Enhanced Attack Report

COLDSTEEL

Generated on 2025-03-27

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 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

Pre-Conditions:

- The malware is capable of exploiting the vulnerability.
- Internet connectivity to reach the public-facing application.
- A public-facing application exists.
- The public-facing application has a known vulnerability.

Post-Conditions:

Attack Step 1.1

==
Name: Exploit Public-Facing Application as used by the malware
Description: COLDSTEEL malware is characterized by its utilization of advanced techniques for stealth and persistence within compromised systems. The malware exhibits variations, including FBI20111024, MileStone 2016, and MileStone 2017, each potentially possessing subtle functional differences. Early versions of COLDSTEEL demonstrated limited support for Windows 10 operating systems and exhibited memory leaks when executed on such platforms. Subsequent variants likely addressed these limitations. The malware heavily relies on the ObRegisterCallbacks technique to intercept system calls pertaining to process and thread handles. Upon detection of a user-mode application attempting to access or duplicate a process or thread handle, COLDSTEEL consults its global process database. If a match is identified, the PROCESS_TERMINATE permission is removed from the DesiredAccess parameter, effectively preventing legitimate tools from terminating targeted processes. This manipulation grants COLDSTEEL enhanced control over compromised systems. Deployment of COLDSTEEL is believed to follow exploitation of the Log4j vulnerability. This suggests that attackers leverage known exploits to gain initial access, subsequently employing COLDSTEEL for further malicious activities. Public-facing applications, potentially vulnerable to Log4j exploits, are targeted by COLDSTEEL as the primary means of initial compromise.

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

Pre-Conditions:

- The malware possesses the capability to execute commands within the Windows Command Shell.
- A compromised Windows machine with an active internet connection.
- The malware has successfully infected the target machine.
- A Windows operating system is present.
- The malware possesses the necessary code to create and execute a service.
- Network connectivity is available for communication with the C2 server.
- A Windows operating system is present.
- The malware has successfully infiltrated the target system.

Post-Conditions:

- Registry entries for malicious processes and services
- Persistence on the system
- Logs containing suspicious activity
- Service hijacking
- Deleted or modified original files
- Altered file timestamps
- Remote control of infected machine
- Network connections to command and control servers
- Modified system services
- Data exfiltration
- System compromise
- New files with malicious code
- Data theft
- Modified system registry entries
- Compromised system functionality
- Remote access by attacker
- System instability
- New service running with suspicious name and description
- Altered system files
- Unusual network traffic to command-and-control servers
- Presence of malicious DLL file
- Data exfiltration logs
- Log files containing suspicious activity
- Backdoors or hidden executables

Attack Step 2.1

==
Name: Command and Scripting Interpreter: Windows Command Shell as used by the malware
Description: COLDSTEEL malware exploits the Windows Command Shell (cmd.exe) to execute arbitrary commands. This functionality enables attackers to retrieve sensitive data, install additional malware, modify system settings, and monitor user activity. To evade detection, certain COLDSTEEL variants employ obfuscation techniques such as copying cmd.exe into less scrutinized processes like dllhost.exe. Beyond basic cmd.exe execution, some COLDSTEEL implementations incorporate

commands for gathering information about running processes, logged-in users, and system configurations. This intelligence can assist attackers in planning subsequent attacks or tailoring their strategies based on the victim's environment. The combination of arbitrary command execution and obfuscation techniques presents a significant security risk, granting attackers broad control over infected systems and enabling sophisticated attacks with limited visibility.

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 provided text constitutes an excerpt from a report detailing the analysis of malware, potentially belonging to the COLDSTEEL family. Analysis conducted on distinct variants of the COLDSTEEL malware family has revealed variations in functionality across different iterations. Designations such as "FBI20111024," "MileStone 2016," and "MileStone 2017" indicate the evolution of these variants over time. Functionality discrepancies observed between variants include potential compatibility issues with Windows 10 operating systems in earlier versions, such as "FBI20111024," while later variants are presumed to address this limitation. Additionally, early versions may exhibit memory management inefficiencies, characterized by a "small memory leak." Obfuscation techniques employed within the malware code, exemplified by the suspicious code snippet provided, suggest an attempt to conceal the malware's true nature. The analysis explicitly states that COLDSTEEL malware utilizes "System Services: Service Execution" for malicious purposes. This implies the exploitation of legitimate Windows services for illicit activities. This exploitation may involve hijacking existing system services or creating new ones under deceptive names, followed by the injection of malicious code into the compromised service. Consequently, the malware achieves persistence by leveraging the automatic startup functionality of services upon Windows boot. The utilization of services often grants elevated privileges and reduces the likelihood of detection by security software, contributing to a stealthy attack vector.

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 ' AND file : parent _ directory _ ref . path = ' C : \ \ Users \ \ < user > \ \ App Data \ \ Roaming ']
- IPv4: 192.95.36.61: [ipv4-addr:value = '192.95.36.61']

Milestone 3

Pre-Conditions:

- A Windows operating system.
- Access to the file system for creating and modifying service files.
- The malware has access to the necessary files and directories.
- The system is running a supported version of Windows.
- A user account with sufficient privileges exists.
- Network connectivity to communicate with the C2 server.

Post-Conditions:

- Loss of system control
- Suspicious DLL file present in user's AppData folder
- Potential for further malware infections
- Modified system registry entries
- Unusual activity logs in event viewer
- Compromised system functionality
- Altered system files
- New Windows service installed
- Network connections to C2 server(s) logged
- Data exfiltration
- Data files transferred from compromised system

Attack Step 3.1

==
Name: Create or Modify System Process: Windows Service as used by the malware
Description: A new service entry is created directly within the registry by the COLDSTEEL malware. This entry specifies the malicious service's name, description, executable path (pointing to its own packed DLL), and startup parameters. The created service entry is utilized to initiate the execution of its own DLL as a Windows service. Upon registration, the operating system automatically commences the service during bootup, ensuring persistence even after system reboots.

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

Pre-Conditions:

- The malware utilizes software packing techniques.
- Tools capable of unpacking packed executables (e.g., Themida).
- The ability to analyze binary files.
- Access to a system infected with the COLDSTEEL malware.
- The malware has the necessary permissions to modify registry keys.
- A Windows operating system is present.
- The malware is running.
- The target system has a registry.
- The malware is running on a compromised system.
- The target files are accessible by the malware.
- A process is running.
- A running operating system with a user account logged in.
- The malware code must be present and executable.
- The malware has access to the current process's token.

Post-Conditions:

- XOR-encrypted communication logs
- Data theft
- Modified system registry entries
- Compromised system functionality
- Remote access by attacker
- System instability
- Themida packer artifacts
- Log4Shell exploit remnants
- Altered system configurations
- Unusual process activity in event logs
- Backdoor executables
- Hidden files and folders
- Network traffic to C2 servers
- New user accounts created
- Installation of additional malware
- Modified registry keys with timestamps indicating changes.
- Deleted files from the infected machine.
- Altered process list with potentially obfuscated entries.
- Modified registry keys with added descriptions for the service.
- Network connections to command and control servers (if any).
- New processes created for service removal and registry modification.
- Obfuscated communication logs.
- Deleted file remnants in system logs and event logs.
- Hidden registry keys from users using Microsoft's Registry Editor.
- Newdev.dll file in AppData\Roaming directory.
- Presence of XOR-encrypted and LZMA-compressed payload.
- Data exfiltration potential.
- Network connections to command-and-control servers.
- Difficulty in detecting and removing the malware due to its sophisticated techniques.

- Altered registry entries related to newdev.dll and ColdSteel components.
- Modified system services.
- Log files containing suspicious activity related to newdev.dll and ColdSteel.
- System instability and performance degradation.
- Traces of process manipulation and file system modifications.
- Compromised system with persistent backdoor access.
- Network traffic to malicious IP addresses listed in the provided context.
- Evidence of data transfer (e.g., modified log files, empty folders).
- Compromised systems with installed malware.
- Logs indicating suspicious activity, such as process creation, file access, and network connections.
- Data exfiltration from compromised systems.
- Potential for further attacks and lateral movement within the network.
- Modified system registry entries.
- Altered firewall rules or security settings.

Attack Step 4.1

==

Name: Obfuscated Files or Information: Software Packing as used by the malware

Description: Software packing, specifically utilizing Themida, is employed by malware variants classified as "Milestone2017" to obfuscate their inherent functionality. This technique involves encasing compiled malware code within a protective layer introduced by Themida during the software packing process. Themida modifies the execution flow of the original code, incorporating intricate instructions that facilitate the unpacking and execution of the malware upon file runtime. Consequently, the analysis of these packed malware variants is significantly impeded due to the convoluted execution path imposed by Themida. This obfuscation strategy enhances the evasion capabilities of "Milestone2017" variants, enabling them to circumvent rudimentary signature-based detection mechanisms employed by antivirus software. Furthermore, the unpacked malware frequently exhibits dynamic behavior, posing challenges for the development of static analysis tools designed for effective detection.

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)

Attack Step 4.2

==

Name: Modify Registry as used by the malware

Description: The "Modify Registry" action (T1112) is executed by COLDSTEEL through direct manipulation of registry keys. This involves the addition of a description to its own service, denoted as "msupdate2." Registry entries pertaining to this service are either modified or newly created, potentially incorporating deceptive descriptions to obfuscate its actual function.

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 ' AND 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 is executed by COLDSTEEL malware variants, specifically the "MileStone2017" variant, through file deletion techniques. Upon system startup, a custom DLL file, disguised as a legitimate system process, is loaded into memory by a Windows service created by the malware. The ServiceMain export within the DLL executes code responsible for indicator removal. Files associated with specific network communication IDs used by the malware, such as "MileStone2017," are targeted for deletion. This includes original DLL files, configuration files, and temporary files utilized by the malware. Additionally, logs and traces of malware activity are potentially deleted to hinder tracking efforts. The objective of this file deletion tactic is to maintain stealth and evade detection by security software and analysts.

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' AND file : parent_directory_ref.path = 'C : \ \ Users \ \ < user > \ \ AppData \ \ Roaming'

Attack Step 4.4

==

Name: Access Token Manipulation: Create Process with Token as used by the malware
Description: Technical analysis indicates potential collaboration among multiple malware groups, evidenced by the sharing of tools and infrastructure such as stolen certificates and command-and-control servers. Samples were observed to be signed hours after compilation, suggesting an attempt to evade detection by utilizing legitimate code tools prior to flagging by security software. The analysis highlights the utilization of the ObRegisterCallbacks function by malware, enabling interception of system calls pertaining to process and thread creation. Process termination permissions (PROCESS_TERMINATE) are manipulated for specific processes, indicating a desire to maintain control over designated targets and prevent their termination by security measures or user actions. Malware creates processes with elevated tokens, such as "ANONYMOUS," potentially executing code with heightened privileges beyond its original permissions. This facilitates further malicious activities, including access to sensitive data or installation of additional malware. Fortinet Solutions FortiEDR is determined to effectively detect and block these threats without requiring specific configuration modifications. This implies the presence of pre-built rules and signatures within the security solution capable of recognizing the aforementioned malicious behaviors.

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

Pre-Conditions:

- The system has active network connections.
- A compromised system with an active internet connection.
- The target system is running Windows.
- The malware is running on a system.
- The malware possesses the necessary permissions to access files and directories.
- The malware has executed successfully.
- Operating System with a file system.
- Malware code capable of performing file and directory operations.
- The target system has processes running.
- Network connectivity (for communication with C2 server, if applicable).
- A running instance of the COLDSTEEL malware.
- The malware is running on a target system.

Post-Conditions:

- System instability or crashes
- Unusual system log entries
- Hidden files and directories
- New processes running with suspicious names
- Compromised system functionality
- Remote access by attacker
- Modified registry entries
- Network connections to command-and-control servers
- Data theft or manipulation
- Modified file timestamps
- Persistence of malicious code
- Presence of malicious DLLs or executables
- Encrypted data remnants

Attack Step 5.1

==
Name: System Information Discovery as used by the malware
Description: System information is acquired by COLDSTEEL malware variants through various methods. Native Windows API calls are employed to retrieve system-wide data such as processor architecture, available memory, and operating system version via functions like GetComputerNameExW() and SystemInfo(). User account information and computer names in ASCII format are obtained using GetUserNameA()/GetComputerNameA(). Windows Management Instrumentation (WMI) is leveraged by COLDSTEEL to query comprehensive system configurations, installed software, network interfaces, and hardware details through queries such as SELECT * FROM Win32_OperatingSystem and SELECT * FROM Win32_NetworkAdapterConfiguration. Specific registry keys are accessed to extract information pertaining to installed applications, user accounts, recent file activity, and system settings. Running processes on the system are enumerated by the malware, revealing active software and potential security vulnerabilities. Network information is gathered using commands such as ipconfig /all for network card details and tracert/traceroute to map network routes.

Certain variants may incorporate custom modules or scripts for targeted data collection tasks not covered by native functions. The collected system information provides attackers with a detailed understanding of the victim's system and network infrastructure, enabling targeted attacks, credential theft, lateral movement, and data exfiltration.

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: [ipv4-addr:value = '192.95.36.61']
- IPv4: 103.224.80.76: [ipv4-addr:value = '103.224.80.76']
- IPv4: 138.128.98.106: [ipv4-addr:value = '138.128.98.106']
- IPv4: 1.9.5.38: [ipv4-addr:value = '1.9.5.38']

Attack Step 5.2

==
Name: File and Directory Discovery as used by the malware
Description: The provided textual evidence strongly suggests the execution of file and directory discovery operations by an unauthorized actor, potentially indicative of malicious intent (e.g., malware). This inference is supported by several factors: - The presence of URLs structured as 104.223.34[.]198/111.php, 104.223.34[.]198/1dll.php, and 104.223.34[.]198/syn.php suggests attempts to access files or scripts within a specific directory structure on a web server. Such behavior is frequently observed in malware seeking sensitive information (e.g., configuration files, credentials) or executing malicious code designed for actions such as data exfiltration, system compromise, or network propagation. - The IP addresses 104.223.34[.]198 and 103.224.80[.]76 are associated with the web server hosting the aforementioned suspicious files, indicating a targeted attack rather than random browsing activity. - The utilization of ".php" extensions suggests that these files are likely PHP scripts, commonly employed on web servers for dynamic content generation and execution. File and directory discovery operations are frequently performed by malware utilizing various techniques: - Operating system APIs: Malicious code may leverage libraries and functions provided by the operating system (e.g., opendir(), readdir() in PHP) to enumerate files and directories within specific paths. - Shell commands: Malware may execute shell commands such as "ls" or "dir" to retrieve directory listings and file information. - Network protocols: Certain malware types utilize network protocols (e.g., FTP or SMB) to scan for files and directories on remote systems.

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 techniques employed by FBI20111024 variants are primarily executed through the utilization of Windows API system calls. These calls encompass functions such as CreateToolhelp32Snapshot for capturing process snapshots and Process32First/Process32Next for iterating through captured processes. Furthermore, GetModuleFileNameEx is leveraged to retrieve executable file paths associated with each identified process. Advanced variants may also employ direct memory reading techniques to extract process information, contingent upon precise knowledge of memory layouts. The significance of process discovery in malware operations lies in its ability to facilitate the identification of vulnerable processes and those handling sensitive data. This information enables malware to target specific processes effectively. Additionally, understanding running processes allows malware to mimic legitimate activity, thereby evading detection mechanisms. Process discovery can also be instrumental in code injection techniques, enabling malware to insert malicious code into legitimate processes for system compromise. Mitigating the risks associated with process discovery necessitates a multi-faceted approach. Regular software updates are crucial to address known vulnerabilities in operating systems and applications. The implementation of reputable anti-malware solutions is essential for proactive threat detection and removal. Exercise caution when downloading files from untrusted sources and avoid clicking on suspicious links. Continuous monitoring of system activity, including unusual program launches or elevated CPU usage, can provide early indicators of potential compromise.

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

Pre-Conditions:

- A system infected with the COLDSTEEL malware is present.

Post-Conditions:

- Unusual network traffic to C2 server IPs
- Loss of sensitive information
- Potential for further malware infections
- Compromised system functionality
- New files with malicious code
- System instability
- Persistence mechanisms (e.g., scheduled tasks, services)
- Hidden artifacts within system files
- Log entries indicating suspicious activity
- Encrypted data transfers
- Data exfiltration
- Modified system configurations
- Modified registry keys

Attack Step 6.1

==
Name: Non-Application Layer Protocol as used by the malware
Description: TCP sockets are utilized by the malware for communication with its command and control (C2) server. A custom message format is employed for these communications, deviating from standard protocols such as HTTP or SMTP.

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.