=================================================
**Name:** System Information Discovery as used by the malware
**Description:** COLDSTEEL malware variants exhibit variations in command structures and functionalities compared to the original Gh0st RAT source code. Certain variants utilize cmd.exe processes mimicking dllhost.exe for security evasion purposes. A distinctive command facilitates the transmission of active system session information to a remote server, potentially for log manipulation or concealment of the initial compromise account. Functional discrepancies exist across variants, including support levels for Windows 10 and potential memory leaks identified in specific variants such as FBI20111024, MileStone 2016, and MileStone 2017. System information discovery is employed by the malware to gather comprehensive data from infected machines. This encompasses operating system details (version, build number, architecture), hardware specifications (CPU details, RAM capacity, available disk space), network configuration (IP address, MAC address, active connections), and user information (logged-in accounts, permissions). The purpose of this extensive system information collection is multifaceted. It enables the assessment of target machine vulnerabilities, facilitates the customization of attacks based on specific configurations and installed software, and contributes to the establishment of a detailed profile for future malicious activities such as data theft or remote control.

### Pre-Conditions:

- A network connection exists between the infected system and the attacker's server.
- The malware is running on a target system.
- The backdoor functionality is active within the malware.

### Post-Conditions:

- System compromise
- Hidden files or folders
- Unusual log entries in event viewer
- Registry modifications
- Backdoor executable file present
- Modified system files
- New user accounts created
- Data exfiltration
- Potential for further malware installation
- Remote control of the system
- Network traffic to malicious IP addresses

### *MITRE Technique*

**ID:** T1082
**Name:** System information discovery
**Description:** An adversary may attempt to get detailed information about the operating system and hardware, including version, patches, hotfixes, service packs, and architecture. Adversaries may use the information from System Information Discovery during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific

actions.
**More info:** https://attack.mitre.org/techniques/T1082/

### *Indicators*

- file : name = ' newdev . dll'
- file : parent_directory_ref.path = ' C : \ \ Users \ \ < user > \ \ AppData \ \ Roaming '
- IPv4: 192.95.36.61: [ ip v 4 - a d d r : v a l u e = ' 1 9 2 . 9 5 . 3 6 . 6 1 ' ]
- IPv4: 103.224.80.76: [ ip v 4 - a d d r : v a l u e = ' 1 0 3 . 2 2 4 . 8 0 . 7 6 ' ]
- IPv4: 138.128.98.106: [ ip v 4 - a d d r : v a l u e = ' 1 3 8 . 1 2 8 . 9 8 . 1 0 6 ' ]
- IPv4: 1.9.5.38: [ ip v 4 - a d d r : v a l u e = ' 1 . 9 . 5 . 3 8 ' ]

# Attack Step 5.2

==================================================
**Name:** File and Directory Discovery as used by the malware
**Description:** The provided textual data suggests a potential scenario of file and directory discovery being conducted by malicious software. Repeated occurrences of "hxxp://" followed by IP addresses and paths such as "/111.php", "/1dll.php", and "/syn.php" indicate the likelihood of malware initiating requests to these locations on the compromised system. The presence of ".php" extensions in the file names suggests that these files are likely scripts, frequently employed for dynamic web content generation. Malware may exploit such scripts to execute commands and interact with the file system. The utilization of varying filenames like "/111.php", "/1dll.php", etc., could be interpreted as attempts to test different file names within directories. These requests might aim to access system files or specific directories known to contain sensitive information. The observed actions, which involve malware probing the file system by sending HTTP requests to various paths and attempting to read content from PHP files, enable the following: identification of existing files and directories; location of sensitive data such as user credentials, configuration information, or other valuable data; and potential creation of new PHP files containing malicious code for future remote access or control. It is important to note that this analysis is contingent upon limited information. A comprehensive understanding of the malware's intentions and actions would necessitate a detailed forensic investigation encompassing system logs, network traffic, and the contents of those PHP files.

## Pre-Conditions:

- A functioning operating system with a file system.
- The malware possesses the necessary permissions to read files and directories.

## Post-Conditions:

### *MITRE Technique*

**ID:** T1083
**Name:** File and directory discovery
**Description:** Adversaries may enumerate files and directories or may search in specific locations of a host or network share for certain information within a file system. Adversaries may use the information from File and Directory Discovery during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.
**More info:** https://attack.mitre.org/techniques/T1083/

### *Indicators*

- file: name = 'newdev.dll'
- file: parent_directory_ref.path = 'C:\Users\\AppData\Roaming'

# Attack Step 5.3

==================================================
**Name:** Process Discovery as used by the malware
**Description:** Process discovery functionalities within the malware FBI20111024 are purportedly executed through the utilization of system calls and potentially Windows API functions. System calls, which constitute low-level requests directed to the operating system kernel by programs, are likely employed by FBI20111024. Functions such as CreateToolhelp32Snapshot and Process32Next are speculated to be utilized for the enumeration of running processes and the retrieval of associated process information, including process ID, name, and path. Furthermore, the malware may leverage higher-level Windows API functions specialized in process management. Examples include EnumProcesses, which retrieves a comprehensive list of active processes, and GetProcessInformation, which facilitates the acquisition of detailed information pertaining to a specific process identified by its ID. The precise methodologies implemented by FBI20111024 necessitate direct code analysis for definitive elucidation. Nevertheless, these system calls and API functions represent commonly observed tools employed by malware engaged in process discovery activities.

## Pre-Conditions:

- A running instance of the COLDSTEEL malware.
- The malware is running.
- Network connectivity (for potential C2 communication).
- The target system has processes executing.

## Post-Conditions:

- System compromise
- Registry modifications related to persistence
- Altered process lists or memory dumps
- New files created by the rootkit
- Modified system files (e.g., tonsiproxy.sys)
- Log entries indicating suspicious activity
- Network reconnaissance
- Data exfiltration
- Unusual network traffic to command and control servers
- Persistence on the system
- Evidence of data transfer (e.g., log files, network captures)
- Firewall rules changes

### *MITRE Technique*

**ID:** T1057
**Name:** Process discovery
**Description:** Adversaries may attempt to get information about running processes on a system. Information obtained could be used to gain an understanding of common software/applications running on systems within the network. Administrator or otherwise elevated access may provide better process details. Adversaries may use the information from Process Discovery during automated discovery to

shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.
**More info:** https://attack.mitre.org/techniques/T1057/

## *Indicators*

- The file name is "newdev.dll".
- The file path is "C:\Users\\AppData\Roaming\newdev.dll".
- The malware communicates with IPv4 addresses: 192.95.36.61, 103.224.80.76, 138.128.98.106, and 1.9.5.38.

# Milestone 6

## Attack Step 6.1

```
==================================================
```
**Name:** Non-Application Layer Protocol as used by the malware
**Description:** TCP sockets are employed by the COLDSTEEL malware for establishing connections with its designated Command and Control (C2) server. A non-standard message format is utilized by the malware, deviating from established protocols such as HTTP or FTP. This characteristic renders communication analysis and detection by conventional security tools more challenging.

## Pre-Conditions:

- The malware is running.
- Network traffic analysis capabilities are available.
- A system with the capability to analyze network packets.
- Access to a network monitoring tool capable of capturing and analyzing raw TCP traffic.
- Connectivity to the network where the malware is operating.

## Post-Conditions:

- System compromise
- Presence of malicious files
- Altered registry settings
- DNS requests to C2 domains
- Network communication with C2 server
- Modified system files
- Log entries indicating suspicious activity
- Service persistence
- Data exfiltration
- Windows service installation

### MITRE Technique

**ID:** T1095
**Name:** Non-application layer protocol
**Description:** Adversaries may use an OSI non-application layer protocol for communication between host and C2 server or among infected hosts within a network. The list of possible protocols is extensive. Specific examples include use of network layer protocols, such as the Internet Control Message Protocol (ICMP), transport layer protocols, such as the User Datagram Protocol (UDP), session layer protocols, such as Socket Secure (SOCKS), as well as redirected/tunneled protocols, such as Serial over LAN (SOL).
**More info:** https://attack.mitre.org/techniques/T1095/

### Indicators

- IPv4: 192.95.36.61: is used by the malware.
- IPv4: 103.224.80.76: is used by the malware.
- IPv4: 138.128.98.106: is used by the malware.
- IPv4: 1.9.5.38: is used by the malware.