POC Ransomware

Luka Wynants

Abstract

*Dit document beschrijft de ontwikkeling en werking van een **Proof-of-Concept (PoC) ransomware**, ontworpen als onderdeel van een breder onderzoek naar ransomware-dreigingen en de effectiviteit van **CyberArk Endpoint Privilege Manager (EPM)** in het beschermen tegen dergelijke aanvallen. De PoC is ontwikkeld om verschillende technieken te testen die door moderne ransomware worden gebruikt om detectie te ontwijken en bestanden te versleutelen.*

*Het PoC-model bestaat uit meerdere componenten: een **builder** voor gepersonaliseerde configuratie, een **dropper** om malafide code te injecteren, een **keygen,** voor asymmetrische encryptie, een **encryptor** die bestanden versleutelt met een combinatie van **AES en RSA** en een **decryptor** voor systeem herstel. Dit onderzoek evalueert verschillende encryptiestrategieën, waaronder **symmetrische, asymmetrische en hybride encryptie**, en analyseert welke het meest efficiënt is voor ransomware-aanvallen.*

Deze documentatie biedt een gedetailleerd overzicht van de architectuur en functionaliteit van de PoC, evenals een uitleg over de technieken die zijn toegepast om detectie te vermijden en encryptie efficiënt uit te voeren.

1.0 Introductie

Ransomware is een voortdurend evoluerende dreiging die al meerdere bedrijven heeft getroffen en enorme sommen losgeld eist. Een goed voorbeeld hiervan is de ransomware-aanval op Colonial Pipeline in mei 2021, waarbij de ransomwaregroep DarkSide het Amerikaanse oliebedrijf lamlegde. Als gevolg hiervan werd de brandstofvoorziening in grote delen van de VS verstoord. Colonial Pipeline zag zich uiteindelijk genoodzaakt om $4,4 miljoen losgeld te betalen om weer toegang te krijgen tot hun systemen. Dit incident toont aan hoe groot de impact van ransomware kan zijn en benadrukt het belang van effectieve beveiligingsmaatregelen zoals CyberArk Endpoint Privilege Manager (EPM).

In het kader van mijn stage bij ActWise richt mijn onderzoek zich op de vraag:

"Hoe effectief is CyberArk Endpoint Privilege Manager (EPM) in het beschermen tegen ransomware-aanvallen?"

Om deze vraag te beantwoorden, is het belangrijk om te begrijpen hoe ransomware zich verspreidt, welke technieken het gebruikt om Windows Defender te omzeilen, en de gebruikte encryptieprocessen.

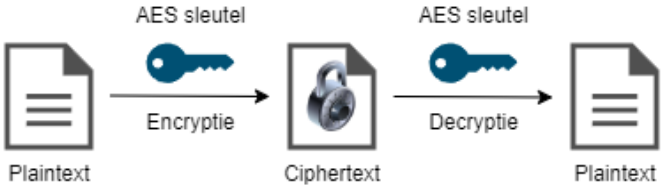
Als eerste stap heb ik bestaande ransomwarevarianten zoals WannaCry, Trigona, Akira en LockBit 3.0 getest binnen een onveilige omgeving zonder EPM. Dit gaf inzicht in hun werking, encryptiemethoden en evasietechnieken. De volgende stap is het ontwikkelen van een eigen PoC ransomware, om deze technieken in een gecontroleerde omgeving te repliceren en te analyseren.

Deze documentatie beschrijft de structuur van de PoC-ransomware. Dit vormt de basis voor verdere testen, waarbij de ransomware zal worden uitgevoerd in een omgeving met EPM, om de effectiviteit van EPM te beoordelen in het stoppen of beperken van de aanval.

2.0 Encryptiestrategieën en Analyse

In deze sectie worden de verschillende encryptiestrategieën die worden gebruikt in ransomware-aanvallen geanalyseerd. Encryptie is het belangrijkste middel voor ransomware om bestanden onbruikbaar te maken voor de gebruiker, wat het belangrijkste doel is van de aanval. We gaan in op de drie meest voorkomende technieken symmetrische encryptie, asymmetrische encryptie, en hybride encryptie.

2.1 Symmetrische encryptie (AES)

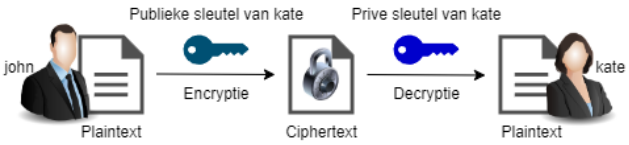


Figuur 1 Symmetrische encryptie

Symmetrische encryptie is een versleutelingstechniek waarbij een dezelfde sleutel wordt gebruikt voor zowel de encryptie als de decryptie van data (Zie Fig. 1.0). Het meest gebruikte algoritme voor symmetrische encryptie is AES (Advanced Encryption Standard). De snelheid van AES maakt het zeer geschikt voor ransomware-aanvallen, waarbij vaak duizenden bestanden snel moeten worden versleuteld.

Hoewel AES snel is, heeft het een kwetsbaarheid: zodra de AES-sleutel wordt verkregen, kunnen alle versleutelde bestanden worden ontsleuteld. Dit maakt het moeilijk om de sleutel veilig over het internet te verzenden of te delen voor communicatie. Dit betekent dat ransomware vaak een extra laag van beveiliging nodig heeft om de AES-sleutel veilig te stellen.

2.2 Asymmetrische encryptie (RSA)

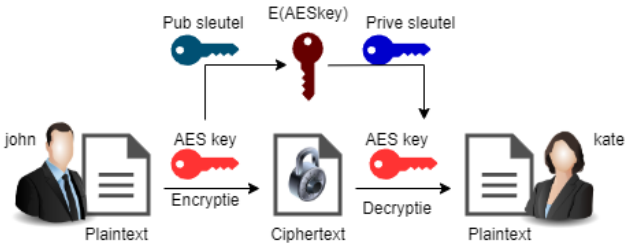


Figuur 2 Asymmetrische encryptie

In tegenstelling tot symmetrische encryptie maakt RSA gebruik van een sleutel paar. Een publieke sleutel wordt gebruikt voor de encryptie van data en een privé sleutel voor de decryptie van data (Zie Fig. 1.1). In tegendeel is deze techniek veel langzamer dan symmetrische encryptie, maar het biedt een extra beveiligingslaag doordat de privé sleutel nooit over het netwerk hoeft te worden verzonden.

2.3 Hybride encryptie (AES + RSA)

De hybride encryptie-methode combineert de voordelen van zowel symmetrische als asymmetrische encryptie. Bij hybride encryptie word symmetrische encryptie gebruikt om de data te versleutelen (omdat het snel en efficiënt is), terwijl asymmetrische encryptie wordt gebruikt om de symmetrische sleutel te encrypteren.



Figuur 3 Hybride encryptie

In figuur 3 wordt dit proces geïllustreerd

1. Er wordt een willekeurige AES-sleutel gegenereerd om de data te versleutelen. De versleutelde data noemen we de *ciphertext*.
2. Vervolgens wordt de AES-sleutel versleuteld met de publieke sleutel van Kate.
3. De versleutelde AES-sleutel en de ciphertext worden naar Kate gestuurd.

Omdat alleen Kate de bijbehorende privésleutel heeft, kan zij als enige de data ontsleutelen. Dit doet ze als volgt:

1. Ze gebruikt haar privésleutel om de versleutelde AES-sleutel te decrypteren en de originele sleutel terug te krijgen.
2. Met deze ontsleutelde AES-sleutel kan ze vervolgens de ciphertext ontcijferen en de oorspronkelijke data herstellen.

Op deze manier biedt hybride encryptie zowel snelheid als veiligheid: AES zorgt voor efficiënte encryptie, terwijl RSA een veilige sleuteluitwisseling mogelijk maakt.

3.0 Evasietechnieken en Windows-misbruik

Evasietechnieken zijn cruciaal voor ransomware omdat ze helpen om detectie te voorkomen door antivirussoftware, EDR (Endpoint Detection and Response), en SIEM-systemen. Moderne beveiligingsoplossingen analyseren signatures, bestandsgedrag, procesinteracties en netwerkverkeer om kwaadaardige activiteiten op te sporen. Door gebruik te maken van technieken zoals obfuscatie, het misbruiken van legitieme Windows-functionaliteiten en het injecteren van code in legitieme processen, kan ransomware onopgemerkt uitvoeren.

3.1 Windows registry

Het Windows-registry is een database waarin Windows en applicaties configuratie-instellingen opslaan. Het bevat informatie over systeeminstellingen, gebruikersvoorkeuren en softwareconfiguraties. De registry is onderverdeeld in verschillende hives (hoofdcategorieën), waarvan de belangrijkste zijn:

- HKEY_LOCAL_MACHINE (HKLM): Bevat instellingen die voor alle gebruikers gelden.
- HKEY_CURRENT_USER (HKCU): Bevat instellingen die specifiek zijn voor de ingelogde gebruiker.
- HKEY_CLASSES_ROOT (HKCR): Bevat informatie over bestandsextensies en COM-objecten

Dit is gevaarlijk sinds het betekend dat een aanvaller de registry ook zou kunnen gebruiken om veranderingen te maken aan windows applicaties, bijvoorbeeld de windows defender registry staat in:
HKLM:\SOFTWARE\Policies\Microsoft\Windows Defender

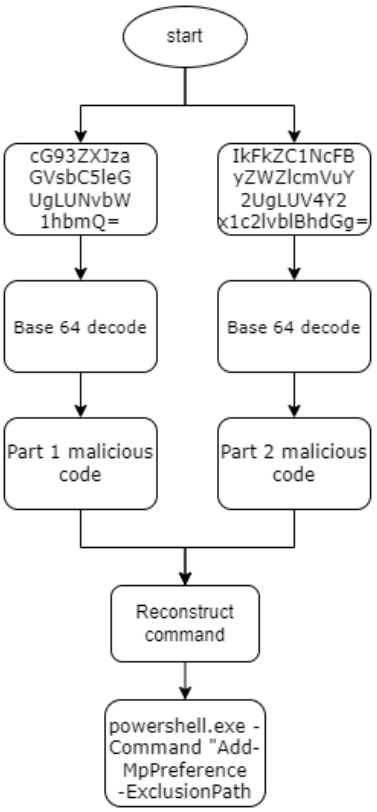
Een aanvaller kan eenvoudig de waardes binnen deze registry-sleutel aanpassen om Windows Defender en real-time protection uit te schakelen. Zoals te zien in Figuur 4, kan dit worden uitgevoerd met Python en de winreg-library.

```python
#windows defender registry key
defender_path = r"SOFTWARE\Policies\Microsoft\Windows Defender"

#open the registry key
with reg.OpenKey(reg.HKEY_LOCAL_MACHINE, defender_path, 0, reg.KEY_SET_VALUE) as key:
    reg.SetValueEx(key, "DisableAntiSpyware", 0, reg.REG_DWORD, 1)

#disabling real time protection
realtime_path = r"SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection"
with reg.CreateKey(reg.HKEY_LOCAL_MACHINE, realtime_path) as key:
    reg.SetValueEx(key, "DisableRealtimeMonitoring", 0, reg.REG_DWORD, 1)
    reg.SetValueEx(key, "DisableBehaviorMonitoring", 0, reg.REG_DWORD, 1)
    reg.SetValueEx(key, "DisableOnAccessProtection", 0, reg.REG_DWORD, 1)
    reg.SetValueEx(key, "DisableScanOnRealtimeEnable", 0, reg.REG_DWORD, 1)
```

Figuur 4 Windows defender uitschakelen met python

3.2 Code Obfuscatie

Een veelgebruikte techniek in ransomware is het verbergen van malafide payloads door ze te encrypteren of encoderen. Hierdoor lijkt de code op onschuldige tekst, maar eenmaal gedecodeerd kan deze kwaadaardige instructies uitvoeren.



Figuur 5 Flowchart van code obfuscatie

Een simpel voorbeeld hiervan is te zien in figuur 5. Er zijn twee strings die op willekeurige tekst lijken, maar deze twee strings worden gedecodeerd en het commando wordt gereconstrueerd naar de volgende commando:

powershell.exe -Command "Add-MpPreference -ExclusionPath

Dit commando wordt gebruikt om een map uit te sluiten van antivirusscanning. Normaal gesproken zou dit opgemerkt worden door Windows antivirus, maar doordat het in real-time wordt gedecodeerd, kan de real-time bescherming het niet detecteren.

4.0 Componenten

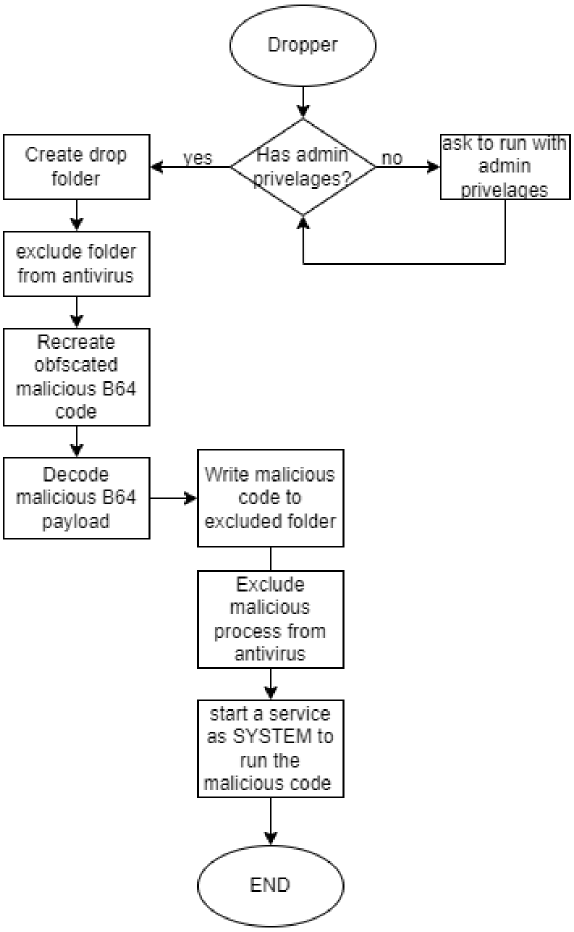
In deze sectie worden alle componenten van de ransomware uitgelegd.

4.1 Builder

De Builder is nog niet volledig afgemaakt, maar het is bedoeld om automatisch een uitvoerbaar bestand gereed te maken van de “dropper”, waarbij je zelf instellingen en configuraties kunt aanpassen. Daarnaast genereert de Builder een nieuw sleutelpaar voor de aanvaller en integreert het de openbare sleutel in de “dropper” executabel.
Aanpasbare instellingen in de configuratie zijn onder andere:

- Het aantal bestanden dat wordt versleuteld voordat de AES-sleutel wordt geroteerd.
- De tijdsduur waarin de ransomware actief moet zijn.
- Specifieke bestanden en mappen die niet versleuteld mogen worden.

4.3 Dropper



Figuur 6 Dropper general flow

De dropper speelt een cruciale rol bij het afleveren en uitvoeren van de ransomware-payload op het doelwit. Dit gebeurt door een combinatie van bestandsmanipulatie, obfuscatie en het omzeilen van Windows Defender-beveiligingen. Het is essentieel dat de dropper onopgemerkt blijft door antivirussoftware om detectie te voorkomen.

De dropper werkt als volgens:

1. Controle op Administratorrechten
Bij het opstarten controleert de dropper of het script met administratorrechten wordt uitgevoerd. Als dit niet het geval is, wordt een UAC-prompt getriggerd om de benodigde rechten te verkrijgen.

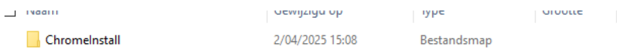
2. Aanmaken van een Verborgen Map
Vervolgens creëert de dropper een nieuwe map op de volgende locatie:

“C:\Users\<gebruiker>\Documents\ChromeInstall”

Deze map dient als opslaglocatie voor de encryptor. Om detectie verder te minimaliseren, wordt een de volgende powershell commando real time gedecodeerd van base64:

Add-MpPreference -ExclusionPath "<path>"

Hiermee wordt de map toegevoegd aan de uitsluitingen van Windows Defender, waardoor bestanden in deze map niet worden gecontroleerd door real-time bescherming. Deze techniek is eerder beschreven in sectie 3.2.

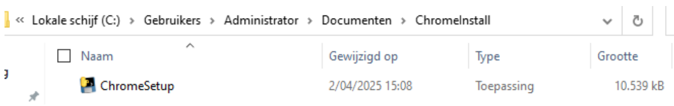


Figuur 7 ChromeInstall map

3. encryptor-decodering
De dropper laadt vervolgens een Base64-gecodeerd bestand (Zie Figuur 8) dat de encryptor executable bevat.

Figuur 8 Binaries.txt payload

Deze inhoud wordt gedecodeerd en opgeslagen als een uitvoerbaar bestand “ChromeSetup.exe”. Dit bestand wordt geplaatst in de eerder aangemaakte verborgen map, klaar voor verdere uitvoering (Zie figuur 9).



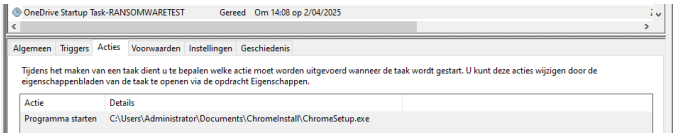
Figuur 9 Decoded malicious “encryptor” executable

4. Uitsluiting van het Uitvoerbare Bestand
Om te voorkomen dat Windows Defender de payload alsnog detecteert en blokkeert, wordt ook het executabel bestand zelf uitgesloten van scanning met de volgende PowerShell-commando:

Add-MpPreference -ExclusionProcess "<process>"

5. Automatische Uitvoering via een Geplande Taak
Tot slot zorgt de dropper ervoor dat de encryptor automatisch wordt uitgevoerd met de hoogste privileges. Dit wordt bereikt door automatisch een taak aan te maken in de Windows Taakplanner (Task Scheduler) (Zie figuur 10):

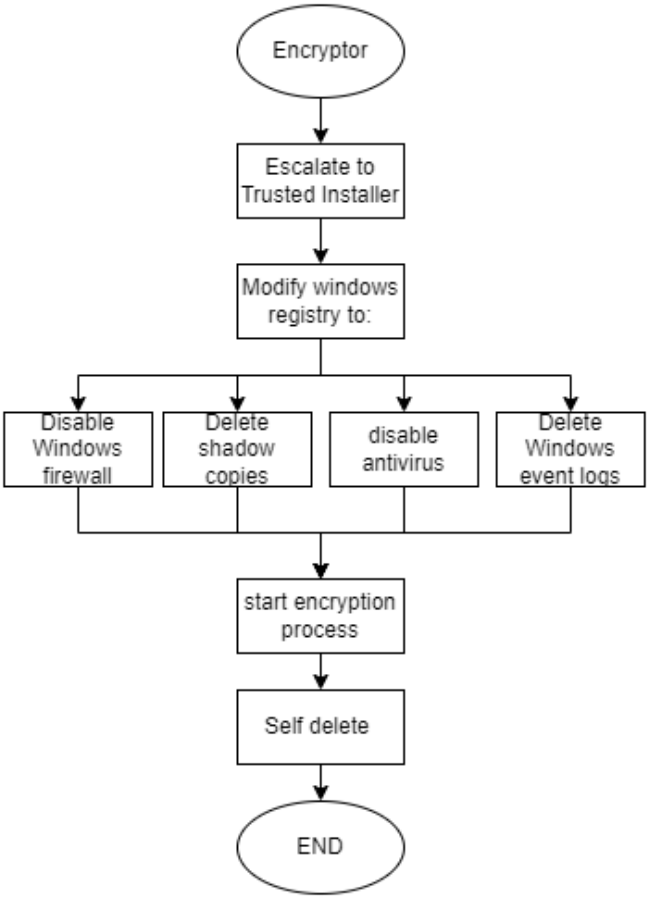
- De taak draait onder het NT AUTHORITY\SYSTEM-account.
- De ChromeSetup.exe automatisch gestart.



Figuur 10 Aangemaakte taak in de windows task scheduler

Door deze methoden te combineren, garandeert de dropper een succesvolle en onopgemerkte infectie, waarbij antivirusmaatregelen worden omzeild en de payload zonder onderbrekingen wordt uitgevoerd.

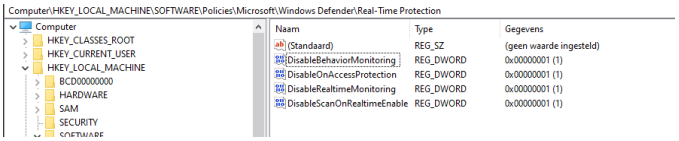
4.4 Encryptor



Figuur 11 Encryptor general flow

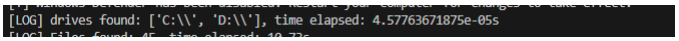
De encryptor is de malicieuze payload. Het voert meerdere taken uit, waaronder het scannen van aangesloten schijven, het zoeken naar bestanden, het versleutelen van bestanden met AES-encryptie en het opslaan van de encryptiesleutels met behulp van een RSA-public key. Daarnaast kan het schaduwkopieën verwijderen, Windows Defender uitschakelen en event logs wissen om detectie en herstel te bemoeilijken. De encryptor werkt als volgens:

1. Uitschakelen van Windows Defender
Als eerste probeert de encryptor Windows Defender volledig uit te schakelen. Het Windows-register wordt aangepast om Windows Defender en de real-time bescherming uit te schakelen.



Figuur 12 Windows registry

2. Schijven scannen (scan_drives)
Vervolgens worden alle beschikbare schijven op de geïnfecteerde machine opgespoord (C:, D:, E:, enz.). Dit gebeurt door een bitmasker van de aangesloten drives op te halen met behulp van ctypes.windll.kernel32.GetLogicalDrives(). Dit bitmasker wordt vertaald naar letters (A-Z), en alle gevonden drives worden opgeslagen in een lijst.



Figuur 13 Gevonden drives

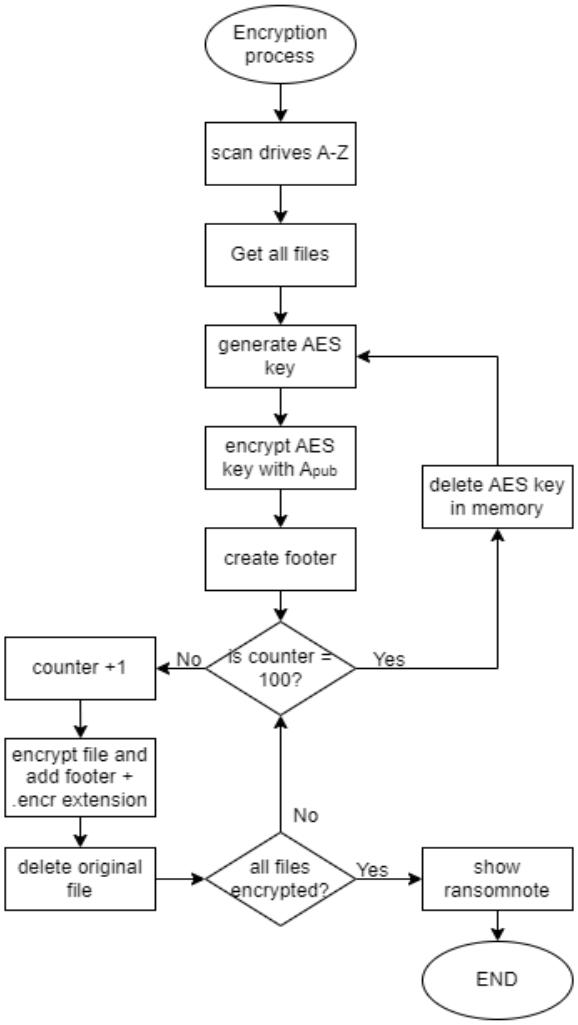
3. Bestanden ophalen
Daarna worden alle schijven simultaan onderzocht via verschillende threads, waarbij alle bestanden die versleuteld kunnen worden, worden verzameld. Er zijn enkele uitsluitingen:

Uitsluiting type	Uitsluiting
Bestand extensies	.exe, .dll, .sys, .lnk, .log
Bestanden	boot.ini, bootmgr, ntldr, BCD, ransom note
Mappen	C:\\Windows, C:\\Program Files, C:\\Program Files (x86), C:\\System Volume Information, C:\\$WINDOWS.~BT

Figuur 15 Uitsluitingen

Deze bestanden en mappen worden overgeslagen omdat ze cruciaal zijn voor het besturingssysteem of door de ransomware zelf zijn gegenereerd, zoals de ransom note. Momenteel wordt alleen de map “Pictures” versleuteld. Dit is nog aanpasbaar; ik heb dit zo ingesteld om de versleuteling in een testomgeving overzichtelijk te houden.

4. Encryptie process



Figuur 14 Encryptie process

De versleuteling verloopt als volgt (Zie Figuur 11):

1. Het optimale aantal threads wordt berekend op basis van de CPU en het beschikbare RAM-geheugen, zodat de encryptie zo efficiënt mogelijk verloopt.
2. Het encryptieproces wordt verdeeld over meerdere threads, waarbij elke thread parallel werkt aan het versleutelen van bestanden.
3. Elke 100 bestanden wordt een nieuwe AES-sleutel gegenereerd en een bijbehorende footer aangemaakt. Dit voorkomt dat alle bestanden met dezelfde sleutel worden versleuteld, wat de beveiliging versterkt.
4. Een bestand wordt geopend in binaire modus en de inhoud wordt versleuteld met AES. De versleutelde data wordt opgeslagen in een nieuw bestand met de extensie ".yZgu0".
5. Na de encryptie wordt de footer toegevoegd aan het bestand. Deze footer bevat de versleutelde AES-sleutel en IV, zodat het bestand later ontsleuteld kan worden (zie figuur voor een voorbeeld).
6. Het originele bestand wordt verwijderd. Zodra 100 bestanden zijn versleuteld, wordt een nieuwe AES-sleutel gegenereerd en gaat het proces verder met de volgende batch bestanden.

5. Verminderen van Herstelopties

Om te voorkomen dat bestanden kunnen worden hersteld worden de Windows Shadow Copies verwijderd. Een shadow copy is een back-up die door Windows automatisch wordt aangemaakt en de mogelijkheid biedt om bestanden te herstellen. Door deze te verwijderen, wordt herstel via shadow copies onmogelijk. Dit gebeurt met het volgende commando:

vssadmin delete shadows /all /quiet

6. Sporen verwijderen

Om forensisch onderzoek te bemoeilijken, worden de Windows Event Logs gewist. Dit zorgt ervoor dat er minder sporen achterblijven van de aanval en het moeilijker wordt om de aanval te reconstrueren.

4.6 Decryptor

De decryptor is verantwoordelijk voor het automatisch opsporen en ontsleutelen van versleutelde bestanden op een systeem. Het maakt gebruik van threading om het proces efficiënt te laten verlopen.

4.7 Custom libraries

Keygen

De KeyGen-library is verantwoordelijk voor het veilig beheren van sleutels. Het wordt geïmporteerd en gebruikt in zowel de Encryptor als Decryptor voor het uitvoeren van de volgende taken:

- Genereren van een RSA-sleutelpaar (2048 bits) en opslaan als .pem.
- Data encrypteren in blokken met de publieke sleutel.
- Data decrypteren met de private sleutel.
- Sleutels laden vanuit bestanden
- Sleutels laden vanuit strings

SymEncryption

De Symencryption-library is verantwoordelijk voor symmetrische encryptie met behulp van AES in CBC-modus. Het genereert in-memory een 128-bit AES-sleutel en IV.
De belangrijkste functies van deze library zijn:

- Genereren van een willekeurige AES-sleutel en IV.
- Encrypteren van data met AES.
- Decrypteren van data met AES.

Deze library wordt gebruikt door de Encryptor (malicious payload) om bestanden te versleutelen met AES voordat de sleutel veilig wordt opgeslagen via de keygen module.

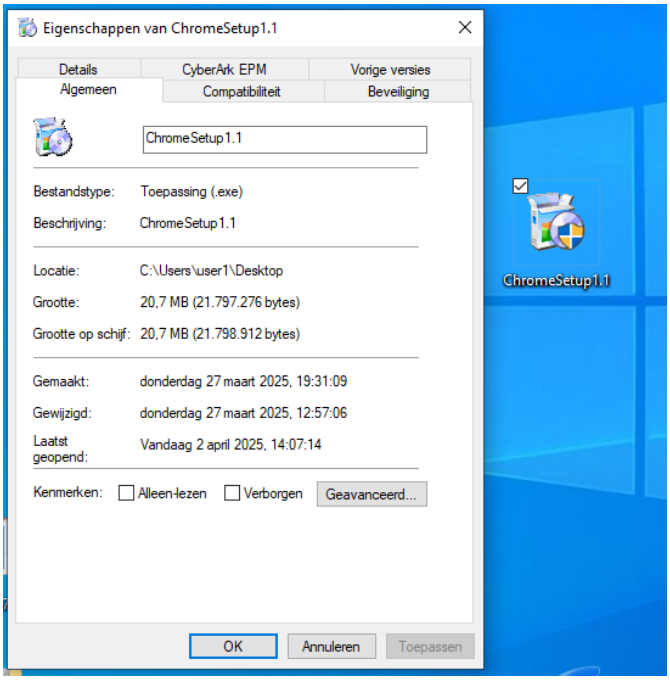
B64

De B64-library biedt functionaliteit om data te coderen en decoderen met Base64. Dit wordt gebruikt door de Dropper, zodat malicious PowerShell-commando's minder snel gedetecteerd worden door beveiligingssoftware. Ook wordt de Base64-gecodeerde payload hiermee weer gedecodeerd vóór uitvoering.
Belangrijkste functies:

- Binaire data naar Base64 coderen.
- Base64-data terug naar binaire vorm decoderen.
- Lezen en schrijven van (gecodeerde) bestanden.

4.8 Executables

Van de originele Python-scripts zijn .exe-bestanden gemaakt. Bij de dropper is het icoon aangepast zodat het lijkt op een legitiem programma. Zoals te zien is in figuur 16, heet het bestand “ChromeSetup.exe” en lijkt het op de officiële Chrome installer, wat helpt bij social engineering.



Figuur 15 ChromeSetup.exe

De encryptor is eerder besproken en is ook gecompileerd naar een executable. Deze executable is base64-gecodeerd en wordt als tekstbestand meegeleverd binnen de dropper. Tijdens de uitvoering decodeert de dropper dit bestand en voert het uit.

5.0 Resultaten

5.1 EDR/ antvirus detectie

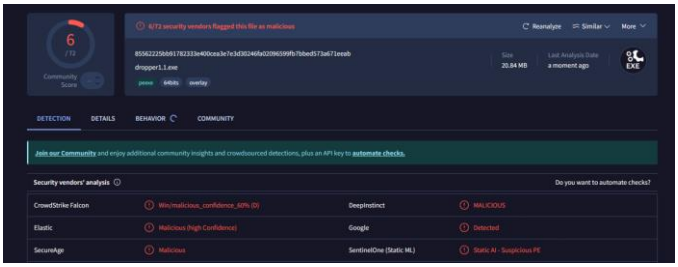
Figuur 17 Windows 10 en 11 Virus & threat protection scan

In Figuur 18 is te zien dat de dropper executable werd gescand met de ingebouwde Windows Virus & Threat Protection op Windows 10 en Windows 11 (versie 24H2). De malware werd niet gedetecteerd op beide systemen. Wel zijn er verschillen in functionaliteit tussen Windows 10 en Windows 11. Dit komt voornamelijk door extra beveiligingsmaatregelen in de nieuwste Windows-versie zoals Tamper Protection. Tamper protection voorkomt dat bepaalde registersleutels via PowerShell worden aangepast. Hierdoor is het uitschakelen van Windows Defender op Windows 11 via register keys niet mogelijk. De tabel in figuur 18 toont welke functionaliteiten nog steeds werken per Windows-versie.

Functie	W10	W11
Onopgemerkt door windows antivirus	✓	✓
Creatie van Verborgen Map	✓	✓
Creatie van Geplande Taak	✓	x
Uitschakelen van Windows Defender	✓	x
Schijven scannen	✓	✓
Bestanden ophalen	✓	✓
Bestanden encrypteren	✓	✓
Shadow kopieën verwijderen	✓	✓
Events verwijderen	✓	✓

Figuur 18 Functionaliteit analyse tabel

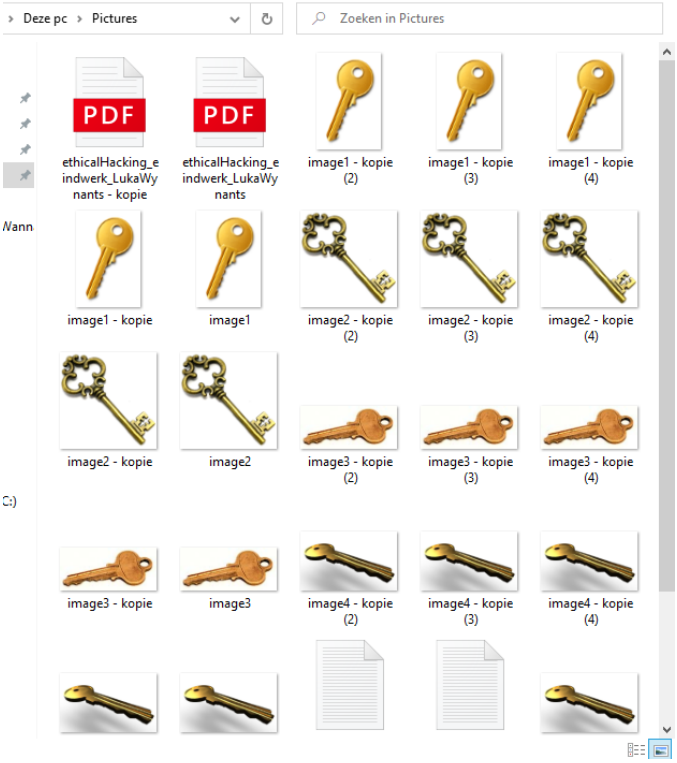
Ook heb ik de dropper laten analyseren door VirusTotal. De VirusTotal analyse van de dropper executable leverde een score van 6/72 op, wat betekent dat slechts 6 van de 72 gebruikte antivirusengines de malware detecteerden als schadelijk. Het is duidelijk dat traditionele antivirussoftware mogelijk niet voldoende bescherming biedt tegen meer geavanceerde bedreigingen.



Figuur 18 Virus total score

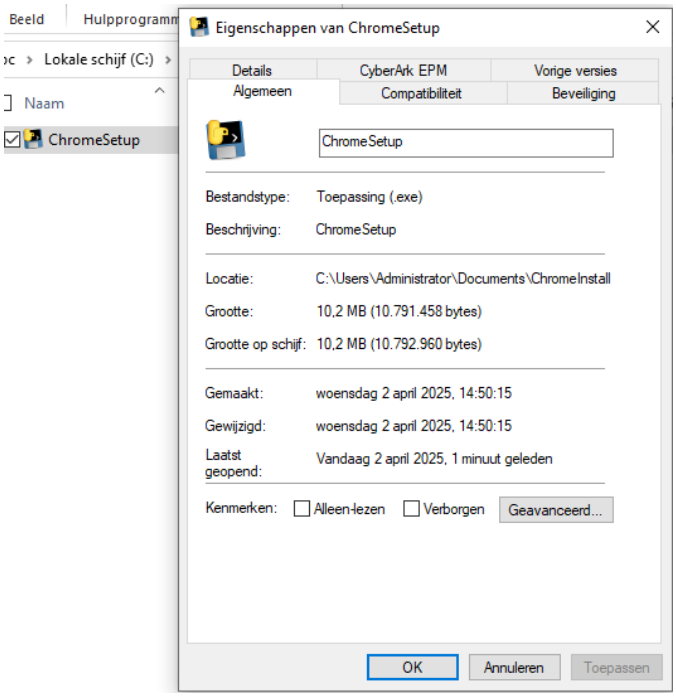
5.3 Uitvoeren

In deze sectie wordt de dropper getest op een Windows 10-machine om te analyseren wat de impact is op het systeem. In Figuur 19 is te zien hoe de *Pictures*-map eruitziet vóór het uitvoeren van de dropper.



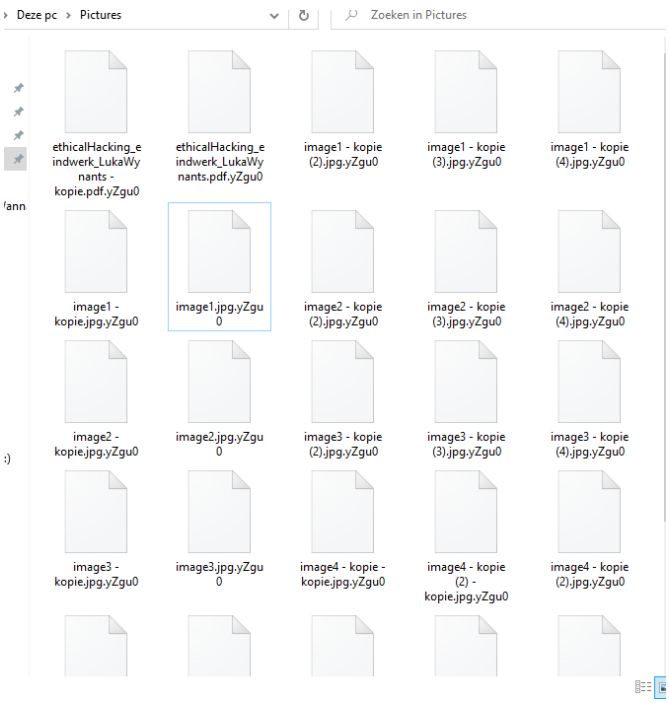
Figuur 19 Pictures folder op windows

Na het uitvoeren van de dropper wordt een map genaamd ChromeInstall aangemaakt. Deze map wordt uitgesloten van Windows Defender-scans. In deze map wordt de geëncodeerde encryptor-executable gedecodeerd en opgeslagen als ‘ChromeSetup’, zoals te zien is in Figuur 20.



Figuur 20 ChromeSetup.exe (encryptor.exe)

De dropper creert automatisch een geplande taak aan die de encryptor uitvoert. Na enkele seconden worden de bestanden in de *Pictures*-map versleuteld. In Figuur 21 is te zien dat alle bestanden nu de extensie “.yZgu0” hebben gekregen, wat aantoont dat ze succesvol zijn versleuteld.



Figuur 21Versleutelde bestanden

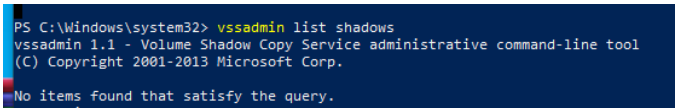
Bij het openen van een versleuteld bestand is de originele inhoud niet meer zichtbaar. De bestanden bevatten onleesbare, versleutelde data (Zie figuur 22).

Figuur 22 Versleuteld bestand

Onderaan elk bestand is een specifieke *footer* toegevoegd. Deze bestaat uit drie delen: de versleutelde data, de AES-sleutel en de IV (initialisatievector). Elk onderdeel is gescheiden door “---” characters gevolgd door een newline. Dankzij deze structuur kan de decryptor de juiste informatie terugvinden om het bestand correct te ontsleutelen. Zie Figuur 23 voor een voorbeeld van zo’n footer.

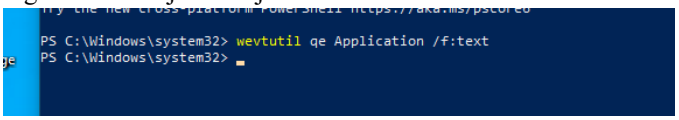
Figuur 23 Versleutelde Footer

Daarnaast zijn de shadow copies van het systeem succesvol verwijderd. Wanneer we het systeem controleren met het commando *“vssadmin list shadows”* zien we dat er geen shadow copies meer aanwezig zijn, wat aantoont dat de malware deze stap heeft uitgevoerd.



Figuur 24 Shadow copies

Ook de Windows Event Logs zijn verwijderd. Het uitvoeren van het commando *“wevtutil qe Application /f:tekst”* toont de application logs, die ervoor vol stonden, als we deze commando uitvoeren na het draaien van de ransomware, wordt er een leeg resultaat getoond, wat bevestigt dat de event logs effectief zijn verwijderd.



Figuur 25 Shadow copies

5.4 Decryptie