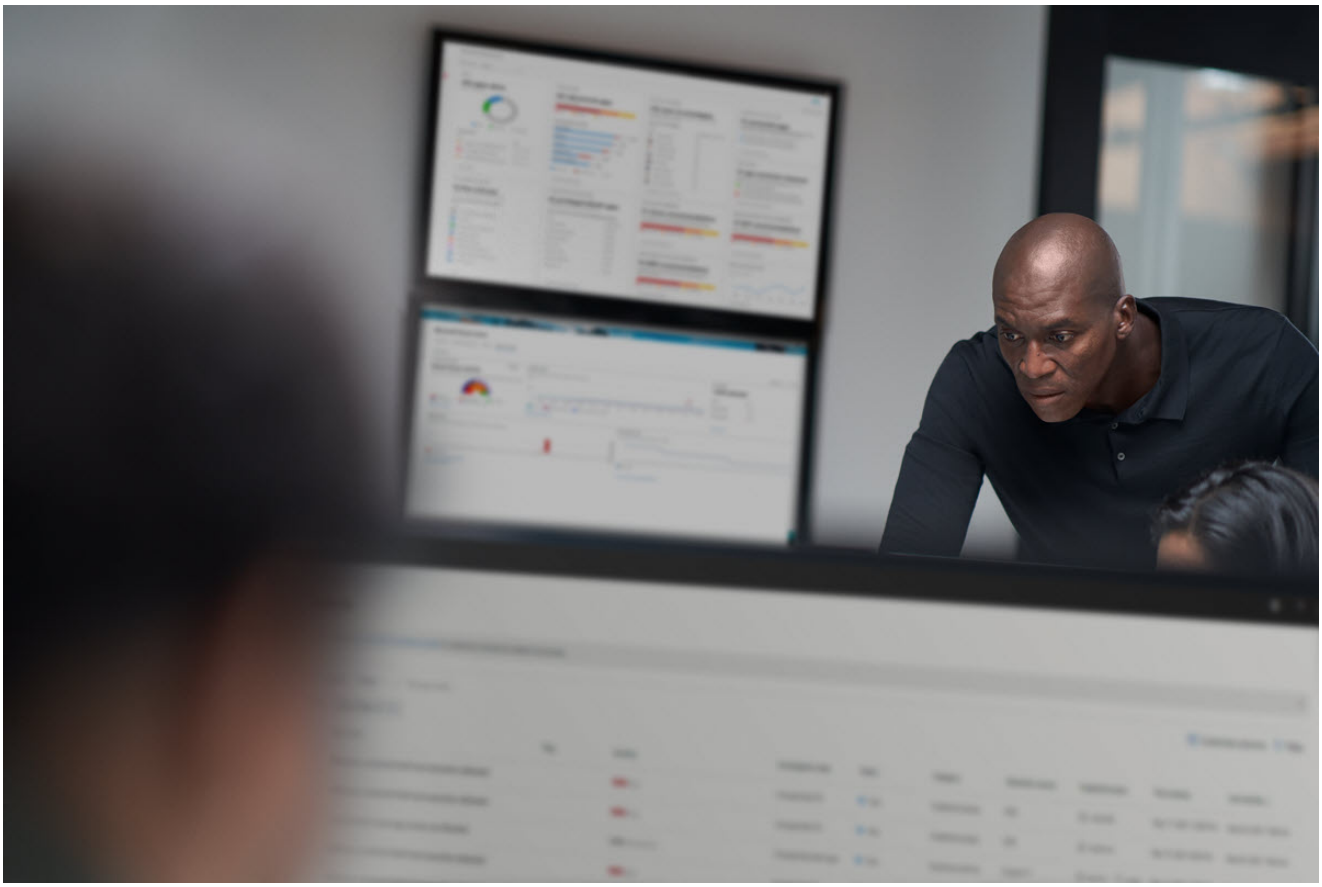
The many lives of BlackCat ransomware

microsoft.com/security/blog/2022/06/13/the-many-lives-of-blackcat-ransomware/

June 13, 2022



The BlackCat ransomware, also known as ALPHV, is a prevalent threat and a prime example of the growing ransomware-as-a-service (RaaS) gig economy. It's noteworthy due to its unconventional programming language (Rust), multiple target devices and possible entry points, and affiliation with prolific threat activity groups. While BlackCat's arrival and execution vary based on the actors deploying it, the outcome is the same—target data is encrypted, exfiltrated, and used for "double extortion," where attackers threaten to release the stolen data to the public if the ransom isn't paid.

First observed in November 2021, BlackCat initially made headlines because it was one of the first ransomware families written in the Rust programming language. By using a modern language for its payload, this ransomware attempts to evade detection, especially by conventional security solutions that might still be catching up in their ability to analyze and parse binaries written in such language. BlackCat can also target multiple devices and operating systems. Microsoft has observed successful attacks against Windows and Linux devices and VMWare instances.

As we previously explained, the RaaS affiliate model consists of multiple players: access brokers, who compromise networks and maintain persistence; RaaS operators, who develop tools; and RaaS affiliates, who perform other activities like moving laterally across the network and exfiltrating data before ultimately launching the ransomware payload. Thus, as a RaaS payload, how BlackCat enters a target organization's network varies, depending on the RaaS affiliate that deploys it. For example, while the common entry vectors for these threat actors include remote desktop applications and compromised credentials, we also saw a threat actor leverage Exchange server vulnerabilities to gain target network access. In addition, at least two known affiliates are now adopting BlackCat: DEV-0237 (known for previously deploying Ryuk, Conti, and Hive) and DEV-0504 (previously deployed Ryuk, REvil, BlackMatter, and Conti).

Such variations and adoptions markedly increase an organization's risk of encountering BlackCat and pose challenges in detecting and defending against it because these actors and groups have different tactics, techniques, and procedures (TTPs). Thus, no two BlackCat "lives" or deployments might look the same. Indeed, based on Microsoft threat data, the impact of this ransomware has been noted in various countries and regions in Africa, the Americas, Asia, and Europe.

Human-operated ransomware attacks like those that deploy BlackCat continue to evolve and remain one of the attackers' preferred methods to monetize their attacks. Organizations should consider complementing their security best practices and policies with a comprehensive solution like Microsoft 365 Defender, which offers protection capabilities that correlate various threat signals to detect and block such attacks and their follow-on activities.

In this blog, we provide details about the ransomware's techniques and capabilities. We also take a deep dive into two incidents we've observed where BlackCat was deployed, as well as additional information about the threat activity groups that now deliver it. Finally, we offer best practices and recommendations to help defenders protect their organizations against this threat, including hunting queries and product-specific mitigations.

BlackCat's anatomy: Payload capabilities

As mentioned earlier, BlackCat is one of the first ransomware written in the Rust programming language. Its use of a modern language exemplifies a recent trend where threat actors switch to languages like Rust or Go for their payloads in their attempt to not only avoid detection by conventional security solutions but also to challenge defenders who may be trying to reverse engineer the said payloads or compare them to similar threats.

BlackCat can target and encrypt Windows and Linux devices and VMWare instances. It has extensive capabilities, including self-propagation configurable by an affiliate for their usage and to environment encountered.

In the instances we've observed where the BlackCat payload did not have administrator privileges, the payload was launched via *dllhost.exe*, which then launched the following commands below (Table 1) via *cmd.exe*. These commands could vary, as the BlackCat payload allows affiliates to customize execution to the environment.

The flags used by the attackers and the options available were the following: *-s -d -f -c*; *–access-token*; *–propagated*; *-no-prop-servers*

```
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
    -h, --help                                     Print help information
        --log-file <LOG_FILE>                      Enable logging to specified file
        --no-net                                   Do not discover network shares
on Windows
        --no-prop                                  Do not self propagate(worm) on
Windows
        --no-prop-servers <NO_PROP_SERVERS>...     Do not propagate to defined
servers
        --no-vm-kill                               Do not stop VMs on ESXi
        --no-vm-kill-names <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 <PATHS>...                         Only process files inside
defined paths
        --propagated                               Run as propagated process
        --ui                                       Show user interface
    -v, --verbose                                  Log to console
```

Figure 1. BlackCat payload deployment options

Command	Description
[service name] /stop	Stops running services to allow encryption of data
vssadmin.exe Delete Shadows /all /quiet	Deletes backups to prevent recovery
wmic.exe Shadowcopy Delete	Deletes shadow copies

wmic csproduct get UUID	Gets the Universally Unique Identifier (UUID) of the target device
reg add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters /v MaxMpxCt /d 65535 /t REG_DWORD /f	Modifies the registry to change MaxMpxCt settings; BlackCat does this to increase the number of outstanding requests allowed (for example, SMB requests when distributing ransomware via its PsExec methodology)
for /F \"tokens=*\" %1 in ('wevtutil.exe el') DO wevtutil.exe cl \"%1\"	Clears event logs
fsutil behavior set SymlinkEvaluation R2L:1	Allows remote-to-local symbolic links; a symbolic link is a file-system object (for example, a file or folder) that points to another file system object, like a shortcut in many ways but more powerful
fsutil behavior set SymlinkEvaluation R2R:1	Allows remote-to-remote symbolic links
net use \\[computer name] /user:[domain]\[user] [password] /persistent:no	Mounts network share

Table 1. List of commands the BlackCat payload can run

User account control (UAC) bypass

BlackCat can bypass UAC, which means the payload will successfully run even if it runs from a non-administrator context. If the ransomware isn't run with administrative privileges, it runs a secondary process under *dllhost.exe* with sufficient permissions needed to encrypt the maximum number of files on the system.

Domain and device enumeration

The ransomware can determine the computer name of the given system, local drives on a device, and the AD domain name and username on a device. The malware can also identify whether a user has domain admin privileges, thus increasing its capability of ransoming more devices.

Self-propagation

BlackCat discovers all servers that are connected to a network. The process first broadcasts NetBIOS Name Service (NBNC) messages to check for these additional devices. The ransomware then attempts to replicate itself on the answering servers using the credentials specified within the config via PsExec.

Hampering recovery efforts

BlackCat has numerous methods to make recovery efforts more difficult. The following are commands that might be launched by the payload, as well as their purposes:

- Modify boot loader
  - “C:\Windows\system32\cmd.exe” /c “bcdedit /set {default}”
  - “C:\Windows\system32\cmd.exe” /c “bcdedit /set {default} recoveryenabled No”
- Delete volume shadow copies
  - “C:\Windows\system32\cmd.exe” /c “vssadmin.exe Delete Shadows /all /quiet”
  - “C:\Windows\system32\cmd.exe” /c “wmic.exe Shadowcopy Delete”
- Clear Windows event logs
  - “C:\Windows\system32\cmd.exe” /c “cmd.exe /c for /F \”tokens=*\” Incorrect function. in (‘ wevtutil.exe el ‘) DO wevtutil.exe cl \”Incorrect function. \””

Slinking its way in: Identifying attacks that can lead to BlackCat ransomware

Consistent with the RaaS model, threat actors utilize BlackCat as an additional payload to their ongoing campaigns. While their TTPs remain largely the same (for example, using tools like Mimikatz and PsExec to deploy the ransomware payload), BlackCat-related compromises have varying entry vectors, depending on the ransomware affiliate conducting the attack. Therefore, the pre-ransom steps of these attacks can also be markedly different.

For example, our research noted that one affiliate that deployed BlackCat leveraged unpatched Exchange servers or used stolen credentials to access target networks. The following sections detail the end-to-end attack chains of these two incidents we’ve observed.

Case study 1: Entry via unpatched Exchange

In one incident we’ve observed, attackers took advantage of an unpatched Exchange server to enter the target organization.

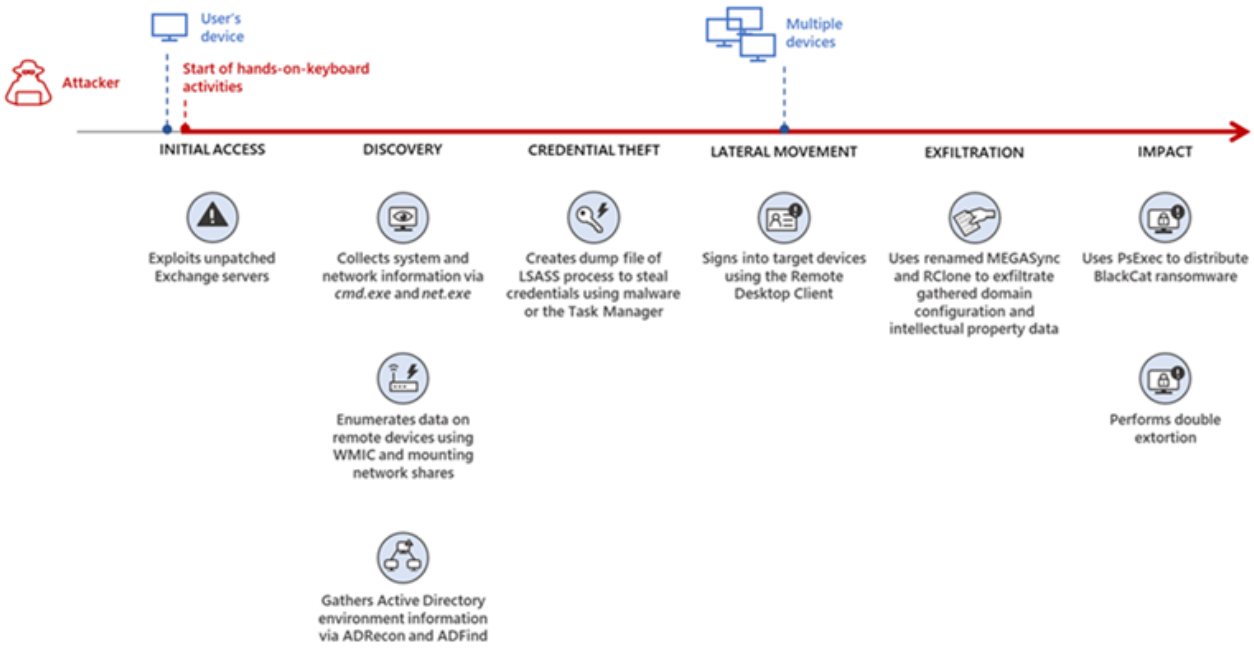


Figure 2. Observed BlackCat ransomware attack chain via Exchange vulnerability exploitation

Discovery

Upon exploiting the Exchange vulnerability, the attackers launched the following discovery commands to gather information about the device they had compromised:

- *cmd.exe* and the commands *ver* and *systeminfo* – to collect operating system information
- *net.exe* – to determine domain computers, domain controllers, and domain admins in the environment

After executing these commands, the attackers navigated through directories and discovered a passwords folder that granted them access to account credentials they could use in the subsequent stages of the attack. They also used the *del* command to delete files related to their initial compromise activity.

The attackers then mounted a network share using *net use* and the stolen credentials and began looking for potential lateral movement targets using a combination of methods. First, they used *WMIC.exe* using the previously gathered device name as the node, launched the command *whoami /all*, and pinged *google.com* to check network connectivity. The output of the results were then written to a *.log* file on the mounted share. Second, the attackers used *PowerShell.exe* with the cmdlet *Get-ADComputer* and a filter to gather the last sign-in event.

Lateral movement

Two and a half days later, the attackers signed into one of the target devices they found during their initial discovery efforts using compromised credentials via interactive sign-in. They opted for a credential theft technique that didn't require dropping a file like Mimikatz that antivirus products might detect. Instead, they opened *Taskmgr.exe*, created a dump file of the *LSASS.exe* process, and saved the file to a ZIP archive.

The attackers continued their previous discovery efforts using a PowerShell script version of ADRecon (*ADRecon.ps1*), which is a tool designed to gather extensive information about an Active Directory (AD) environment. The attacker followed up this action with a net scanning tool that opened connections to devices in the organization on server message block (SMB) and remote desktop protocol (RDP). For discovered devices, the attackers attempted to navigate to various network shares and used the Remote Desktop client (*mstsc.exe*) to sign into these devices, once again using the compromised account credentials.

These behaviors continued for days, with the attackers signing into numerous devices throughout the organization, dumping credentials, and determining what devices they could access.

Collection and exfiltration

On many of the devices the attackers signed into, efforts were made to collect and exfiltrate extensive amounts of data from the organization, including domain settings and information and intellectual property. To do this, the attackers used both MEGAsync and Rclone, which were renamed as legitimate Windows process names (for example, *winlogon.exe*, *mstsc.exe*).

Exfiltration of domain information to identify targets for lateral movement

Collecting domain information allowed the attackers to progress further in their attack because the said information could identify potential targets for lateral movement or those that would help the attackers distribute their ransomware payload. To do this, the attackers once again used *ADRecon.ps1*with numerous PowerShell cmdlets such as the following:

- *Get-ADRGPO* – gets group policy objects (GPO) in a domain
- *Get-ADRDNSZone* – gets all DNS zones and records in a domain
- *Get-ADRGPLink* – gets all group policy links applied to a scope of management in a domain

Additionally, the attackers dropped and used *ADFind.exe* commands to gather information on persons, computers, organizational units, and trust information, as well as pinged dozens of devices to check connectivity.

Exfiltration for double extortion

Intellectual property theft likely allowed the attackers to threaten the release of information if the subsequent ransom wasn't paid—a practice known as "double extortion." To steal intellectual property, the attackers targeted and collected data from SQL databases. They also navigated through directories and project folders, among others, of each device they could access, then exfiltrated the data they found in those.

The exfiltration occurred for multiple days on multiple devices, which allowed the attackers to gather large volumes of information that they could then use for double extortion.

Encryption and ransom

It was a full two weeks from the initial compromise before the attackers progressed to ransomware deployment, thus highlighting the need for triaging and scoping out alert activity to understand accounts and the scope of access an attacker gained from their activity. Distribution of the ransomware payload using *PsExec.exe* proved to be the most common attack method.

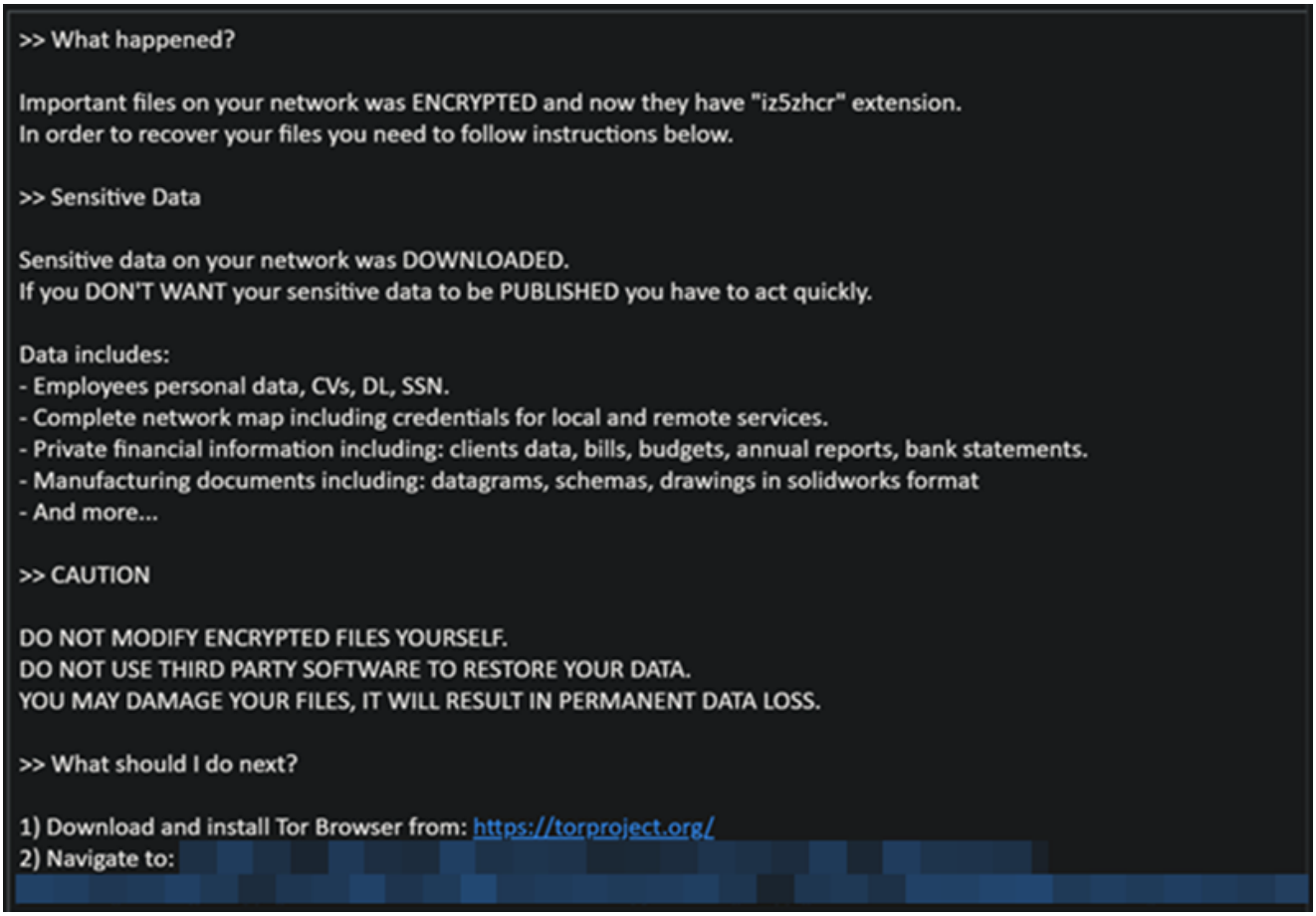


Figure 3. Ransom note displayed by BlackCat upon successful infection

Case study 2: Entry via compromised credentials

In another incident we observed, we found that a ransomware affiliate gained initial access to the environment via an internet-facing Remote Desktop server using compromised credentials to sign in.

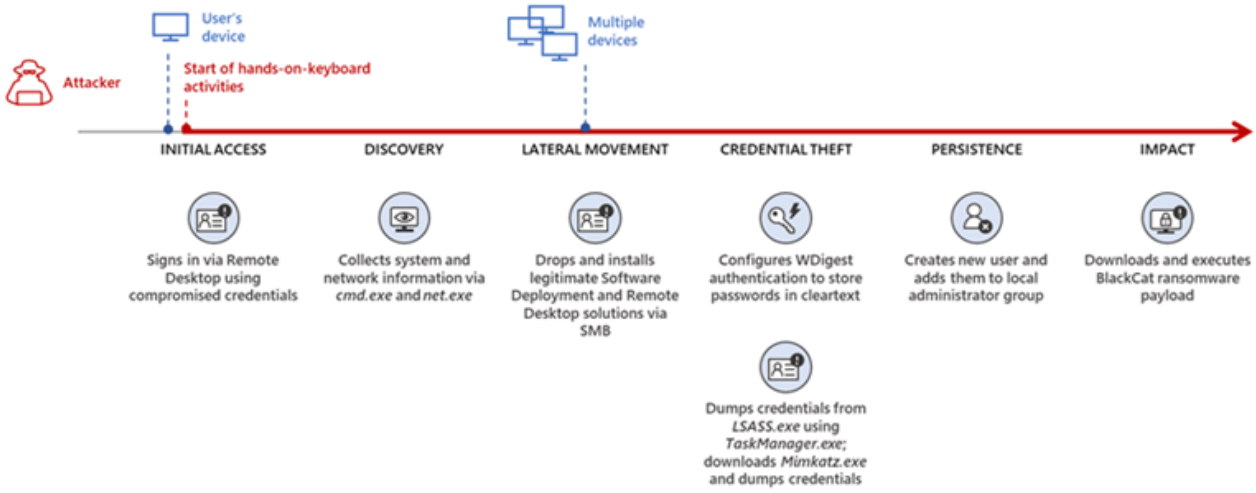


Figure 4. Observed BlackCat ransomware attack chain via stolen credentials

Lateral movement

Once the attackers gained access to the target environment, they then used SMB to copy over and launch the Total Deployment Software administrative tool, allowing remote automated software deployment. Once this tool was installed, the attackers used it to install ScreenConnect (now known as ConnectWise), a remote desktop software application.

Credential theft

ScreenConnect was used to establish a remote session on the device, allowing attackers interactive control. With the device in their control, the attackers used *cmd.exe* to update the Registry to allow cleartext authentication via WDigest, and thus saved the attackers time by not having to crack password hashes. Shortly later, they used the Task Manager to dump the *LSASS.exe* process to steal the password, now in cleartext.

Eight hours later, the attackers reconnected to the device and stole credentials again. This time, however, they dropped and launched Mimikatz for the credential theft routine, likely because it can grab credentials beyond those stored in *LSASS.exe*. The attackers then signed out.

Persistence and encryption

A day later, the attackers returned to the environment using ScreenConnect. They used PowerShell to launch a command prompt process and then added a user account to the device using *net.exe*. The new user was then added to the local administrator group via *net.exe*.

Afterward, the attackers signed in using their newly created user account and began dropping and launching the ransomware payload. This account would also serve as a means of additional persistence beyond ScreenConnect and their other footholds in the environment to allow them to re-establish their presence, if needed. Ransomware adversaries are not above ransoming the same organization twice if access is not fully remediated.

Chrome.exe was used to navigate to a domain hosting the BlackCat payload. Notably, the folder structure included the organization name, indicating that this was a pre-staged payload specifically for the organization. Finally, the attackers launched the BlackCat payload on the device to encrypt its data.

Ransomware affiliates deploying BlackCat

Apart from the incidents discussed earlier, we've also observed two of the most prolific affiliate groups associated with ransomware deployments have switched to deploying BlackCat. Payload switching is typical for some RaaS affiliates to ensure business continuity or if there's a possibility of better profit. Unfortunately for organizations, such adoption further adds to the challenge of detecting related threats.

Microsoft tracks one of these affiliate groups as DEV-0237. Also known as FIN12, DEV-0237 is notable for its distribution of Hive, Conti, and Ryuk ransomware. We've observed that this group added BlackCat to their list of distributed payloads beginning March 2022. Their switch to BlackCat from their last used payload (Hive) is suspected to be due to the public discourse around the latter's decryption methodologies.

DEV-0504 is another active affiliate group that we've seen switching to BlackCat for their ransomware attacks. Like many RaaS affiliate groups, the following TTPs might be observed in a DEV-0504 attack:

- Entry vector that can involve the affiliate remotely signing into devices with compromised credentials, such as into devices running software solutions that allow for remote work
- The attackers' use of their access to conduct discovery on the domain
- Lateral movement that potentially uses the initial compromised account
- Credential theft with tools like Mimikatz and Rubeus

DEV-0504 typically exfiltrates data on devices they compromise from the organization using a malicious tool such as StealBit—often named "send.exe" or "sender.exe". PsExec is then used to distribute the ransomware payload. The group has been observed delivering the following ransom families before their adoption of BlackCat beginning December 2021:

- BlackMatter
- Conti
- LockBit 2.0

- Revil
- Ryuk

Defending against BlackCat ransomware

Today's ransomware attacks have become more impactful because of their growing industrialization through the RaaS affiliate model and the increasing trend of double extortion. The incidents we've observed related to the BlackCat ransomware leverage these two factors, making this threat durable against conventional security and defense approaches that only focus on detecting the ransomware payloads. Detecting threats like BlackCat, while good, is no longer enough as human-operated ransomware continues to grow, evolve, and adapt to the networks they're deployed or the attackers they work for.

Instead, organizations must shift their defensive strategies to prevent the end-to-end attack chain. As noted above, while attackers' entry points may vary, their TTPs remain largely the same. In addition, these types of attacks continue to take advantage of an organization's poor credential hygiene and legacy configurations or misconfigurations to succeed. Therefore, defenders should address these common paths and weaknesses by hardening their networks through various best practices such as access monitoring and proper patch management. We provide detailed steps on building these defensive strategies against ransomware in this blog.

In the BlackCat-related incidents we've observed, the common entry points for ransomware affiliates were via compromised credentials to access internet-facing remote access software and unpatched Exchange servers. Therefore, defenders should review their organization's identity posture, carefully monitor external access, and locate vulnerable Exchange servers in their environment to update as soon as possible. The financial impact, reputation damage, and other repercussions that stem from attacks involving ransomware like BlackCat are not worth forgoing downtime, service interruption, and other pain points related to applying security updates and implementing best practices.

Leveraging Microsoft 365 Defender's comprehensive threat defense capabilities

Microsoft 365 Defender helps protect organizations from attacks that deliver the BlackCat ransomware and other similar threats by providing cross-domain visibility and coordinated threat defense. It uses multiple layers of dynamic protection technologies and correlates threat data from email, endpoints, identities, and cloud apps. Microsoft Defender for Endpoint detects tools like Mimikatz, the actual BlackCat payload, and subsequent attacker behavior. Threat and vulnerability management capabilities also help discover vulnerable or misconfigured devices across different platforms; such capabilities could help detect and block possible exploitation attempts on vulnerable devices, such as those running Exchange. Finally, advanced hunting lets defenders create custom detections to proactively surface this ransomware and other related threats.

Additional mitigations and recommendations

Defenders can also follow the following steps to reduce the impact of this ransomware:

- Turn on Microsoft Defender Antivirus. Turn on cloud-delivered protection in Microsoft Defender Antivirus or the equivalent for your antivirus product to cover rapidly evolving attacker tools and techniques. Cloud-based machine learning protections block a large amount of new and unknown variants.
- Enforce strong, randomized local administrator passwords. Use tools like Local Administrator Password Solution (LAPS).
- Require multifactor authentication (MFA) for local device access, RDP access, and remote connections through virtual private networks (VPNs) and Outlook Web Access. Solutions like Windows Hello or Fast ID Online (FIDO) v2.0 security keys let users sign in using biometrics and/or a physical key or device.
- Turn on Microsoft Defender Firewall.
- Implement controlled folder access to help prevent files from being altered or encrypted by ransomware. Set controlled folder access to Enabled or Audit mode.
- Investigate and remediate vulnerabilities in Exchange servers. Also, determine if implementing the Exchange Emergency Mitigation service is feasible for your environment. This service helps keep your Exchange servers secure by applying mitigations to address potential threats against your servers.

Microsoft 365 Defender customers can also apply the additional mitigations below:

- Use advanced protection against ransomware.
- Turn on tamper protection in Microsoft Defender for Endpoint to prevent malicious changes to security settings. Enable network protection in Microsoft Defender for Endpoint and Microsoft 365 Defender to prevent applications or users from accessing malicious domains and other malicious content on the internet.
- Ensure Exchange servers have applied the mitigations referenced in the related Threat Analytics report.
- Turn on the following attack surface reduction rules to block or audit activity associated with this threat:
    - Block credential stealing from the Windows local security authority subsystem (*lsass.exe*)
    - Block process creations originating from PSExec and WMI commands
    - Block executable files from running unless they meet a prevalence, age, or trusted list criterion

For a full list of ransomware mitigations regardless of threat, refer to this article: Rapidly protect against ransomware and extortion.

Learn how you can stop attacks through automated, cross-domain security and built-in AI with Microsoft Defender 365.

Microsoft 365 Defender Threat Intelligence Team

Appendix

Microsoft 365 Defender detections

Microsoft Defender Antivirus

Microsoft Defender for Endpoint EDR

Alerts with the following titles in the security center can indicate threat activity on your network:

- An active ‘BlackCat’ ransomware was detected
- ‘BlackCat’ ransomware was detected
- BlackCat ransomware

Hunting queries

Microsoft 365 Defender

To locate possible ransomware activity, run the following queries.

Suspicious process execution in PerfLogs path

Use this query to look for processes executing in PerfLogs—a common path used to place the ransomware payloads.

```
DeviceProcessEvents
| where InitiatingProcessFolderPath has "PerfLogs"
| where InitiatingProcessFileName matches regex "[a-z]{3}.exe"
| extend Length = strlen(InitiatingProcessFileName)
| where Length == 7
```

Suspicious registry modification of MaxMpxCt parameters

Use this query to look for suspicious running processes that modify registry settings to increase the number of outstanding requests allowed (for example, SMB requests when distributing ransomware via its PsExec methodology).

```
DeviceProcessEvents
| where ProcessCommandLine has_all("LanmanServer", "parameters", "MaxMpxCt", "65535")
```

Suspicious command line indicative of BlackCat ransom payload execution

Use these queries to look for instances of the BlackCat payload executing based on a required command argument for it to successfully encrypt ‘–access-token’.

```
DeviceProcessEvents
| where ProcessCommandLine has_all("--access-token", "-v")
| extend CommandArguments = split(ProcessCommandLine, " ")
| mv-expand CommandArguments
| where CommandArguments matches regex "^[A-Fa-f0-9]{64}$"

DeviceProcessEvents
| where InitiatingProcessCommandLine has "--access-token"
| where ProcessCommandLine has "get uuid"
```

Suspected data exfiltration

Use this query to look for command lines that indicate data exfiltration and the indication that an attacker may attempt double extortion.

```
DeviceNetworkEvents
| where InitiatingProcessCommandLine has_all("copy", "--max-age", "--ignore-
existing", "--multi-thread-streams", "--transfers") and InitiatingProcessCommandLine
has_any("ftp", "ssh", "-q")
```