Oorn: nuovo toolkit di Keylogging sviluppato in outsourcing

Pubblicato: 08/10/2019 16:57 - Ultimo aggiornamento: 08/10/2019 17:04

Categorie: Cyber Crime, Threat
Tipo di Informazione: Strategico, Tattico, Tecnico
Distribuzione: **TLP:RED**[1] - Declassificato TLP WHITE in data 22/11/2019

Genesi

Dalle attività di tracciamento del **CIOC di TS-WAY** sono stati reperiti i componenti di un nuovo **Keylogger** dell'attore Italiano Oorn.

Assessment

L'avversario Oorn ha arricchito il suo arsenale con un nuovo toolkit, il cui componente principale è un **Keylogger**. Questo programma è in grado di rubare i dati digitati dall'utente e quelli salvati nella clipboard, oltre a raccogliere alcune informazioni basilari riguardanti la macchina infetta.

Il canale di comunicazione verso l'esterno sfrutta il **protocollo SMTP**, inviando i dati all'avversario sotto forma di una serie di email con indirizzo di destinazione noto e già osservato in precedenti campagne.

Il toolkit include inoltre un **launcher** con funzionalità di **cifratura e decifratura dei file** attraverso chiave hardcodata e algoritmo AES256, per rendere più difficile da rilevare il payload deployato sulla macchina.

I due componenti sono stati ribattezzati rispettivamente **Josar** e **JoLuncher**, prendendo spunto dalla stringa PDB e da caratteristiche proprie dei binari.

Lo **sviluppo** di questi strumenti è **esternalizzato a sviluppatori esteri** e in questo caso è stato affidato a un *programmatore palestinese*.

In sintesi

- Oorn ha aggiunto ai suoi strumenti un Keylogger con esfiltrazione dati via SMTP.
- Si iniziano a notare tecniche di offuscamento dei binari più avanzate, tra cui cifratura con AES256.
- Lo sviluppo degli strumenti è esternalizzato a sviluppatori esteri, con pagamento a progetto.

Analisi Tecnica

Dalle attività di tracciamento del **CIOC di TS-WAY** sono stati reperiti diversi sample appartenenti a un **nuovo toolset malevolo** dell'attore Italiano **Oorn**.

Il componente principale del toolkit, con nome *'syshost.exe'* per imitare un servizio lecito di Windows, è un **Keylogger con data exfiltration via STMP**. Il programma raccoglie le digitazioni dell'utente e i dati salvati nella clipboard e li invia a un indirizzo email posseduto dall'attaccante. L'email di destinazione era già stata osservata in precedenti campagne e script utilizzati dall'avversario (tra cui CyclicOrder già documentato in un precedente Insight Special Report).

Il sample analizzato dal CIOC, oltre che mandare le informazioni raccolte alla email dell'avversario, le *invia anche a quella dello sviluppatore*. Si tratta con molta probabilità di una **versione di test**, caricata anche su sistemi di verifica online per comprenderne la detection da parte dei principali motori antivirus.

Il malware raccoglie inoltre alcune informazioni relative alla macchina infetta:

- Username
- Computer name
- Local IP address
- External IP address (attraverso una serie di servizi e API web)

Parti del codice sono state scritte ex-novo, come la parte di comunicazione via SMTP, mentre altre, come le funzioni di base per la realizzazione del keylogger, appartengono al tool open-source di amministrazione remota **Quasar**.

L'altro componente principale del toolkit, avente nome *'Luncher.exe'*, si occupa di installare ed eseguire il keylogger.

Le modalità di persistenza utilizzate sono al momento due:

- se il launcher viene eseguito come utente **Amministratore** verrà predisposto uno **scheduled task al logon** dell'utente e verrà scritto un valore nella chiave *"Software\Microsoft\Windows\CurrentVersion\Run"* sia nella sezione di registro *"Local Machine"* che *"Current User"*.
- altrimenti, in caso di **utente non privilegiato**, verrà scritta un'unica chiave sempre sotto *"Software\Microsoft\Windows\CurrentVersion\Run"* ma solo nella sezione di registro *"Current User"*.

Il codice che si occupa della persistenza è riportato qui di seguito:

```
public static void AddToStartUp(string exePath)
{
    if (Tools.IsUserAdmin())
    {
        try
        {
            Process process = Process.Start(new ProcessStartInfo("schtasks")
            {
                Arguments = string.Concat(new string[]
                {
                    "/create /tn \"",
                    Tools.startup_key,
                    "\" /sc ONLOGON /tr \"",
                    exePath,
                    "\" /rl HIGHEST /f"
                }),
                UseShellExecute = false,
                CreateNoWindow = true
            });
            process.WaitForExit(1000);
            if (process.ExitCode == 0)
            {
                return;
            }
        }
        catch (Exception)
        {
        }
        if (!Tools.AddRegistryKeyValue(RegistryHive.LocalMachine,
"Software\\Microsoft\\Windows\\CurrentVersion\\Run", Tools.startup_key,
exePath, true))
        {
            Tools.AddRegistryKeyValue(RegistryHive.CurrentUser,
"Software\\Microsoft\\Windows\\CurrentVersion\\Run", Tools.startup_key,
exePath, true);
            return;
        }
    }
    else
    {
        Tools.AddRegistryKeyValue(RegistryHive.CurrentUser,
"Software\\Microsoft\\Windows\\CurrentVersion\\Run", Tools.startup_key,
exePath, true);
    }
}
```

Il launcher include inoltre **funzionalità di cifratura e decifratura** di files tramite **AES256** e chiave di decifratura hardcodata, e può indicare l'intenzione di distribuire in futuro il payload in forma cifrata per bypassare le normali signatures antivirus.

Il codice preposto al deploy del keylogger e alla decifratura è il seguente:

```
private void button1_Click(object sender, EventArgs e)
{
    File.WriteAllBytes("syshost_exe_bin",
this.aes256.Encrypt(File.ReadAllBytes("syshost.exe")));
    File.WriteAllBytes("syshost_exe_config_bin",
this.aes256.Encrypt(File.ReadAllBytes("syshost.exe.config")));
    File.WriteAllBytes("Gma_System_MouseKeyHook_dll_bin",
this.aes256.Encrypt(File.ReadAllBytes("Gma.System.MouseKeyHook.dll")));
}

private void button2_Click(object sender, EventArgs e)
{
    File.WriteAllBytes("decrypt\\syshost.exe",
this.aes256.Decrypt(File.ReadAllBytes("syshost.exe.bin")));
    File.WriteAllBytes("decrypt\\syshost.exe.config",
this.aes256.Decrypt(File.ReadAllBytes("syshost.exe.config.bin")));
    File.WriteAllBytes("decrypt\\Gma.System.MouseKeyHook.dll",
this.aes256.Decrypt(File.ReadAllBytes("Gma.System.MouseKeyHook.dll.bin")));
}
```

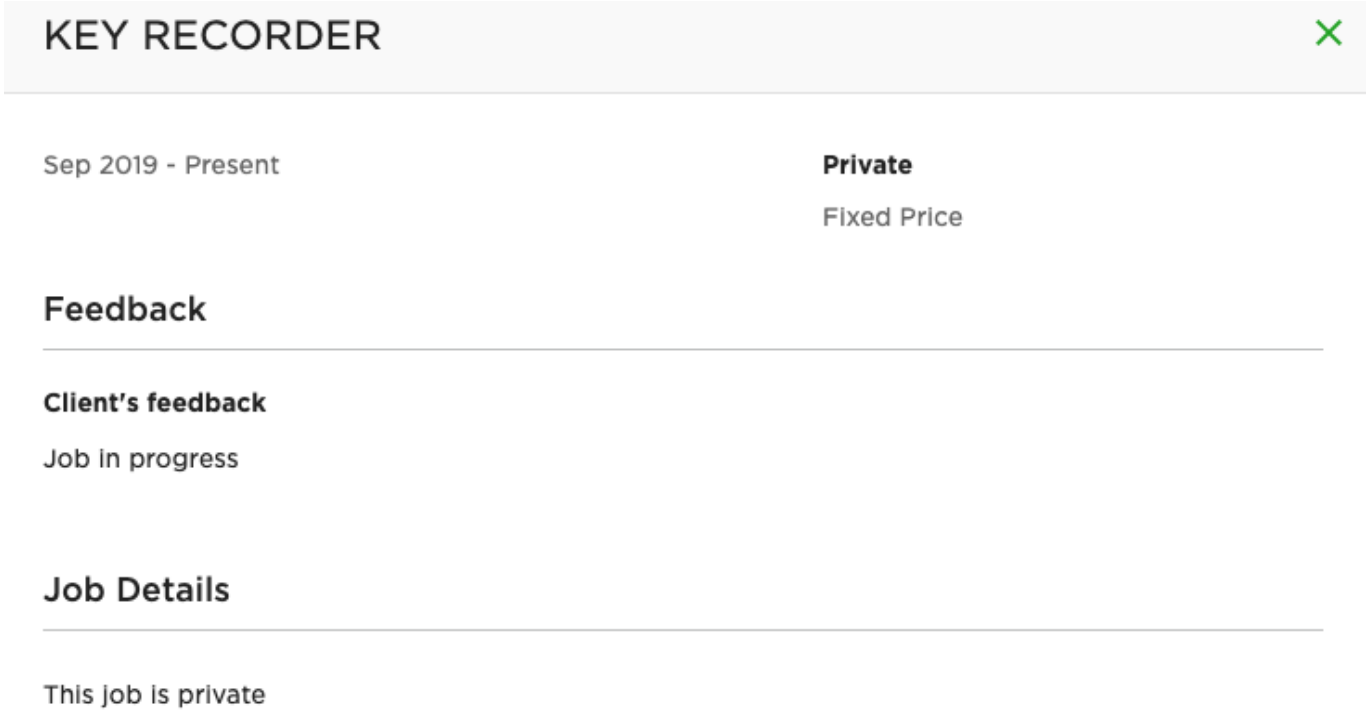
Il keylogger e stato ribattezzato con il nome di **Josar**, fondendo lo username presente nel PDB con *“Quasar”*, da cui sono state prese porzioni di codice. I launcher è invece stato identificato con il nome di **JoLuncher**, unendo come prima lo username del PDB con il nome di test dello strumento, che contiene un errore ortografico nella parola *“Launcher”*.

Le stringhe PDB per i due binari sono rispettivamente:

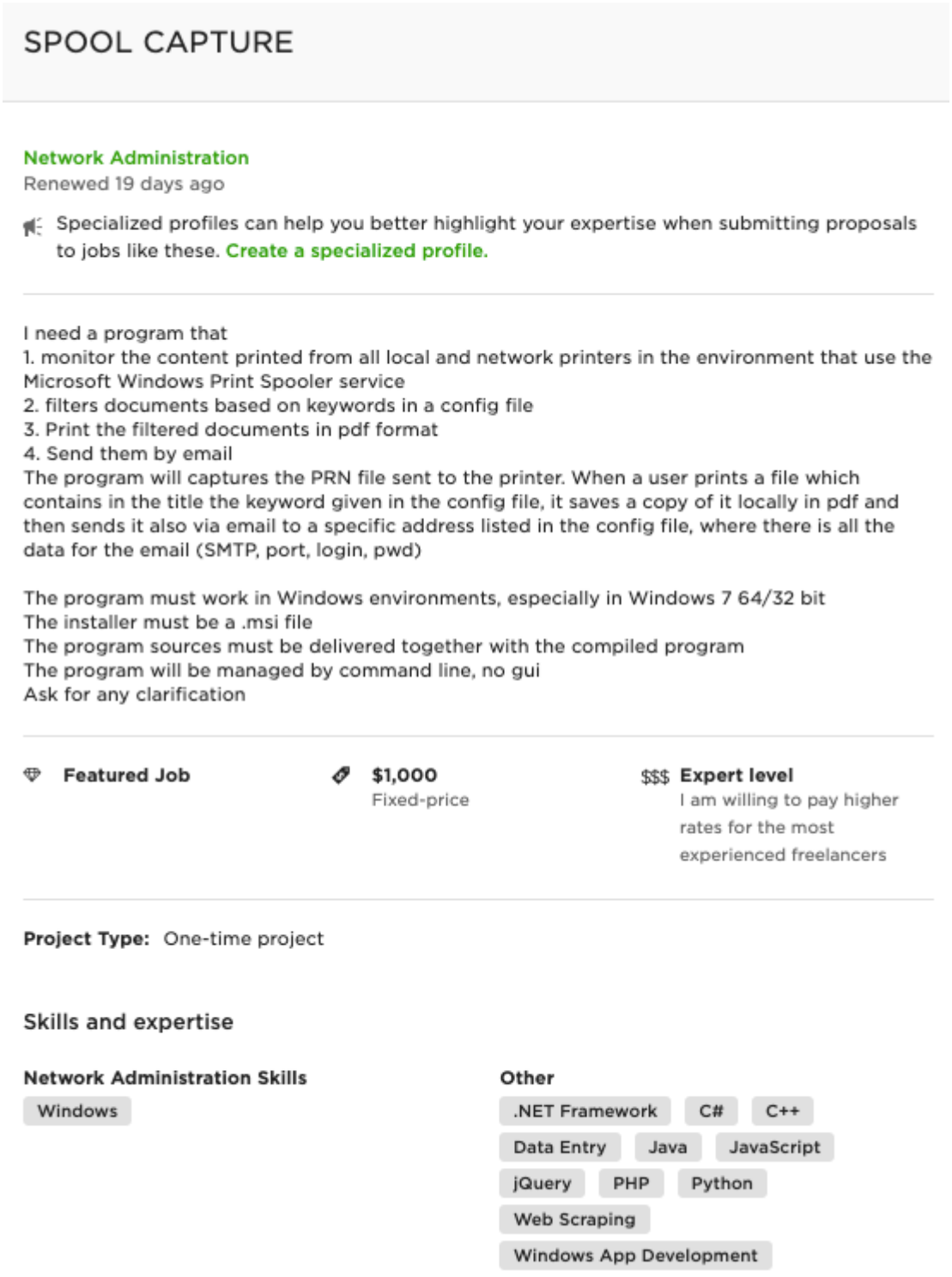
- *C:\Users\Ashraf\Source\Repos\Josef\KeyLogger\Keylogger\Luncher\obj\Release\Luncher.pdb*
- *C:\Users\Ashraf\Source\Repos\Josef\KeyLogger\Keylogger\Keylogger\obj\Debug\syshost*

Dalle stesse stringhe di debug, oltre che dall’email di test usata nel keylogger, è stato possibile capire che lo **sviluppo del malware è stato svolto da sviluppatori esteri**, *in questo caso un programmatore palestinese*, che offrono i loro servizi su piattaforme web pubbliche con pagamento a progetto.

Attraverso operazioni di ricerca è stato possibile individuare alcuni lavori che l’avversario ha commissionato su queste piattaforme, tra cui quello relativa al toolkit analizzato in questo report:



L’avversario pare inoltre interessato ad ampliare il proprio arsenale con malware non ortodossi e molto specializzati:



SPOOL CAPTURE

Network Administration
Renewed 19 days ago

Specialized profiles can help you better highlight your expertise when submitting proposals to jobs like these. Create a specialized profile.

I need a program that
1. monitor the content printed from all local and network printers in the environment that use the Microsoft Windows Print Spooler service
2. filters documents based on keywords in a config file
3. Print the filtered documents in pdf format
4. Send them by email
The program will captures the PRN file sent to the printer. When a user prints a file which contains in the title the keyword given in the config file, it saves a copy of it locally in pdf and then sends it also via email to a specific address listed in the config file, where there is all the data for the email (SMTP, port, login, pwd)

The program must work in Windows environments, especially in Windows 7 64/32 bit
The installer must be a .msi file
The program sources must be delivered together with the compiled program
The program will be managed by command line, no gui
Ask for any clarification

Featured Job
$1,000 Fixed-price
$$$ Expert level I am willing to pay higher rates for the most experienced freelancers

Project Type: One-time project

Skills and expertise

Network Administration Skills
Windows

Other
.NET Framework C# C++ Data Entry Java JavaScript jQuery PHP Python Web Scraping Windows App Development

Questo programma di *"Spool Capture"*, non ancora ultimato, permetterà ad Oorn di esfiltrare via email tutti i **documenti mandati in stampa** sulle macchine delle vittime, **selezionandoli in base a una lista di keyword** ed esportandoli in formato PDF.

Ci si attende l'uso *in-the-wild* di questo e di possibili altri strumenti che utilizzano l'**esfiltrazione di dati via SMTP**, *modus operandi* che contraddistingue sempre più questo

avversario.

Eventi IOC

1. [753851] Malicious Keylogger sample and related Tools

Note

[1] Per maggiori informazioni riguardo le TLP si prega di consultare https://www.us-cert.gov/tlp

Contatti

intel@ts-way.com