Apr 14 2024

NETMANAGEIT

Intelligence Report

Smoke and mirrors: A strange signed backdoor

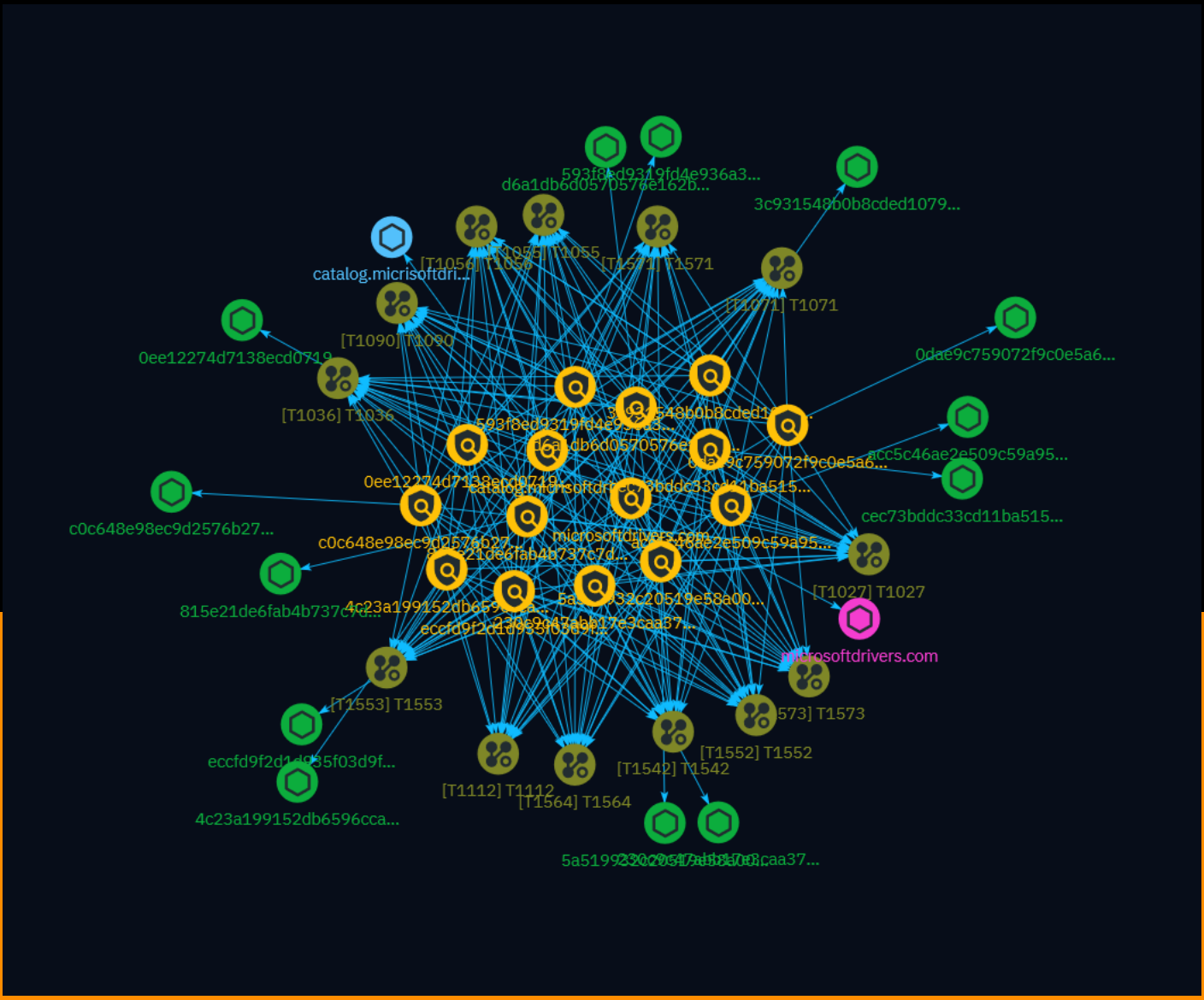

Table of contents

Overview

Entities

Observables

External References

Overview

Description

Sophos X-Ops discovered a curious backdoored and signed executable masquerading as something else. The file was bundled with LaiXi Android Screen Mirroring software. Technical analysis revealed it installs a service called CatalogWatcher and embeds a tiny proxy server, likely to monitor and intercept traffic. Variants were found dating back to early 2023, some signed with valid Microsoft certificates. Sophos and Microsoft worked together to revoke the certificates used.

Confidence

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

100 / 100

Content

N/A

Indicator

Name
catalog.micrisoftdrivers.com
Pattern Type
stix
Pattern
[hostname:value = 'catalog.micrisoftdrivers.com']
Name
microsoftdrivers.com
Pattern Type
stix
Pattern
[domain-name:value = 'microsoftdrivers.com']
Name
d6a1db6d0570576e162bc1c1f9b4e262b92723dbabdde85b27f014a59bbff70c

Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'd6a1db6d0570576e162bc1c1f9b4e262b92723dbabdde85b27f014a59bbff70c']
Name
acc5c46ae2e509c59a952269622b4e6b5fa6cf9d03260bfebdfaa86c734ee6ea
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'acc5c46ae2e509c59a952269622b4e6b5fa6cf9d03260bfebdfaa86c734ee6ea']
Name
c0c648e98ec9d2576b275d55f22b8273a6d2549f117f83a0bcc940194f1d0773
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'c0c648e98ec9d2576b275d55f22b8273a6d2549f117f83a0bcc940194f1d0773']

Name

cec73bddc33cd11ba515e39983e81569d9586abdaabbdd5955389735e826c3c7

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'cec73bddc33cd11ba515e39983e81569d9586abdaabbdd5955389735e826c3c7']

Name

815e21de6fab4b737c7dd844e584c1fc5505e6b180aecdd209fbd9b4ed14e4b2

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '815e21de6fab4b737c7dd844e584c1fc5505e6b180aecdd209fbd9b4ed14e4b2']

Name

593f8ed9319fd4e936a36bc6d0f163b9d43220e61221801ad0af8b1db35a0de5

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '593f8ed9319fd4e936a36bc6d0f163b9d43220e61221801ad0af8b1db35a0de5']

Name

5a519932c20519e58a004ddbfee6c0ed46f1cee8d7c04f362f3545335904bae2

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '5a519932c20519e58a004ddbfee6c0ed46f1cee8d7c04f362f3545335904bae2']

Name

4c23a199152db6596ccafb5ea2363500e2e1df04961a4ede05168999da87d39a

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '4c23a199152db6596ccafb5ea2363500e2e1df04961a4ede05168999da87d39a']

Name

3c931548b0b8cded10793e5517e0a06183b76fa47d2460d28935e28b012e426c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '3c931548b0b8cded10793e5517e0a06183b76fa47d2460d28935e28b012e426c']

Name

230c9c47abb17e3caa37bcb1b8e49b30e671e6c50e88f334107e3350bee13385

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '230c9c47abb17e3caa37bcb1b8e49b30e671e6c50e88f334107e3350bee13385']

Name

0ee12274d7138ecd0719f6cb3800a04a6667968c1be70918e31c6f75de7da1ba

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '0ee12274d7138ecd0719f6cb3800a04a6667968c1be70918e31c6f75de7da1ba']

Name

0dae9c759072f9c0e5a61a9de24a89e76da35ffab8ff9610cc90df417c741f3f

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '0dae9c759072f9c0e5a61a9de24a89e76da35ffab8ff9610cc90df417c741f3f']

Name

eccfd9f2d1d935f03d9fbdb4605281c7a8c23b3791dc33ae8d3c75e0b8fbaec6

Description

Created by VirusTotal connector as the positive count was >= 10

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'eccfd9f2d1d935f03d9fbdb4605281c7a8c23b3791dc33ae8d3c75e0b8fbaec6']

Attack-Pattern

Name
T1056
ID
T1056
Description
Adversaries may use methods of capturing user input to obtain credentials or collect information. During normal system usage, users often provide credentials to various different locations, such as login pages/portals or system dialog boxes. Input capture mechanisms may be transparent to the user (e.g. [Credential API Hooking](https://attack.mitre.org/techniques/T1056/004)) or rely on deceiving the user into providing input into what they believe to be a genuine service (e.g. [Web Portal Capture](https://attack.mitre.org/techniques/T1056/003)).
Name
T1573
ID
T1573
Description

Adversaries may employ a known encryption algorithm to conceal command and control traffic rather than relying on any inherent protections provided by a communication protocol. Despite the use of a secure algorithm, these implementations may be vulnerable to reverse engineering if secret keys are encoded and/or generated within malware samples/configuration files.

Name

T1553

ID

T1553

Description

Adversaries may undermine security controls that will either warn users of untrusted activity or prevent execution of untrusted programs. Operating systems and security products may contain mechanisms to identify programs or websites as possessing some level of trust. Examples of such features would include a program being allowed to run because it is signed by a valid code signing certificate, a program prompting the user with a warning because it has an attribute set from being downloaded from the Internet, or getting an indication that you are about to connect to an untrusted site. Adversaries may attempt to subvert these trust mechanisms. The method adversaries use will depend on the specific mechanism they seek to subvert. Adversaries may conduct [File and Directory Permissions Modification](https://attack.mitre.org/techniques/T1222) or [Modify Registry](https://attack.mitre.org/techniques/T1112) in support of subverting these controls. (Citation: SpectorOps Subverting Trust Sept 2017) Adversaries may also create or steal code signing certificates to acquire trust on target systems.(Citation: Securelist Digital Certificates)(Citation: Symantec Digital Certificates)

Name

T1571

ID

T1571

Description

Adversaries may communicate using a protocol and port pairing that are typically not associated. For example, HTTPS over port 8088(Citation: Symantec Elfin Mar 2019) or port 587(Citation: Fortinet Agent Tesla April 2018) as opposed to the traditional port 443. Adversaries may make changes to the standard port used by a protocol to bypass filtering or muddle analysis/parsing of network data. Adversaries may also make changes to victim systems to abuse non-standard ports. For example, Registry keys and other configuration settings can be used to modify protocol and port pairings.(Citation: change_rdp_port_conti)

Name

T1542

ID

T1542

Description

Adversaries may abuse Pre-OS Boot mechanisms as a way to establish persistence on a system. During the booting process of a computer, firmware and various startup services are loaded before the operating system. These programs control flow of execution before the operating system takes control.(Citation: Wikipedia Booting) Adversaries may overwrite data in boot drivers or firmware such as BIOS (Basic Input/Output System) and The Unified Extensible Firmware Interface (UEFI) to persist on systems at a layer below the operating system. This can be particularly difficult to detect as malware at this level will not be detected by host software-based defenses.

Name

T1090

ID

T1090

Description

Adversaries may use a connection proxy to direct network traffic between systems or act as an intermediary for network communications to a command and control server to avoid direct connections to their infrastructure. Many tools exist that enable traffic redirection through proxies or port redirection, including [HTRAN](https://attack.mitre.org/software/S0040), ZXProxy, and ZXPortMap. (Citation: Trend Micro APT Attack Tools) Adversaries use these types of proxies to manage command and control communications, reduce the number of simultaneous outbound network connections, provide resiliency in the face of connection loss, or to ride over existing trusted communications paths between victims to avoid suspicion. Adversaries may chain together multiple proxies to further disguise the source of malicious traffic. Adversaries can also take advantage of routing schemes in Content Delivery Networks (CDNs) to proxy command and control traffic.

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

T1112

ID

T1112

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. Access to specific areas of the Registry depends on account permissions, some requiring administrator-level access. The built-in Windows command-line utility [Reg](https://attack.mitre.org/software/S0075) may be used for local or remote Registry modification. (Citation: Microsoft Reg) Other tools may also be used, such as a remote access tool, which may contain functionality to interact with the Registry through the Windows API. Registry modifications may also include actions to hide keys, such as prepending key names with a null character, which will cause an error and/or be ignored when read via [Reg](https://attack.mitre.org/software/S0075) or other utilities using the Win32 API. (Citation: Microsoft Reghide NOV 2006) Adversaries may abuse these pseudo-hidden keys to conceal payloads/commands used to maintain persistence. (Citation: TrendMicro POWELIKS AUG 2014) (Citation: SpectorOps Hiding Reg Jul 2017) The Registry of a remote system may be modified to aid in execution of files as part of lateral movement. It requires the remote Registry service to be running on the target system. (Citation: Microsoft Remote) Often [Valid Accounts](https://attack.mitre.org/techniques/T1078) are required, along with access to the remote system's [SMB/Windows Admin Shares](https://attack.mitre.org/techniques/T1021/002) for RPC communication.

Name

T1055

ID

T1055

Description

Adversaries may inject code into processes in order to evade process-based defenses as well as possibly elevate privileges. Process injection is a method of executing arbitrary code in the address space of a separate live process. Running code in the context of another process may allow access to the process's memory, system/network resources, and possibly elevated privileges. Execution via process injection may also evade detection from security products since the execution is masked under a legitimate process. There are many different ways to inject code into a process, many of which abuse legitimate functionalities. These implementations exist for every major OS but are typically platform specific. More sophisticated samples may perform multiple process injections to segment modules and further evade detection, utilizing named pipes or other inter-process communication (IPC) mechanisms as a communication channel.

Name

T1552

ID

T1552

Description

Adversaries may search compromised systems to find and obtain insecurely stored credentials. These credentials can be stored and/or misplaced in many locations on a system, including plaintext files (e.g. [Bash History](https://attack.mitre.org/techniques/T1552/003)), operating system or application-specific repositories (e.g. [Credentials in Registry](https://attack.mitre.org/techniques/T1552/002)), or other specialized files/artifacts (e.g. [Private Keys](https://attack.mitre.org/techniques/T1552/004)).

Name

T1036

ID

T1036

Description

Adversaries may attempt to manipulate features of their artifacts to make them appear legitimate or benign to users and/or security tools. Masquerading occurs when the name or location of an object, legitimate or malicious, is manipulated or abused for the sake of evading defenses and observation. This may include manipulating file metadata, tricking users into misidentifying the file type, and giving legitimate task or service names. Renaming abusable system utilities to evade security monitoring is also a form of [Masquerading](https://attack.mitre.org/techniques/T1036).(Citation: LOLBAS Main Site) Masquerading may also include the use of [Proxy](https://attack.mitre.org/techniques/T1090) or VPNs to disguise IP addresses, which can allow adversaries to blend in with normal network traffic and bypass conditional access policies or anti-abuse protections.

Name

T1071

ID

T1071

Description

Adversaries may communicate using OSI application layer protocols 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. Adversaries may utilize many different protocols, including those used for web browsing, transferring files, electronic mail, or DNS. For connections that occur internally within an enclave (such as those between a proxy or pivot node and other nodes), commonly used protocols are SMB, SSH, or RDP.

Name
T1564
ID
T1564
Description
Adversaries may attempt to hide artifacts associated with their behaviors to evade detection. Operating systems may have features to hide various artifacts, such as important system files and administrative task execution, to avoid disrupting user work environments and prevent users from changing files or features on the system. Adversaries may abuse these features to hide artifacts such as files, directories, user accounts, or other system activity to evade detection.(Citation: Sofacy Komplex Trojan) (Citation: Cybereason OSX Pirrit)(Citation: MalwareBytes ADS July 2015) Adversaries may also attempt to hide artifacts associated with malicious behavior by creating computing regions that are isolated from common security instrumentation, such as through the use of virtualization technology.(Citation: Sophos Ragnar May 2020)

Hostname

Value
catalog.micrisoftdrivers.com

StixFile

Value
eccfd9f2d1d935f03d9fbdb4605281c7a8c23b3791dc33ae8d3c75e0b8fbaec6
d6a1db6d0570576e162bc1c1f9b4e262b92723dbabdde85b27f014a59bbff70c
cec73bddc33cd11ba515e39983e81569d9586abdaabbdd5955389735e826c3c7
c0c648e98ec9d2576b275d55f22b8273a6d2549f117f83a0bcc940194f1d0773
acc5c46ae2e509c59a952269622b4e6b5fa6cf9d03260bfebdfaa86c734ee6ea
5a519932c20519e58a004ddbfee6c0ed46f1cee8d7c04f362f3545335904bae2
815e21de6fab4b737c7dd844e584c1fc5505e6b180aecdd209fbd9b4ed14e4b2
593f8ed9319fd4e936a36bc6d0f163b9d43220e61221801ad0af8b1db35a0de5
4c23a199152db6596ccafb5ea2363500e2e1df04961a4ede05168999da87d39a
3c931548b0b8cded10793e5517e0a06183b76fa47d2460d28935e28b012e426c
230c9c47abb17e3caa37bcb1b8e49b30e671e6c50e88f334107e3350bee13385
0ee12274d7138ecd0719f6cb3800a04a6667968c1be70918e31c6f75de7da1ba
0dae9c759072f9c0e5a61a9de24a89e76da35ffab8ff9610cc90df417c741f3f

Domain-Name

Value
microsoftdrivers.com

External References

- https://news.sophos.com/en-us/2024/04/09/smoke-and-screen-mirrors-a-strange-signed-backdoor/
- https://otx.alienvault.com/pulse/6616d0684e735780ef0610fa