

Table of contents

Overview

● Description	4
● Confidence	4
● Content	5

Entities

● Attack-Pattern	6
● Indicator	18
● Intrusion-Set	20
● Malware	21
● indicates	22
● uses	27
● based-on	30



Observables

● StixFile	31
------------	----



External References

● External References	32
-----------------------	----

Overview

Description

This report provides an in-depth analysis of a newly discovered information stealer named Kematian-Stealer, actively developed on GitHub and distributed as open-source software. The malware employs various techniques to collect sensitive data from compromised systems, evade detection, and maintain persistence, posing a significant threat to individuals and organizations.

Confidence

This value represents the confidence in the correctness of the data contained within this report.

100 / 100

Content

N/A

Attack-Pattern

Name

T1564.001

ID

T1564.001

Description

Adversaries may set files and directories to be hidden to evade detection mechanisms. To prevent normal users from accidentally changing special files on a system, most operating systems have the concept of a 'hidden' file. These files don't show up when a user browses the file system with a GUI or when using normal commands on the command line. Users must explicitly ask to show the hidden files either via a series of Graphical User Interface (GUI) prompts or with command line switches (`dir /a`` for Windows and `ls -a`` for Linux and macOS). On Linux and Mac, users can mark specific files as hidden simply by putting a "." as the first character in the file or folder name (Citation: Sofacy Komplex Trojan) (Citation: Antiquated Mac Malware). Files and folders that start with a period, ".", are by default hidden from being viewed in the Finder application and standard command-line utilities like "ls". Users must specifically change settings to have these files viewable. Files on macOS can also be marked with the UF_HIDDEN flag which prevents them from being seen in Finder.app, but still allows them to be seen in Terminal.app (Citation: WireLurker). On Windows, users can mark specific files as hidden by using the attrib.exe binary. Many applications create these hidden files and folders to store information so that it doesn't clutter up the user's workspace. For example, SSH utilities create a .ssh folder that's hidden and contains the user's known hosts and keys. Adversaries can use this to their advantage to hide files and folders anywhere on the system and evading a typical user or system analysis that does not incorporate investigation of hidden files.

Name

T1053

ID

T1053

Description

Adversaries may abuse task scheduling functionality to facilitate initial or recurring execution of malicious code. Utilities exist within all major operating systems to schedule programs or scripts to be executed at a specified date and time. A task can also be scheduled on a remote system, provided the proper authentication is met (ex: RPC and file and printer sharing in Windows environments). Scheduling a task on a remote system typically may require being a member of an admin or otherwise privileged group on the remote system.(Citation: TechNet Task Scheduler Security) Adversaries may use task scheduling to execute programs at system startup or on a scheduled basis for persistence. These mechanisms can also be abused to run a process under the context of a specified account (such as one with elevated permissions/privileges). Similar to [System Binary Proxy Execution](<https://attack.mitre.org/techniques/T1218>), adversaries have also abused task scheduling to potentially mask one-time execution under a trusted system process. (Citation: ProofPoint Serpent)

Name

T1048

ID

T1048

Description

Adversaries may steal data by exfiltrating it over a different protocol than that of the existing command and control channel. The data may also be sent to an alternate network location from the main command and control server. Alternate protocols include FTP, SMTP, HTTP/S, DNS, SMB, or any other network protocol not being used as the main

command and control channel. Adversaries may also opt to encrypt and/or obfuscate these alternate channels. [Exfiltration Over Alternative Protocol](<https://attack.mitre.org/techniques/T1048>) can be done using various common operating system utilities such as [Net](<https://attack.mitre.org/software/S0039>)/SMB or FTP.(Citation: Palo Alto OilRig Oct 2016) On macOS and Linux `curl` may be used to invoke protocols such as HTTP/S or FTP/S to exfiltrate data from a system.(Citation: 20 macOS Common Tools and Techniques) Many IaaS and SaaS platforms (such as Microsoft Exchange, Microsoft SharePoint, GitHub, and AWS S3) support the direct download of files, emails, source code, and other sensitive information via the web console or [Cloud API](<https://attack.mitre.org/techniques/T1059/009>).

Name

T1566

ID

T1566

Description

Adversaries may send phishing messages to gain access to victim systems. All forms of phishing are electronically delivered social engineering. Phishing can be targeted, known as spearphishing. In spearphishing, a specific individual, company, or industry will be targeted by the adversary. More generally, adversaries can conduct non-targeted phishing, such as in mass malware spam campaigns. Adversaries may send victims emails containing malicious attachments or links, typically to execute malicious code on victim systems. Phishing may also be conducted via third-party services, like social media platforms. Phishing may also involve social engineering techniques, such as posing as a trusted source, as well as evasive techniques such as removing or manipulating emails or metadata/headers from compromised accounts being abused to send messages (e.g., [Email Hiding Rules](<https://attack.mitre.org/techniques/T1564/008>)).(Citation: Microsoft OAuth Spam 2022)(Citation: Palo Alto Unit 42 VBA Infostealer 2014) Another way to accomplish this is by forging or spoofing(Citation: Proofpoint-spoof) the identity of the sender which can be used to fool both the human recipient as well as automated security tools.(Citation: cyberproof-double-bounce) Victims may also receive phishing messages that instruct them to call a phone number where they are directed to visit a malicious URL, download malware,(Citation: sygnia Luna Month)(Citation: CISA Remote Monitoring and Management Software) or install adversary-accessible remote management tools onto

their computer (i.e., [User Execution](https://attack.mitre.org/techniques/T1204)).(Citation: Unit42 Luna Moth)

Name

T1204.002

ID

T1204.002

Description

An adversary may rely upon a user opening a malicious file in order to gain execution. Users may be subjected to social engineering to get them to open a file that will lead to code execution. This user action will typically be observed as follow-on behavior from [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001). Adversaries may use several types of files that require a user to execute them, including .doc, .pdf, .xls, .rtf, .scr, .exe, .lnk, .pif, and .cpl. Adversaries may employ various forms of [Masquerading](https://attack.mitre.org/techniques/T1036) and [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to increase the likelihood that a user will open and successfully execute a malicious file. These methods may include using a familiar naming convention and/or password protecting the file and supplying instructions to a user on how to open it.(Citation: Password Protected Word Docs) While [Malicious File](https://attack.mitre.org/techniques/T1204/002) frequently occurs shortly after Initial Access it may occur at other phases of an intrusion, such as when an adversary places a file in a shared directory or on a user's desktop hoping that a user will click on it. This activity may also be seen shortly after [Internal Spearphishing](https://attack.mitre.org/techniques/T1534).

Name

T1566.001

ID

T1566.001

Description

Adversaries may send spearphishing emails with a malicious attachment in an attempt to gain access to victim systems. Spearphishing attachment is a specific variant of spearphishing. Spearphishing attachment is different from other forms of spearphishing in that it employs the use of malware attached to an email. All forms of spearphishing are electronically delivered social engineering targeted at a specific individual, company, or industry. In this scenario, adversaries attach a file to the spearphishing email and usually rely upon [User Execution](<https://attack.mitre.org/techniques/T1204>) to gain execution. (Citation: Unit 42 DarkHydrus July 2018) Spearphishing may also involve social engineering techniques, such as posing as a trusted source. There are many options for the attachment such as Microsoft Office documents, executables, PDFs, or archived files. Upon opening the attachment (and potentially clicking past protections), the adversary's payload exploits a vulnerability or directly executes on the user's system. The text of the spearphishing email usually tries to give a plausible reason why the file should be opened, and may explain how to bypass system protections in order to do so. The email may also contain instructions on how to decrypt an attachment, such as a zip file password, in order to evade email boundary defenses. Adversaries frequently manipulate file extensions and icons in order to make attached executables appear to be document files, or files exploiting one application appear to be a file for a different one.

Name

T1005

ID

T1005

Description

Adversaries may search local system sources, such as file systems and configuration files or local databases, to find files of interest and sensitive data prior to Exfiltration. Adversaries may do this using a [Command and Scripting Interpreter](<https://attack.mitre.org/techniques/T1059>), such as [cmd](<https://attack.mitre.org/software/S0106>) as well as a [Network Device CLI](<https://attack.mitre.org/techniques/T1059/008>), which have functionality to interact with the file system to gather information. (Citation: show_run_config_cmd_cisco) Adversaries may also use [Automated Collection](<https://attack.mitre.org/techniques/T1119>) on the local system.

Name

T1027

ID

T1027

Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behavior that can be used across different platforms and the network to evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](<https://attack.mitre.org/techniques/T1140>) for [User Execution](<https://attack.mitre.org/techniques/T1204>). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016) Adversaries may also use compressed or archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](<https://attack.mitre.org/techniques/T1027/010>) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](<https://attack.mitre.org/techniques/T1059>). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

Name

T1105

ID

T1105

Description

Adversaries may transfer tools or other files from an external system into a compromised environment. Tools or files may be copied from an external adversary-controlled system to the victim network through the command and control channel or through alternate protocols such as [ftp](https://attack.mitre.org/software/S0095). Once present, adversaries may also transfer/spread tools between victim devices within a compromised environment (i.e. [Lateral Tool Transfer](https://attack.mitre.org/techniques/T1570)). On Windows, adversaries may use various utilities to download tools, such as `copy`, `finger`, [certutil](https://attack.mitre.org/software/S0160), and [PowerShell](https://attack.mitre.org/techniques/T1059/001) commands such as `EX(New-Object Net.WebClient).downloadString()` and `Invoke-WebRequest`. On Linux and macOS systems, a variety of utilities also exist, such as `curl`, `scp`, `sftp`, `tftp`, `rsync`, `finger`, and `wget`. (Citation: t1105_lolbas) Adversaries may also abuse installers and package managers, such as `yum` or `winget`, to download tools to victim hosts. Adversaries have also abused file application features, such as the Windows `search-ms` protocol handler, to deliver malicious files to victims through remote file searches invoked by [User Execution](https://attack.mitre.org/techniques/T1204) (typically after interacting with [Phishing](https://attack.mitre.org/techniques/T1566) lures). (Citation: T1105: Trellix_search-ms) Files can also be transferred using various [Web Service](https://attack.mitre.org/techniques/T1102)s as well as native or otherwise present tools on the victim system. (Citation: PTSecurity Cobalt Dec 2016) In some cases, adversaries may be able to leverage services that sync between a web-based and an on-premises client, such as Dropbox or OneDrive, to transfer files onto victim systems. For example, by compromising a cloud account and logging into the service's web portal, an adversary may be able to trigger an automatic syncing process that transfers the file onto the victim's machine. (Citation: Dropbox Malware Sync)

Name

T1485

ID

T1485

Description

Adversaries may destroy data and files on specific systems or in large numbers on a network to interrupt availability to systems, services, and network resources. Data destruction is likely to render stored data irrecoverable by forensic techniques through overwriting files or data on local and remote drives.(Citation: Symantec Shamoon 2012) (Citation: FireEye Shamoon Nov 2016)(Citation: Palo Alto Shamoon Nov 2016)(Citation: Kaspersky StoneDrill 2017)(Citation: Unit 42 Shamoon3 2018)(Citation: Talos Olympic Destroyer 2018) Common operating system file deletion commands such as ``del`` and ``rm`` often only remove pointers to files without wiping the contents of the files themselves, making the files recoverable by proper forensic methodology. This behavior is distinct from [Disk Content Wipe](<https://attack.mitre.org/techniques/T1561/001>) and [Disk Structure Wipe](<https://attack.mitre.org/techniques/T1561/002>) because individual files are destroyed rather than sections of a storage disk or the disk's logical structure. Adversaries may attempt to overwrite files and directories with randomly generated data to make it irrecoverable.(Citation: Kaspersky StoneDrill 2017)(Citation: Unit 42 Shamoon3 2018) In some cases politically oriented image files have been used to overwrite data. (Citation: FireEye Shamoon Nov 2016)(Citation: Palo Alto Shamoon Nov 2016)(Citation: Kaspersky StoneDrill 2017) To maximize impact on the target organization in operations where network-wide availability interruption is the goal, malware designed for destroying data may have worm-like features to propagate across a network by leveraging additional techniques like [Valid Accounts](<https://attack.mitre.org/techniques/T1078>), [OS Credential Dumping](<https://attack.mitre.org/techniques/T1003>), and [SMB/Windows Admin Shares] (<https://attack.mitre.org/techniques/T1021/002>).(Citation: Symantec Shamoon 2012) (Citation: FireEye Shamoon Nov 2016)(Citation: Palo Alto Shamoon Nov 2016)(Citation: Kaspersky StoneDrill 2017)(Citation: Talos Olympic Destroyer 2018). In cloud environments, adversaries may leverage access to delete cloud storage, cloud storage accounts, machine images, and other infrastructure crucial to operations to damage an organization or their customers.(Citation: Data Destruction - Threat Post)(Citation: DOJ - Cisco Insider)

Name

T1059

ID

T1059

Description

Adversaries may abuse command and script interpreters to execute commands, scripts, or binaries. These interfaces and languages provide ways of interacting with computer

systems and are a common feature across many different platforms. Most systems come with some built-in command-line interface and scripting capabilities, for example, macOS and Linux distributions include some flavor of [Unix Shell](<https://attack.mitre.org/techniques/T1059/004>) while Windows installations include the [Windows Command Shell](<https://attack.mitre.org/techniques/T1059/003>) and [PowerShell](<https://attack.mitre.org/techniques/T1059/001>). There are also cross-platform interpreters such as [Python](<https://attack.mitre.org/techniques/T1059/006>), as well as those commonly associated with client applications such as [JavaScript](<https://attack.mitre.org/techniques/T1059/007>) and [Visual Basic](<https://attack.mitre.org/techniques/T1059/005>). Adversaries may abuse these technologies in various ways as a means of executing arbitrary commands. Commands and scripts can be embedded in [Initial Access](<https://attack.mitre.org/tactics/TA0001>) payloads delivered to victims as lure documents or as secondary payloads downloaded from an existing C2. Adversaries may also execute commands through interactive terminals/shells, as well as utilize various [Remote Services](<https://attack.mitre.org/techniques/T1021>) in order to achieve remote Execution. (Citation: Powershell Remote Commands)(Citation: Cisco IOS Software Integrity Assurance - Command History)(Citation: Remote Shell Execution in Python)

Name

T1087

ID

T1087

Description

Adversaries may attempt to get a listing of valid accounts, usernames, or email addresses on a system or within a compromised environment. This information can help adversaries determine which accounts exist, which can aid in follow-on behavior such as brute-forcing, spear-phishing attacks, or account takeovers (e.g., [Valid Accounts](<https://attack.mitre.org/techniques/T1078>)). Adversaries may use several methods to enumerate accounts, including abuse of existing tools, built-in commands, and potential misconfigurations that leak account names and roles or permissions in the targeted environment. For examples, cloud environments typically provide easily accessible interfaces to obtain user lists.(Citation: AWS List Users)(Citation: Google Cloud - IAM Service Accounts List API) On hosts, adversaries can use default [PowerShell](<https://attack.mitre.org/techniques/T1059/001>) and other command line functionality to identify

accounts. Information about email addresses and accounts may also be extracted by searching an infected system's files.

Name

T1083

ID

T1083

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](<https://attack.mitre.org/techniques/T1083>) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Many command shell utilities can be used to obtain this information. Examples include ``dir``, ``tree``, ``ls``, ``find``, and ``locate``.(Citation: Windows Commands JPCERT) Custom tools may also be used to gather file and directory information and interact with the [Native API](<https://attack.mitre.org/techniques/T1106>). Adversaries may also leverage a [Network Device CLI](<https://attack.mitre.org/techniques/T1059/008>) on network devices to gather file and directory information (e.g. ``dir``, ``show flash``, and/or ``nvram``). (Citation: US-CERT-TA18-106A) Some files and directories may require elevated or specific user permissions to access.

Name

T1113

ID

T1113

Description

Adversaries may attempt to take screen captures of the desktop to gather information over the course of an operation. Screen capturing functionality may be included as a

feature of a remote access tool used in post-compromise operations. Taking a screenshot is also typically possible through native utilities or API calls, such as `CopyFromScreen``, `xwd``, or `screencapture``.(Citation: CopyFromScreen .NET)(Citation: Antiquated Mac Malware)

Name

T1082

ID

T1082

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](https://attack.mitre.org/techniques/T1082) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Tools such as [Systeminfo](https://attack.mitre.org/software/S0096) can be used to gather detailed system information. If running with privileged access, a breakdown of system data can be gathered through the `systemsetup`` configuration tool on macOS. As an example, adversaries with user-level access can execute the `df -aH`` command to obtain currently mounted disks and associated freely available space. Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather detailed system information (e.g. `show version``). (Citation: US-CERT-TA18-106A) [System Information Discovery](https://attack.mitre.org/techniques/T1082) combined with information gathered from other forms of discovery and reconnaissance can drive payload development and concealment. (Citation: OSX.FairyTale)(Citation: 20 macOS Common Tools and Techniques) Infrastructure as a Service (IaaS) cloud providers such as AWS, GCP, and Azure allow access to instance and virtual machine information via APIs. Successful authenticated API calls can return data such as the operating system platform and status of a particular instance or the model view of a virtual machine. (Citation: Amazon Describe Instance)(Citation: Google Instances Resource)(Citation: Microsoft Virtual Machine API)

Name

T1041

ID

T1041

Description

Adversaries may steal data by exfiltrating it over an existing command and control channel. Stolen data is encoded into the normal communications channel using the same protocol as command and control communications.

Indicator

Name

cf2affe891d09bf79c912e9dc1e6cd4d4f8fc4f9579876c54d11a4344d5446fb

Pattern Type

stix

Pattern

```
[file:hashes!'SHA-256' =  
'cf2affe891d09bf79c912e9dc1e6cd4d4f8fc4f9579876c54d11a4344d5446fb']
```

Name

e7cec26b659674a5cb75cf03e6046f2e1f25b52fe0ba36214a8e4864e0cccb53

Pattern Type

stix

Pattern

```
[file:hashes!'SHA-256' =  
'e7cec26b659674a5cb75cf03e6046f2e1f25b52fe0ba36214a8e4864e0cccb53']
```

Name

8a00861d8549920dd29ffa597f4cf137834458cc596e4ed2faf09336e778c0e3

Pattern Type

stix

Pattern

[file:hashes!'SHA-256' =
'8a00861d8549920dd29ffa597f4cf137834458cc596e4ed2faf09336e778c0e3']

Name

8efbf26a851ce6e6e060a808633b968b6337a8e2807178876d03d663083dcf5e

Pattern Type

stix

Pattern

[file:hashes!'SHA-256' =
'8efbf26a851ce6e6e060a808633b968b6337a8e2807178876d03d663083dcf5e']

Intrusion-Set

Name

KDot227

Malware

Name

Kematian-Stealer

indicates

Name
Name
Name
Name
Name
Name
Name
Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

uses

Name
Name
Name
Name
Name
Name
Name
Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

based-on

Name
Name
Name

StixFile

Value

8efbf26a851ce6e6e060a808633b968b6337a8e2807178876d03d663083dcf5e

8a00861d8549920dd29ffa597f4cf137834458cc596e4ed2faf09336e778c0e3

cf2affe891d09bf79c912e9dc1e6cd4d4f8fc4f9579876c54d11a4344d5446fb

e7cec26b659674a5cb75cf03e6046f2e1f25b52fe0ba36214a8e4864e0cccb53

External References

-
- <https://www.cyfirma.com/research/kematian-stealer-a-deep-dive-into-a-new-information-stealer>
-
- <https://otx.alienvault.com/pulse/668e5daf4bc143e93672d0de>