aDvens

Security for the greater good

Renseignement sur les menaces
Bulletin du mois de juin 2024

2024-07-04 | **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 présentant un intérêt, en complément de celles déjà publiées.

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

- le maliciel GOMIR exploité par le groupe nord-coréen Kimsuky.
- un état de la menace à quelques jours de l'ouverture des jeux olympiques 2024.

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. CVE-2024-4577

Le 6 juin 2024, PHP a publié plusieurs bulletins de sécurité concernant une vulnérabilité critique dans PHP CGI et publié des correctifs appropriés. Cette vulnérabilité existe depuis 2012, il s'agit d'un contournement de la CVE-2012-1823.

Une erreur dans PHP CGI lorsque celui-ci est installé sur un serveur Windows permet à un attaquant non authentifié, en envoyant des requêtes spécifiquement forgées, d'exécuter du code arbitraire sur le système.

2.1.1. Type de vulnérabilité

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

2.1.2. Risque

- Exécution de code arbitraire

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	Élevé

2.1.4. Produits impactés

- PHP versions 5.X à 8.3.X

2.1.5. Recommandations

Mettre à jour PHP vers la version 8.1.29, 8.2.20 or 8.3.8 ou ultérieure.

Les versions 5.x et 8.0.x ne sont plus prises en charge. Il est recommandé de remplacer les produits vulnérables par d'autres versions disposant d'un correctif.

Des informations complémentaires sont disponibles dans les bulletins de PHP consacrés aux différentes versions du produit.

- Version 8.3.8
- Version 8.2.20
- Version 8.1.29

2.1.6. Preuve de concept

Une preuve de concept est disponible en sources ouvertes.

2.2. CVE-2024-29973

Le 4 juin 2024, Zyxel a publié un bulletin de sécurité afin de corriger la vulnérabilité CVE-2024-29973 touchant plusieurs serveurs NAS.

Un défaut dans le paramètre « setCookie » de Zyxel NAS326 et NAS542 permet à un attaquant non authentifié d'envoyer des requêtes HTTP POST spécialement forgées, dans le but d'exécuter du code arbitraire.

2.2.1. Type de vulnérabilité

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

2.2.2. Risque

- Exécution de code arbitraire

2.2.3. 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

2.2.4. Produit impacté

- Zyxel NAS326 aux versions 5.21(AAZF.16)C0 et antérieures
- Zyxel NAS542 aux versions 5.21(ABAG.13)C0 et antérieures

2.2.5. Recommandations

- Mettre à jour le NAS326 vers la version 5.21(AAZF.17)C0 ou ultérieure.
- Mettre à jour le NAS542 vers la version 5.21(ABAG.14)C0 ou ultérieure.

Zyxel précise que les NAS326 et NAS542 sont normalement des produits dont le support n'est plus fourni depuis le 31 décembre 2023.

- Des informations complémentaires sont disponibles dans le bulletin de Zyxel.

2.2.6. Preuve de concept

Une preuve de concept est disponible en sources ouvertes.

2.3. CVE-2024-28995

Le 5 juin 2024, la société SolarWind publie un bulletin d'alerte concernant une vulnérabilité critique de type "traversée de repertoire" affectant le logiciel SolarWind Serv-U.

Cette vulnérabilité dans SolarWinds Serv-U permet à un attaquant non authentifié d'envoyer des requêtes spécialement forgées, dans le but de porter atteinte à la confidentialité des données sur la machine hôte.

2.3.1. Type de vulnérabilité

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

2.3.2. Risque

- Atteinte à la confidentialité des données

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

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

2.3.4. Produit impacté

- SolarWinds Serv-U dans sa version 15.4.2 HF 1 et versions antérieures

2.3.5. Recommandations

- Mettre à jour SolarWinds Serv-U vers la version 15.4.2 HF 2 ou ultérieure.
- Des informations complémentaires sont disponibles dans le bulletin de SolarWinds.

2.3.6. Preuve de concept

Une preuve de concept est disponible en sources ouvertes.

3. Virologie : analyse d'un échantillon Gomir (APT Kimsuky)

3.1. Une porte dérobée sophistiquée

Découvert en mai 2024, Gomir est une porte dérobée utilisée par l'APT Kimsuky (Corée du Nord). Développé en **Go** au format **ELF 32**, celle-ci est conçue pour être utilisée sur le système d'exploitation **Linux**.

Gomir a été distribué lors d'une campagne de cyber-espionnage ciblant des organisations localisées en Corée du Sud et aux Etats-Unis.

L'analyse de Gomir révèle la sophistication de ce maliciel ainsi que des similitudes avec Gobear (destinée aux environnements Microsoft), une autre porte dérobée intégrée à l'arsenal de l'APT Kimsuky.

3.2. Fonctionnalités

Ci-dessous, les principales fonctionnalités du logiciel malveillant Gomir.

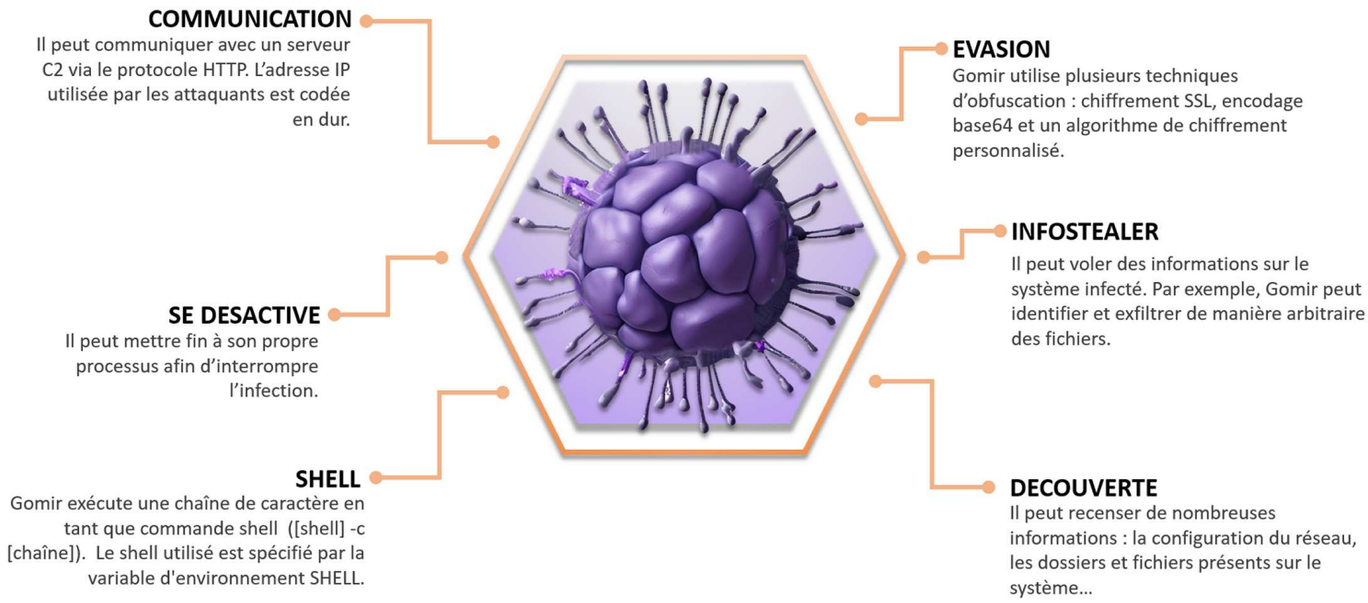
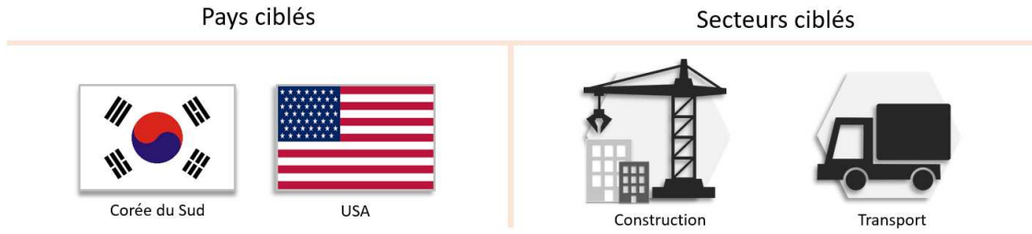


Figure 1. Les fonctionnalités de Gomir : une porte dérobée multifonction.

3.3. Victimologie



3.4. Infectiologie

3.4.1. Chaine d'infection : synthèse

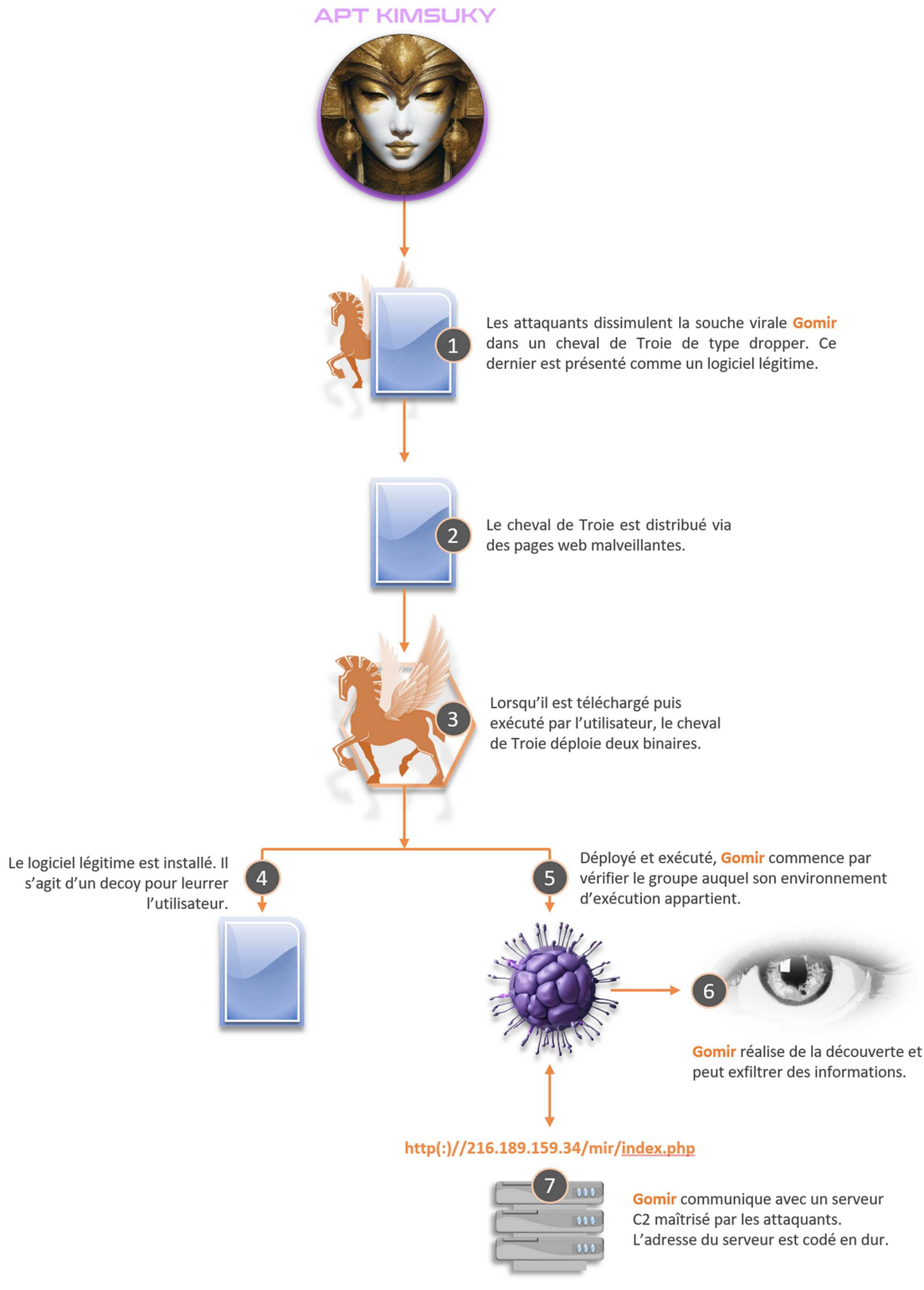


Figure 2. Synthèse infographie de la chaine d'infection.

3.4.2. Chaine d'infection : analyse détaillée

Vecteur d'infection

Le principal vecteur d'infection utilisé par les attaquants est la distribution de chevaux de Troie de type dropper. Ces derniers sont présentés comme étant des logiciels légitimes, ils embarquent deux binaires :

- Le logiciel légitime : celui-ci est déployé puis exécuté pour leurrer (decoy) l'utilisateur.
- La souche virale Gomir : une porte dérobée sophistiquée qui est installée de manière discrète.

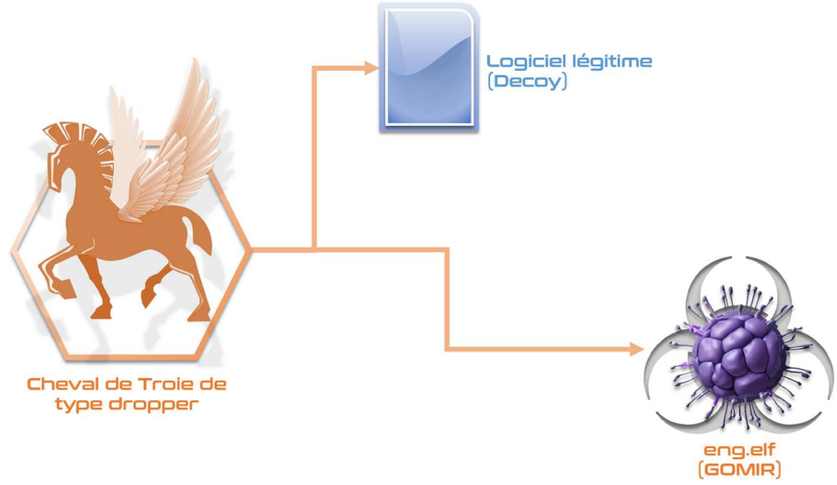


Figure 3. Principal vecteur d'infection : cheval de Troie dropper.

Depuis le début de l'année 2024, différents logiciels légitimes ont été exploités de manière malveillante par l'APT Kimsuky pour dissimuler les souches virales (Troll Stealer,Gobear et Gomir).

- **TrustPKI** et **NX_PRNMAN** de la société *SGA Solutions*.
- **Wizvera VeraPort** de la société *Wizvera*.
- **Humetro** de la société *Humetro Busan Kr*.

Par ailleurs, la chaine d'approvisionnement de *Wizvera VeraPort* a été victime de multiples cyberattaques attribuées au groupe nord coréen APT Lazarus en 2020.

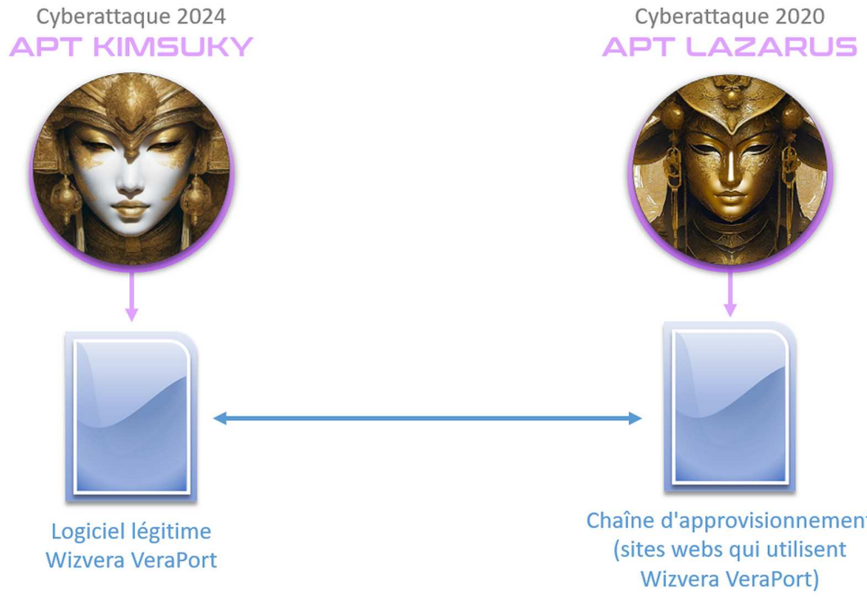


Figure 4. Wizvera VeraPort : une cible répétée de la menace nord-coréenne.

3.4.3. Analyse de la souche virale

Exécution et vérification du groupe

Lorsqu'il est déployé et exécuté par le cheval de Troie de type dropper, Gomir commence par vérifier le groupe auquel appartient son processus d'exécution. Pour cela, c'est la fonction suivante qui est utilisée:

```
if ( syscall_rawSyscallNoError(202, 0, 0, 0) )
```

Ce qui permet de résoudre la fonction **getegid32()**.

Si son processus appartient au groupe 0 (privilège root), la porte dérobée Gomir garantit sa persistance lors de l'installation via **systemd**. Dans le cas contraire, elle sera maintenue via le planificateur de tâches **crontab**.

Installation avec persistance via Systemd

Si son processus appartient au groupe 0 (privilège root), alors Gomir se copie dans le dossier :

```
/var/log/syslogd
```

```
                    01 00 00
082de325 89 54 24 20     MOV        dword ptr [ESP + local_14],EDX
082de329 89 4c 24 24     MOV        dword ptr [ESP + local_10],ECX
082de32d 8d 05 c7        LEA        EAX,[DAT_083471c7]                    = 2Fh    /
         71 34 08
082de333 89 44 24 08     MOV        dword ptr [ESP + local_2c],EAX=>DAT_083471c7    = 2Fh    /
082de337 c7 44 24        MOV        dword ptr [ESP + local_28],0x10
```

Figure 5. GHIDRA - CodeBrowser : l'instruction LEA de la fonction 082de32d charge le registre EAX avec les données DAT_083471c7.

```
                    DAT_083471c7                                    XREF[4]:    FUN_082de2e0:082de32d(*),
                                                                                FUN_082de2e0:082de333(*),
                                                                                FUN_082de2e0:082de350(*),
                                                                                FUN_082de2e0:082de356(*)
083471c7 2f              ??         2Fh    /
083471c8 76              ??         76h    v
083471c9 61              ??         61h    a
083471ca 72              ??         72h    r
083471cb 2f              ??         2Fh    /
083471cc 6c              ??         6Ch    l
083471cd 6f              ??         6Fh    o
083471ce 67              ??         67h    g
083471cf 2f              ??         2Fh    /
083471d0 73              ??         73h    s
083471d1 79              ??         79h    y
083471d2 73              ??         73h    s
083471d3 6c              ??         6Ch    l
083471d4 6f              ??         6Fh    o
083471d5 67              ??         67h    g
083471d6 64              ??         64h    d
```

Figure 6. GHIDRA - CodeBrowser : les données DAT_083471c7, celles-ci correspondent à /var/log/syslogd.

Gomir crée un fichier dans le dossier suivant :

```
/etc/systemd/system/syslogd.service
```

Ce dernier contient les informations ci-dessous :

```
[Unit]
After=network.target
Description=syslogd
[Service]
ExecStart=/bin/sh -c "/var/log/syslogd"
Restart=always
[Install]
WantedBy=multi-user.target
```

```
082de3d7 c6 44 24 1f 01    MOV     byte ptr [ESP + local_15],0x1
082de3dc 8d 0d 4b 8f 35 08 LEA     ECX,[DAT_08358f4b]
082de3e2 f7 d9             NEG     ECX
082de3e4 90                NOP
```

Figure 7. GHIDRA - CodeBrowser : l'instruction LEA de la fonction 082de3dc charge le registre ECX avec les données DAT_08358f4b.

```
                    DAT_08358f4b                                XREF[3]:  FUN_082de2e0:082de3dc(*),
                                                                          FUN_082de2e0:082de3f4(*),
                                                                          FUN_082de2e0:082de3fa(*)
08358f4b 0a          ??    0Ah
08358f4c 5b          ??    5Bh    [
08358f4d 55          ??    55h    U
08358f4e 6e          ??    6Eh    n
08358f4f 69          ??    69h    i
08358f50 74          ??    74h    t
08358f51 5d          ??    5Dh    ]
08358f52 0a          ??    0Ah
08358f53 41          ??    41h    A
08358f54 66          ??    66h    f
08358f55 74          ??    74h    t
08358f56 65          ??    65h    e
08358f57 72          ??    72h    r
08358f58 3d          ??    3Dh    =
08358f59 6e          ??    6Eh    n
08358f5a 65          ??    65h    e
08358f5b 74          ??    74h    t
08358f5c 77          ??    77h    w
08358f5d 6f          ??    6Fh    o
08358f5e 72          ??    72h    r
08358f5f 6b          ??    6Bh    k
08358f60 2e          ??    2Eh    .
08358f61 74          ??    74h    t
08358f62 61          ??    61h    a
08358f63 72          ??    72h    r
08358f64 67          ??    67h    g
08358f65 65          ??    65h    e
08358f66 74          ??    74h    t
08358f67 0a          ??    0Ah
08358f68 44          ??    44h    D
08358f69 65          ??    65h    e
08358f6a 73          ??    73h    s
08358f6b 63          ??    63h    c
08358f6c 72          ??    72h    r
08358f6d 69          ??    69h    i
08358f6e 70          ??    70h    p
08358f6f 74          ??    74h    t
08358f70 69          ??    69h    i
08358f71 6f          ??    6Fh    o
08358f72 6e          ??    6Eh    n
08358f73 3d          ??    3Dh    =
08358f74 73          ??    73h    s
08358f75 79          ??    79h    y
08358f76 73          ??    73h    s
08358f77 6c          ??    6Ch    l
08358f78 6f          ??    6Fh    o
08358f79 67          ??    67h    g
08358f7a 64          ??    64h    d
08358f7b 0a          ??    0Ah
08358f7c 0a          ??    0Ah
```

Figure 8. GHIDRA - CodeBrowser : les données DAT_08358f4b, celles-ci correspondent au contenu de l'artéfact syslogd.service

Le fichier **syslogd** permet sur les environnements linux de configurer la journalisation des évènements. Afin que les modifications soient prises en compte, Gomir exécute les commandes suivantes :

```
${SHELL} -c systemctl daemon-reload
${SHELL} -c systemctl reenable syslogd
${SHELL} -c systemctl start syslogd
```

Lorsque ce service est exécuté, Gomir interrompt son processus et se supprime pour ne pas laisser de trace.

Installation avec persistance via crontab

Si son processus n'appartient pas au groupe 0 (privilège root), Gomir utilise crontab pour établir sa persistance. Pour cela, le fichier **cron.txt** est créé dans le dossier où Gomir est présent.

Le fichier **cron.txt** contient le code suivant :

```
@reboot [Chemin_du_processus]
```

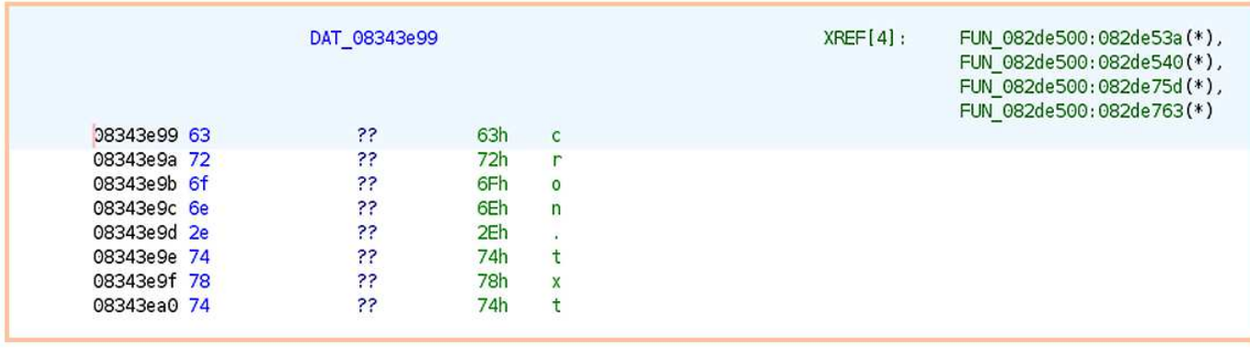


Figure 9. GHIDRA - CodeBrowser : l'intitulé de l'artéfact (cron.txt) est codé en dur dans les données DAT_08343e99.

Gomir tente de répertorier toutes les entrées crontab existantes, puis les concatène dans le fichier **cron.txt**. Après le chargement de la nouvelle configuration de crontab (commande ci-dessous), le fichier est supprimé.

```
/bin/sh -c crontab -l
${SHELL} -c crontab cron.txt
```

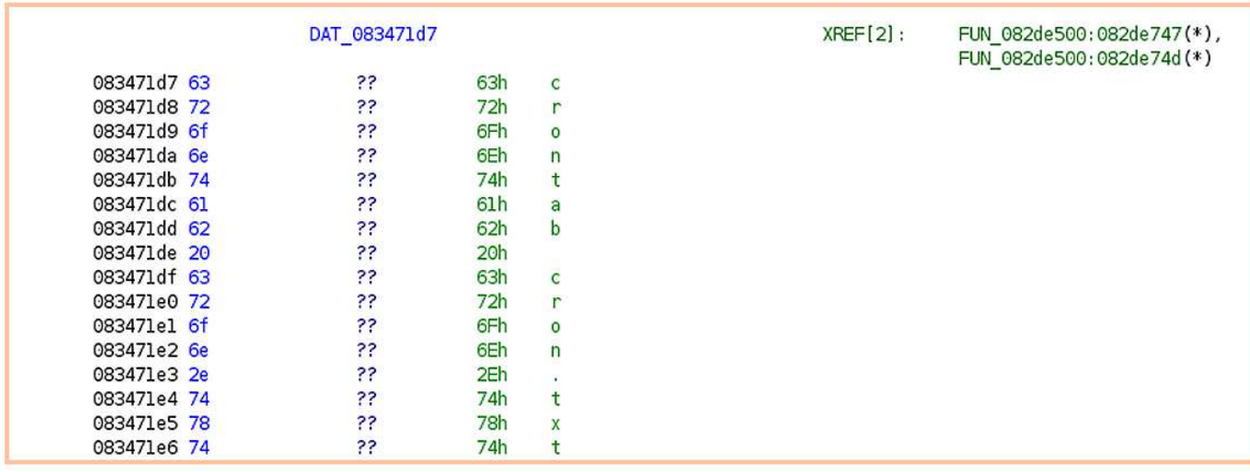


Figure 10. GHIDRA - CodeBrowser : la chaine de caractère de la commande (crontab cron.txt) est codée en dur dans les données DAT_083471d7.

Identifiant de la victime

Lors de l'infection du système, Gomir génère un identifiant de la victime via la fonction **generate_infection_id** :

```
def generate_infection_id(hostname, username): hexdigest = hashlib.md5(hostname + username).hexdigest()
return "g-" + hexdigest[:10]
```

Cet identifiant est utilisé lors de la communication avec le serveur C2.

Communication avec le serveur C2

Gomir communique avec un serveur C2 dont l'adresse est codée en dure. La communication est réalisée via des requêtes HTTP.

```
http(:)//216(.)189.159.34/mir/index.php
```

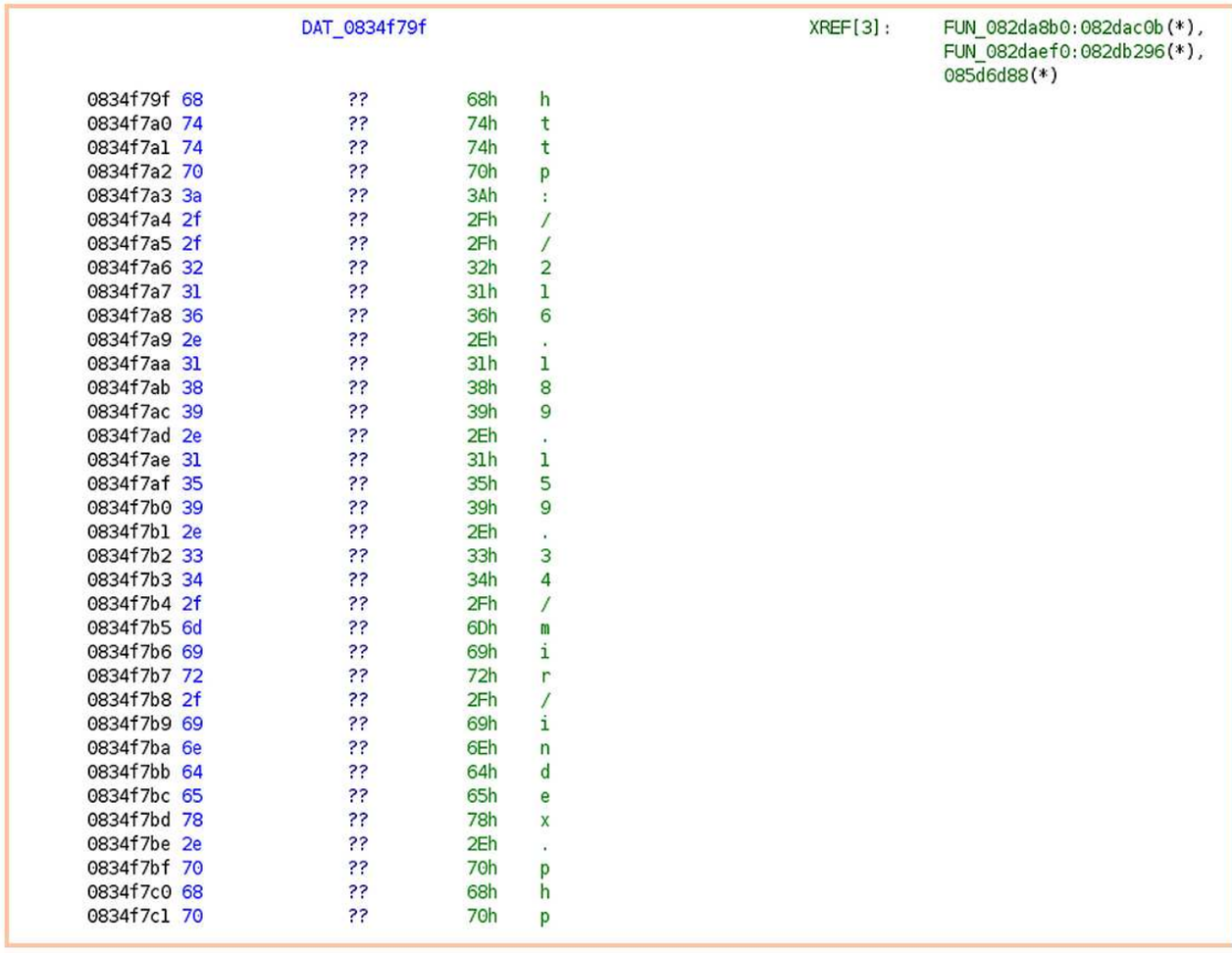


Figure 11. GHIDRA - CodeBrowser : l'adresse du serveur C2 est codée en dur dans la souche virale. Données DAT_0834f79f.

Afin de recevoir de nouvelles instructions, Gomir envoie une requête HTTP de type POST au serveur C2. Le corps de la requête est structuré de la manière suivante :

```
a\w{9}=2&b\w{9}=[Identifiant victime]1&c\w{9}=
```

Instructions C2

Gomir peut recevoir 17 instructions du serveur C2 :

OEPARATION	INSTRUCTIONS
1	Suspend la communication avec le serveur C&C pour une durée arbitraire.
2	Exécute une chaîne de caractères en tant que commande shell ([shell] -c [chaine]). Le shell utilisé est spécifié par la variable d'environnement "SHELL". Si cette variable n'est pas disponible un shell de secours est configuré via l'opération 10.
3	Indique le répertoire de travail actuel.
4	Modifie le répertoire de travail actuel et précise le nouveau chemin d'accès du répertoire de travail.
5	Teste la connectivité TCP de points de terminaison du réseau.
6	Arrête son processus : interrompant ainsi la porte dérobée Gomir
7	Indique le chemin de l'exécutable de son propre processus.
8	Collecte des statistiques sur une arborescence de répertoires et génère des rapports sur le nombre total de sous-répertoires et de fichiers ainsi que les tailles totales des fichiers
9	Collecte des détails de configuration de l'ordinateur compromis (nom d'hôte, nom d'utilisateur,CPU, RAM ainsi que des informations sur la configuration du réseau).
10	Configure un shell de secours à utiliser lors de l'exécution de l'opération 02. La valeur de configuration initiale est "/bin/sh".
11	Configure une page de codes pour interpréter la sortie de la commande shell de l'opération 02.
12	Suspend la communication avec le serveur C2 jusqu'à une date/heure arbitraire.
13	Répond par le message "Not implemented on Linux!"
14	Démarre un proxy inversé en se connectant à un terminal de contrôle arbitraire. La communication avec ce terminal de contrôle est chiffrée à l'aide du protocole SSL. La porte dérobée agit comme un client proxy. Cela permet à l'attaquant distant d'initier des connexions à des terminaux arbitraires sur le réseau de la victime.
15	Indique les terminaux de contrôle du proxy inverse.
30	Crée un fichier arbitraire.
31	Exfiltre un fichier arbitraire.

Eléments Github

Les attaquants semblent exploiter via Gomir plusieurs éléments issus de différents projets *Github*.

Projet : klauspost/cpuid

083f2b...	ds "github.com/klauspost/cpuid.glob..func1"	"github.com/klauspost/cpuid.glob..func1"	string
083f2b...	ds "github.com/klauspost/cpuid.init.0"	"github.com/klauspost/cpuid.init.0"	string
083f2b...	ds "github.com/klauspost/cpuid.initCPU"	"github.com/klauspost/cpuid.initCPU"	string
083f2b...	ds "github.com/klauspost/cpuid.Detect"	"github.com/klauspost/cpuid.Detect"	string
083f2b...	ds "github.com/klauspost/cpuid.(*flagSet).unset"	"github.com/klauspost/cpuid.(*flagSet).unset"	string
083f2bff	ds "github.com/klauspost/cpuid.CPUInfo.FeatureSet"	"github.com/klauspost/cpuid.CPUInfo.FeatureSet"	string
083f2c...	ds "github.com/klauspost/cpuid.(*flagSet).nEnabled"	"github.com/klauspost/cpuid.(*flagSet).nEnabled"	string
083f2c...	ds "github.com/klauspost/cpuid.(*CPUInfo).frequencies"	"github.com/klauspost/cpuid.(*CPUInfo).frequencies"	string
083f2c...	ds "github.com/klauspost/cpuid.maxFunctionID"	"github.com/klauspost/cpuid.maxFunctionID"	string
083f2ccd	ds "github.com/klauspost/cpuid.ParseFeature"	"github.com/klauspost/cpuid.ParseFeature"	string
083f2cf5	ds "github.com/klauspost/cpuid.flagSet.Strings"	"github.com/klauspost/cpuid.flagSet.Strings"	string
083f2d...	ds "github.com/klauspost/cpuid.(*flagSet).inSet"	"github.com/klauspost/cpuid.(*flagSet).inSet"	string
083f2d...	ds "github.com/klauspost/cpuid.brandName"	"github.com/klauspost/cpuid.brandName"	string
083f2d...	ds "github.com/klauspost/cpuid.maxExtendedFunction"	"github.com/klauspost/cpuid.maxExtendedFunction"	string
083f2d...	ds "github.com/klauspost/cpuid.threadsPerCore"	"github.com/klauspost/cpuid.threadsPerCore"	string
083f2d...	ds "github.com/klauspost/cpuid.logicalCores"	"github.com/klauspost/cpuid.logicalCores"	string
083f2df2	ds "github.com/klauspost/cpuid.familyModel"	"github.com/klauspost/cpuid.familyModel"	string
083f2e...	ds "github.com/klauspost/cpuid.physicalCores"	"github.com/klauspost/cpuid.physicalCores"	string
083f2e...	ds "github.com/klauspost/cpuid.vendorID"	"github.com/klauspost/cpuid.vendorID"	string
083f2e...	ds "github.com/klauspost/cpuid.cacheLine"	"github.com/klauspost/cpuid.cacheLine"	string
083f2e...	ds "github.com/klauspost/cpuid.(*CPUInfo).cacheSize"	"github.com/klauspost/cpuid.(*CPUInfo).cacheSize"	string
083f2e...	ds "github.com/klauspost/cpuid.(*CPUInfo).Has"	"github.com/klauspost/cpuid.(*CPUInfo).Has"	string
083f2e...	ds "github.com/klauspost/cpuid.hasSGX"	"github.com/klauspost/cpuid.hasSGX"	string
083f2f07	ds "github.com/klauspost/cpuid.support"	"github.com/klauspost/cpuid.support"	string
083f2f2a	ds "github.com/klauspost/cpuid.(*flagSet).setIf"	"github.com/klauspost/cpuid.(*flagSet).setIf"	string
083f2f56	ds "github.com/klauspost/cpuid.(*flagSet).set"	"github.com/klauspost/cpuid.(*flagSet).set"	string
083f2f80	ds "github.com/klauspost/cpuid.valAsString"	"github.com/klauspost/cpuid.valAsString"	string
083f2fa7	ds "github.com/klauspost/cpuid.addInfo"	"github.com/klauspost/cpuid.addInfo"	string
083f2fca	ds "github.com/klauspost/cpuid.FeatureID.String"	"github.com/klauspost/cpuid.FeatureID.String"	string
083f2ff6	ds "github.com/klauspost/cpuid.init"	"github.com/klauspost/cpuid.init"	string
083f30...	ds "github.com/klauspost/cpuid.CombineFeatures"	"github.com/klauspost/cpuid.CombineFeatures"	string
083f30...	ds "github.com/klauspost/cpuid.map.init.0"	"github.com/klauspost/cpuid.map.init.0"	string
083f30...	ds "github.com/klauspost/cpuid.asmCpuid"	"github.com/klauspost/cpuid.asmCpuid"	string
083f30...	ds "github.com/klauspost/cpuid.asmCpuidex"	"github.com/klauspost/cpuid.asmCpuidex"	string
083f30...	ds "github.com/klauspost/cpuid.asmXgetbv"	"github.com/klauspost/cpuid.asmXgetbv"	string
083f30...	ds "github.com/klauspost/cpuid.asmRdtscpAsm"	"github.com/klauspost/cpuid.asmRdtscpAsm"	string

Figure 12. Elément identifié.

- Source : https://github.com/klauspost/cpuid
- Utilisation : une bibliothèque Golang qui permet de récupérer des informations sur le microprocesseur.

Projet : pbnjay/memory

083f30...	ds "github.com/klauspost/cpuid.asmRdtscpAsm"	"github.com/klauspost/cpuid.asmRdtscpAsm"	string
083f30fe	ds "github.com/klauspost/cpuid.asmDarwinHasAVX512"	"github.com/klauspost/cpuid.asmDarwinHasAVX512"	string
083f31...	ds "github.com/pbnjay/memory.sysTotalMemory"	"github.com/pbnjay/memory.sysTotalMemory"	string
083f31...	ds "github.com/armon/go-socks5.NoAuthAuthenticator.GetCode"	"github.com/armon/go-socks5.NoAuthAuthenticator.GetCode"	string
083f31...	ds "github.com/armon/go-socks5.NoAuthAuthenticator.Authen...	"github.com/armon/go-socks5.NoAuthAuthenticator.Authenticate"	string

Figure 13. Elément identifié.

- Source : https://github.com/pbnjay/memory
- Utilisation : une bibliothèque Golang qui permet de récupérer des informations sur la mémoire du système.

Projet : go-humanize

083f18...	ds "github.com/saintfish/chardet.NewTextDetector"	"github.com/saintfish/chardet.NewTextDetector"	string
083f18...	ds "github.com/saintfish/chardet.(*Detector).DetectBest"	"github.com/saintfish/chardet.(*Detector).DetectBest"	string
083f28...	ds "github.com/dustin/go-humanize.logn"	"github.com/dustin/go-humanize.logn"	string
083f28...	ds "github.com/dustin/go-humanize.humanateBytes"	"github.com/dustin/go-humanize.humanateBytes"	string
083f290f	ds "github.com/dustin/go-humanize.revfmap"	"github.com/dustin/go-humanize.revfmap"	string
083f29...	ds "github.com/dustin/go-humanize.init.0"	"github.com/dustin/go-humanize.init.0"	string
083f29...	ds "github.com/dustin/go-humanize.map.init.2"	"github.com/dustin/go-humanize.map.init.2"	string
083f29...	ds "github.com/dustin/go-humanize.init"	"github.com/dustin/go-humanize.init"	string
083f2b...	ds "github.com/klauspost/cpuid.glob..func1"	"github.com/klauspost/cpuid.glob..func1"	string
083f2b...	ds "github.com/klauspost/cpuid.init.0"	"github.com/klauspost/cpuid.init.0"	string

Figure 14. Elément identifié.

- Source : https://github.com/dustin/go-humanize
- Utilisation : ensemble de fonctions qui permettent de formater des informations sur la taille du système de fichiers.

Ci-dessous, les librairies provenant de divers dépôts github qui sont intégrées dans la souche virale Gomir :

083f4f18	ds	"github.com/hashicorp/yamux.(*Stream).SetWriteDeadline"	"github.com/hashicorp/yamux.(*Stream).SetWriteDeadline" string
083f4f4e	ds	"github.com/hashicorp/yamux.glob..func1"	"github.com/hashicorp/yamux.glob..func1" string
083f4f75	ds	"github.com/hashicorp/yamux.init"	"github.com/hashicorp/yamux.init" string
083f4f95	ds	"type:.eq.github.com/hashicorp/yamux.NetError"	"type:.eq.github.com/hashicorp/yamux.NetError" string
083f4fc2	ds	"type:.eq.github.com/hashicorp/yamux.Config"	"type:.eq.github.com/hashicorp/yamux.Config" string
083f4fed	ds	"github.com/hashicorp/yamux.(*header).String"	"github.com/hashicorp/yamux.(*header).String" string
083f50...	ds	"github.com/hashicorp/yamux.(*Stream).closeTimeout-fm"	"github.com/hashicorp/yamux.(*Stream).closeTimeout-fm" string
083f50...	ds	"github.com/hashicorp/yamux.(*Session).Accept"	"github.com/hashicorp/yamux.(*Session).Accept" string
083f59...	ds	"github.com/dustin/go-humanize.Bytes"	"github.com/dustin/go-humanize.Bytes" string
083f59...	ds	"github.com/pbnjay/memory.TotalMemory"	"github.com/pbnjay/memory.TotalMemory" string
083fbe...	ds	"github.com/saintfish/chardet/2022.go"	"github.com/saintfish/chardet/2022.go" string
083fbe...	ds	"github.com/saintfish/chardet/detector.go"	"github.com/saintfish/chardet/detector.go" string
083fbe...	ds	"github.com/saintfish/chardet/multi_byte.go"	"github.com/saintfish/chardet/multi_byte.go" string
083fbebf	ds	"github.com/saintfish/chardet/recognizer.go"	"github.com/saintfish/chardet/recognizer.go" string
083fbe...	ds	"github.com/saintfish/chardet/single_byte.go"	"github.com/saintfish/chardet/single_byte.go" string
083fbf16	ds	"github.com/saintfish/chardet/unicode.go"	"github.com/saintfish/chardet/unicode.go" string
083fbf3e	ds	"github.com/saintfish/chardet/utf8.go"	"github.com/saintfish/chardet/utf8.go" string
083fc7...	ds	"github.com/dustin/go-humanize/bytes.go"	"github.com/dustin/go-humanize/bytes.go" string
083fc7...	ds	"github.com/dustin/go-humanize/si.go"	"github.com/dustin/go-humanize/si.go" string
083fc7fd	ds	"github.com/dustin/go-humanize/bigbytes.go"	"github.com/dustin/go-humanize/bigbytes.go" string
083fc8...	ds	"github.com/klauspost/cpuid/cpuid.go"	"github.com/klauspost/cpuid/cpuid.go" string
083fc8...	ds	"github.com/klauspost/cpuid/detect_x86.go"	"github.com/klauspost/cpuid/detect_x86.go" string
083fc8...	ds	"github.com/klauspost/cpuid/featureid_string.go"	"github.com/klauspost/cpuid/featureid_string.go" string
083fc8...	ds	"github.com/klauspost/cpuid/cpuid_386.s"	"github.com/klauspost/cpuid/cpuid_386.s" string
083fc8...	ds	"github.com/pbnjay/memory/memory_linux.go"	"github.com/pbnjay/memory/memory_linux.go" string
083fc9...	ds	"github.com/armon/go-socks5/auth.go"	"github.com/armon/go-socks5/auth.go" string
083fc9...	ds	"github.com/armon/go-socks5/request.go"	"github.com/armon/go-socks5/request.go" string
083fc9...	ds	"github.com/armon/go-socks5/resolver.go"	"github.com/armon/go-socks5/resolver.go" string
083fc9...	ds	"github.com/armon/go-socks5/ruleset.go"	"github.com/armon/go-socks5/ruleset.go" string
083fc9...	ds	"github.com/armon/go-socks5/socks5.go"	"github.com/armon/go-socks5/socks5.go" string
083fc9dc	ds	"github.com/hashicorp/yamux/addr.go"	"github.com/hashicorp/yamux/addr.go" string
083fc9ff	ds	"github.com/hashicorp/yamux/const.go"	"github.com/hashicorp/yamux/const.go" string
083fca...	ds	"github.com/hashicorp/yamux/mux.go"	"github.com/hashicorp/yamux/mux.go" string
083fca...	ds	"github.com/hashicorp/yamux/session.go"	"github.com/hashicorp/yamux/session.go" string
083fca...	ds	"github.com/hashicorp/yamux/util.go"	"github.com/hashicorp/yamux/util.go" string
083fca...	ds	"github.com/hashicorp/yamux/stream.go"	"github.com/hashicorp/yamux/stream.go" string

Figure 15. Intégration d'éléments de projets Github dans Gomir : example 1.

083f30...	ds	"github.com/klauspost/cpuid.asmRdtscpAsm"	"github.com/klauspost/cpuid.asmRdtscpAsm" string
083f30fe	ds	"github.com/klauspost/cpuid.asmDarwinHasAVX512"	"github.com/klauspost/cpuid.asmDarwinHasAVX512" string
083f31...	ds	"github.com/pbnjay/memory.sysTotalMemory"	"github.com/pbnjay/memory.sysTotalMemory" string
083f31...	ds	"github.com/armon/go-socks5.NoAuthAuthenticator.GetCode"	"github.com/armon/go-socks5.NoAuthAuthenticator.GetCode" string
083f31...	ds	"github.com/armon/go-socks5.NoAuthAuthenticator.Authen...	"github.com/armon/go-socks5.NoAuthAuthenticator.Authenticate" string
083f31...	ds	"github.com/armon/go-socks5.UserPassAuthenticator.GetC...	"github.com/armon/go-socks5.UserPassAuthenticator.GetCode" string
083f32...	ds	"github.com/armon/go-socks5.UserPassAuthenticator.Auth...	"github.com/armon/go-socks5.UserPassAuthenticator.Authentica... string
083f32...	ds	"github.com/armon/go-socks5.(*Server).authenticate"	"github.com/armon/go-socks5.(*Server).authenticate" string
083f32...	ds	"github.com/armon/go-socks5.noAcceptableAuth"	"github.com/armon/go-socks5.noAcceptableAuth" string
083f32...	ds	"github.com/armon/go-socks5.readMethods"	"github.com/armon/go-socks5.readMethods" string
083f32...	ds	"github.com/armon/go-socks5.(*AddrSpec).String"	"github.com/armon/go-socks5.(*AddrSpec).String" string
083f330f	ds	"github.com/armon/go-socks5.AddrSpec.Address"	"github.com/armon/go-socks5.AddrSpec.Address" string
083f33...	ds	"github.com/armon/go-socks5.NewRequest"	"github.com/armon/go-socks5.NewRequest" string
083f33...	ds	"github.com/armon/go-socks5.(*Server).handleRequest"	"github.com/armon/go-socks5.(*Server).handleRequest" string
083f33...	ds	"github.com/armon/go-socks5.(*Server).handleConnect"	"github.com/armon/go-socks5.(*Server).handleConnect" string
083f33...	ds	"github.com/armon/go-socks5.(*Server).handleConnect.fu...	"github.com/armon/go-socks5.(*Server).handleConnect.func4" string
083f34...	ds	"github.com/armon/go-socks5.(*Server).handleConnect.fu...	"github.com/armon/go-socks5.(*Server).handleConnect.func3" string
083f34...	ds	"github.com/armon/go-socks5.(*Server).handleConnect.fu...	"github.com/armon/go-socks5.(*Server).handleConnect.func2" string
083f34...	ds	"github.com/armon/go-socks5.(*Server).handleBind"	"github.com/armon/go-socks5.(*Server).handleBind" string
083f34...	ds	"github.com/armon/go-socks5.(*Server).handleAssociate"	"github.com/armon/go-socks5.(*Server).handleAssociate" string
083f34...	ds	"github.com/armon/go-socks5.readAddrSpec"	"github.com/armon/go-socks5.readAddrSpec" string
083f34ff	ds	"github.com/armon/go-socks5.sendReply"	"github.com/armon/go-socks5.sendReply" string
083f35...	ds	"github.com/armon/go-socks5.proxy"	"github.com/armon/go-socks5.proxy" string
083f35...	ds	"github.com/armon/go-socks5.DNSResolver.Resolve"	"github.com/armon/go-socks5.DNSResolver.Resolve" string
083f35...	ds	"github.com/armon/go-socks5.(*PermitCommand).Allow"	"github.com/armon/go-socks5.(*PermitCommand).Allow" string
083f35...	ds	"github.com/armon/go-socks5.New"	"github.com/armon/go-socks5.New" string
083f35...	ds	"github.com/armon/go-socks5.PermitAll"	"github.com/armon/go-socks5.PermitAll" string
083f35...	ds	"github.com/armon/go-socks5.(*Server).ServeConn"	"github.com/armon/go-socks5.(*Server).ServeConn" string
083f36...	ds	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logg...	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logger).Prin... string
083f36...	ds	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logg...	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logger).Prin... string
083f36...	ds	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logg...	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logger).Prin... string
083f37...	ds	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logg...	"github.com/armon/go-socks5.(*Server).ServeConn.(*Logger).Prin... string
083f37...	ds	"github.com/armon/go-socks5.(*Server).ServeConn.func5"	"github.com/armon/go-socks5.(*Server).ServeConn.func5" string
083f37...	ds	"github.com/armon/go-socks5.(*Server).handleConnect.fu...	"github.com/armon/go-socks5.(*Server).handleConnect.func1" string
083f37...	ds	"github.com/armon/go-socks5.init"	"github.com/armon/go-socks5.init" string
083f37...	ds	"type:.eq.github.com/armon/go-socks5.Request"	"type:.eq.github.com/armon/go-socks5.Request" string

Figure 16. Intégration d'éléments de projets Github dans Gomir : example 2.

3.5. Lignée virologique

3.5.1. Des similitudes au sein l'arsenal de Kimsuky

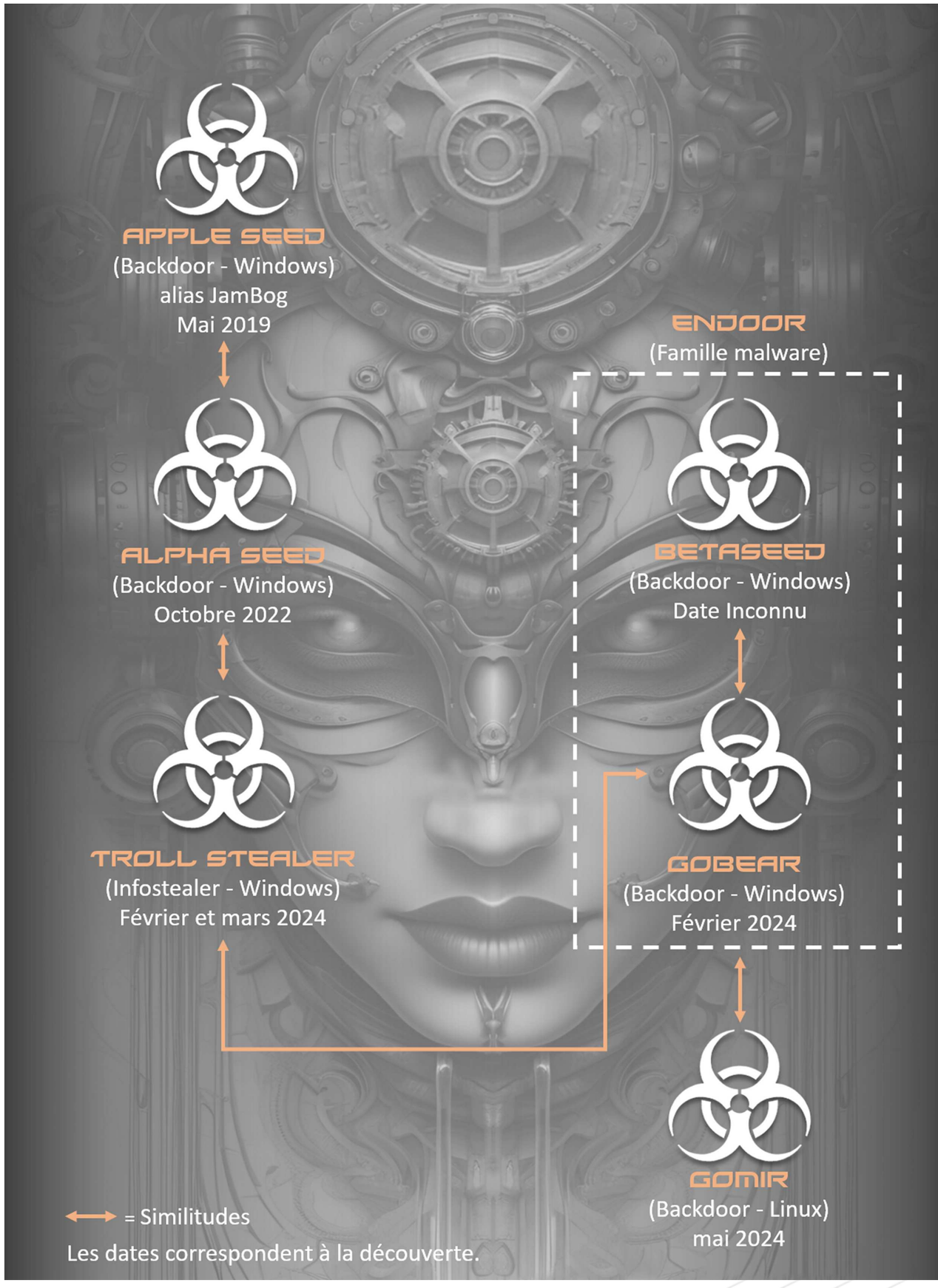


Figure 17. Infographie non exhaustive des similitudes avec l'arsenal d'APT Kimsuky.

Plusieurs similitudes ont été identifiées entre différentes souches virales appartenant à l'arsenal de l'APT Kimsuky.

Troll Stealer et Apple Seed

- L'emplacement et l'intitulé de la souche virale sont identiques. Par ailleurs, le nommage du mutex et plusieurs fonctions sont semblables.

Troll Stealer et Alpha Seed

- Le chiffrement et le déchiffrement des données sont identiques

Gobear et Gomir

- Les deux souches sont structurellement presque identiques.

Gobear et Troll Stealer

- Ils ont le même certificat *D2innovation Co.,LTD*

Gobear et Betaseed

- Certaines fonctions ont le même intitulé.

Apple Seed et Alpha Seed

- Alpha Seed est une version développée en Go d'Apple Seed.

3.5.2. Gomir : déployé par Chalubo en 2023 ?

Gomir a fait l'objet d'analyses et publications au cours du mois de mai 2024. La date exacte de l'émergence ne semble pas connue. Cependant, une information intéressante a été relevée lors d'une recherche en sources ouvertes : Gomir aurait été déployé par le cheval de Troie Chalubo lors d'une cyberattaque d'envergure en octobre 2023.

Selon un rapport publié par *Black Lotus Lab* de la société *Lumen Technologies*, plus de 600 000 routeurs aux États-Unis d'Amérique ont été mis hors-service lors d'une cyberattaque qui s'est déroulée du 25 ou 27 octobre 2023. Dans un premier temps, des attaquants inconnus ont utilisé le cheval de Troie Chalubo comme logiciel malveillant de primo-infection. Des implants additionnels ont été déployés sur les routeurs pour réaliser une opération de sabotage. Parmi les souches additionnelles identifiées, l'empreinte de l'une d'elles correspond au SHA256 de Gomir.

L'empreinte (sha256) 30584f13c0a9d0c86562c803de350432d5a0607a06b24481ad4d92cdf7288213 correspond à l'implant Gomir déployé lors de campagnes d'attaque en 2023 et 2024.

Il est possible que Gomir soit **plus ancien qu'il n'y parait** et que son utilisation **ne se limite pas qu'au cyber-espionnage mais aussi au cyber-sabotage**.

D'autres empreintes (Gobear et Troll Stealer) ont aussi été identifiées lors du cyber-sabotage qui s'est déroulé en octobre 2023.

L'hypothése que l'**APT Kimsuky soit potentiellement l'auteur de ce sabotage est considéré comme probable**.

3.6. APT Kimsuky - Evolution de TTP

3.6.1. Chevaux de Troie avec decoy

Depuis le début de l'année 2024, l'APT Kimsuky utiliserait **une nouvelle technique pour déployer son arsenal**. Les attaquants forgent et distribuent via des pages web malveillantes **des chevaux de Troie de type dropper**: ces derniers déploient un decoy (application légitime) et la souche virale sur le système de l'utilisateur. **Aucun courriel d'hameçonnage** ne semble avoir été utilisé pour le déploiement de Troll Stealer, Gobear et Gomir lors des campagnes de cyber-espionnage à l'encontre de la Corée du Sud.

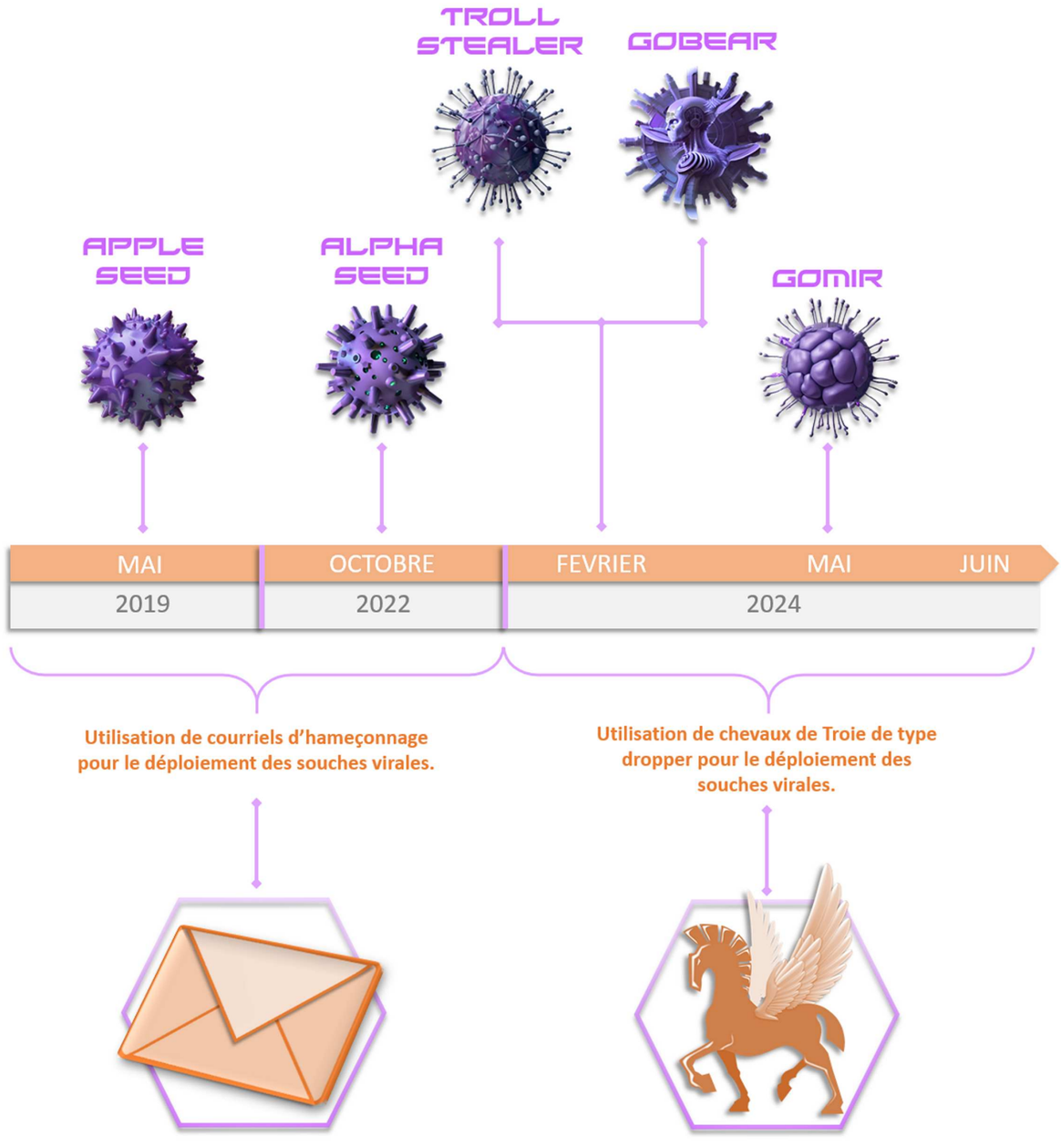


Figure 18. APT Kimsuky : évolution de TTP.

3.7. APT Kimsuky - Modèle diamant

L'APT Kimsuky (alias APT 43, TA406, Thallium, Black Banshee, Velvet Chollima…) est une menace avancée et persistante d'origine nord-coréenne

Figure 19. Modèle diamant de l'APT Kimsuky.

3.8. MITRE ATT&CK

RESOURCE DEVELOPMENT

T1583.001 Acquire Infrastructure: Domains. T1583.004 Acquire Infrastructure: Server.
T1587.001 Develop Capabilities: Malware. T1587.002 Develop Capabilities: Code Signing Certificates.
T1608.001 Stage capabilities: Upload Malware. T1608.003 Stage Capabilities: Install Digital Certificate.

INITIAL ACCESS

T1566.003 Phishing: Spearphishing via Service.

EXECUTION

T1059.004 Command and Scripting Interpreter: Unix Shell (Linux). T1204.001 User Execution: Malicious Link. T1204.002 User Execution: Malicious File. T1053.005 Scheduled Task/Job: Scheduled Task. T1053.003 Scheduled Task/Job: Cron.

PERSISTENCE

T1547 Boot or Logon Autostart Execution. T1053.005 Scheduled Task/Job: Scheduled Task. T1053.003 Scheduled Task/Job: Cron.
T1043.002 Create or Modify System Process: Systemd Service. T1546.016 Event Triggered Execution: Installer Packages.

PRIVILEGE ESCALATION

T1547 Boot or Logon Autostart Execution. T1053.005 Scheduled Task/Job: Scheduled Task. T1053.003 Scheduled Task/Job: Cron.
T1043.002 Create or Modify System Process: Systemd Service. T1546.016 Event Triggered Execution: Installer Packages.

DEFENSE EVASION

T1140 Deobfuscate / Decode Files or Information. T1564.001 Hide Artifacts: Hidden Files and Directories. T1070.004 Indicator Removal: File Deletion. T1036.008 Masquerading: Masquerade File Type. T1027 Obfuscated Files or Information. T1027.002 Obfuscated Files or Information: Software Packing. T1027.005 Obfuscated Files or Information: Indicator Removal from Tools. T1027.010 Obfuscated Files or Information: Command Obfuscation.T1027.009 Obfuscated Files or Information: Embedded Payloads.T1027.013 Obfuscated Files or Information: Encrypted/Encoded File. T1553.002 Subvert Trust Controls: Code Signing.

DISCOVERY

T1082 System Information Discovery. T1518.001 Software Discovery: Security Software Discovery. T1135 Network Share Discovery. T1120 Peripheral Device Discovery. T1518.016 System Network Configuration Discovery. T15049 System Network Connections Discovery. T1033 System Owner/User Discovery. T1007 System Service Discovery. T1124 System Time Discovery. T1069.001 Permission Groups Discovery: Local Groups. T1016.001 System Network Configuration Discovery: Internet Connection Discovery. T1083 File and Directory Discovery.

COLLECTION

T1005 Data from Local System.

COMMAND AND CONTROL

T1071.001 Application Layer Protocol: Web Protocols. T1132.001 Data Encoding: Standard Encoding. T1573 Encrypted Channel. T1095 Non-Application Layer Protocol. T1090.001 Proxy: Internal Proxy (reverse proxy).

EXFILTRATION

T1041 Exfiltration Over C2 Channel. T1041 Scheduled Transfer.

Figure 20. TTPS GOMIR (APT KIMSUKY)

3.9. IOCs

GOMIR

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	SHA256	30584f13c0a9d0c86562c803de350432d5a0607a06b24481ad4d92cdf7288213	GOMIR (Souche virale)
TLP:CLEAR	SHA1	93edc15a20aac8b5193e5b22e35dbb09848e2ca0	GOMIR (Souche virale)
TLP:CLEAR	MD5	e562cf30d17d47347c7e6ffd249fc190	GOMIR (Souche virale)
TLP:CLEAR	IP	216(.)189.159.34	C2 GOMIR

GOBEAR

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	SHA256	7BD723B5E4F7B3C645AC04E763DFC913060EAF6E136EECC4EE0653AD2056F3A0	Trojan Dropper GOBEAR
TLP:CLEAR	SHA1	1DD417D7373DF9B8F5B76E7EB8FE87B7C37F0CC8	Trojan Dropper GOBEAR
TLP:CLEAR	MD5	B74EFD8470206A20175D723C14C2E872	Trojan Dropper GOBEAR

TROLL STEALER

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	SHA256	d7f3ecd8939ae8b170b641448ff12ade2163baad05ca6595547f8794b5ad013b	Troll Stealer (Souche virale)
TLP:CLEAR	SHA256	36ea1b317b46c55ed01dd860131a7f6a216de71958520d7d558711e13693c9dc	Troll Stealer (Souche virale)
TLP:CLEAR	MD5	19c2decfa7271fa30e48d4750c1d18c1	Trojan Dropper NX_PRNMANS.EXE
TLP:CLEAR	SHA1	e6be97ca9e79b45c671c6531908f70b353d47994	Trojan Dropper NX_PRNMANS.EXE
TLP:CLEAR	SHA256	6eebb5ed0d0b5553e40a7b1ad739589709d077aab4cbea1c64713c48ce9c96f9	Trojan Dropper NX_PRNMANS.EXE
TLP:CLEAR	MD5	7b6d02a459fdaa4caa1a5bf741c4bd42	Trojan Dropper NXTPKIENT.exe
TLP:CLEAR	SHA1	4eea45c22881a092ac7a8b0a5379076d5803e83e	Trojan Dropper NXTPKIENT.exe
TLP:CLEAR	SHA256	f8ab78e1db3a3cc3793f7680a90dc1d8ce087226ef59950b7acd6bb1beffd6e3	Trojan Dropper NXTPKIENT.exe
TLP:CLEAR	MD5	27ef6917fe32685fdf9b755eb8e97565	Trojan Dropper XOWizmxM6U.exe
TLP:CLEAR	SHA1	6d531b021b20febf1dafa730582944eb82d9c6f3	Trojan Dropper XOWizmxM6U.exe
TLP:CLEAR	SHA256	2e0ffaab995f22b7684052e53b8c64b9283b5e81503b88664785fe6d6569a55e	Trojan Dropper XOWizmxM6U.exe
TLP:CLEAR	MD5	7457dc037c4a5f3713d9243a0dfb1a2c	Troll Stealer (Souche virale)
TLP:CLEAR	SHA1	4c8b7d968806f8108ccde6ac07a37b8174ac44bf	Troll Stealer (Souche virale)
TLP:CLEAR	SHA256	ff3718ae6bd59ad479e375c602a81811718dfb2669c2d1de497f02baf7b4adca	Troll Stealer (Souche virale)
TLP:CLEAR	MD5	c8e7b0d3b6afa22e801cacaf16b37355	Troll Stealer (Souche virale)
TLP:CLEAR	SHA256	955cb4f01eb18f0d259fcb962e36a339e8fe082963dfd9f72d3851210f7d2d3b	Troll Stealer (Souche virale)
TLP:CLEAR	MD5	88f183304b99c897aacfa321d58e1840	Troll Stealer (Souche virale)

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	SHA256	bc4c1c869a03045e0b594a258ec3801369b0dcabac193e90f0a684900e9a582d	Troll Stealer (Souche virale)
TLP:CLEAR	URL	hxxp(:)//ai.kostin.p-e(.)kr/index.php	
TLP:CLEAR	URL	hxxp(:)//ar.kostin.p-e(.)kr/index.php	
TLP:CLEAR	URL	hxxp(:)//ai.negapa.p-e(.)kr/index.php	
TLP:CLEAR	URL	hxxp(:)//ol.negapa.p-e(.)kr/index.php	
TLP:CLEAR	URL	hxxp(:)//ai.limsjo.p-e(.)kr/index.php	
TLP:CLEAR	URL	hxxp(:)//qi.limsjo.p-e(.)kr/index.php	
TLP:CLEAR	URL	hxxp(:)//coolsystem(.)co.kr/admin/mail/index.php	
TLP:CLEAR	Domaine	ai.kostin.p-e(.)kr	
TLP:CLEAR	Domaine	ar.kostin.p-e(.)kr	
TLP:CLEAR	Domaine	ai.negapa.p-e(.)kr	
TLP:CLEAR	Domaine	ol.negapa.p-e(.)kr	
TLP:CLEAR	Domaine	ai.limsjo.p-e(.)kr	
TLP:CLEAR	Domaine	qi.limsjo.p-e(.)kr	
TLP:CLEAR	IP	216.189.159(.)197	C2 TROLL STEALER

3.10. YARA

3.10.1. YARA 1

YARA - ShadowStackre
Source : https://www.shadowstackre.com/analysis/gomir

```
rule GomirBackdoor {
meta:
       description = "Rule to detect Gomir Backdoor"
       author = "ShadowStackRe.com"
       date = "2024-05-22"
       Rule_Version = "v1"
       malware_type = "backdoor"
       malware_family = "gomir"
       License = "MIT License, https://opensource.org/license/mit/"
       Hash = "30584f13c0a9d0c86562c803de350432d5a0607a06b24481ad4d92cdf7288213"
strings:
       $strCronText = "cron.txt"
       $strHttpResPathMIR = "mir/"
       $strSystemDSvc = "syslogd.service"
       $strSocksList = "Socks list"
       $strCmdPath = "CmdPath:"
       $strCodePage = "Codepage:"
       $strNextConnTime = "Next Connection Time:"
       $strTCPOpenedIndicator = {
       C7 44 24 29 5B 2B 5D 20
       C7 44 24 2C 20 4F 70 65
       C7 44 24 30 6E 65 64 2E
       }
condition:
       all of them and filesize < 6MB
}
```

3.10.2. YARA 2

YARA - aDvens

```
rule GOMIR_Specific_strings {
meta:
author = "aDvens-CTI"
source = "aDvens"
status = "RELEASED"
sharing = "TLP:CLEAR"
malware = "GOMIR"
description = "Yara_rule_that_detects_GOMIR_Backdoor_June_2024."
info = "GOMIR_Backdoor_malware_used_by_APT_KIMSUKY"
strings:
$GOMIR_string1 = "cron.txt"
$GOMIR_string2 = "/var/log/syslogd"
$GOMIR_string3 = "216.189.159.34"
condition:
$GOMIR_string1 and $GOMIR_string2 and $GOMIR_string3
}
```

4. JO2024 : Analyse de la menace et du malware AcidPour

Le 26 juillet 2024, la France inaugure les Jeux Olympiques avec une parade des Nations sur la Seine, devant accueillir 600 000 spectateurs. Cet événement est perçu comme une opportunité pour la France de briller sur la scène mondiale, d'invoquer une trêve pacifique et l'esprit de fraternité de Pierre de Coubertin.

En résulte dans le discours public un consensus médiatique prônant une stricte séparation entre géopolitique et sport. Cependant, les Jeux Olympiques modernes, depuis leur réhabilitation en 1894, servent de vitrine aux nations pour afficher leur puissance et transmettre des messages politiques. L'idéal olympique ne doit pas occulter les menaces pesant sur l'organisation des Jeux et la France. Il est essentiel de déconstruire l'idée que les jeux politiques n'ont pas leur place dans les jeux sportifs et de rester conscients des menaces cyber qui planent sur l'édition JOP2024.

4.1. Géopolitique du sport

Dans la déclinaison moderne des Jeux Olympiques, la compétition n'oppose plus des Grecs entre eux au sein d'un sanctuaire unique, mais des nations différentes, souvent antagonistes, qui les accueillent de manière successive. Les Jeux Olympiques modernes vont ensuite suivre de manière logique et naturelle les modes de pensées, surtout à mesure que le sport prend une importance croissante dans la vie domestique et l'espace public.

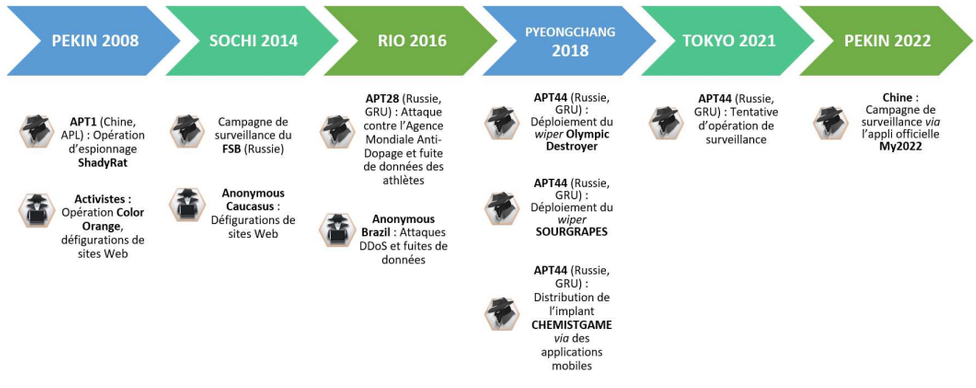
Au cours des 100 dernières années, 4 exemples marquants nous permettent de suivre le lien indissociable entre compétition sportive mondiale et géopolitique :

- Les Jeux de Berlin de 1936. Est-il besoin de rappeler le cadre, et le contexte.
- Les Jeux de Moscou de 1980, boycottés par les pays du Bloc de l'Ouest.
- Les Jeux suivants, de Los Angeles de 1984, boycottés par les pays du Bloc Soviétique.
- Plus contemporain, les Jeux de Pékin de 2008 sont un moyen pour la Chine d'exposer publiquement, notamment au cours de sa cérémonie d'ouverture grandiose, son retour sur la scène mondiale en qualité de géant économique et politique, concurrent des Etats-Unis.

En 2022 et 2023, les évènements de la Coupe du Monde de football au Qatar, ou l'interdiction de participation des athlètes russes et biélorusses aux JOP2024, démontrent que le sport peut être à la fois moteur et prétexte d'enjeux politiques.

4.2. Campagnes cyber des précédentes éditions

En conséquence, il est attendu que la menace *cyber* va s'inviter cet été, comme vecteur de relais des ambitions politiques des pays participants, ou ne participants officiellement pas d'ailleurs. Depuis qu'elle est devenu un outil efficient pour les nations, et une menace crédible, il n'y a pas eu de Jeux sans campagnes d'attaque cyber depuis Pékin en 2008, que ce soit pour des motifs divers ; déstabilisation, sabotage, espionnage, ou appât du gain.



4.3. Présentation de la menace

Tout évènement important, fréquemment médiatisé, est à la fois cause et prétexte d'attaques cyber. Les Jeux Olympiques de Paris sont potentiellement l'objet de campagnes d'attaques, quels qu'en soit la finalité, et le prétexte de vecteurs d'accès initiaux, tels que les envois massifs de courriels d'hameçonnage.

La prévention de cet évènement est extrêmement complexe en raison de la multiplicité des acteurs impliqués : Opérateurs d'Importance Vitale (OIVs), infrastructures, sites de compétition, collectivités locales accueillant des épreuves, entreprises partenaires, relais médiatiques, sous-traitants, etc…

A cette complexité s'ajoute une actualité mondiale à flux tendus, avec plusieurs théâtres d'opérations armées et conflits en cours, sur lesquels la France tient des positions officielles. Ces dernières dessinent différents parties prenantes, hostiles, inamicales, opportunistes, actives ou en embuscade.

4.3.1. Contexte mondial

- **Théâtre ukrainien** : La France s'est positionnée fermement, à travers son gouvernement, dans la guerre actuelle entre l'Ukraine et la Russie. Les déclarations d'envois de troupes françaises sur le front, ou de la non-participation des athlètes russes et biélorusses « non-bienvenus » ont entraîné des tensions très importantes entre la France et la Russie, ainsi que quelques menaces explicites de certains décideurs à Moscou.

Fédération de Russie

Très élevée

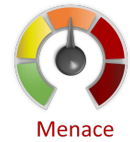
- **Théâtre Pacifique** : La Chine poursuit chaque jour son ambition de concurrencer les Etats-Unis sur des aspects économiques, militaires et diplomatiques. Si la France reste en arrière-plan dans cet affrontement entre puissances, la Chine veille en revanche farouchement dans l'Hexagone sur les discours autour des droits de l'Homme, des Ouïghours, et surtout de Taïwan. La France ne reconnait pas la souveraineté de l'île du « Taipei chinois », cependant des inexactitudes de langage ou certaines positions médiatiques pendant l'évènement peuvent susciter l'ire de Pékin et des représailles, comme pendant les Jeux Olympiques de Tokyo en 2022. Enfin, si la Chine n'a pas habitué à de vastes campagnes de déstabilisation, elle est en revanche un acteur opportuniste et quasi systématique en matière d'espionnage.

République Populaire de Chine

Elevée

- **Théâtre Proche-Orient** : Les attaques du Hamas contre des civils Israéliens en octobre 2023, et la réplique israélienne sur la bande de Gaza a entrainé un émoi au niveau mondial. L'implication récente de l'Iran en avril 2024 a encore complexifié l'affrontement. La France a conservé une position neutre en appelant à une trêve, et s'est peu exposé à des déclarations de représailles.

Groupes activistes nationalistes arabes, ou musulmans



Modérée

4.3.2. Cybercriminalité

Les différents groupes composant le paysage souterrain de la cybercriminalité étant tous en compétition entre eux, un évènement mondial est une opportunité à ne pas laisser passer afin de construire ou affermir sa réputation. De plus, ces groupes privés et lucratifs se distinguent par une mentalité opportuniste, les Jeux peuvent donc servir d'objet et de thème pour de nombreux courriels d'hameçonnage et de documents leurres.

Il est à noter qu'en termes d'impact, ces cartels représentent la première menace tous secteurs confondus. Leurs opérateurs sont compétents, matures, sophistiqués et s'appuient sur des infrastructures efficaces et des méthodes éprouvées. Enfin, la majorité de ces groupes d'attaquants provient de Fédération de Russie. Le ciblage de leurs victimes, orienté vers le reste de l'Asie et l'Occident, relève d'une convention informelle avec l'exécutif russe, et rejoint donc les intérêts de Moscou. De plus, leur domiciliation à l'intérieur des frontières de la Fédération de Russie les préserve d'opérations de police et d'éventuelles arrestations.

Lockbit

Le cas du groupe Lockbit en est un bon exemple. Le 19 février 2024, l'opération policière internationale Cronos permet de mettre hors ligne une partie des infrastructures du groupe *ransomware* le plus actif depuis 2022. Les publications des forces de l'ordre sur le site vitrine de Lockbit portent un coup très sérieux à la réputation du groupe, et donc à son modèle économique. Cependant, son fondateur, LockbitSupp, résident russe, et les principaux développeurs de la marque n'ont pas été inquiétés dans l'opération. Si la saisie de l'infrastructure a porté un arrêt brutal à l'activité, le produit Lockbit3.0 reste pourtant une marque de référence pour de nombreux affiliés cybercriminels. Après une phase de silence et de désorganisation, le *ransomware* est revenu en force et en agressivité. D'importantes attaques ont frappé des victimes françaises :

- 30/04/2024 : Centre Hospitalier de Cannes,
- 06/05/2024 : Agence des Espaces Verts d'Ile-de-France.

A l'heure de cette rédaction, Lockbit s'est senti assez confiant pour revendiquer le 24 juin 2024 l'exfiltration de 33To de données bancaires de la Réserve Fédérale des USA. Celles-ci ont été publiées le 26/06, et s'avèrent finalement appartenir à la Evolve Bank and Trust, qui a reçu une ordonnance de cessation de la Banque Fédérale des Etats-Unis. La France, dont la Gendarmerie Nationale a participé à l'opération Cronos, pourrait être fortement prise pour cible par le groupe pendant la période des Jeux Olympiques.

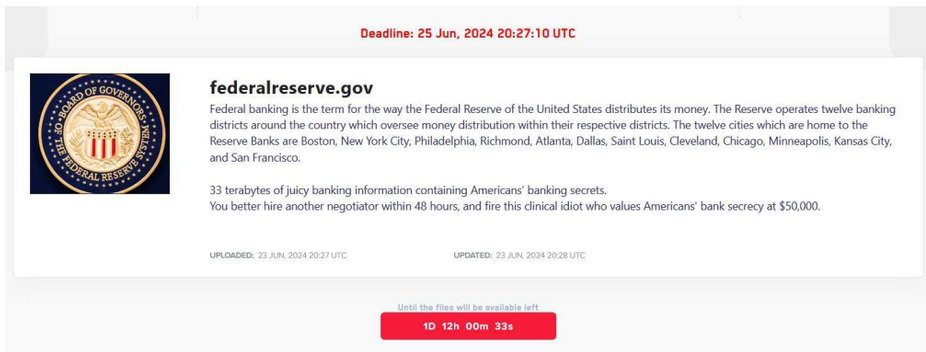
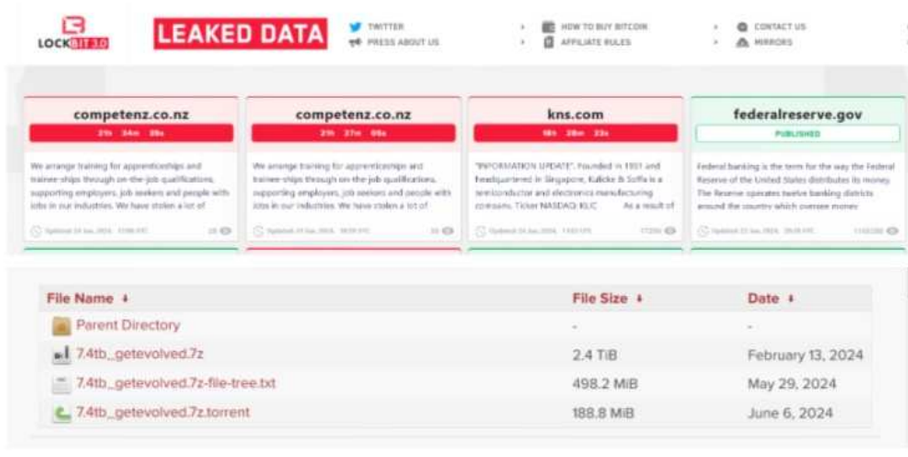


Figure 21. Source : Lockbit.



- **Cibles :** Grandes villes, municipalités moyennes et petites, ministères et institutions publiques, sociétés d'exploitation touristique, entreprises partenaires, acteurs privés de tous secteurs.

4.3.3. Hacktivisme

Les groupes hacktivistes opèrent quant à eux pour des motifs politiques, souvent nationalistes ou religieux dans le cas des groupes ciblant la France. Leur mode opératoire est de provoquer des attaques DDoS contre des sites Internet ou certaines plateformes. Bien que l'impact matériel soit nul, l'inaccessibilité temporaire des ressources en ligne et le relais médiatique de ces attaques sont importants, pouvant avoir un impact psychologique significatif sur les populations.

Cette portée d'attaque pourrait être encore plus importante dans le cas des JOP2024, par exemple avec le ciblage de chaînes de télévision, de plateformes de vidéo à la demande et des billeteries en ligne. Le 19 juin dernier, la chaîne polonaise TV Spot a subi une attaque pendant la retransmission du match de la Pologne face aux Pays-Bas, privant les spectateurs de la première mi-temps. La Pologne a officiellement attribué l'attaque à la Russie.

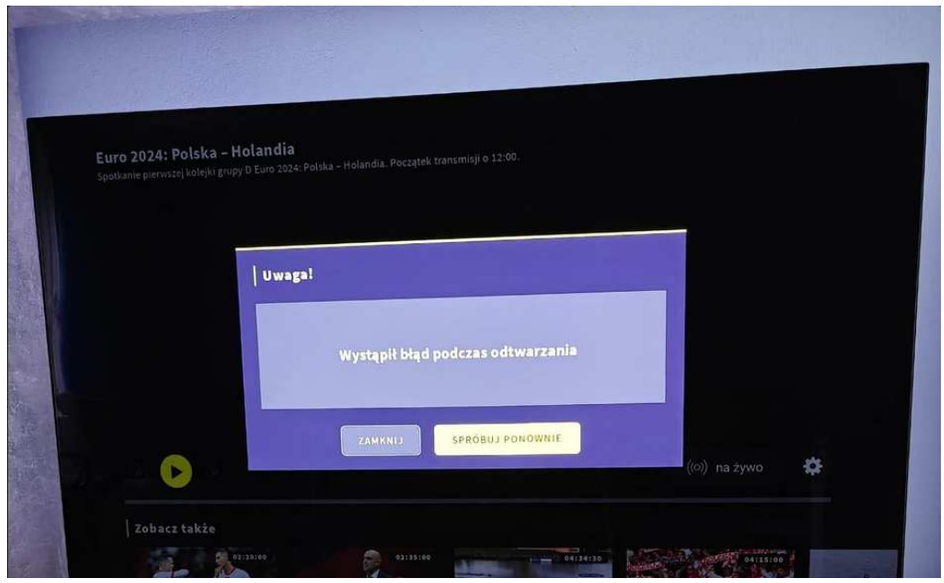


Figure 22. Source : Natemat.pl.

NoName057(16)

La France est régulièrement ciblée par des groupes pro-russes depuis l'offensive militaire en Ukraine de 2022. La dernière attaque en date a été orchestrée par le groupe NoName057(16) et a ciblé une quinzaine de sites gouvernementaux le 15 juin. Ce collectif est très actif et fait figure de pointe de diamant avec le développement du projet DDoSia, une boîte à outils d'attaque par déni de service distribué (DDoS), utilisable par n'importe quel affilié.

Figure 23. Source : CyberArmyofRussia_Reborn.

D'autres groupes avec un spectre pro-Palestine ont ciblé la France ces derniers mois, à titre d'exemple :

- LulzSec Muslims : collectif inspiré de Killnet,
- Türk Hack Team : groupe pro-Turquie coordonnant des attaques contre des pays sympathisants de la cause kurde.

Au Proche Orient, l'intensité des affrontements n'a pas drastiquement baissé, et des revendications politiques et médiatiques ont pris le relais de ce conflit en France, qui ne reconnait pas d'Etat de Palestine. Il n'est pas exclu que ces derniers groupes prennent part à des déstabilisations pendant les Jeux Olympiques en conséquence.

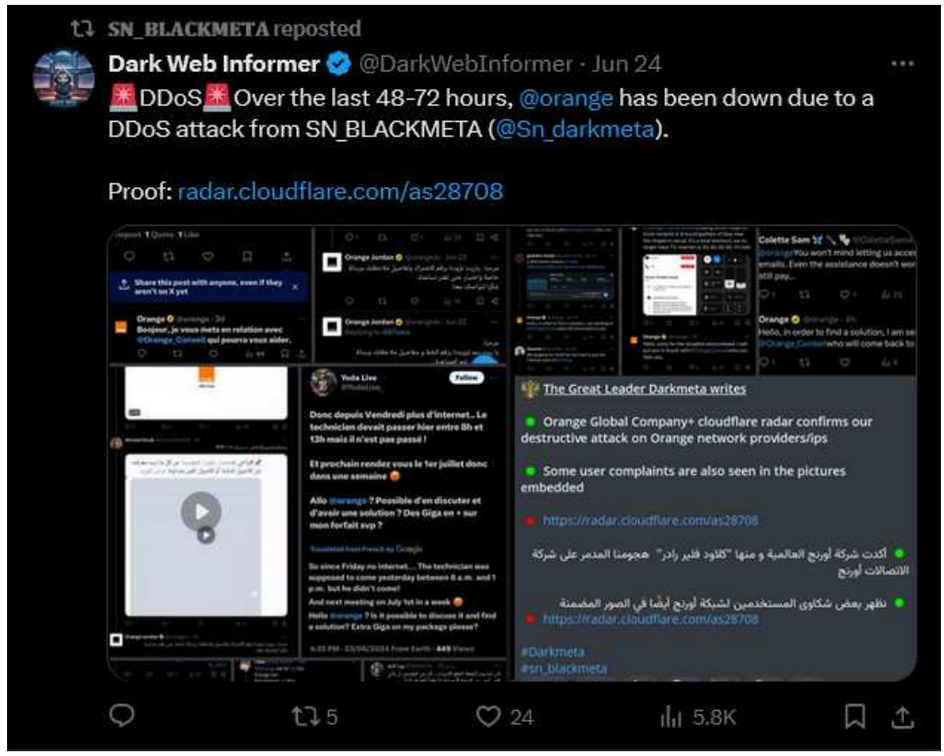


Figure 24. Source : Attaque DDoS contre Orange le 24/06/2024 par le groupe SN_BLACKMETA (source : X).

- **Cibles :** Services publics, billetteries et comités olympiques, chaînes de télévision et services de vidéo à la demande.

4.3.4. Perturbations et sabotage

Russie

La Russie maintient une pression cyber constante sur ses adversaires sur la scène mondiale. Les différentes APTs rattachées à l'appareil de renseignement russe sont tout à la fois capables de mener des actes de guerre, de destruction, d'espionnage, de surveillance et de déstabilisations. Le CERT-FR a rappelé dans un rapport le 19 juin dernier les tentatives d'attaques récurrentes du groupe NOBELIUM, affilié aux services de renseignements extérieurs russes (SVR), contre des Ministères français, notamment les Ministères de la Culture et des Affaires Etrangères.

Olympic Destroyer

Dans le cas des Jeux Olympiques, la Russie s'est déjà distinguée par l'emploi de *wipers* sur les infrastructures hôtes. La cérémonie d'ouverture des Jeux de Pyeongchang, en Corée du Sud en 2018, a été perturbée par un *wiper*, qui sera plus tard baptisé Olympic Destroyer. Celui-ci a mis hors ligne le site officiel des jeux, désactivé le réseau Wifi du stade, le système de vidéosurveillance, ainsi que plusieurs drones utilisés pour la captation d'images.

Le *malware* avait été distribué par courriels d'harponnage, puis déployait deux utilitaires dédiés au vol de mots de passe stockés dans les navigateurs et ceux utilisés sur le système. Il détruit les sauvegardes et copies conservées automatiquement par le système, puis l'outil de récupération Windows proposé avant la séquence de démarrage. Olympic Destroyer supprime ensuite ses traces, et désactive l'ensemble des services liés à Windows avant de mettre hors tension la machine, qui ne peut ensuite plus redémarrer.

La campagne militaire en Ukraine entamée en 2022 est marquante par l'utilisation massive et inédite de nombreux *wipers* contre les infrastructures ukrainiennes : WhisperGate, HermecticWiper, HermeticRansom (faux *ransomware*), AcidRain, IsaacWiper, DesertBlade, CaddyWiper, DoubleZero, ArguePatch, Industroyer 2, Prestige (faux *ransomware*), NikoWiper, Somnia, RansomBoggs (faux *ransomware*), sDelete, AWFULSHRED, BidSwipe, SwiftSlicer. La Russie a donc pu profiter de son engagement en Ukraine pour développer une bonne maturité de ces malwares, depuis Pyeongchang en 2018.

- **Cibles :** Infrastructures.

4.4. Analyse d'AcidPour

4.4.1. AcidRain, le grand frère

Un des *scenarii* redoutés dans le cadre de ces Jeux Olympique est la réutilisation d'une de ces armes de destruction, ou d'un variant, en mesure de paralyser et annihiler des infrastructures (physiques ou de réseau).

Parmi ceux-là, AcidPour a été identifié en mars 2024 par des chercheurs en sécurité de SentinelOne. Ce dernier est lui-même une variante du tristement célèbre AcidRain, utilisé le 24 février 2022 le jour même de l'offensive en Ukraine. L'attaque avait ciblé le réseau KA-SAT de l'opérateur Viasat (Eutelsat) et a touché les communications de plusieurs milliers de clients en Ukraine mais également en Europe. Un dégât collatéral a été la désactivation du contrôle à distance de 5800 éoliennes en Allemagne. Les attaquants avaient exploité l'accès au VPN Skylogic, avant de se latéraliser et d'exécuter des commandes légitimes sur des modems SurfBeam. Ces commandes destructrices ont écrasé les données de la mémoire flash de ces modems.

Si le *malware* est exécuté avec un compte *root*, les périphériques de disque (/dev/sdX, /dev/loopX,/dev/block/mtdblockX,/dev/block/mmcblkX.) sont effacés. Les périphériques de mémoire '/dev/mtdX' sont effacés *via* les utilitaires MEMWRITEOOB et ioctl. A l'issue de ces effacements, un redémarrage de l'appareil est déclenché.

4.4.2. AcidPour

AcidPour, téléchargé pour la première fois le 16 mars 2024 en Ukraine, présente quelques similitudes, comme les chemins pris pour cible sur les machines infectées, et possède 30% de code en commun avec AcidRain. Cette proximité se dessine dans le mécanisme de redémarrage, la logique et le mécanisme d'effacement basé sur la fonction IOCTL utilisé à la fois par AcidRain et le *plugin* VPNFilter « dstr ». En revanche, si AcidRain peut cibler des systèmes Linux avec l'architecture MIPS, AcidPour peut dorénavant cibler des systèmes Linux avec une architecture x86, en plus d'embarquer de nouvelles fonctionnalités.

Parmi ces nouvelles fonctionnalités, AcidPour étend le champ des périphériques ciblés pour inclure des processus *Unsorted Block Image* (UBI) et *Device Mapper* (DM).

AcidRain prend en charge les périphériques suivants :

- **/dev/sd** :* Un périphérique de bloc générique,
- **/dev/mtdblock** :* Mémoire Flash (commune dans les routeurs et les appareils IoT),
- **/dev/block/mtdblock** :* Un autre moyen potentiel d'accéder à la mémoire flash,
- **/dev/mtd** :* Le fichier de périphérique pour la mémoire flash qui prend en charge les opérations de fichiers,
- **/dev/mmcblk** :* Pour les cartes SD/MMC,
- **/dev/block/mmcblk** :* Un autre moyen potentiel d'accéder aux cartes SD/MMC,
- **/dev/loop** :* Périphériques de bloc virtuel.

AcidPour étend ces fonctionnalités et inclut :

- **/dev/dm-XX :** *Framework* de mappage de périphérique, rendant vulnérables les réseaux de stockage (SAN) et les stockages en réseau (NAS),
- **/dev/ubiXX :** L'interface UBI est un système de gestion d'usure de la mémoire flash. Elle est courante dans des systèmes embarqués comme des appareils mobiles, de l'IoT, et même, parfois, des systèmes de contrôles industriels (ICS).
- **Auto-suppression :** Cette nouvelle version démarre avec une fonction d'auto-destruction, en cartographiant le fichier d'origine en mémoire, puis en l'écrasant avec une séquence d'octets allant de 0 à 255 suivie d'un « OK ».

Il est intéressant de noter qu'AcidPour est développé en C, comme CaddyWiper, utilisé contre des centrales électriques en Ukraine (*voir bulletin mensuel de novembre 2023*) par le renseignement militaire russe. Ces nouvelles fonctionnalités semblent indiquer sa prochaine utilisation contre des systèmes industriels, utilisés à la fois dans des manufactures, des centrales, ou des infrastructures publiques …

4.4.3. Attributions

Le CERT-UA a attribué l'exploitation d'AcidPour à UAC-0165, un sous-groupe d'APT44 (ex-Sandworm). De plus, la découverte d'AcidPour par les chercheurs en sécurité de SentinelOne coïncide avec une attaque revendiquée par Solntsepek le 13 mars 2024, soit 3 jours avant. Cette dernière attaque a visé 4 opérateurs en Ukraine, Triacom, Misto TV, Linktelecom et КИМ, avec des réseaux paralysés pendant une semaine :

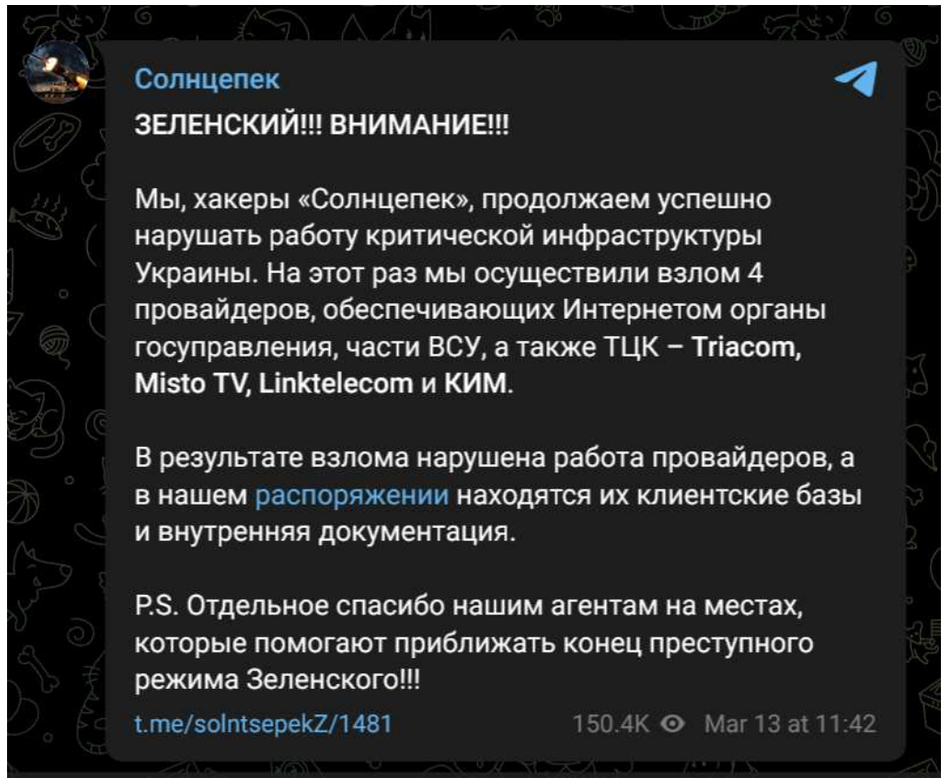


Figure 25. Source : Solntsepek.

4.5. Conclusion

- La géopolitique s'est toujours invitée dans les compétitions sportives depuis le début du sport moderne, au XXème siècle. Le principe selon lequel la politique n'a pas sa place dans le sport peut sembler être une réponse aux ambitions agressives des pays opposés à la France sur la scène mondiale, mais cette affirmation n'est pas fondée.
- Cette posture, bien que noble en apparence, ne doit pas occulter la menace qui pèse actuellement sur la France et sur ces Jeux Olympiques, avec des adversaires affirment ouvertement leur hostilité.
- Un des scénarios les plus redoutés est l'utilisation d'un ou plusieurs wipers contre les infrastructures des Jeux Olympiques, comme cela s'est produit lors des Jeux de Pyeongchang. À cette époque, les groupes APT affiliés au renseignement militaire russe n'étaient pas impliqués dans la campagne en cours en Ukraine. Cependant, la prolifération de ces logiciels malveillants destructeurs observée dans ce conflit récent suscite des inquiétudes quant à leur possible utilisation lors des Jeux Olympiques. De plus, les divers groupes du GRU ont tiré parti de cette expérience pour développer un modèle d'intrusion et d'attaque en cinq phases, conçu pour des opérations cyber offensives de haute intensité visant à augmenter la rapidité, l'ampleur et l'intensité des attaques tout en minimisant les risques de détection.

4.6. IoCs

TLP	TYPE	VALEUR	COMMENTAIRE
TLP:CLEAR	Fichier	tmphluyl8zn	AcidPour (Echantillon)
TLP:CLEAR	SHA256	30584f13c0a9d0c86562c803de350432d5a0607a06b24481ad4d92cdf7288213	AcidPour (Echantillon)
TLP:CLEAR	SHA1	b5de486086eb2579097c141199d13b0838e7b631	AcidPour (Echantillon)
TLP:CLEAR	MD5	1bde1e4ecc8a85cffef1cd4e5379aa44	AcidPour (Echantillon)
TLP:CLEAR	IP	185[.]61.137.155	Domaine Solntsepek
TLP:CLEAR	IP	solntsepek[.]com	Domaine Solntsepek
TLP:CLEAR	IP	solntsepek[.]info	Domaine Solntsepek
TLP:CLEAR	IP	solntsepek[.]org	Domaine Solntsepek
TLP:CLEAR	IP	solntsepek[.]ru	Domaine Solntsepek
TLP:CLEAR	Fichier	acid_rain.elf	AcidPour (Echantillon)
TLP:CLEAR	SHA256	9b4dfaca873961174ba935fddaf696145afe7bbf5734509f95feb54f3584fd9a	AcidPour (Echantillon)

5. Références

CVE

- https://community.zyxel.com/en/discussion/23278/zyxel-security-advisory-for-multiple-vulnerabilities-in-nas-products
- https://www.php.net/archive/2024.php#2024-06-06-2
- https://www.solarwinds.com/trust-center/security-advisories/cve-2024-28995

GOMIR (APT KIMSUKY)

- https://www.shadowstackre.com/analysis/gomir
- https://www.virustotal.com/gui/file/30584f13c0a9d0c86562c803de350432d5a0607a06b24481ad4d92cdf7288213/details
- https://www.joesandbox.com/analysis/1445737/0/html
- https://app.any.run/tasks/78586403-ddc5-4880-ac3f-2875a5bdd7d5?_gl=1*107l1ba*_gcl_au*Njc2MjgyOTU5LjE3MTQwMzAzMTc.*_ga*MTcwNjY5NzU3Ny4xNzE0MDMwMzE3*_ga_53KB74YDZR*MTcxNzQ5OTg0Ny4yLjEuMTcxNzUwMDExMC4wLjAuMTM1MjQ3OTk5/
- https://symantec-enterprise-blogs.security.com/threat-intelligence/springtail-kimsuky-backdoor-espionage
- https://thehackernews.com/2024/05/kimsuky-apt-deploying-linux-backdoor.html
- https://bazaar.abuse.ch/sample/30584f13c0a9d0c86562c803de350432d5a0607a06b24481ad4d92cdf7288213/
- https://www.welivesecurity.com/fr/2020/11/17/attaque-lazarus-coree-du-sud/
- https://any.run/report/7bd723b5e4f7b3c645ac04e763dfc913060eaf6e136eecc4ee0653ad2056f3a0/1d49cc21-50f1-4784-8216-decee3dbaf0d
- https://www.mphasis.com/content/dam/mphasis-com/global/en/home/services/cybersecurity/june-3-2-mysterious-threat-actor-used-chalubo-malware.pdf
- https://asec.ahnlab.com/en/61934/

JO2024 : Analyse d'AcidPour

- https://blog.sekoia.io/securing-gold-assessing-cyber-threats-on-paris-2024/#h-history-of-cyber-operations-impacting-olympic-games
- https://cloud.google.com/blog/topics/threat-intelligence/cyber-threats-2024-paris-olympics?hl=en
- https://www.france24.com/fr/info-en-continu/20240226-ni-r%C3%A9sign%C3%A9s-ni-d%C3%A9faitistes-r%C3%A9union-%C3%A0-paris-des-alli%C3%A9s-de-l-ukraine
- https://www.leparisien.fr/jo-paris-2024/jo-2024-les-athletes-russes-et-bielorusses-pas-les-bienvenus-a-paris-dit-anne-hidalgo-depuis-kiev-30-03-2024-MYSOLNPL5RBKLIZUPDD6ZCM2WE.php
- https://www.francetvinfo.fr/les-jeux-olympiques/jo-2021-a-tokyo-taiwan-ou-taipei-chinois-la-question-est-sensible-depuis-quarante-ans-et-pekin-veille-au-grain_4715377.html
- https://x.com/AlvieriD/status/1805074447130636320
- https://www.numerama.com/cyberguerre/1633112-les-polices-de-11-pays-dont-la-france-abattent-le-site-de-lockbit-le-plus-important-gang-de-hackers.html
- https://www.trendmicro.com/en_us/research/24/d/operation-cronos-aftermath.html
- https://www.lemondeinformatique.fr/actualites/lire-lockbit-30-revendique-un-vol-de-donnees-de-la-fed-94113.html
- https://x.com/DarkWebInformer/status/1805256295248769119
- https://next.ink/141115/une-attaque-ddos-aurait-vise-une-dizaine-de-sites-gouvernementaux-francais/
- https://www.numerama.com/cyberguerre/1764372-euro-2024-la-pologne-accuse-des-hackers-russes-davoir-perturbe-la-diffusion-dun-match.html
- https://www.cert.ssi.gouv.fr/cti/CERTFR-2024-CTI-006/
- https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/overview-of-the-cyber-weapons-used-in-the-ukraine-russia-war/
- https://www.bleepingcomputer.com/news/security/new-acidpour-data-wiper-targets-linux-x86-network-devices/
- https://www.sentinelone.com/labs/acidpour-new-embedded-wiper-variant-of-acidrain-appears-in-ukraine/
- https://cip.gov.ua/en/news/yak-zminyuyutsya-taktiki-cili-i-spromozhnosti-khakerskikh-grup-uryadu-rf-ta-kontrolovanikh-nim-ugrupovan-zvit
- https://www.wired.com/story/ukraine-kyivstar-solntsepek-sandworm-gru/
- https://cloud.google.com/blog/topics/threat-intelligence/gru-disruptive-playbook/?hl=en