Aug 08 2023

NETMANAGEIT

Intelligence Report

Reptile Malware Targeting Linux Systems

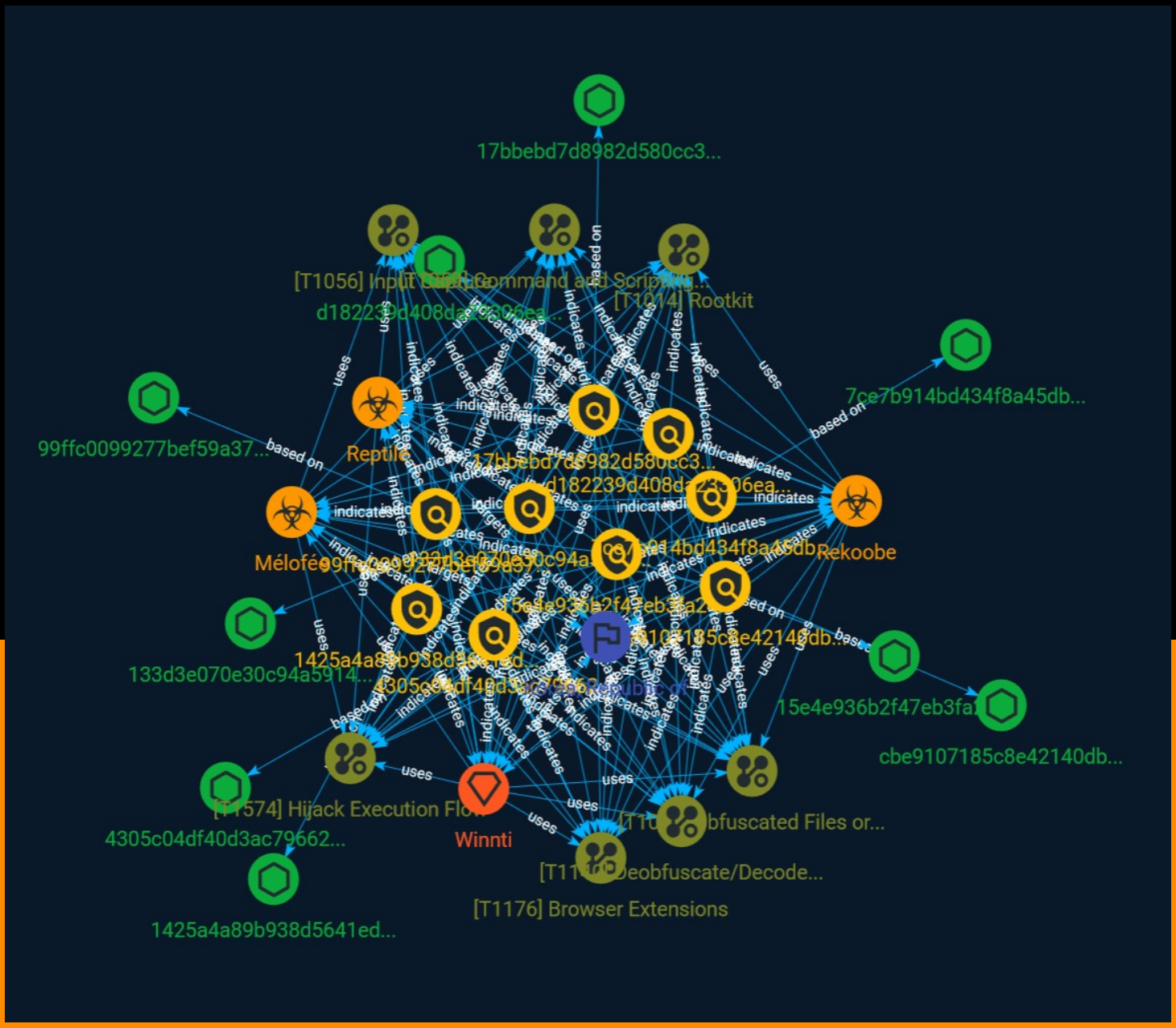

Table of contents

Overview

Entities

Observables

External References

Overview

Description

Reptile is an open-source Linux kernel module rootkit that has been used in a series of attacks targeting companies in South Korea, according to a report by security firm ExaTrack.

Confidence

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

15 / 100

Indicator

Name

133d3e070e30c94a591450b0930daf9f751debc0f4384fac6ace63f60a383818

Description

is__elf SHA256 of f8247453077dd6c5c1471edd01733d7f

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '133d3e070e30c94a591450b0930daf9f751debc0f4384fac6ace63f60a383818']

Name

15e4e936b2f47eb3fa2455b7c22b2714bebe9f8c01b24bbf7cb5f9559999d292

Description

is__elf SHA256 of bb2a0bac5451f8acb229d17c97891eaf

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '15e4e936b2f47eb3fa2455b7c22b2714bebe9f8c01b24bbf7cb5f9559999d292']

Name

17bbebd7d8982d580cc3dea35d988ae2bfd62d708b69662419c41682274e0a14

Description

is__elf SHA256 of cb61b3624885deed6b2181b15db86f4d

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '17bbebd7d8982d580cc3dea35d988ae2bfd62d708b69662419c41682274e0a14']

Name

7ce7b914bd434f8a45db1cb3ec783237a5485b7abcee4df06275ea274e095295

Description

is__elf SHA256 of 977bb7fa58e6dfe80f4bea1a04900276

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '7ce7b914bd434f8a45db1cb3ec783237a5485b7abcee4df06275ea274e095295']

Name

1425a4a89b938d5641ed438333708d1728cfed8c124451180d011f6bbb409976

Description

SHA256 of 1957e405e7326bd2c91d20da1599d18e

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '1425a4a89b938d5641ed438333708d1728cfed8c124451180d011f6bbb409976']

Name

d182239d408da23306ea6b0f5f129ef401565a4d7ab4fe33506f8ac0a08d37ba

Description

is__elf SHA256 of d1abb8c012cc8864dcc109b5a15003ac

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'd182239d408da23306ea6b0f5f129ef401565a4d7ab4fe33506f8ac0a08d37ba']

Name

4305c04df40d3ac7966289cc0a81cedbdd4eee2f92324b26fe26f57f57265bca

Description

is__elf SHA256 of c3c332627e68ce7673ca6f0d273b282e

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '4305c04df40d3ac7966289cc0a81cedbdd4eee2f92324b26fe26f57f57265bca']

Name

99ffc0099277bef59a37a4cfcf4cdd71df13ad33d1c7bf943dc87f803e75dd2c

Description

is__elf SHA256 of 246c5bec21c0a87657786d5d9b53fe38

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '99ffc0099277bef59a37a4cfcf4cdd71df13ad33d1c7bf943dc87f803e75dd2c']

Name

cbe9107185c8e42140dbd1294d8c20849134dd122cc64348f1bfcc90401379ec

Description

is__elf SHA256 of 5b788feef374bbac8a572adaf1da3d38

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'cbe9107185c8e42140dbd1294d8c20849134dd122cc64348f1bfcc90401379ec']

Malware

Name
Mélofée
Name
Rekoobe
Name
Reptile

Intrusion-Set

Name
Winnti

Country

Name
Korea, Republic of

Attack-Pattern

Name
Rootkit
ID
T1014
Description
Adversaries may use rootkits to hide the presence of programs, files, network connections, services, drivers, and other system components. Rootkits are programs that hide the existence of malware by intercepting/hooking and modifying operating system API calls that supply system information. (Citation: Symantec Windows Rootkits) Rootkits or rootkit enabling functionality may reside at the user or kernel level in the operating system or lower, to include a hypervisor, Master Boot Record, or [System Firmware](https://attack.mitre.org/techniques/T1542/001). (Citation: Wikipedia Rootkit) Rootkits have been seen for Windows, Linux, and Mac OS X systems. (Citation: CrowdStrike Linux Rootkit) (Citation: BlackHat Mac OSX Rootkit)
Name
Browser Extensions
ID
T1176

Description

Adversaries may abuse Internet browser extensions to establish persistent access to victim systems. Browser extensions or plugins are small programs that can add functionality and customize aspects of Internet browsers. They can be installed directly or through a browser's app store and generally have access and permissions to everything that the browser can access.(Citation: Wikipedia Browser Extension)(Citation: Chrome Extensions Definition) Malicious extensions can be installed into a browser through malicious app store downloads masquerading as legitimate extensions, through social engineering, or by an adversary that has already compromised a system. Security can be limited on browser app stores so it may not be difficult for malicious extensions to defeat automated scanners.(Citation: Malicious Chrome Extension Numbers) Depending on the browser, adversaries may also manipulate an extension's update url to install updates from an adversary controlled server or manipulate the mobile configuration file to silently install additional extensions. Previous to macOS 11, adversaries could silently install browser extensions via the command line using the `profiles` tool to install malicious `.mobileconfig` files. In macOS 11+, the use of the `profiles` tool can no longer install configuration profiles, however `.mobileconfig` files can be planted and installed with user interaction.(Citation: xorrior chrome extensions macOS) Once the extension is installed, it can browse to websites in the background, steal all information that a user enters into a browser (including credentials), and be used as an installer for a RAT for persistence. (Citation: Chrome Extension Crypto Miner)(Citation: ICEBRG Chrome Extensions)(Citation: Banker Google Chrome Extension Steals Creds)(Citation: Catch All Chrome Extension) There have also been instances of botnets using a persistent backdoor through malicious Chrome extensions.(Citation: Stantinko Botnet) There have also been similar examples of extensions being used for command & control.(Citation: Chrome Extension C2 Malware)

Name

Input Capture

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

Obfuscated Files or Information

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

Hijack Execution Flow

ID

T1574

Description

Adversaries may execute their own malicious payloads by hijacking the way operating systems run programs. Hijacking execution flow can be for the purposes of persistence, since this hijacked execution may reoccur over time. Adversaries may also use these mechanisms to elevate privileges or evade defenses, such as application control or other restrictions on execution. There are many ways an adversary may hijack the flow of execution, including by manipulating how the operating system locates programs to be executed. How the operating system locates libraries to be used by a program can also be intercepted. Locations where the operating system looks for programs/resources, such as file directories and in the case of Windows the Registry, could also be poisoned to include malicious payloads.

Name

Command and Scripting Interpreter

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

Deobfuscate/Decode Files or Information

ID

T1140

Description

Adversaries may use [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to hide artifacts of an intrusion from analysis. They may require separate mechanisms to decode or deobfuscate that information depending on how they intend to use it. Methods for doing that include built-in functionality of malware or by using utilities present on the system. One such example is the use of [certutil](https://attack.mitre.org/software/S0160) to decode a remote access tool portable executable file that has been hidden inside a certificate file.(Citation: Malwarebytes Targeted Attack against Saudi Arabia) Another example is using the Windows `copy /b` command to reassemble binary fragments into a malicious payload.(Citation: Carbon Black Obfuscation Sept 2016) Sometimes a user's action may be required to open it for deobfuscation or decryption as part of [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)

StixFile

Value
d182239d408da23306ea6b0f5f129ef401565a4d7ab4fe33506f8ac0a08d37ba
7ce7b914bd434f8a45db1cb3ec783237a5485b7abcee4df06275ea274e095295
1425a4a89b938d5641ed438333708d1728cfed8c124451180d011f6bbb409976
4305c04df40d3ac7966289cc0a81cedbdd4eee2f92324b26fe26f57f57265bca
15e4e936b2f47eb3fa2455b7c22b2714bebe9f8c01b24bbf7cb5f9559999d292
133d3e070e30c94a591450b0930daf9f751debc0f4384fac6ace63f60a383818
17bbebd7d8982d580cc3dea35d988ae2bfd62d708b69662419c41682274e0a14
cbe9107185c8e42140dbd1294d8c20849134dd122cc64348f1bfcc90401379ec
99ffc0099277bef59a37a4cfcf4cdd71df13ad33d1c7bf943dc87f803e75dd2c

External References

- https://otx.alienvault.com/pulse/64d267daf70e411d6f941487
- https://asec.ahnlab.com/en/55785/