Elastic Security Labs discovers the LOBSHOT malware

elastic.co/de/security-labs/elastic-security-labs-discovers-lobshot-malware

An analysis of LOBSHOT, an hVNC malware family spreading through Google Ads.

Von

Daniel Stepanic

25. April 2023
Deutsch

Key takeaways

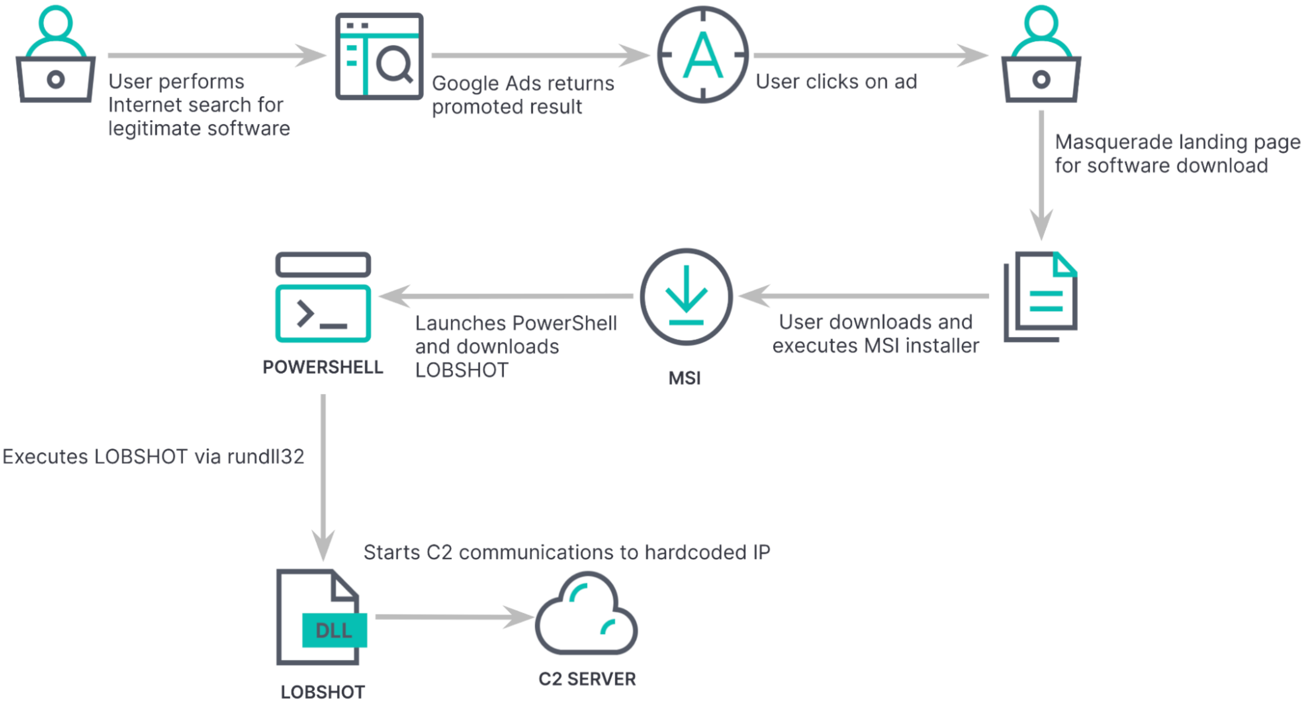
- Adversaries continue to abuse and increase reach through malvertising such as Google Ads by impersonating legitimate software
- Elastic Security Labs is shedding light on an undiscovered hVNC malware that has been quietly collecting a large install base
- This malware we are calling LOBSHOT appears to be leveraged for financial purposes employing banking trojan and info-stealing capabilities

Preamble

Elastic Security Labs along with the research community noticed a large spike in the adoption of malvertising earlier this year. Attackers promoted their malware using an elaborate scheme of fake websites through Google Ads and embedding backdoors in what appears to users as legitimate installers. In this post, we will highlight one malware family we observed from this spike we're calling LOBSHOT. LOBSHOT continues to collect victims while staying under the radar.

One of LOBSHOT's core capabilities is around its hVNC (Hidden Virtual Network Computing) component. These kinds of modules allow for direct and unobserved access to the machine. This feature continues to be successful in bypassing fraud detection systems and is often baked into many popular families as plugins.

We will walk through the LOBSHOT infection chain and its behaviors. Additionally, we will provide a YARA signature and configuration extractor for this family.


User performs Internet search for legitimate software
Google Ads returns promoted result
User clicks on ad
Masquerade landing page for software download
User downloads and executes MSI installer
MSI
Launches PowerShell and downloads LOBSHOT
POWERSHELL
Executes LOBSHOT via rundll32
Starts C2 communications to hardcoded IP
DLL
LOBSHOT
C2 SERVER

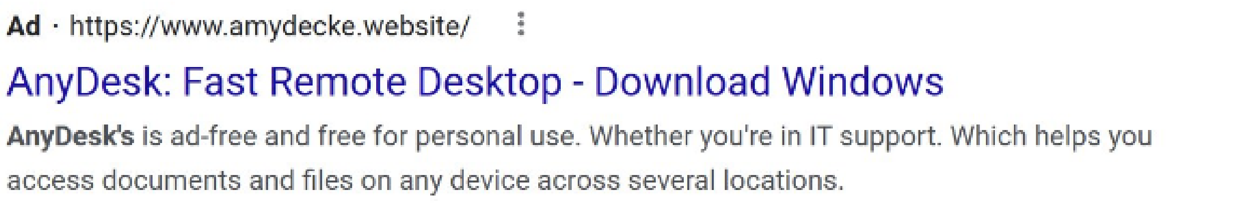
LOBSHOT infection chain

Throughout our analysis, we observed infrastructure known to belong to TA505. TA505 is a well-known cybercrime group associated with Dridex, Locky, and Necurs campaigns. A loader documented by Proofpoint, known as Get2, has also been tied to the same domains in the past that we observed with LOBSHOT. We assess with moderate confidence that LOBSHOT is a new malware capability leveraged by TA505 starting in 2022.

Campaign context

Earlier this year, Elastic Security Labs observed multiple infections with an interesting chain of events that resulted in the execution of an unknown hVNC malware, which we are calling LOBSHOT. Around this same time, similar infection chains were observed in the security community with commonalities of users searching for legitimate software downloads that ended up getting served illegitimate software from promoted ads from Google [1, 2, 3, 4].

In one example, the malicious ad was for a legitimate remote desktop solution, AnyDesk. Careful examination of the URL goes to **https://www.amydecke[.]website** instead of the legitimate AnyDesk URL, **https://www.anydesk[.]com**.



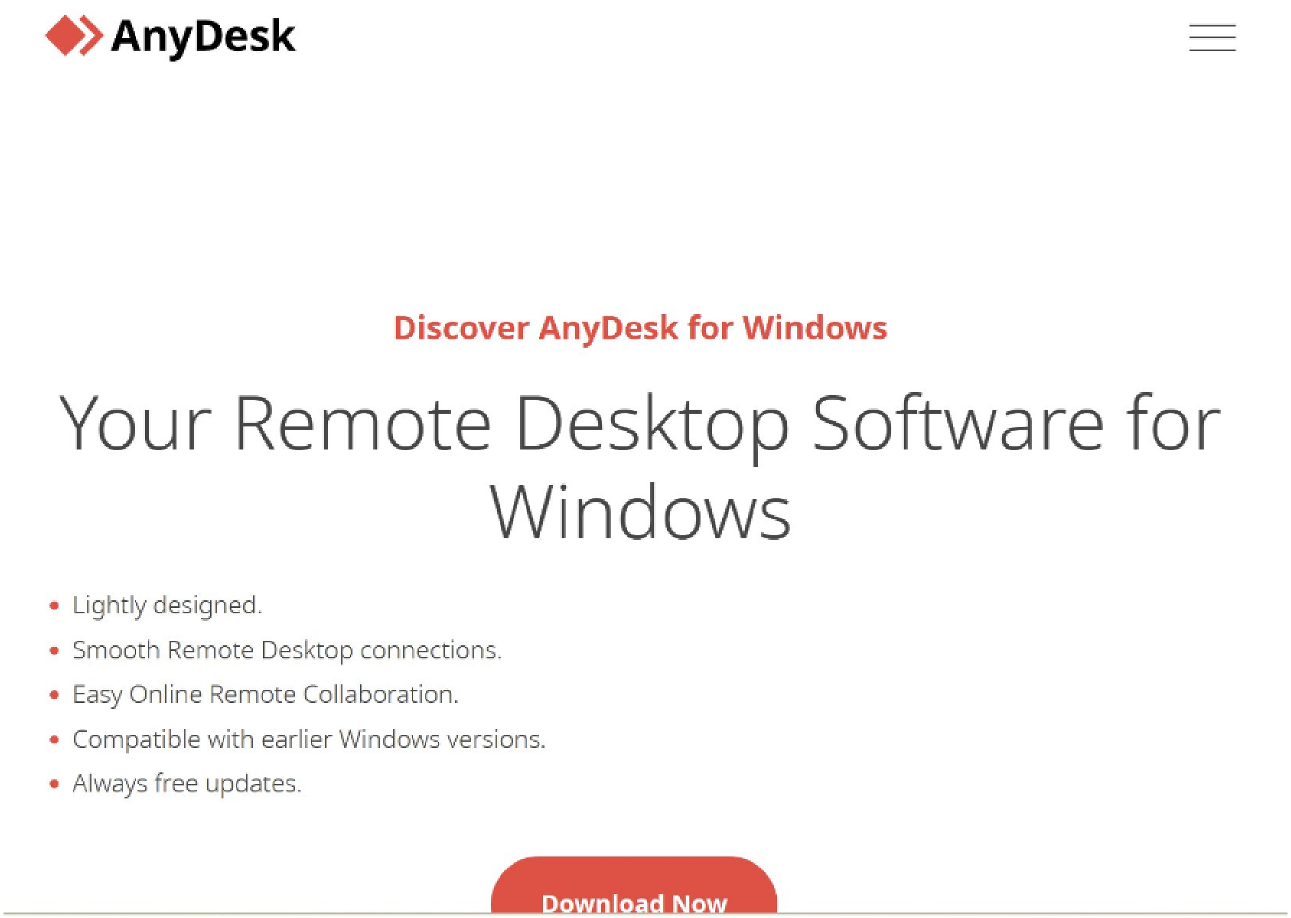
Ad · https://www.amydecke.website/

AnyDesk: Fast Remote Desktop - Download Windows

AnyDesk's is ad-free and free for personal use. Whether you're in IT support. Which helps you access documents and files on any device across several locations.

Malicious Google Ad

The landing pages were very convincing with similar branding as the legitimate software and included Download Now buttons that pointed to an MSI installer.



AnyDesk

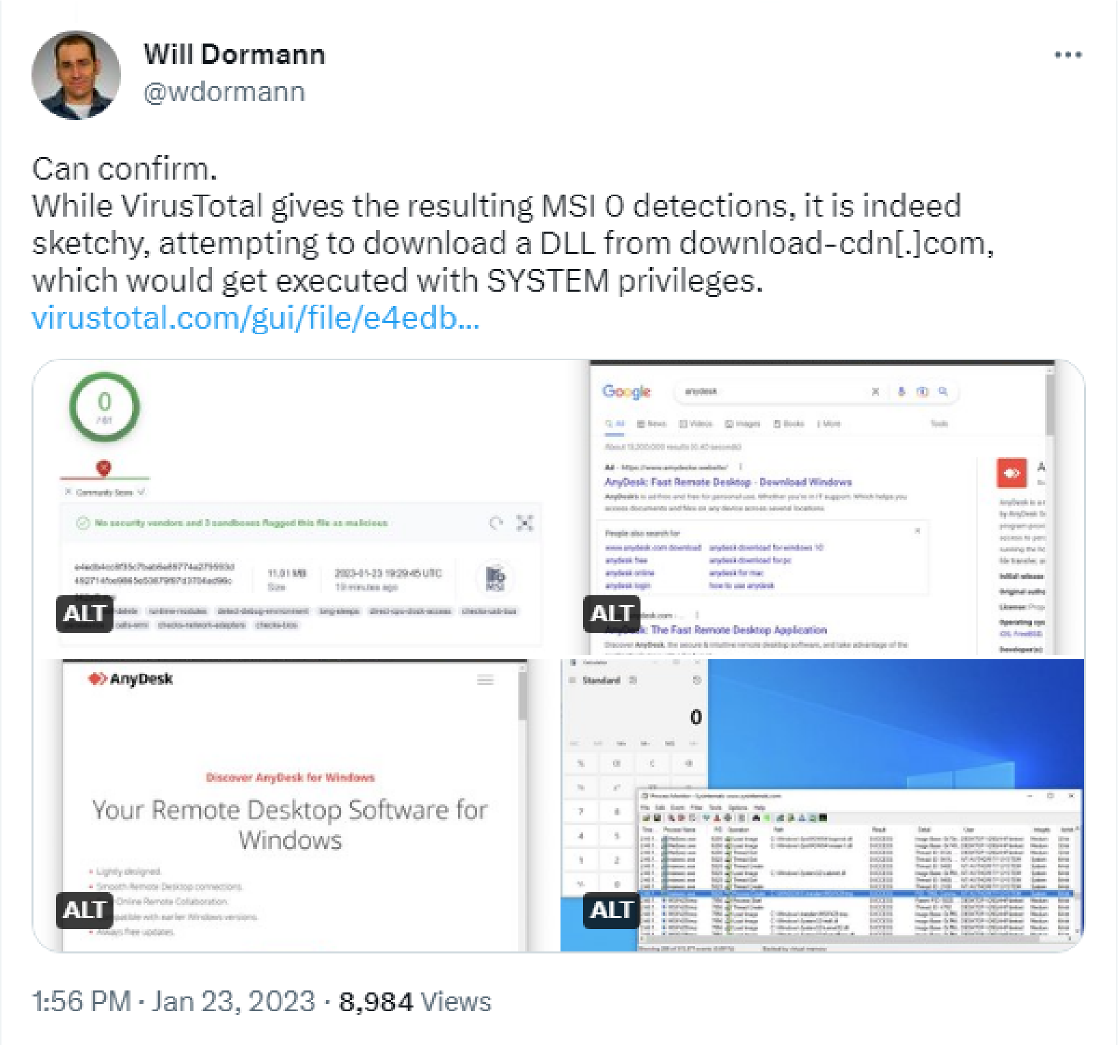
Discover AnyDesk for Windows

Your Remote Desktop Software for Windows

- Lightly designed.
- Smooth Remote Desktop connections.
- Easy Online Remote Collaboration.
- Compatible with earlier Windows versions.
- Always free updates.

Fake AnyDesk landing page for installer

Thanks to security researcher Will Dormann, we were able to view the screenshots from the AnyDesk campaign.



Will Dormann
@wdormann

Can confirm.
While VirusTotal gives the resulting MSI 0 detections, it is indeed sketchy, attempting to download a DLL from download-cdn[.]com, which would get executed with SYSTEM privileges.
virustotal.com/gui/file/e4edb...

0
/ 61
No security vendors and 3 sandboxes flagged this file as malicious
ALT
Google
anydesk
AnyDesk: Fast Remote Desktop - Download Windows
The Fast Remote Desktop Application
ALT
AnyDesk
Discover AnyDesk for Windows
Your Remote Desktop Software for Windows
ALT
Standard
0
ALT

1:56 PM · Jan 23, 2023 · 8,984 Views

Tweet

referencing AnyDesk infection chain

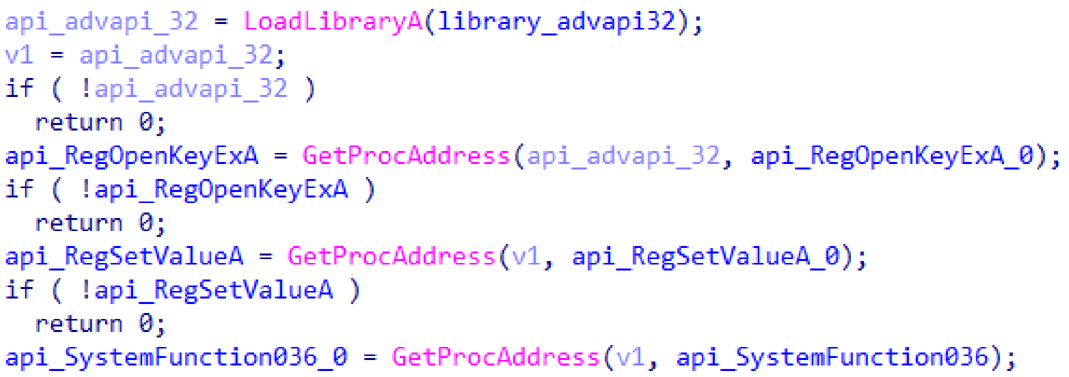
At the time of publication, we haven't seen any previous public information about LOBSHOT, so we will focus our research on LOBSHOT's functionality and capabilities.

LOBSHOT code analysis

To focus on the LOBSHOT malware, we will skip the initial infection chain. For these initial details, here is a good sandbox report to look over for general TTPs. We have observed over 500 unique LOBSHOT samples since last July. The samples we have observed are compiled as 32-bit DLLs or 32-bit executables typically ranging around **93** KB to **124** KB. Consider the following sample representative of LOBSHOT for purposes of this analysis.

Dynamic API resolution

In our LOBSHOT sample, like most malware we see today, it employs dynamic import resolution to evade security products and slow down the rapid identification of its capabilities. This process involves resolving the names of the Windows APIs that the malware needs at runtime as opposed to placing the imports into the program ahead of time.

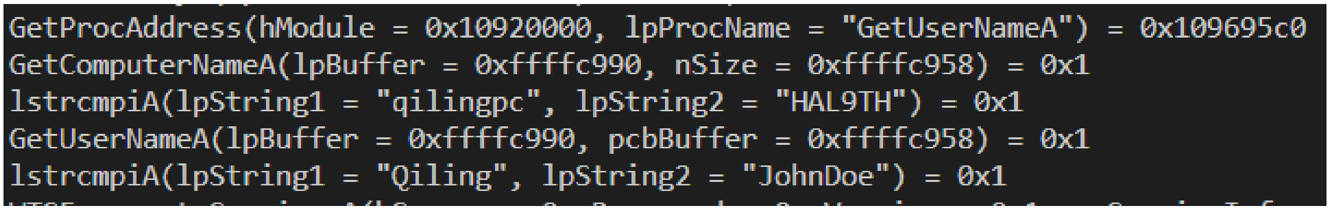


```
api_advapi_32 = LoadLibraryA(library_advapi32);
v1 = api_advapi_32;
if ( !api_advapi_32 )
  return 0;
api_RegOpenKeyExA = GetProcAddress(api_advapi_32, api_RegOpenKeyExA_0);
if ( !api_RegOpenKeyExA )
  return 0;
api_RegSetValueA = GetProcAddress(v1, api_RegSetValueA_0);
if ( !api_RegSetValueA )
  return 0;
api_SystemFunction036_0 = GetProcAddress(v1, api_SystemFunction036);
```

Resolving Windows Registry APIs through LoadLibraryA/GetProcessAddress

Defender emulation check

After the initial libraries are loaded, LOBSHOT performs a Windows Defender anti-emulation check by verifying if the computer name matches the string **HAL9TH** and if the username matches **JohnDoe**. These are hard-coded values within the emulation layer of Defender, if they are present, the malware immediately stops running. This kind of verification has been incorporated in many other stealers including Arkei, Vidar, and Oski. Below is the emulation output using the Qilingframework highlighting these verification checks.

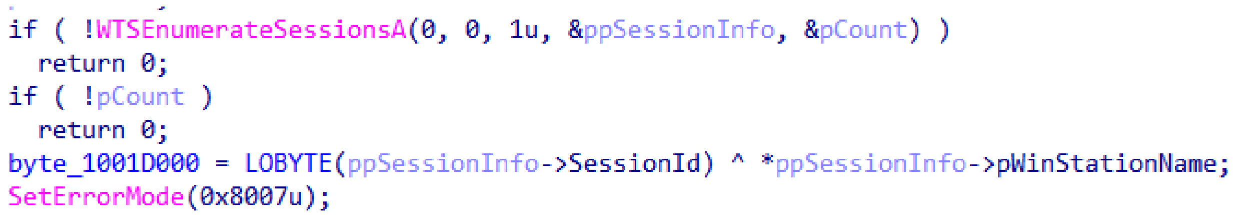


```
GetProcAddress(hModule = 0x10920000, lpProcName = "GetUserNameA") = 0x109695c0
GetComputerNameA(lpBuffer = 0xffffc990, nSize = 0xffffc958) = 0x1
lstrcmpiA(lpString1 = "qilingpc", lpString2 = "HAL9TH") = 0x1
GetUserNameA(lpBuffer = 0xffffc990, pcbBuffer = 0xffffc958) = 0x1
lstrcmpiA(lpString1 = "Qiling", lpString2 = "JohnDoe") = 0x1
```

Defender checks via Qiling

String obfuscation

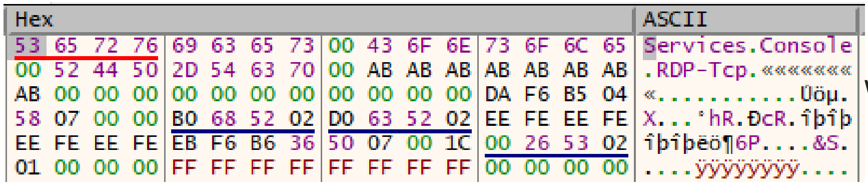
This malware hides its primary strings through a straightforward encryption function using different bitwise operators. To perform the string decryption, LOBSHOT uses an initial seed from the WTS_SESSION_INFO structure from a call to **WTSEnumerateSessionsA.**



```
if ( !WTSEnumerateSessionsA(0, 0, 1u, &ppSessionInfo, &pCount) )
  return 0;
if ( !pCount )
  return 0;
byte_1001D000 = LOBYTE(ppSessionInfo->SessionId) ^ *ppSessionInfo->pWinStationName;
SetErrorMode(0x8007u);
```

LOBSHOT calling WTSEnumerateSessionsA

In this case, the malware developer sets up the initial seed by performing an XOR on the **SessionID** (always a **0**) and the **S** char from “Services”.



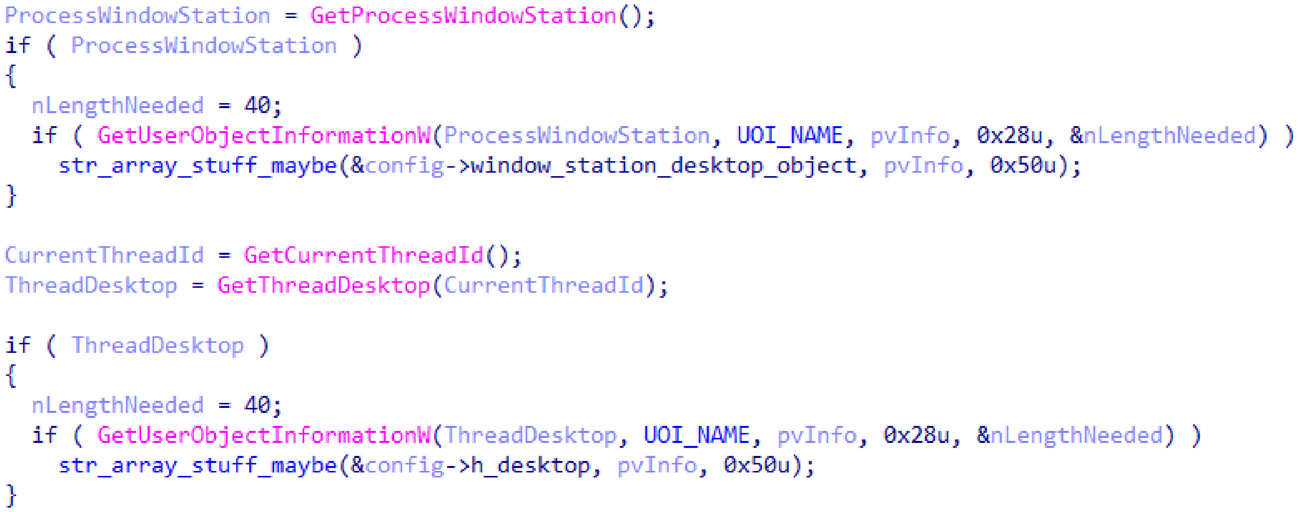
```
Hex                                                  ASCII
53 65 72 76 69 63 65 73 00 43 6F 6E 73 6F 6C 65      Services.Console
00 52 44 50 2D 54 63 70 00 AB AB AB AB AB AB AB      .RDP-Tcp.«««««««
AB 00 00 00 00 00 00 00 00 00 00 00 DA F6 B5 04      «...........Úöµ.
58 07 00 00 B0 68 52 02 D0 63 52 02 EE FE EE FE      X...°hR.ÐcR.îþîþ
EE FE EE FE EB F6 B6 36 50 07 00 1C 00 26 53 02      îþîþëö¶6P....&S.
01 00 00 00 FF FF FF FF FF FF FF FF 00 00 00 00      ....ÿÿÿÿÿÿÿÿ....
```

WTS_SESSION_INFO structure used as the initial seed for string decryption

Initial enumeration

Before sending any outbound network requests, LOBSHOT builds a custom structure containing enumerated data from the machine including:

- GUID of machine derived from **SOFTWARE\Microsoft\Cryptography\MachineGuid**
- Windows edition, username, computer name
- A VM check, number of processes running, process ID, parent process of malware
- Windows desktop object details
  - Screen height/width
  - Display device information
  - Handles to the desktop objects and windows
  - DPI for the display(s)

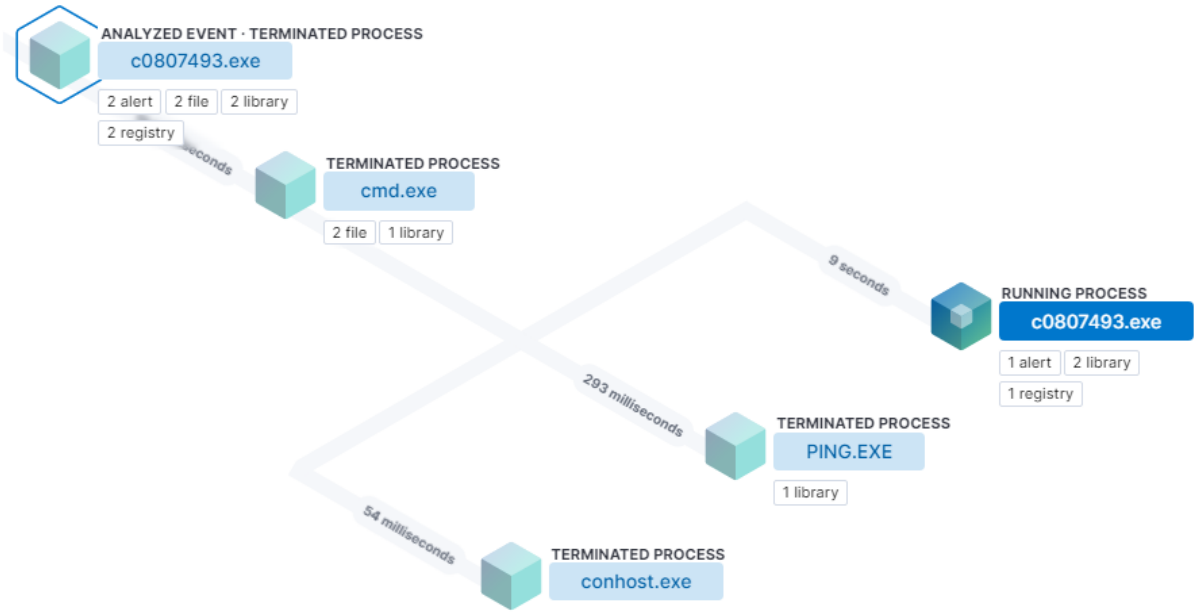


```
ProcessWindowStation = GetProcessWindowStation();
if ( ProcessWindowStation )
{
  nLengthNeeded = 40;
  if ( GetUserObjectInformationW(ProcessWindowStation, UOI_NAME, pvInfo, 0x28u, &nLengthNeeded) )
    str_array_stuff_maybe(&config->window_station_desktop_object, pvInfo, 0x50u);
}

CurrentThreadId = GetCurrentThreadId();
ThreadDesktop = GetThreadDesktop(CurrentThreadId);

if ( ThreadDesktop )
{
  nLengthNeeded = 40;
  if ( GetUserObjectInformationW(ThreadDesktop, UOI_NAME, pvInfo, 0x28u, &nLengthNeeded) )
    str_array_stuff_maybe(&config->h_desktop, pvInfo, 0x50u);
}
```

Malware retrieving Windows desktop object information

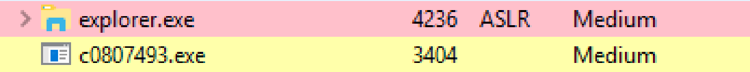
Execution flow

After LOBSHOT is executed, it moves a copy of itself to the **C:\ProgramData** folder, spawning a new process using **explorer.exe**, terminating the original process, and finally deleting the original file. This design choice is used in an attempt to break the process tree ancestry; making it harder to spot for analysts.



The LOBSHOT process tree as observed with Elastic Defend

Below is a screenshot after the initial execution, the malware is now parentless and running from the **C:\ProgramData** directory.

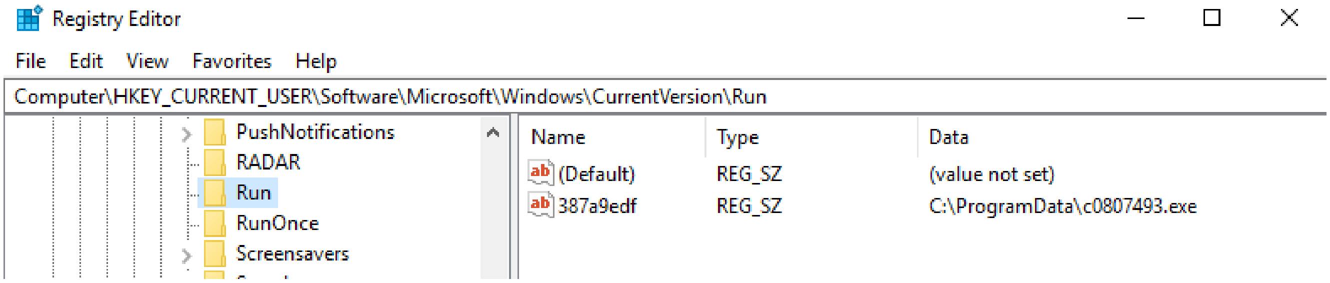


```
explorer.exe     4236  ASLR  Medium
c0807493.exe     3404        Medium
```

LOBSHOT running without a parent process

Persistence

For persistence, LOBSHOT leverages the Registry run key persistence method. In our sample, this is placed in the **HKEY_CURRENT_USER** Registry hive with a randomly generated name pointing to the malware located in **C:\ProgramData.**

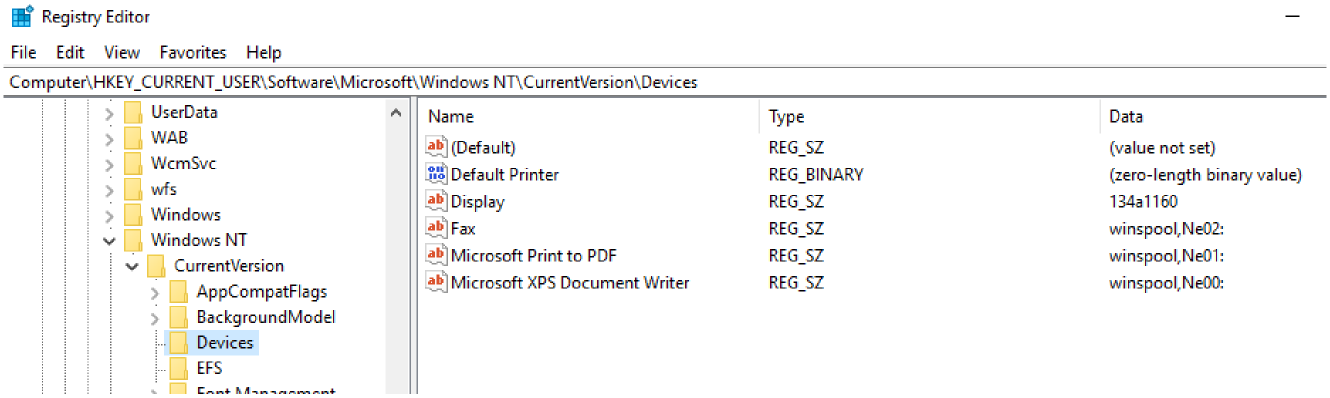


Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run

Name	Type	Data
(Default)	REG_SZ	(value not set)
387a9edf	REG_SZ	C:\ProgramData\c0807493.exe

Registry key persistence

In addition, it sets Registry key data under the **Software\Microsoft\Windows NT\CurrentVersion\Devices** key path which is used to check for a hardcoded global identifier key that would indicate the system had already been infected. In our sample, the **Display** value is set to the string **134a1160**. The results from the stealer feature are recorded inside the **Default Printer** value. We'll discuss the stealer functionality in the next section.



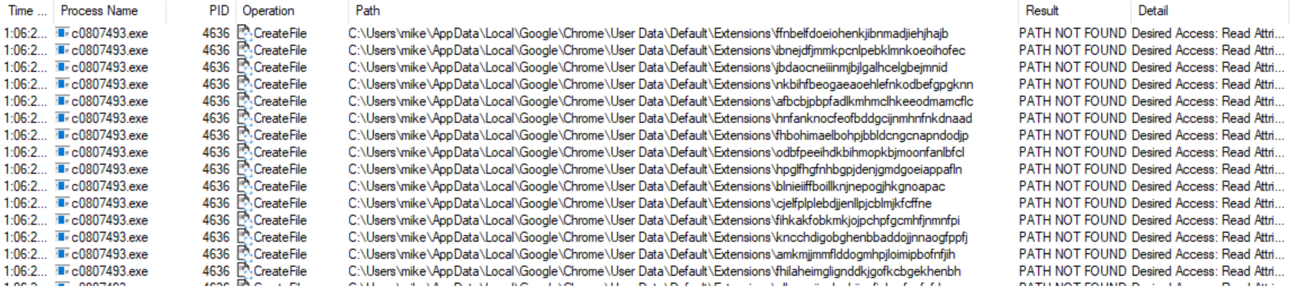
Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Devices

Name	Type	Data
(Default)	REG_SZ	(value not set)
Default Printer	REG_BINARY	(zero-length binary value)
Display	REG_SZ	134a1160
Fax	REG_SZ	winspool,Ne02:
Microsoft Print to PDF	REG_SZ	winspool,Ne01:
Microsoft XPS Document Writer	REG_SZ	winspool,Ne00:

Registry hive used to store data

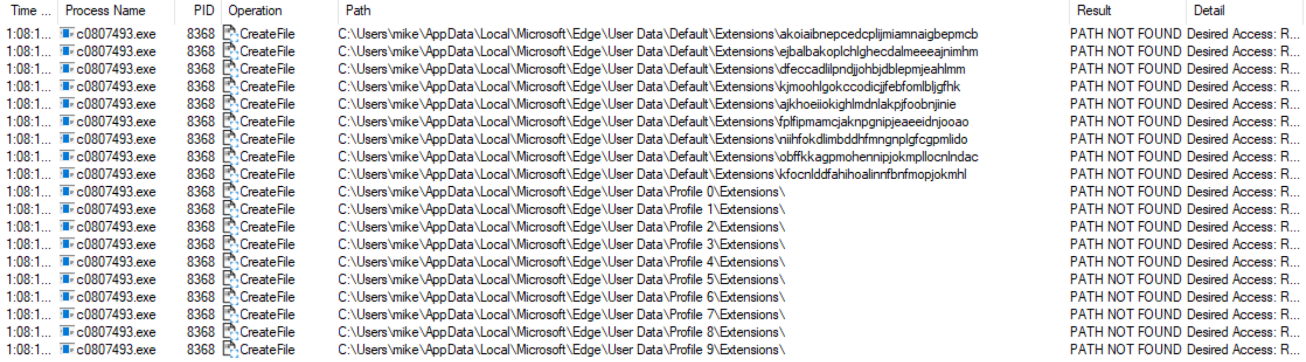
Stealer functionality

With the persistence mechanism established, LOBSHOT starts a new thread kicking off the stealer functionality. It starts by targeting specific Google Chrome extensions that deal with cryptocurrency wallets. Below are Procmon outputs showing LOBSHOT trying to access 32 Chrome wallet extensions, nine Edge wallet extensions, and 11 Firefox wallet extensions.



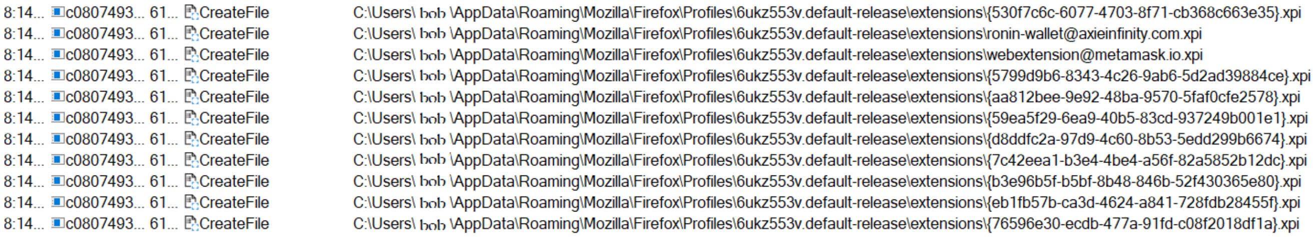
```
Time ...   Process Name      PID  Operation   Path                                                                                                        Result          Detail
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\ffnbelfdoeiohenkjibnmadjiehjhajb  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\ibnejdfjmmkpcnlpebklmnkoeoihofec  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\jbdaocneiiinmjbjlgalhcelgbejmnid  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\nkbihfbeogaeaoehlefnkodbefgpgknn  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\afbcbjpbpfadlkmhmclhkeeodmamcflc  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\hnfanknocfeofbddgcijnmhnfnkdnaad  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\fhbohimaelbohpjbbldcngcnapndodjp  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\odbfpeeihdkbihmopkbjmoonfanlbfcl  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\hpglfhgfnhbgpjdenjgmdgoeiappafln  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\blnieiiffboillknjnepogjhkgnoapac  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\cjelfplplebdjjenllpjcblmjkfcffne  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\fihkakfobkmkjojpchpfgcmhfjnmnfpi  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\kncchdigobghenbbaddojjnnaogfppfj  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\amkmjjmmflddogmhpjloimipbofnfjih  PATH NOT FOUND  Desired Access: Read Attri...
1:06:2...  c0807493.exe      4636 CreateFile  C:\Users\mike\AppData\Local\Google\Chrome\User Data\Default\Extensions\fhilaheimglignddkjgofkcbgekhenbh  PATH NOT FOUND  Desired Access: Read Attri...
```

Chrome extensions related to cryptocurrency wallets



```
Time ...   Process Name   PID  Operation   Path                                                                                                      Result          Detail
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\akoiaibnepcedcplijmiamnaigbepmcb  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\ejbalbakoplchlghecdalmeeeajnimhm  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\dfeccadlilpndjjohbjdblepmjeahlmm  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\kjmoohlgokccodicjjfebfomlbljgfhk  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\ajkhoeiiokighlmdnlakpjfoobnjinie  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\fplfipmamcjaknpgnipjeaeeidnjooao  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\niihfokdlimbddhfmngnplgfcgpmlido  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\obffkkagpmohennipjokmpllocnlndac  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Default\Extensions\kfocnlddfahihoalinnfbnfmopjokmhl  PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 0\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 1\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 2\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 3\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 4\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 5\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 6\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 7\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 8\Extensions\                               PATH NOT FOUND  Desired Access: R...
1:08:1...  c0807493.exe   8368 CreateFile  C:\Users\mike\AppData\Local\Microsoft\Edge\User Data\Profile 9\Extensions\                               PATH NOT FOUND  Desired Access: R...
```

Edge extensions related to cryptocurrency wallets

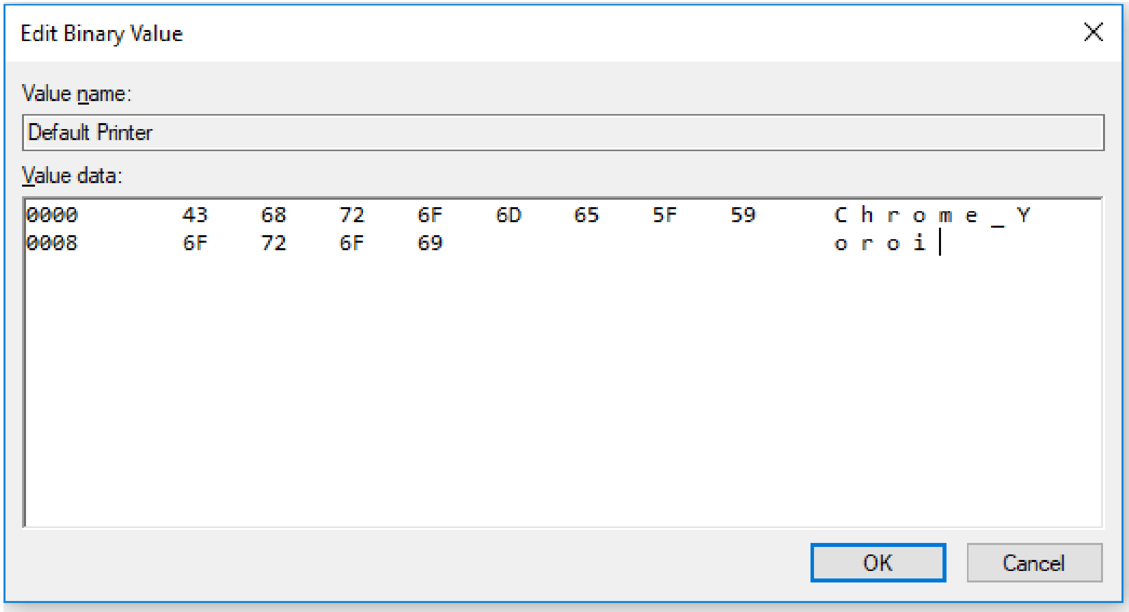


```
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{530f7c6c-6077-4703-8f71-cb368c663e35}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\ronin-wallet@axieinfinity.com.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\webextension@metamask.io.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{5799d9b6-8343-4c26-9ab6-5d2ad39884ce}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{aa812bee-9e92-48ba-9570-5faf0cfe2578}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{59ea5f29-6ea9-40b5-83cd-937249b001e1}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{d8ddfc2a-97d9-4c60-8b53-5edd299b6674}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{7c42eea1-b3e4-4be4-a56f-82a5852b12dc}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{b3e96b5f-b5bf-8b48-846b-52f430365e80}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{eb1fb57b-ca3d-4624-a841-728fdb28455f}.xpi
8:14...  c0807493...  61...  CreateFile  C:\Users\ bob \AppData\Roaming\Mozilla\Firefox\Profiles\6ukz553v.default-release\extensions\{76596e30-ecdb-477a-91fd-c08f2018df1a}.xpi
```

Firefox extensions related to cryptocurrency wallets

For the complete listing of the different cryptocurrencies mapped to their extension IDs, see the appendix.

If there is a match with any of these extensions, the results are inserted in the **Software\Microsoft\Windows NT\CurrentVersion\Devices** Registry key value as binary data with the format of browser name_extension name. Below is an example after the registry modification showing: **Chrome_Yoroi**.



Value name: Default Printer

Value data:

```
0000        43    68    72    6F    6D    65    5F    59      C h r o m e _ Y
0008        6F    72    6F    69                              o r o i
```

Stealer component writing found wallet to registry

After the browser extensions are enumerated, there is a check for a hardcoded file titled **hmr_*.dat** inside the **C:\ProgramData** directory. If this file exists it will be executed with **rundll32.exe** with the following command-line arguments:

```
rundll32.exe "C:\ProgramData\hmr_1.dat", #1 hmod
```

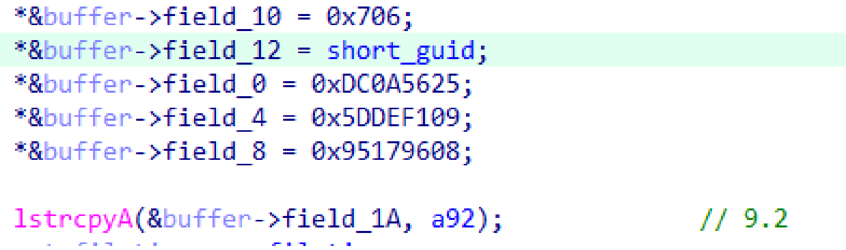
While we didn't observe this behavior directly, this function appears to show off a feature baked in by the developer allowing the ability for additional execution options such as loading their own DLL.

Network communications

For each LOBSHOT sample we have reviewed, there is a hardcoded IP and port that is decrypted from the binary that is used as the primary C2. The malware beacons every 5 seconds communicating by using the following calls:

- **ws2_32.socket**
- **ws2_32.connect**
- **ws2_32.send**
- **ws2_32.select**
- **ws2_32.recv**
- **ws2_32.shutdown**
- **ws2_32.closesocket**

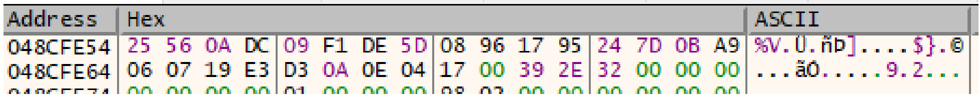
On these outbound requests, it sends pseudorandom hard-coded data along with a shortened GUID value and version number of the module.



```
*&buffer->field_10 = 0x706;
*&buffer->field_12 = short_guid;
*&buffer->field_0 = 0xDC0A5625;
*&buffer->field_4 = 0x5DDEF109;
*&buffer->field_8 = 0x95179608;

lstrcpyA(&buffer->field_1A, a92);              // 9.2
```

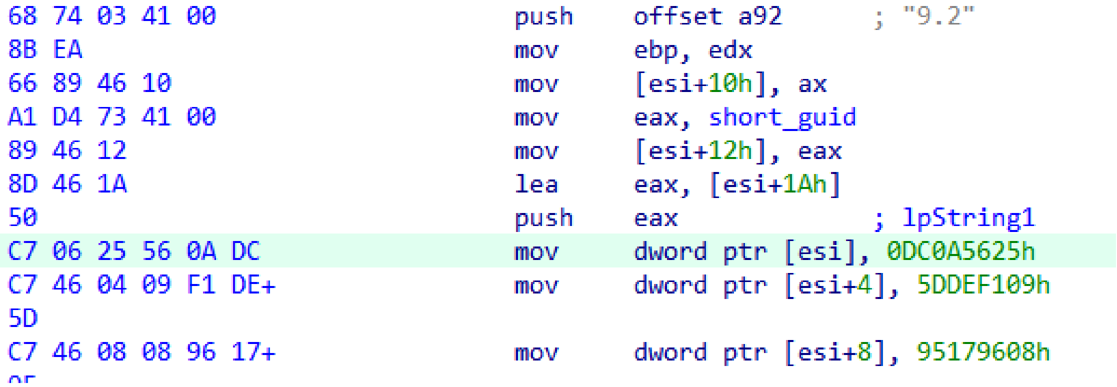
Hardcoded values and version in request

Below is an example of the send request buffer sent during the initial outbound requests showing the above-hardcoded values and version number.



```
Address  | Hex                                             | ASCII
048CFE54 | 25 56 0A DC|09 F1 DE 5D|08 96 17 95|24 7D 0B A9 | %V.Ü.ñÞ]....$}.©
048CFE64 | 06 07 19 E3|D3 0A 0E 04|17 00 39 2E|32 00 00 00 | ...ãÓ.....9.2...
```

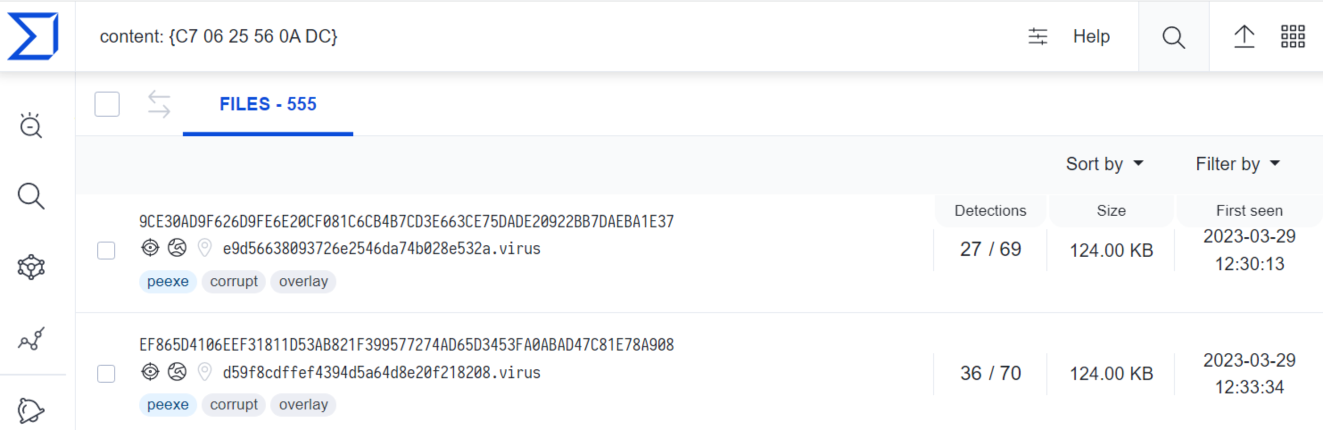
Request buffer on outbound network traffic



```
68 74 03 41 00          push    offset a92      ; "9.2"
8B EA                   mov     ebp, edx
66 89 46 10             mov     [esi+10h], ax
A1 D4 73 41 00          mov     eax, short_guid
89 46 12                mov     [esi+12h], eax
8D 46 1A                lea     eax, [esi+1Ah]
50                      push    eax             ; lpString1
C7 06 25 56 0A DC       mov     dword ptr [esi], 0DC0A5625h
C7 46 04 09 F1 DE+      mov     dword ptr [esi+4], 5DDEF109h
5D
C7 46 08 08 96 17+      mov     dword ptr [esi+8], 95179608h
```

Hardcoded values within outbound network traffic request

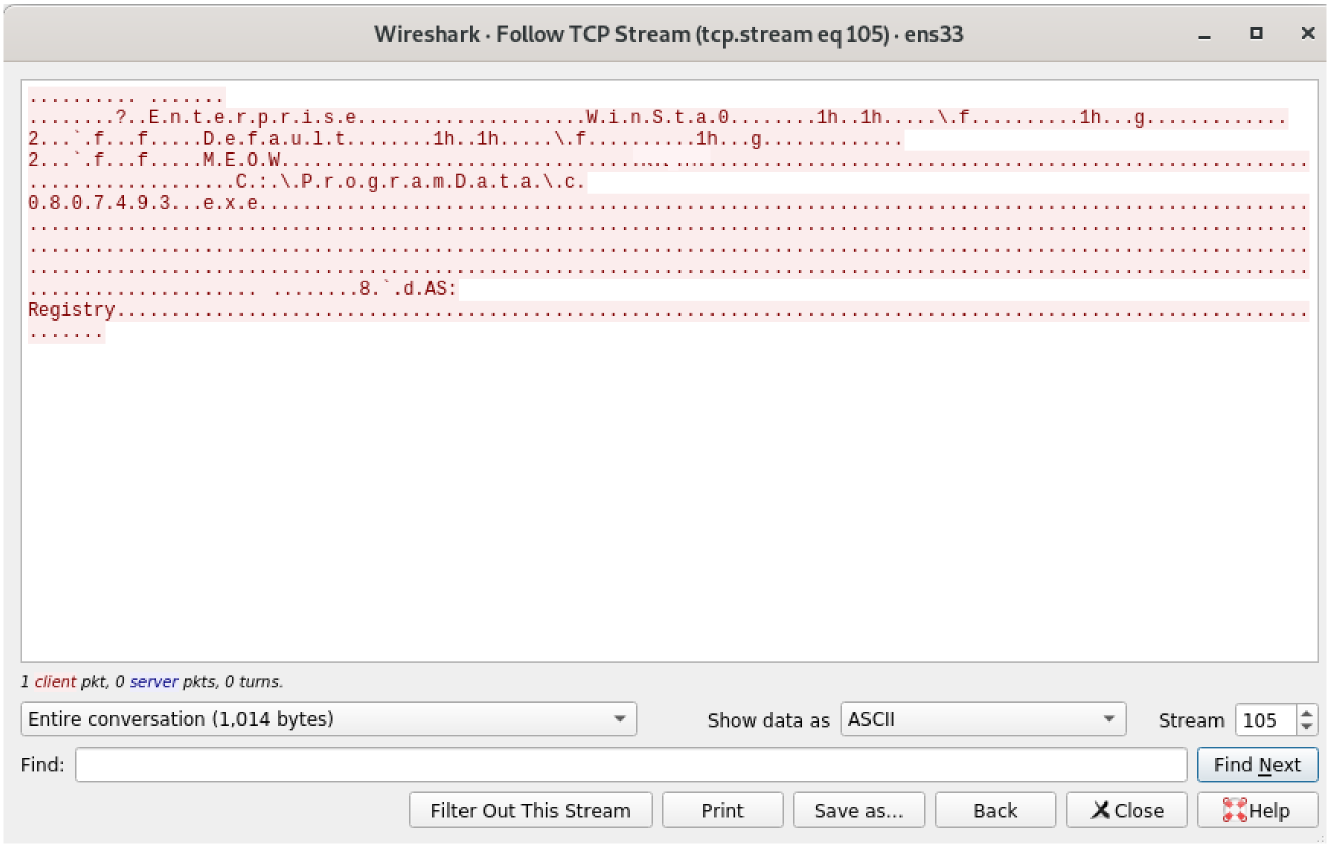
Searching for the above **mov** instruction paired with the first **DWORD** of the hardcoded value (**C7 06 25 56 0A DC**) shows over **550** samples in VirusTotal within the last year. With some of the first samples showing up in late July 2022. The prevalence of these hardcoded values shows that it has been actively used and under development for a long period of time, and will likely continue to be used in the future.



	Detections	Size	First seen
9CE30AD9F626D9FE6E20CF081C6CB4B7CD3E663CE75DADE20922BB7DAEBA1E37 e9d56638093726e2546da74b028e532a.virus peexe corrupt overlay	27 / 69	124.00 KB	2023-03-29 12:30:13
EF865D4106EEF31811D53AB821F399577274AD65D3453FA0ABAD47C81E78A908 d59f8cdffef4394d5a64d8e20f218208.virus peexe corrupt overlay	36 / 70	124.00 KB	2023-03-29 12:33:34

VirusTotal VTGrep search on hardcoded bytes

After this initial handshake, LOBSHOT will send the previous custom data structure containing the enumerated data such as the hostname, username, windows objects, etc. over this port.


Wireshark · Follow TCP Stream (tcp.stream eq 105) · ens33

```
.......... .......
........?..E.n.t.e.r.p.r.i.s.e.....................W.i.n.S.t.a.0........1h..1h.....\.f..........1h...g.............
2...`.f...f.....D.e.f.a.u.l.t........1h..1h.....\.f..........1h...g............
2...`.f...f.....M.E.O.W..............................................................................................
...................C.:.\.P.r.o.g.r.a.m.D.a.t.a.\.c.
0.8.0.7.4.9.3...e.x.e................................................................................................
.....................................................................................................................
.....................................................................................................................
.....................................................................................................................
..................... ........8.`.d.AS:
Registry.............................................................................................................
.......
```

1 client pkt, 0 server pkts, 0 turns.

Entire conversation (1,014 bytes) Show data as ASCII Stream 105

Find:

The Wireshark output of outbound requests containing victim host information

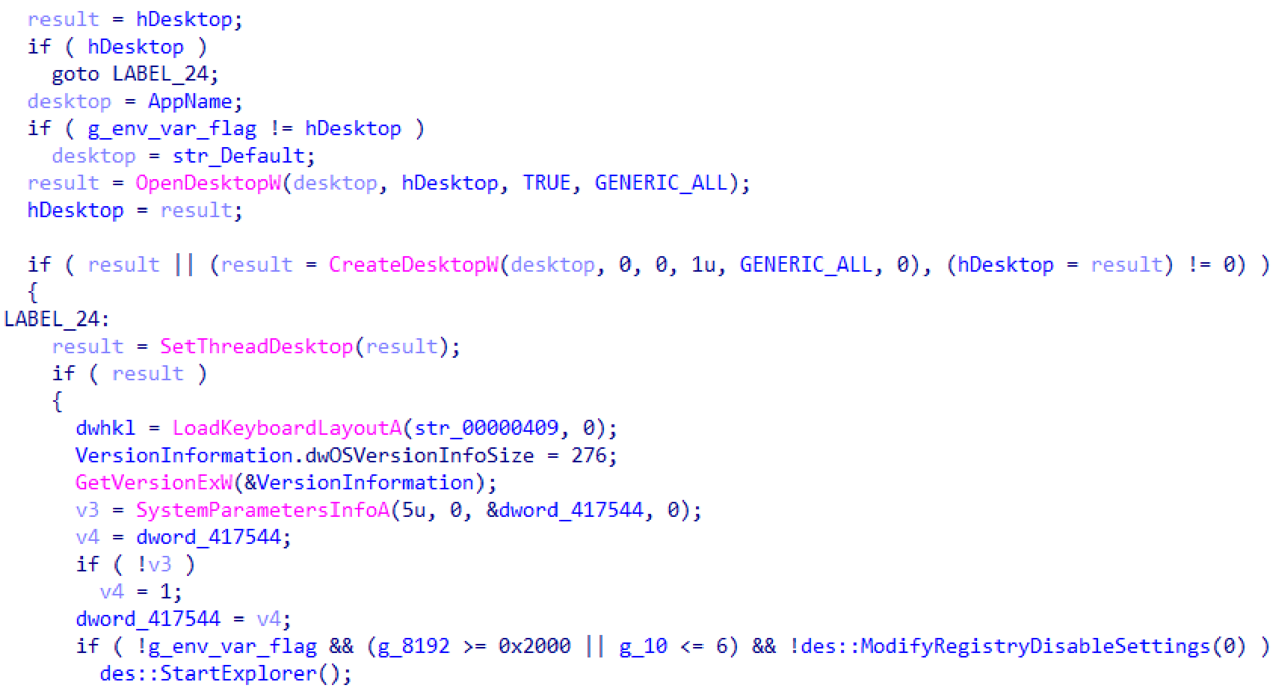
Capabilities

One of LOBSHOT’s core capabilities is around its hVNC (Hidden Virtual Network Computing) module.

hVNC overview

Unlike traditional VNC (Virtual Network Computing) where the software provides remote access to a machine with the user’s consent and the visibility of the actions taken on the machine can be clearly observed. hVNC acts in the opposite way designed to stay stealthy where all actions by an attacker are taking place on the same machine, but can’t be visibly observed by the victim. hVNC became a popular solution within the banking trojan space to bypass device and fraud detection solutions. More details on hVNC can be found here.

LOBSHOT implements the hVNC feature by generating a hidden desktop using the **CreateDesktopW** Windows API and then assigning the desktop to the malware using the **SetThreadDesktop** API**.** A new Windows **explorer.exe** process is then created under the context of the new hidden desktop.



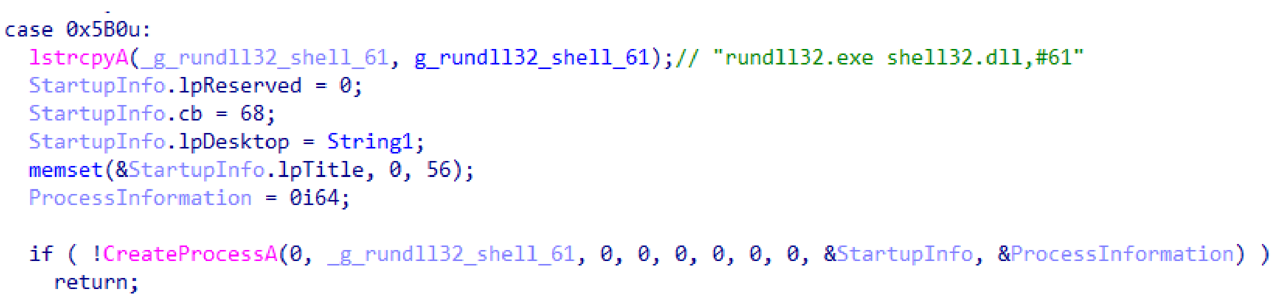
```
  result = hDesktop;
  if ( hDesktop )
    goto LABEL_24;
  desktop = AppName;
  if ( g_env_var_flag != hDesktop )
    desktop = str_Default;
  result = OpenDesktopW(desktop, hDesktop, TRUE, GENERIC_ALL);
  hDesktop = result;

  if ( result || (result = CreateDesktopW(desktop, 0, 0, 1u, GENERIC_ALL, 0), (hDesktop = result) != 0) )
  {
LABEL_24:
    result = SetThreadDesktop(result);
    if ( result )
    {
      dwhkl = LoadKeyboardLayoutA(str_00000409, 0);
      VersionInformation.dwOSVersionInfoSize = 276;
      GetVersionExW(&VersionInformation);
      v3 = SystemParametersInfoA(5u, 0, &dword_417544, 0);
      v4 = dword_417544;
      if ( !v3 )
        v4 = 1;
      dword_417544 = v4;
      if ( !g_env_var_flag && (g_8192 >= 0x2000 || g_10 <= 6) && !des::ModifyRegistryDisableSettings(0) )
        des::StartExplorer();
```

LOBSHOT’s hidden desktop creation

At this stage, the victim machine will start sending screen captures that represent the hidden desktop that is sent to a listening client controlled by the attacker. The attacker interacts with the client by controlling the keyboard, clicking buttons, and moving the mouse, these capabilities provide the attacker full remote control of the device.

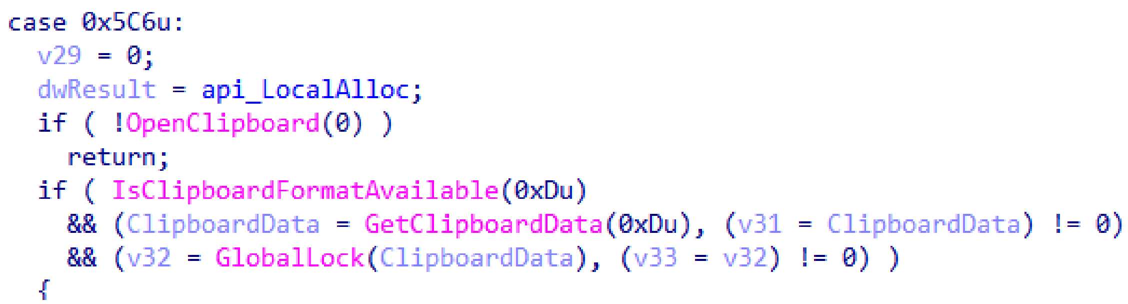
Within LOBSHOT’s hVNC module, there is a built-in GUI menu that allows the attacker to run the following commands quickly:

- Start new **explorer.exe** process
- Start Windows Run command
- Start new Windows process with provided command
- Start Browsers (Internet Explorer, Edge, Firefox)
- Terminate existing explorer.exe processes and start new explorer.exe process
- Tamper with Windows sound settings
- Set/retrieve Clipboard text
- Activate Start Menu
- Modify DPI Awareness settings



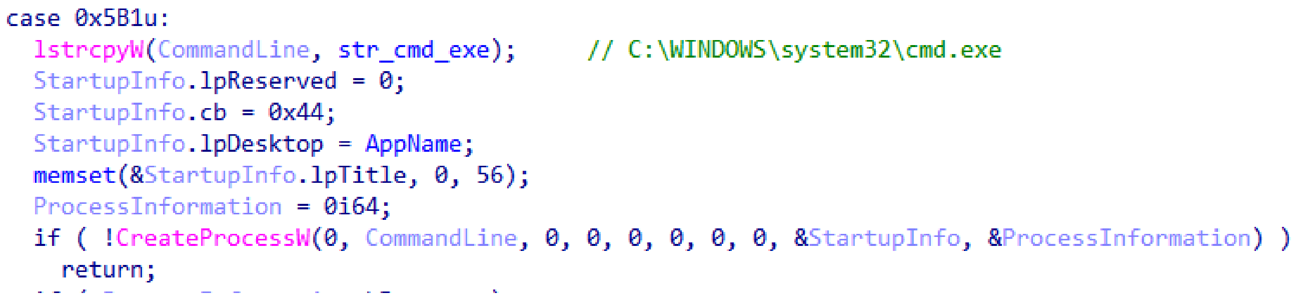
```
  case 0x5B0u:
    lstrcpyA(_g_rundll32_shell_61, g_rundll32_shell_61);// "rundll32.exe shell32.dll,#61"
    StartupInfo.lpReserved = 0;
    StartupInfo.cb = 68;
    StartupInfo.lpDesktop = String1;
    memset(&StartupInfo.lpTitle, 0, 56);
    ProcessInformation = 0i64;

    if ( !CreateProcessA(0, _g_rundll32_shell_61, 0, 0, 0, 0, 0, 0, &StartupInfo, &ProcessInformation) )
      return;
```

Execute the run dialog inside the hVNC module



```
case 0x5C6u:
  v29 = 0;
  dwResult = api_LocalAlloc;
  if ( !OpenClipboard(0) )
    return;
  if ( IsClipboardFormatAvailable(0xDu)
    && (ClipboardData = GetClipboardData(0xDu), (v31 = ClipboardData) != 0)
    && (v32 = GlobalLock(ClipboardData), (v33 = v32) != 0) )
  {
```

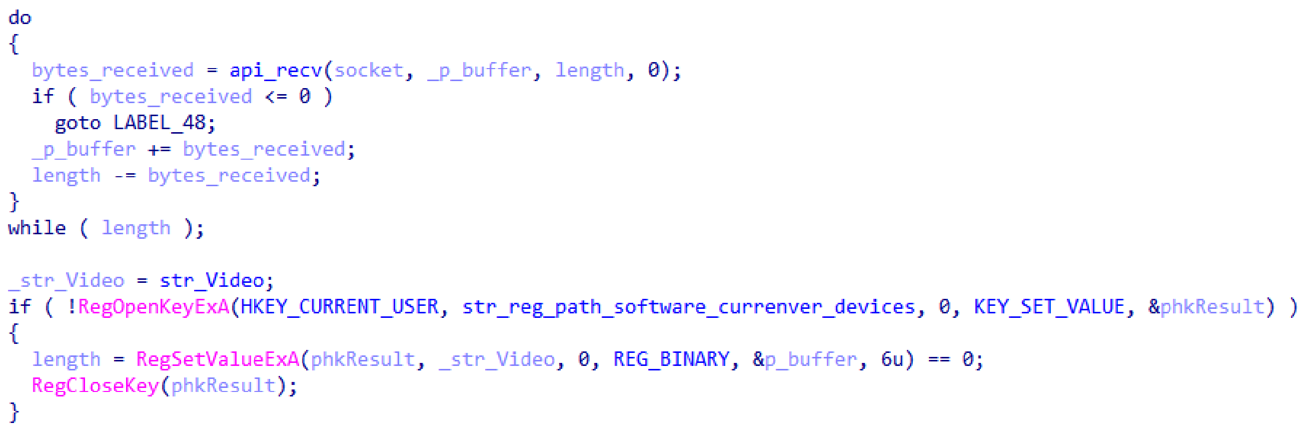
Clipboard grabber inside the hVNC module



```
case 0x5B1u:
  lstrcpyW(CommandLine, str_cmd_exe);     // C:\WINDOWS\system32\cmd.exe
  StartupInfo.lpReserved = 0;
  StartupInfo.cb = 0x44;
  StartupInfo.lpDesktop = AppName;
  memset(&StartupInfo.lpTitle, 0, 56);
  ProcessInformation = 0i64;
  if ( !CreateProcessW(0, CommandLine, 0, 0, 0, 0, 0, 0, &StartupInfo, &ProcessInformation) )
    return;
```

CMD execution inside the hVNC module

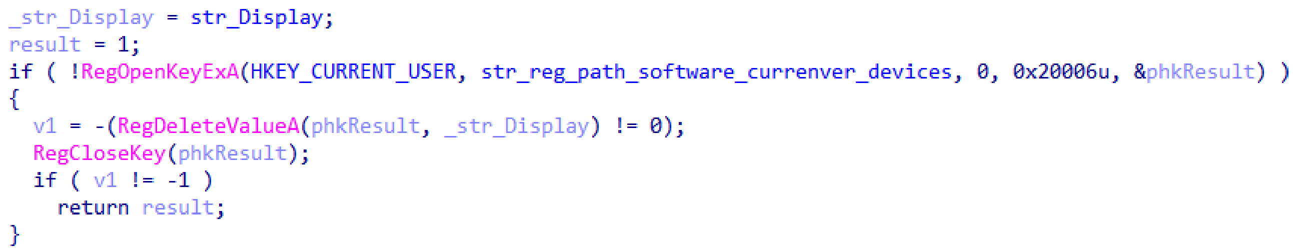
While the main functionality is centered on LOBSHOT's hVNC module, it does have additional capabilities. One example is its ability to swap out its C2 provided by an operator; it manages this by writing the new C2 details into the registry key path **Software\Microsoft\Windows NT\CurrentVersion\Devices** under the **Video** value.



```
do
{
  bytes_received = api_recv(socket, _p_buffer, length, 0);
  if ( bytes_received <= 0 )
    goto LABEL_48;
  _p_buffer += bytes_received;
  length -= bytes_received;
}
while ( length );

_str_Video = str_Video;
if ( !RegOpenKeyExA(HKEY_CURRENT_USER, str_reg_path_software_currenver_devices, 0, KEY_SET_VALUE, &phkResult) )
{
  length = RegSetValueExA(phkResult, _str_Video, 0, REG_BINARY, &p_buffer, 6u) == 0;
  RegCloseKey(phkResult);
}
```

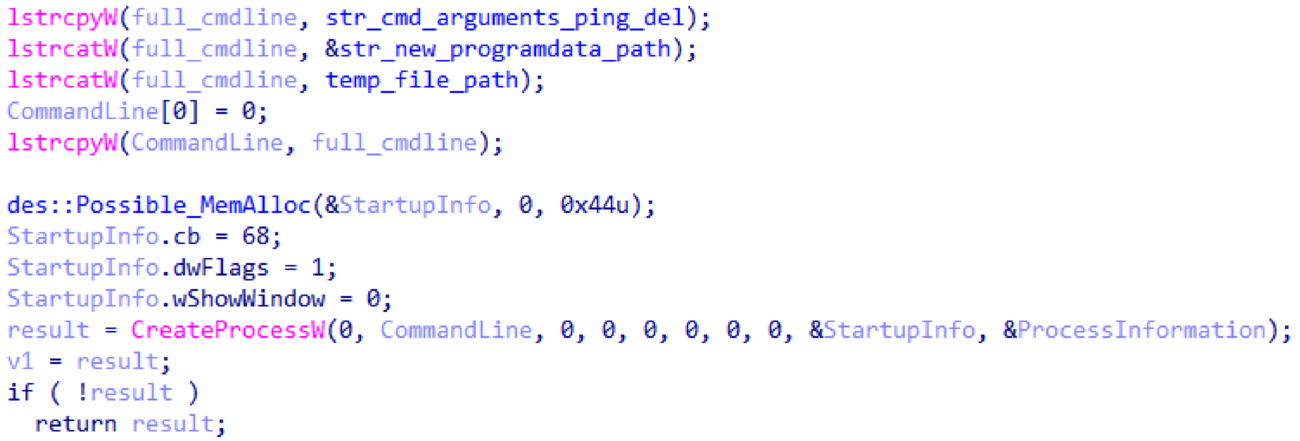
Updating C2 through registry modification

LOBSHOT also includes an update mechanism where it will remove previous modifications to the registry such as removing the "Display" value and Run key persistence, starting a new process, and finally exiting the existing process.



```
_str_Display = str_Display;
result = 1;
if ( !RegOpenKeyExA(HKEY_CURRENT_USER, str_reg_path_software_currenver_devices, 0, 0x20006u, &phkResult) )
{
  v1 = -(RegDeleteValueA(phkResult, _str_Display) != 0);
  RegCloseKey(phkResult);
  if ( v1 != -1 )
    return result;
}
```

Remove existing registry key

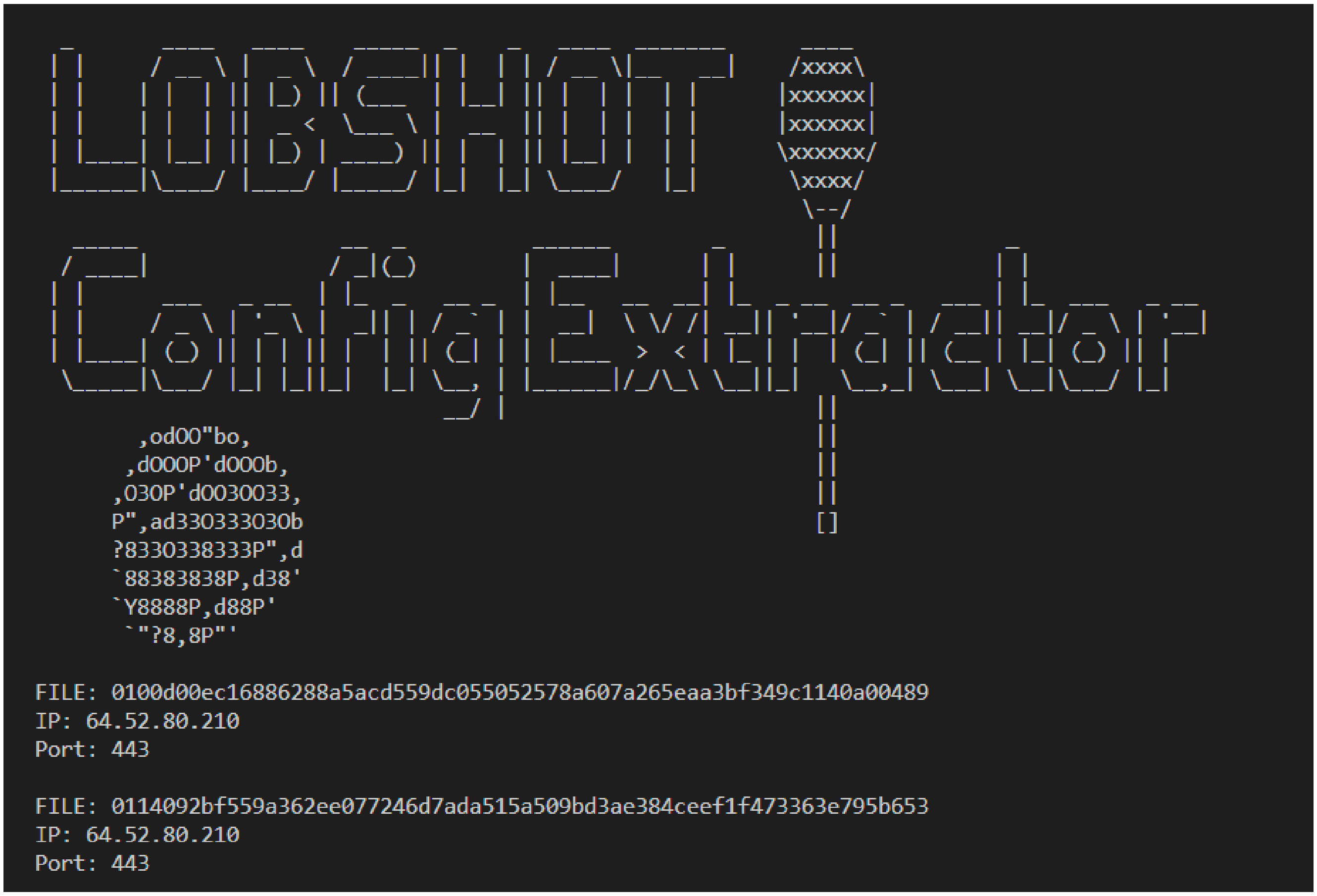


```
lstrcpyW(full_cmdline, str_cmd_arguments_ping_del);
lstrcatW(full_cmdline, &str_new_programdata_path);
lstrcatW(full_cmdline, temp_file_path);
CommandLine[0] = 0;
lstrcpyW(CommandLine, full_cmdline);

des::Possible_MemAlloc(&StartupInfo, 0, 0x44u);
StartupInfo.cb = 68;
StartupInfo.dwFlags = 1;
StartupInfo.wShowWindow = 0;
result = CreateProcessW(0, CommandLine, 0, 0, 0, 0, 0, 0, &StartupInfo, &ProcessInformation);
v1 = result;
if ( !result )
  return result;
```

Spawn new LOBSHOT process

LOBSHOT configuration extractor

Elastic Security Labs has released an open source tool, under the Elastic 2.0 license, that will allow for configurations to be extracted from LOBSHOT samples. The tool can be downloaded here.



```
  _      ____  ____   _____ _    _  ____ _______   ____
 | |    / __ \|  _ \ / ____| |  | |/ __ \__   __| /xxxx\
 | |   | |  | | |_) | (___ | |__| | |  | | | |   |xxxxxx|
 | |   | |  | |  _ < \___ \|  __  | |  | | | |   |xxxxxx|
 | |___| |__| | |_) |____) | |  | | |__| | | |   \xxxxxx/
 |______\____/|____/|_____/|_|  |_|\____/  |_|    \xxxx/
                                                   \--/
   _____             __ _       ______      _       ||         _
  / ____|           / _(_)     |  ____|    | |      ||        | |
 | |     ___  _ __ | |_ _  __ _| |__  __  _| |_ _ __ __ _  ___| |_ ___  _ __
 | |    / _ \| '_ \|  _| |/ _` |  __| \ \/ / __| '__/ _` |/ __| __/ _ \| '__|
 | |___| (_) | | | | | | | (_| | |____ >  <| |_| | | (_| | (__| || (_) | |
  \_____\___/|_| |_|_| |_|\__, |______/_/\_\\__|_|  \__,_|\___|\__\___/|_|
                           __/ |                    ||
       ,od00"bo,                                    ||
      ,d000P'd000b,                                 ||
     ,030P'd0030033,                                ||
     P",ad330333030b                                []
     ?8330338333P",d
     `88383838P,d38'
     `Y8888P,d88P'
      `"?8,8P"'

FILE: 0100d00ec16886288a5acd559dc055052578a607a265eaa3bf349c1140a00489
IP: 64.52.80.210
Port: 443

FILE: 0114092bf559a362ee077246d7ada515a509bd3ae384ceef1f473363e795b653
IP: 64.52.80.210
Port: 443
```

The extractor can run at the individual file or directory level, examples are below:

- **python lobshot_config_extractor.py --file sample.bin**
- **python lobshot_config_extractor.py --directory samples**

Summary

Threat groups are continuing to leverage malvertising techniques to masquerade legitimate software with backdoors like LOBSHOT. These kinds of malware seem small, but end up packing significant functionality which helps threat actors move quickly during the initial access stages with fully interactive remote control capabilities. We are continuing to see new samples related to this family each week, and expect it to be around for some time.

Detection logic

Detection

EQL query

Using the Timeline section of the Security Solution in Kibana under the “Correlation” tab, you can use the below EQL queries to hunt for behaviors similar

The following EQL query can be used to detect suspicious grandparent, parent, child relationships observed with LOBSHOT.

```
sequence by host.id, user.id with maxspan=1m
  [process where event.type == "start" and not startsWith~(process.executable,
process.parent.executable)] by process.parent.name, process.entity_id
  [file where event.type == "deletion"] by file.name, process.entity_id
  [process where event.type == "start" and not startsWith~(process.executable,
process.parent.executable)] by process.name, process.parent.entity_id
until [process where event.type == "end"] by process.name, process.entity_id
```

YARA rule

```
rule Windows_Trojan_Lobshot {
    meta:
        author = "Elastic Security"
        creation_date = "2023-04-18"
        last_modified = "2023-04-18"
        license = "Elastic License v2"
        os = "Windows"
        threat_name = "Windows.Trojan.Lobshot"
        reference_sample =
"e4ea88887753a936eaf3361dcc00380b88b0c210dcbde24f8f7ce27991856bf6"
    strings:
        $str0 = "HVNC Remote Control" ascii fullword
        $str1 = " Error # %d - %08lx" ascii fullword
        $str2 = "Set  clipboard text failed." ascii fullword
        $str3 = "OK %08lx %08lx %d" ascii fullword
        $str4 = "\") & (rundll32.exe \"" wide fullword
        $str5 = "%LOCALAPPDATA%\\svc.db" wide fullword
        $str6 = "cmd.exe /c (ping -n 10 127.0.0.1) & (del /F /Q \"" wide fullword
        $seq_str_decrypt = { 8A 5A ?? 8D 52 ?? 80 EB ?? 85 FF 74 ?? C0 E0 ?? 2C ?? 0A C3 32
C1 32 C7 88 06 32 E8 83 C6 ?? 83 C5 ?? EB ?? }
        $seq_emu_check = { 8B 35 ?? ?? ?? ?? 8D 44 24 ?? 50 8D 44 24 ?? C7 44 24 ?? 48 41 4C
39 50 C7 44 24 ?? 54 48 00 00 FF D6 }
        $seq_enum_xor = { FF 15 ?? ?? ?? ?? 84 C0 0F 84 ?? ?? ?? ?? 83 7C 24 ?? 00 0F 84 ??
?? ?? ?? 8B 4C 24 ?? 68 07 80 00 00 8B 41 ?? 8A 00 32 01 A2 ?? ?? ?? ?? }
        $seq_create_guid = { 8D 48 ?? 80 F9 ?? 77 ?? 2C ?? C1 E2 ?? 46 0F B6 C8 0B D1 83 FE
?? 7C ?? 5F 8B C2 5E C3 }
    condition:
        2 of ($seq*) or 5 of ($str*)
}Weitere Informationen
```

Observed adversary tactics and techniques

Elastic uses the MITRE ATT&CK framework to document common tactics, techniques, and procedures that advanced persistent threats use against enterprise networks.

Observations

All observables are also available for download in both ECS and STIX format. Additionally, we have created a VirusTotal Collection with all indicators.

Indicator	Type	Reference
95.217.125.200	IP Address	LOBSHOT C2
e4ea88887753a936eaf3361dcc00380b88b0c210dcbde24f8f7ce27991856bf6	SHA-256	LOBSHOT

Appendix

Chrome wallet extensions

Wallet name	Extension ID
Yoroi	ffnbelfdoeiohenkjibnmadjiehjhajb
TronLink	ibnejdfjmmkpcnlpebklmnkoeoihofec
Nifty Wallet	jbdaocneiiinmjbjlgalhcelgbejmnid
MetaMask	nkbihfbeogaeaoehlefnkodbefgpgknn
Math Wallet	afbcbjpbpfadlkmhmclhkeeodmamcflc
Coinbase Wallet	hnfanknocfeofbddgcijnmhnfnkdnaad
Binance Wallet	fhbohimaelbohpjbbldcngcnapndodjp
Brave Wallet	odbfpeeihdkbihmopkbjmoonfanlbfcl
Guarda	hpglfhgfnhbgpjdenjgmdgoeiappafln
Equal Wallet	blnieiiffboillknjnepogjhkgnoapac
Jaxx Liberty	cjelfplplebdjjenllpjcblmjkfcffne
BitApp Wallet	fihkakfobkmkjojpchpfgcmhfjnmnfpi
iWallet	kncchdigobghenbbaddojjnnaogfppfj
Wombat	amkmjjmmflddogmhpjloimipbofnfjih
Oxygen	fhilaheimglignddkjgofkcbgekhenbh
MyEtherWallet	nlbmnnijcnlegkjjpcfjclmcfggfefdm
GuildWallet	nanjmdknhkinifnkgdcggcfnhdaammmj
Saturn Wallet	nkddgncdjgjfcddamfgcmfnlhccnimig
Ronin Wallet	fnjhmkhhmkbjkkabndcnnogagogbneec
Station Wallet	aiifbnbfobpmeekipheeijimdpnlpgpp
Harmony	fnnegphlobjdpkhecapkijjdkgcjhkib
Coin98 Wallet	aeachknmefphepccionboohckonoeemg
EVER Wallet	cgeeodpfagjceefieflmdfphplkenlfk
KardiaChain Wallet	pdadjkfkgcafgbceimcpbkalnfnepbnk
Phantom	bfnaelmomeimhlpmgjnjophhpkkoljpa
Pali Wallet	mgffkfbidihjpoaomajlbgchddlicgpn
BOLT X	aodkkagnadcbobfpggfnjeongemjbjca

Wallet name	Extension ID
Liquality Wallet	kpfopkelmapcoipemfendmdcghnegimn
XDEFI Wallet	hmeobnfnfcmdkdcmlblgagmfpfboieaf
Nami	lpfcbjknijpeeillifnkikgncikgfhdo
MultiversX DeFi Wallet	dngmlblcodfobpdpecaadgfbcggfjfnm

Edge wallet extensions

Wallet name	Extension ID
Yoroi	akoiaibnepcedcplijmiamnaigbepmcb
MetaMask	ejbalbakoplchlghecdalmeeeajnimhm
Math Wallet	dfeccadlilpndjjohbjdblepmjeahlmm
Ronin Wallet	kjmoohlgokccodicjjfebfomlbljgfhk
Station Wallet	ajkhoeiiokighlmdnlakpjfoobnjinie
BDLT Wallet	fplfipmamcjaknpgnipjeaeeidnjooao
Glow	niihfokdlimbddhfmngnplgfcgpmlido
OneKey	obffkkagpmohennipjokmpllocnlndac
MetaWallet	kfocnlddfahihoalinnfbnfmopjokmhl

Firefox wallet extensions

Wallet name	Extension ID
Yoroi	{530f7c6c-6077-4703-8f71-cb368c663e35}.xpi
Ronin Wallet	ronin-wallet@axieinfinity.com.xpi
MetaMask	webextension@metamask.io.xpi
TronLink	{5799d9b6-8343-4c26-9ab6-5d2ad39884ce}.xpi
	{aa812bee-9e92-48ba-9570-5faf0cfe2578}.xpi
	{59ea5f29-6ea9-40b5-83cd-937249b001e1}.xpi
	{d8ddfc2a-97d9-4c60-8b53-5edd299b6674}.xpi
Phantom	{7c42eea1-b3e4-4be4-a56f-82a5852b12dc}.xpi
	{b3e96b5f-b5bf-8b48-846b-52f430365e80}.xpi

Wallet name	Extension ID
	{eb1fb57b-ca3d-4624-a841-728fdb28455f}.xpi
	{76596e30-ecdb-477a-91fd-c08f2018df1a}.xpi