



TLP: CLEAR



Ransomware Detection and Analysis Report

CERTMH_CTI_2025_32

11/07/2025

Table of Contents



01	Executive Summary
02	Introduction
03	IOCs
04	Analysis Methodology
05	Static Analysis
06	Dynamic Analysis
07	MITRE ATT&CK Mapping
08	Possible Adversaries
09	Recommendations

Executive Summary

This report presents a comprehensive analysis of a malware sample proactively intercepted by our SOC team during an attempted intrusion. The malware, delivered via the HTTP protocol on port 80 from the IP address 192[.]166[.]244[.]243, located in Hong Kong, was successfully captured. The prompt detection and containment ensured that no systems were affected. This early intervention enabled a detailed examination of the malware's behavior, vectors, and indicators of compromise (IOCs), supporting continuous improvement of defensive posture.

The malware exploited a known vulnerability in **Apache version 2.4.50** using a malicious **POST** request. This allowed the attacker to gain unauthorized access by establishing a reverse shell connection to the target system. Once access was obtained, the malware was downloaded into the `/tmp` directory, after which a thorough static analysis was conducted. The analysis revealed that the malware had been packed using **UPX(Ultimate Packer for Executables)** version 4.22. Upon unpacking, several significant details were discovered, including FTP credentials that pointed to an external server at the IP address 152[.]32[.]185[.]104. Additionally, an onion link was found, which was likely used for command-and-control communication following the infection. There was also clear evidence of the malware attempting to enumerate system information.

Further dynamic analysis confirmed that the malware operated as ransomware. It encrypted most of the user's files on the infected system, saved detailed system information in a file located at `/tmp/112.txt`, and dropped a ransom note in the directory path `/home/${USERNAME}/README`. The behavior exhibited by the malware closely matched several techniques from the MITRE ATT&CK framework. These included the use of command and scripting interpreters, obfuscation of files or information, data exfiltration using alternative protocols, and the encryption of data for impact.

The attacker relied on hardcoded FTP credentials to attempt data exfiltration and utilized public key encryption to secure the encrypted files, making recovery without the private key practically impossible. These tactics reflect a well-planned and persistent approach aimed not only at encrypting data for ransom but also at stealing information stealthily.

Based on the tactics, techniques, and procedures observed, the malware campaign appears to be the work of a sophisticated threat actor. It bears resemblance to operations carried out by advanced persistent threat groups such as Ke3chang, also known as **APT 15**, and **APT 28**, also known as Fancy Bear. Both groups are known for their involvement in cyber espionage and ransomware attacks using similar methods and infrastructure.

Introduction

Objectives of the report

This report is designed to provide an in-depth analysis of a malware sample that was captured during an active monitoring operation. The primary objectives of this analysis are to assess the functionality, behavior, and overall impact of the malware, with a particular focus on identifying critical Indicators of Compromise. It also aims to understand the exploitation techniques used by the adversary to gain unauthorized access and maintain control over the target system. The report utilizes a combination of static and dynamic analysis methodologies, offering a thorough examination of the malware from both structural and behavioral perspectives. By analyzing the static file components along with its real-time execution patterns, the report delivers a complete risk assessment of the malware's capabilities. Additionally, it provides insights into how the malware interacts within the environment, how it avoids detection mechanisms, and the strategies it uses to establish and maintain persistence within the compromised system.

Scope of Analysis

The scope of this analysis encompasses a thorough examination of both static and dynamic aspects.

The static analysis involves dissecting the malware's components, file structure, and any modifications made to the system at the file level. This will help in identifying key characteristics of the malware that might indicate malicious intent or potential vulnerabilities that it targets. Dynamic analysis, on the other hand, captures the malware's real-time behavior during execution, including system changes, network communications, and its interaction with other processes. This will provide insights into how the malware propagates, executes commands, and attempts to remain undetected.

The analysis includes a detailed review of the initial attack vector, focusing on how the malware was delivered and its subsequent behavior upon activation. A key component of this report is mapping the malware's tactics, techniques, and procedures (TTPs) to the MITRE ATT&CK framework. This mapping helps contextualize the threat within a broader strategic framework, offering a more comprehensive understanding of potential adversaries and their methods. By combining pre-execution findings with runtime behavior, the report aims to provide a holistic view of the threat landscape and its implications for defense strategies.

Introduction

Attack Vector

Parameter	Details
Monitoring Tag	HTTP
Source IP Address	192[.]166[.]244[.]243
Source IP Location	Hong Kong
Protocol	HTTP
Destination Port	80
Vulnerability Exploited	Apache 2.4.50 (via POST request)
Malware Upload Path	/tmp

The malware was delivered through the HTTP protocol over port 80. The source of the attack was identified as the IP address 192[.]166[.]244[.]243, registered in Hong Kong.

The attacker exploited a known vulnerability in Apache version 2.4.50 by sending a specially crafted POST request payload using the path `/cgi-bin/./%32%65/./%32%65/./%32%65/./%32%65/./%32%65/./%32%65/./%32%65/bin/sh`. This allowed the adversary to upload and execute the malware in the `/tmp` directory of the targeted system.

IoCs

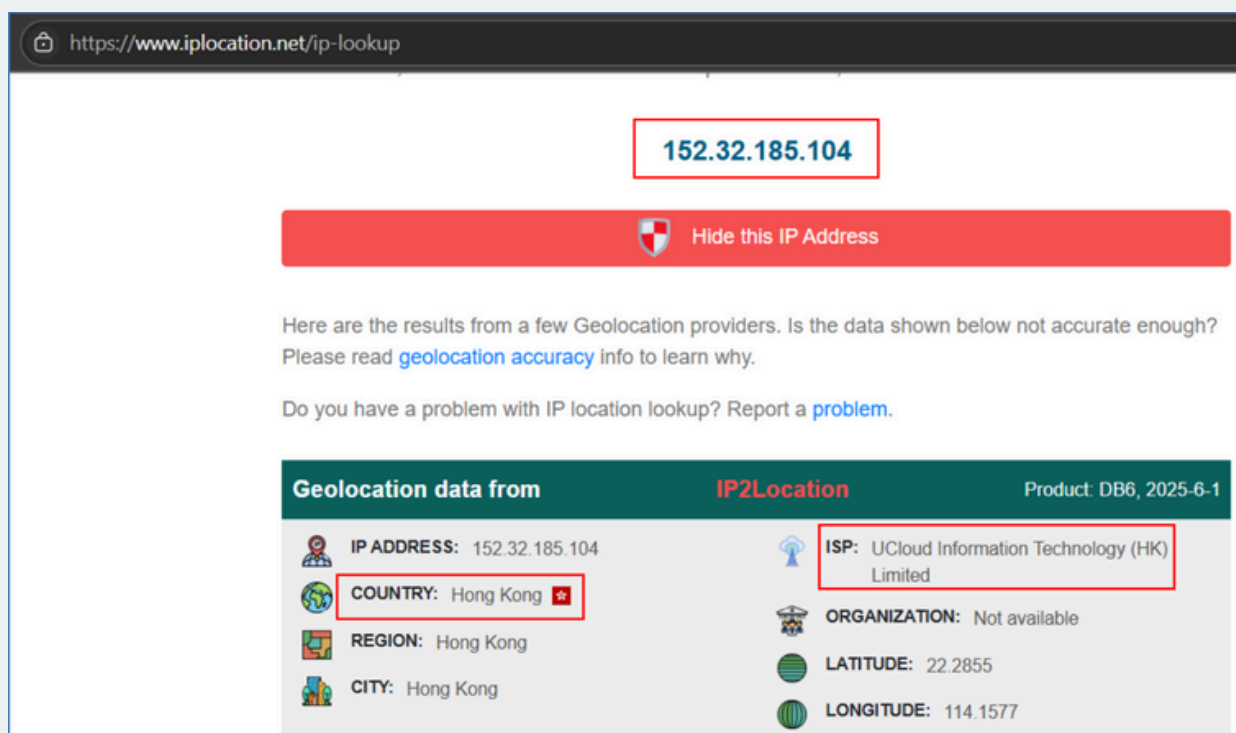
Malware Capture

The malware was captured through HTTP traffic and delivered to a decoy system configured to simulate realistic web services. The malicious payload was stored in the /tmp directory of the system. The event was recorded in a controlled, isolated environment specifically designed to analyze malicious activity without risk of further propagation or damage. This approach ensured safe handling and accurate assessment of the malware sample.

Connections

During the behavioral analysis of the malware, it was observed establishing outbound connections to external IP addresses. One of the key connections was made to IP address 152[.]32[.]185[.]104. This IP belongs to an autonomous system registered in Hong Kong, matching the geographical origin of the initial attack source, 192[.]166[.]244[.]243. The connection was likely used to facilitate data exfiltration or maintain remote control over the compromised system, consistent with post-exploitation tactics.

IP Address	ASN	ISP	Country
152[.]32[.]185[.]104	135377	Ucloud Information Technology Hk	Hong Kong



The screenshot shows the IP2Location website interface. At the top, the IP address 152.32.185.104 is entered and highlighted. Below it, a red button says "Hide this IP Address". The main content area displays geolocation data from IP2Location, with a product version of DB6, 2025-6-1. The data is organized into two columns. The left column lists: IP ADDRESS: 152.32.185.104, COUNTRY: Hong Kong (marked with a star), REGION: Hong Kong, and CITY: Hong Kong. The right column lists: ISP: UCloud Information Technology (HK) Limited, ORGANIZATION: Not available, LATITUDE: 22.2855, and LONGITUDE: 114.1577. The COUNTRY and ISP fields are highlighted with red boxes in the original image.

152[.]32[.]185[.]104 IP Location Lookup Results

IoCs

Type	Indicator	Description
Malware File	main	The main malware executable file.
File Hash (MD5)	a548bf0527e16515d2d0e98951fb2be9	MD5 hash of the main malware file.
File Hash (SHA1)	cbe8a1dee2c079065532cae33215b00d81f3c228	SHA1 hash of the main malware file.
File Hash (SHA256)	744d5f0e9f1cffff51fcee7b19dd4068473a69ff79f3e81a2bce54cd655	SHA256 hash of the main malware file.
FTP Server IP	152[.]32[.]185[.]104	IP address of the FTP server used for command-and-control.
FTP Port	21	Port used by the malware to communicate with the FTP server.
FTP Username	admin112	Username used to connect to the FTP server.
FTP Password	b7t16016qg9	Password used to authenticate to the FTP server.
Ransom Note Path	/home/\${USERNAME}/README	Path where the ransomware leaves the ransom note.
Ransom Note Hash (MD5)	0889666c8bc99616c56c47a965f9ceb0	MD5 hash of the ransom note file.
Dropped File Path	/tmp/112.txt	Path of a dropped text file with system information.
Dropped File Hash (MD5)	1340107ce24c38fb69074c2f5078dbd8	MD5 hash of the 112.txt file.
Dropped File Path	/tmp/README	Path where an additional ransom note file was dropped.
Public Key	<i>(Found within binary strings)</i>	Public key likely used for encryption of victim files.
Attack Source IP	192[.]166[.]244[.]243	Source IP address of the attacker.
Attack Source Location	Hong Kong	Geolocation of the attacker's IP address.
User-Agent	-	Specific User-Agent used in the HTTP request (can be extracted)
Exploit Path	/cgi-bin/.%32%65/.%32%65/.%32	Exploit path used to trigger Apache vulnerability (CVE-2021-

Analysis Methodology

Objectives of the report

The malware analysis was performed in a controlled and isolated environment to ensure no cross-contamination with live systems.

VM Instance Configurations

The analysis was carried out on an Ubuntu 22.04 virtual machine (VM) running on a Windows host machine using Hyper-V. The virtual machine configuration was as follows:

Component	Specification
Operating System	Ubuntu 22.04
RAM	8 GB
Storage	80 GB
CPU	6 vCPUs
Network Adapter	Not Connected

This configuration was chosen to provide adequate resources for analyzing the malware's behavior without risking the security of the primary host machine. The network adapter was disconnected to prevent any communication with external systems during analysis, except for the initial interaction with the malware's FTP server.

Static Analysis

File Information

Attribute	Value
File Name	main
File Size	18,424 Bytes
File Type	ELF 64-bit LSB shared object, x86-64
MD5 Hash	a548bf0527e16515d2d0e98951fb2be9
SHA1 Hash	cbe8a1dee2c079065532cae33215b00d81f3c228
SHA256 Hash	744d5f0e9f1cffff51fcee7b19dd4068473a69ff79f3e81a2bce54cd655b7de1
Malware Category	Ransomware
Packer Used	UPX 4.22

Packer

During static analysis, it was discovered that the malware binary had been packed using UPX 4.22, a well-known executable packer. UPX (Ultimate Packer for Executables) is often used by attackers to compress and obfuscate the contents of a binary file, making it harder to analyze. The purpose of packing the malware is to avoid detection by security tools and analysts, as it prevents the file from being immediately recognized as malicious.

The packed binary file was unpacked using the same version of UPX (4.22) to reveal its actual contents. Upon unpacking, several interesting strings were discovered inside the binary. These strings pointed to various functionalities of the malware, such as system information enumeration, FTP credentials, encryption-related public keys, and even a ransom note.

Here is a detailed breakdown of the findings:

1. UPX Version Detection:

The first step in understanding the packed binary was identifying the version of UPX used. During a string analysis of the binary, the version "UPX 4.22" was found embedded in the file.

Static Analysis

This indicated that the binary had been packed using this particular version of UPX, and it allowed us to use UPX 4.22 to unpack the file for further analysis.

```
remnux@remnux:~/treacle$ strings main_original | grep -i "upx"
UPX!
$Info: This file is packed with the UPX executable packer http://upx.sf.net $
$Id: UPX 4.22 Copyright (C) 1996-2024 the UPX Team. All Rights Reserved. $
UPX!
UPX!
```

UPX Version

UPX 4.22 Version Found in Binary Strings

The screenshot above shows the string "UPX 4.22" found within the binary. This version string was located during the static analysis of the packed file. Knowing the exact packing tool and version helped in unpacking the binary successfully.

2. Unpacked Binary Contents:

After unpacking the binary, we were able to extract several interesting strings that indicated the malware's capabilities. These strings revealed details about the malware's behavior, including commands related to system information collection, encrypted file handling, and FTP communication.

```
remnux@remnux:~/treacle$ ../upx/upx-4.2.2-amd64_linux/upx -d main
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2024
UPX 4.2.2 Markus Oberhumer, Laszlo Molnar & John Reiser Jan 3rd 2024
```

File size	Ratio	Format	Name
48883 <- 18424	37.69%	linux/amd64	main

Unpacked 1 file.

Unpacked

Binary Unpacked with UPX 4.22 Version

The unpacked binary showed several system-related strings, such as commands to retrieve system information, libraries the malware interacts with, and encrypted data handling mechanisms. These strings were crucial in understanding the malware's operation and potential targets.

3. System Information Strings:

Among the strings found within the unpacked binary were several related to system information enumeration. These strings suggested that the malware was likely performing an inventory of the infected system, which could be used to determine the environment or configure the attack based on the system's specifications.

Static Analysis

```
c8uBf.
[]A\
[]A\
Bext4
PATH
/etc/passwd
Invalid
System Name: %s
Node Name (Hostname): %s
Release: %s
Machine Architecture: %s
Kernel Version: %s
CPU Architecture: %s
Number of CPU Cores: %d
Total Memory: %ld
Free Memory: %ld
Available Memory: %ld
Total Disk Space: %ld
Free Disk Space: %ld
File System Type: %s
Process ID (PID): %d
Parent Process ID (PPID): %d
Current User: %s
User ID (UID): %d
Group ID (GID): %d
Load Average (1-minute): %f
Load Average (5-minute): %f
Load Average (15-minute): %f
Environment Variables: %s
/etc/passwd file: %s
%s/%s
exts
threads
%s.%s
```

System Information Related Strings Found

The screenshot above displays the system information-related strings found in the unpacked binary. These strings included references to system libraries and executable paths, which the malware likely accessed to gather data about the infected machine.

4. FTP Credentials and Communication:

One of the most concerning findings was the discovery of hardcoded FTP credentials in the unpacked binary. The malware contained an FTP URL along with a username and password. These credentials were likely used for exfiltrating stolen data or for further command-and-control communication with the attacker's server.

```
Target file extensions
Address for the FTP(S) server
ftp://152.32.185.104
FTP(S) username
admin112
FTP(S) password
b7t16016qg9
```

FTP URL and Its Credentials Found

Static Analysis

The screenshot above highlights the FTP URL along with the username and password found in the unpacked binary. This indicates that the malware attempts to communicate with a remote FTP server (IP: 152.32.185.104) for further malicious actions, such as uploading or downloading files.

5. Public Key for Encryption:

Another significant finding was the presence of a public key within the binary. This key is likely used for asymmetric encryption, indicating that the malware encrypts files on the victim's system before demanding a ransom. The presence of this key suggests that the ransomware may use a public-private key pair to encrypt and decrypt files, making it harder for the victim to recover data without paying the ransom.

```
-----BEGIN PUBLIC KEY-----
MIIEIjANBgkqhkiG9w0BAQEFAAOCAg8AMIIECgKCBAEAYBK3TrFE9kjiJisQgUm
zhVFAHlripVhmYtFiCil7TTOxb/Fb0gXLpxehw1QKrES/QW2l0d9g9x12M48fUeA
LYudJdiHNLsr49A1gz+kIWVLg2yAVrFqzHku0vLlmEw32ByDxYy8Uf2MlxEzkgdk
RQLVzbWRI/JdcToqMfuE/Nut/vCN2QXBtNXzIc6q2oAhHaWzEpuVZ0hNPfPojmUV
3ea9XD7mDF3NfkdxcMz4aa7+he8ZkDLid7yG+vBmKEpflhisdx60kyz7mV9t5hHI
9fmj/w23fS5uaMSYblHri2ZLH7wo7XzqaERN2Zr9fLjxFnFwPNnglG50FfebvloK
9TaeUl3xv1cFMKvmF9R867uBEXuCFEGfVWUGoDbCy/KyS+KajkbxrpsyskC35I4
YzlJe9JS3wvcNbZmtipmbM0xhw0PoZlB2T36+31Xw/nAde6TbDDy9AXHT3oLFnd/
8aR7MURa4naXLF69bytGRSGp8e2MRcliVBByl8CFBzp4SyAEbQ7g0aqAPB23rkRY
5s+p732whp4wD6amGfgI2NV6ufbEJXKDUQUhCygSyerua+VwFVZRS3A3TCKP40
rBMAkqWAHUPDveoWLQR5YDhZbZHSANq8jDLyd5v/ZpGITS4xyroA7t9GDf+1Xwb
+GybxWRBFfn12AeRudKo12Hm3s6CXkitVpY159ngc3hZ4aHjUEBIXmEEFnreFa+8
+LoA8h+9uReKYIoRJY+Pzd84E+cNEZdhAkiwonUeVQsfV8ecL5b9TQmTiJo/W0/S
vjkgKDf73SjGGLRWfb5tQxQm739VP0LdEEMfZKM/wY+Y3PpAYMfp1P26LC0LFwrb
w9gpoKkM+ZufByoWkhtTZ3+VcKSvbfyah+3v8sFJIT+/smYMBv/vHnkf40tNQEA
mPz/02nkKba6eBVqLcbZ5G7y0h2JzGXRTA0gagNX+Q9aXWHPsDaDyUjze6592lL0
dFY7qVDHCexuj8jfIuzSMLAZNr4QiZqTAASG4SDMpifF4sWMxvDm+vpts/VVfzuG
MN6CvVSgo8ZcmqSCI8VxmFQ7oMS6V6bMTN6NTE/TT8Tkzn7gbC6z0UcPRIHphqv5
syRLB7sRYbt7TERPSYJtdul7Df0eYLpksSrvX0UVG2HpI35vWiVmJ0HwPbs4MfB5
mYpsuTkUg4oQTfC/xa5Iz0bTTjvssMf/2UBxidxVvgZb3Ni0bZAWQL8/LBnTpJ3n
UwBYXL6q7gq6TjXBDqyy54p7t6P5e80Fs6YC6kSDXn/jYXovP0dUtat+uoBZGp3X
VyzP2iLGo3DR9wo0Ro0CKhqOI/qyNlwF0GaGuPdW5FPAvdExX0V1v3gYX4sFoKX+
/QIDAQAB
-----END PUBLIC KEY-----
*****
```

Public Key Strings Found

The screenshot above shows the public key string found in the binary. This public key is essential for the encryption process carried out by the ransomware. It is used to encrypt files on the infected machine, rendering them inaccessible without the corresponding private key.

Static Analysis

6. Ransom Note and Onion Link:

Finally, the unpacked binary also contained strings related to a ransom note. This note provided instructions for the victim to contact the attacker via an onion link, which suggests the use of the Tor network to maintain anonymity. The ransom note typically demands payment in cryptocurrency and provides instructions on how to make the payment.

```
*****
ID - 112
Your computer and servers are encrypted.
We have downloaded all you data. Contact us or we sell your data.
Using Tor Browser:
1) Download tor browser from this site: https://torproject.org
2) Install Tor Broser
3) Open website - fi0k2pble3kjlC487ofs8ew9g97m6ufryxgzqtf3blzqe53iuy2y4yu1.onion
4) Use your ID
```

Tor link in ransom note

Ransom Note with Onion Link Strings Found

The screenshot shows the ransom note strings found within the unpacked binary. This ransom note also included an onion link, which is indicative of the use of the Tor network for secure, anonymous communication between the victim and the attacker. The note likely contains instructions on how to pay the ransom to decrypt the files.

Dynamic Analysis

Behavior Analysis

During dynamic analysis in a isolated VM, malware exhibited behaviors of a ransomware. It could be executed on user-level. It encrypted most of the accessible files. It created a file 112.txt on path **/tmp** and left a ransom note with filename README on path **/home/\${USERNAME}/**.

S.No.	Behavior/Category	Source	Details
1	Binary is Packed	UPX Parser	Entropy: 7.868573292817385
2	Binary is Stripped	Static Parser	Malware is stripped (symbols removed for obfuscation/evasion).
3	Binary Deletes itself after execution	—	After the malware execution is complete, it deletes itself.
4	Calls Native Functions (syscalls)	API Call	Invoked syscalls: read, fstat, mmap, mprotect, close, openat, stat, writev, exit_group, procexit, execve, open, readlink
5	Reads Files	API Call	/home/ubuntu/main reads: <ul style="list-style-type: none"> • /lib/x86_64-linux-gnu/libdl.so.2 • /lib/x86_64-linux-gnu/libz.so.1 • /lib/x86_64-linux-gnu/libpthread.so.0 • /lib/x86_64-linux-gnu/libc.so.6 • /home/ubuntu/main
6	Calls a System Call Typically Used to Get Directory Entries	API Call	/home/ubuntu/main executes: getdents64
7	Binary is statically linked	—	The binary has been stripped of all debug symbols and its shared objects are also statically linked to the binary.
8	Ability to modify file	Indicator Combinations	Analysis contains indicators for crypto or data obfuscation (e.g., base64, decrypt) which can encrypt data.

Dynamic Analysis

Dropped files

File Path	File Type	MD5 Hash	Details
/tmp/112.txt	Text File	1340107ce24c38fb69074c2f5078dbd8	Contains malware-enumerated system
/home/\${USERNAME}/REA DME	Text File	0889666c8bc99616c56c47a965f9ceb0	Contains ransom note with instructions to contact

```
ubuntu2@ubuntu2:~/Downloads$ cat /tmp/112.txt
System Name: Linux
Node Name (Hostname): 4p
Release: 6.8.0-60-generic
Version: #63~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC Tue Apr 22 19:00:15 UTC 2
Machine Architecture: x86_64
Kernel Version: 6.8.0-60-generic
CPU Architecture: x86_64
Number of CPU Cores: 6
Total Memory: 5034332160
Free Memory: 3321999360
Available Memory: 3368464384
Total Disk Space: 83423059968
Free Disk Space: 71444738048
File System Type: ext4
Process ID (PID): 2695
Parent Process ID (PPID): 1938
Current User: ubuntu2
Current Working Directory: /home/ubuntu2/Desktop
User ID (UID): 1000
Group ID (GID): 1000
Load Average (1-minute): 0.000000
Load Average (5-minute): 0.000000
Load Average (15-minute): 0.000000
Environment Variables: /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin
/etc/passwd file: root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
```

112.txt file data contains malware enumerated system information

Dynamic Analysis

```
ubuntu2@ubuntu2:~/Downloads$ cat ../README
*****
Hi!
*****

ID - 112

Your computer and servers are encrypted.

We have downloaded all you data. Contact us or we sell your data.

Using Tor Browser:
1) Download tor browser from this site: https://torproject.org
2) Install Tor Broser
3) Open website - fi0k2pble3kjl487ofs8ew9g97m6ufryxgzqtf3b1zqe53iuy2y4yu1.onion
4) Use your ID
```

README file contains ransom note

Network Analysis

During network analysis as part of network analysis, malware is making ftp connection with IP **152.32.185.104** on port 21 with username **admin112** and password **b7t16016qg9**.

IP Address	Port	ASN	ISP	Country
152.32.185.104	21	135377	Ucloud Information Technology Hk	Hong Kong

The FTP connection attempt was unsuccessful, possibly indicating an issue with the server or a failed connection attempt by the malware.

10	0.000225458	192.168.0.141	152.32.185.104	TCP	74	54152 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627985030 TSecr=0 WS=128
11	0.000339971	192.168.0.141	152.32.185.104	TCP	74	54160 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627985030 TSecr=0 WS=128
12	0.000355140	192.168.0.141	152.32.185.104	TCP	74	54170 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627985030 TSecr=0 WS=128
13	0.000429585	192.168.0.141	152.32.185.104	TCP	74	54186 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627985030 TSecr=0 WS=128
14	0.018646087	192.168.0.141	152.32.185.104	TCP	74	54192 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627985049 TSecr=0 WS=128
15	1.050626374	192.168.0.141	152.32.185.104	TCP	74	[TCP Retransmission] 54138 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627986081 TSecr=0 WS=128
16	1.050639640	192.168.0.141	152.32.185.104	TCP	74	[TCP Retransmission] 54098 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627986081 TSecr=0 WS=128
17	1.050645391	192.168.0.141	152.32.185.104	TCP	74	[TCP Retransmission] 54082 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627986081 TSecr=0 WS=128
18	1.050703363	192.168.0.141	152.32.185.104	TCP	74	[TCP Retransmission] 54152 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627986081 TSecr=0 WS=128
19	1.050711289	192.168.0.141	152.32.185.104	TCP	74	[TCP Retransmission] 54122 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627986081 TSecr=0 WS=128
20	1.050717160	192.168.0.141	152.32.185.104	TCP	74	[TCP Retransmission] 54100 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1627986081 TSecr=0 WS=128

unsuccessful
connections

Unsuccessful FTP connection

MITRE ATT&CK Mapping

Tactic	Technique ID	Technique Name	Description
Initial Access	T1190	Exploit Public-Facing Application	The attacker exploited a vulnerability (CVE-2021-42013) in Apache 2.4.50 by sending a malicious POST request to gain unauthorized access to the system through the /bin/sh shell.
Execution	T1059	Command and Scripting Interpreter	After gaining access, the attacker used the /bin/sh shell to execute commands, allowing interactive control of the target system through native shell utilities.
	T1106	Native API	The malware utilized direct system calls such as read, write, mmap, execve, etc., to interact with the OS, manipulate memory, and execute files, bypassing high-level defenses.
Defense Evasion	T1027	Obfuscated Files or Information	The binary was packed using UPX 4.22 to obfuscate its contents, making it difficult for antivirus and static analysis tools to detect or analyze the malware.
Credential Access	T1003	OS Credential Dumping	The malware accessed the /etc/passwd file to gather user account information, likely aiding in privilege escalation or lateral movement.
Discovery	T1083	File and Directory Discovery	System calls such as getdents64 were used to enumerate directories and identify potential targets for encryption or exfiltration.
	T1124	System Time Discovery	The malware collected system time information, possibly to determine when to launch encryption or tailor the attack based on time zones or region.
Collection	T1119	Automated Collection	Malware identified and encrypted files automatically, effectively collecting critical data for ransom purposes. Dropped files like 112.txt and README supported this functionality.

MITRE ATT&CK Mapping

Tactic	Technique ID	Technique Name	Description
Command and Control	T1071.001	Application Layer Protocol: Web Protocols (FTP)	The malware attempted to connect to an FTP server at 152.32.185.104 using hardcoded credentials (admin112 / b7t16016qg9) to establish C2 communication and potentially exfiltrate data.
Exfiltration	T1048	Exfiltration Over Alternative Protocol	An FTP-based exfiltration attempt was made for the file 112.txt, which contained system information. Though unsuccessful, this indicates intent to steal and exfiltrate data.
Impact	T1486	Data Encrypted for Impact	As ransomware, the malware encrypted user files and dropped a ransom note to coerce payment in exchange for a decryption key. This aligns directly with the goal of disrupting access to data for ransom.

Possible Adversaries

Based on the TTPs identified in this malware analysis, two potential adversary groups emerge: Ke3chang (APT15) and APT28 (Fancy Bear).

Ke3chang (APT15), a Chinese cyber espionage group, has been active since at least 2011 and typically targets government, defense, and corporate sectors across the globe. They are known for exploiting web application vulnerabilities (e.g., the Apache vulnerability in this case) and using obfuscated malware to avoid detection. Their tactics also include data exfiltration via FTP, which aligns with the malware's use of FTP to potentially exfiltrate sensitive system information. The exploitation of public-facing applications and the use of shell scripts for command execution also mirrors their typical behavior.

APT28 (Fancy Bear), a Russian group linked to the GRU, has been involved in numerous high-profile cyber-espionage campaigns, particularly against government and military targets. They also exploit web-based vulnerabilities and have employed similar obfuscation tactics, such as packing malware to evade detection. Their focus on data exfiltration, including using protocols like FTP, also matches the malware's behavior. Given their known operations in political espionage and cyber-warfare, APT28 remains another strong candidate for this attack.

Recommendations

1. Deploy Web Application Firewalls (WAF) with Exploit Detection for CVEs

Implementation Strategy:

- Configure WAF rules to detect and block malicious POST payloads.
- Integrate WAF with threat intelligence feeds to auto-update rules.
- Regularly patch all web-facing services, especially Apache servers, through patch management.

2. Detect and Block Reverse Shell & FTP Traffic

Implementation Strategy:

- Use network intrusion detection/prevention systems (NIDS/NIPS) to monitor and block reverse shell behavior and unencrypted outbound FTP connections.
- Block outbound traffic to port 21 (FTP) and other non-essential ports by default.
- Deploy host-based firewalls or EDRs to alert on suspicious processes invoking /bin/sh or making network connections to foreign IPs.

3. Implement Anti-Packing and Binary Obfuscation Detection in Endpoint Protection

Implementation Strategy:

- Ensure endpoint protection software includes heuristics for detecting packed binaries (e.g., UPX) and stripped executables.
- Use sandbox-based malware detonation solutions to dynamically analyze suspicious executables before execution.
- Establish a workflow to automatically unpack binaries and scan them with both static and dynamic analyzers before allowing execution.

4. Restrict Execution & Access Controls in Temp Directories

Implementation Strategy:

- Apply policies on systems to prevent execution of binaries from /tmp or /var/tmp directories.
- Enable noexec mount options on /tmp, and log any execution attempts for real-time alerts.
- Monitor for the creation of files like README or 112.txt to detect ransomware behavior early.