

New LockBit 5.0 Targets Windows, Linux, ESXi

By Sarah Pearl Camiling, Jacob Santos Sep 25, 2025 Read time: 7 min (2008 words)

Published: 2025-09-25 · Archived: 2026-04-05 12:36:25 UTC

Ransomware

Trend™ Research analyzed source binaries from the latest activity from notorious LockBit ransomware with their 5.0 version that exhibits advanced obfuscation, anti-analysis techniques, and seamless cross-platform capabilities for Windows, Linux, and ESXi systems.

Key takeaways:

- The LockBit 5.0 Windows variant uses heavy obfuscation and packing by loading its payload through DLL reflection while implementing anti-analysis technique. The Linux variant has similar functionality with command-line options for targeting specific directories and file types. The ESXi variant specifically targets VMware virtualization infrastructure, designed to encrypt virtual machines.
- The new variants use randomized 16-character file extensions, has Russian language system avoidance, and event log clearing post-encryption.
- LockBit 5.0 also has a dedicated ESXi that targets VMware's ESXi virtualization infrastructure.
- The existence of Windows, Linux, and ESXi variants confirms LockBit's continued cross-platform strategy, enabling simultaneous attacks across entire enterprise networks including virtualized environments. Heavy obfuscation and technical improvements across all variants make LockBit 5.0 significantly more dangerous than its predecessors.
- Trend Vision One™ detects and blocks the specific IoCs mentioned in this blog, and offers customers access to hunting queries, threat insights, and intelligence reports related to LockBit 5.0.

Trend™ Research has identified and analyzed the source binaries of a new [LockBitnews article](#) version in the wild, which is the latest from the group's [activities](#) following the February 2024 law enforcement operation (Operation Cronos) that disrupted their infrastructure. In early September, the LockBit ransomware group [reportedly](#) resurfaced for their sixth anniversary, announcing the release of "LockBit 5.0". Trend Research discovered a binary available in the wild and began analysis that initially discovered a Windows variant and confirmed the existence of Linux and ESXi variants of LockBit 5.0.

This latest news continues the group's established cross-platform strategy seen [since LockBit 2.0 in 2021](#).

Trend Research analysis found that the Windows binary uses heavy obfuscation and packing: it loads its payload through DLL reflection while implementing anti-analysis techniques like ETW patching and terminating security services. Meanwhile, the newly discovered Linux variant maintains similar functionality with command-line options for targeting specific directories and file types. The ESXi variant specifically targets VMware virtualization environments, designed to encrypt entire virtual machine infrastructures in a single attack.

Our investigation also reveals that these newer versions share key behaviors: randomized 16-character file extensions, Russian language system avoidance through geolocation checks, and event log clearing post-encryption. The 5.0 version also shares code characteristics with LockBit 4.0, including identical hashing algorithms and API resolution methods, confirming this is an evolution of the original codebase rather than an imitation.

LockBit 5.0 Windows analysis

The Windows version of Lockbit 5.0 uses the `-h` parameter to display help information; the new version features a better user interface with clean formatting, which has not been seen in previous versions. It describes various options and settings for executing the ransomware, including basic options like specifying directories to encrypt or bypass, operation modes such as invisible mode and verbose mode, notes settings, encryption settings, filtering options, and examples of usage. The detailed commands and parameters illustrate the flexibility and customization available to the attacker.

```

Administrator: C:\Windows\system32\cmd.exe
LOCKBITS.0 ChuongDong Locker v1.01 Windows x64

USAGE
chuongdong64.exe [options]
* Command line length is limited to 500 characters.

BASIC OPTIONS
-h Show this help
-p <dirs> Semicolon-separated list of directories to encrypt
-b <dirs> Semicolon-separated list of directories to bypass

OPERATION MODES
-i Invisible mode (don't change extensions, no notes, don't change modification date)
-v Run in verbose visible mode with status bar in console
  * Not available when using -p
-d Run in visible mode with debug output

NOTES SETTINGS
-n <0/1/2> Notes storage mode (0: none, 1: everywhere, 2: C:\ only)
  * This option is ignored when using -i (invisible mode)

ENCRYPTION SETTINGS
-m <mode> Encryption mode (all/local/net)
-f Fast encryption mode
-w Enable wipe free space after encryption

FILTERING
-k Don't delete .exe
-nomutex Allow multiple instances
-t <seconds> Set timeout before starting encryption

EXAMPLES
chuongdong64.exe
  Encrypt entire system with default settings

chuongdong64.exe -p "C:\Users;X:\remote"
  Encrypt C:\Users and X:\remote directories

chuongdong64.exe -m local -k
  Encrypt local files only, don't delete executable

chuongdong64.exe -t 300
  Wait 5 minutes before starting encryption
    
```

Figure 1. Help command shows the parameters and their respective uses

Table 1 shows the command line arguments observed in Trend Research threat hunting analysis and their respective descriptions.

Option	Description
--------	-------------

Basic Options	
-h	Show help
-d <dirs>	Semicolon-separated list of directories to encrypt
-b <dirs>	Semicolon-separated list of directories to bypass
Operation Modes	
-i	Invisible mode (don't change extensions, no notes, don't change modification date)
-p	Run in verbose visible mode with status bar in console (not available when using -i)
-v	Run in visible mode with debug output
Notes Settings	
-n <0/1/2>	Notes storage mode:
	0: None
	1: Everywhere
	2: C:\ only
	Ignored when using -i (invisible mode)
Encryption Settings	
-m <mode> -w	Encryption mode:
	all: Encrypt all files
	local: Encrypt local files
	net: Encrypt network files
	Enable wipe free space after encryption
Filtering	
-k	Don't delete .exe files
-nomutex	Allow multiple instances
Timeout	
-t <seconds>	Set timeout before starting encryption

Table 1. A summary of the command line arguments in our analysis of the LockBit 5.0 Windows version.

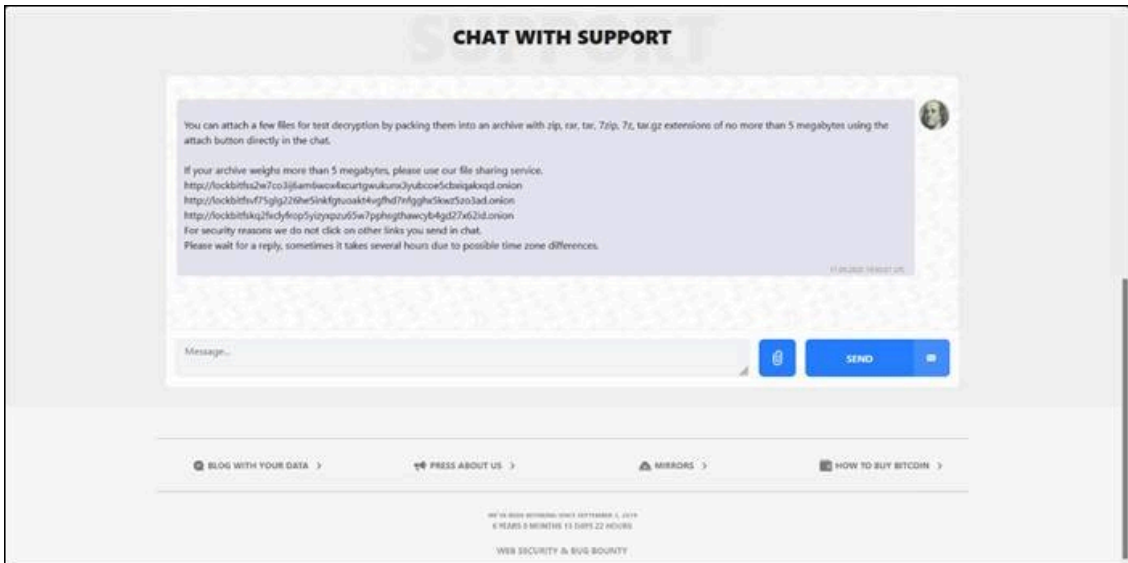


Figure 4. The data leak site provides a direct communication channel with the victims in the "Chat with Support" section.

The encryption process appends randomized 16-character extensions to files, complicating recovery efforts. Unlike some ransomware variants that use common infection markers, LockBit 5.0 omits traditional markers at file endings. However, our analysis revealed consistent patterns including the original file size embedded in the encrypted file footer.

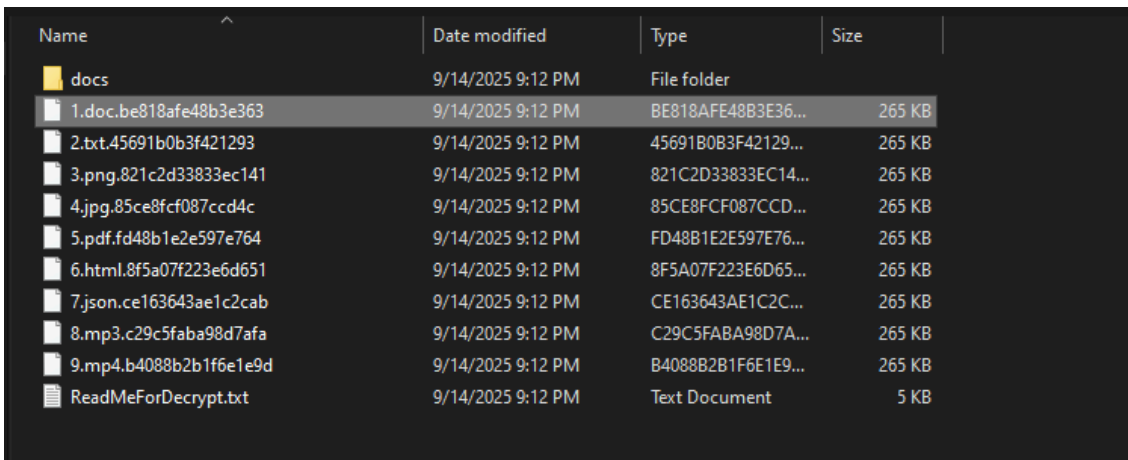


Figure 5. LockBit 5.0 encrypted files are appended with unique and seemingly randomly generated extensions with 16 characters which complicates the decryption process.



Figure 6. End part of encrypted file A



Figure 7. End part of encrypted file B

The sample Trend Research analyzed employs heavy obfuscation through packing. During debugging, we discovered it functions as a binary loader, decrypting a PE binary in memory and loading it via DLL reflection methods. This sophisticated loading mechanism significantly complicates static analysis.

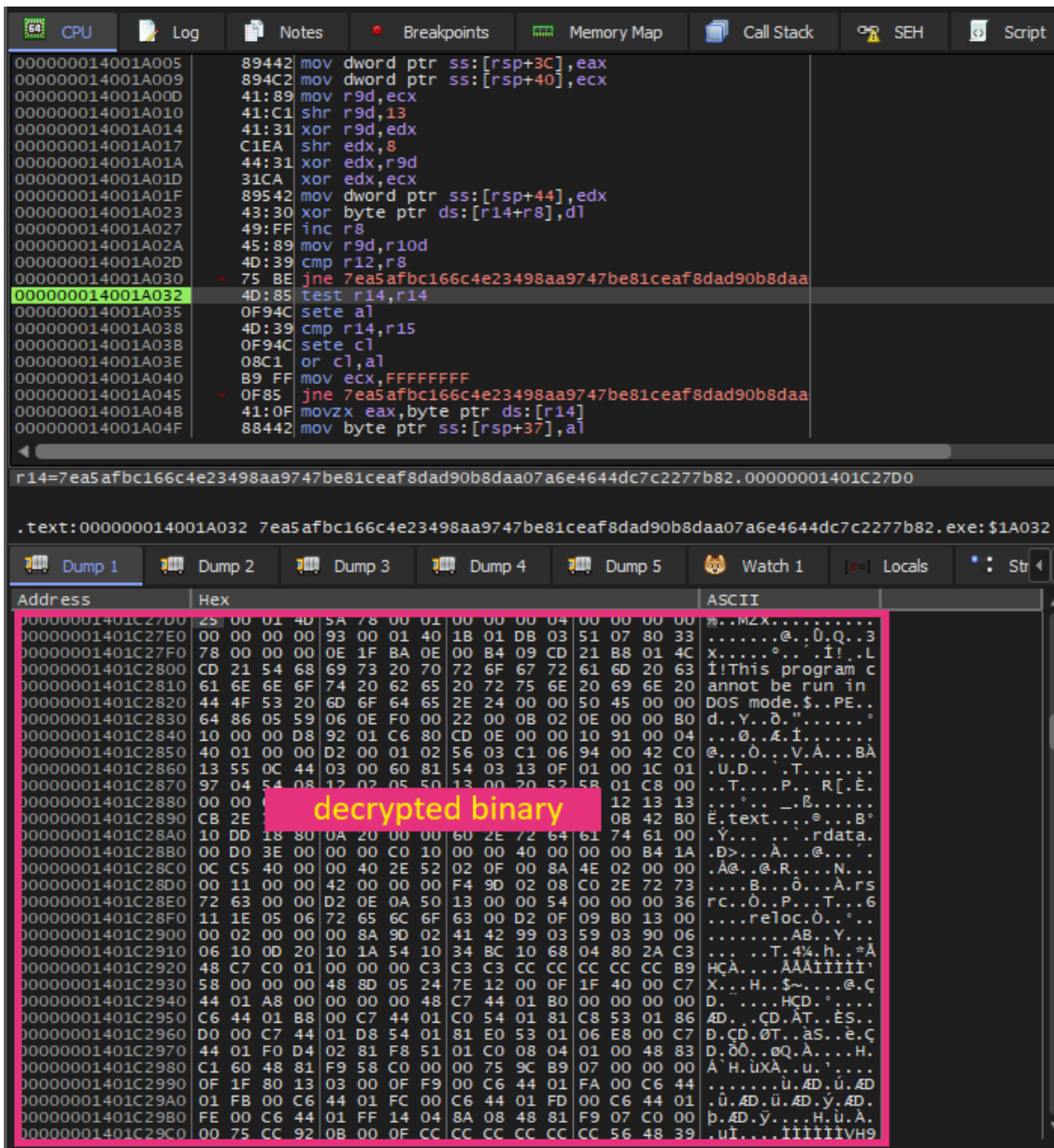


Figure 8. Decrypted PE binary in the memory of loader

Aside from that, the malware implements multiple anti-forensics techniques. It patches the `EtwEventWrite` API by overwriting it with a `0xC3` (return) instruction, disabling Windows Event Tracing capabilities. Additionally, it terminates security-related services by comparing hashed service names against a hardcoded list of 63 values, then clears all event logs using the `EvtClearLog` API after encryption completion.

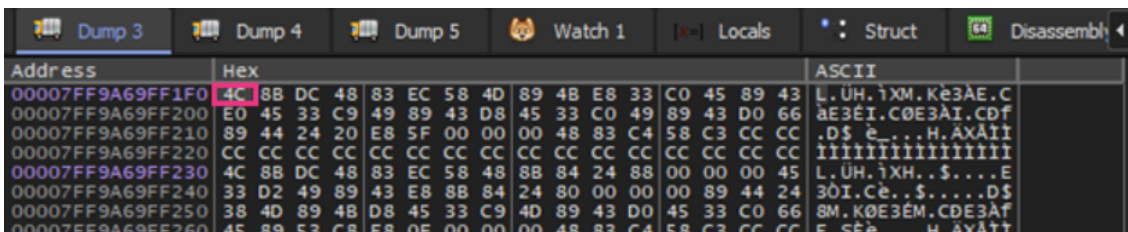


Figure 9. Before patching of `EtwEventWrite`

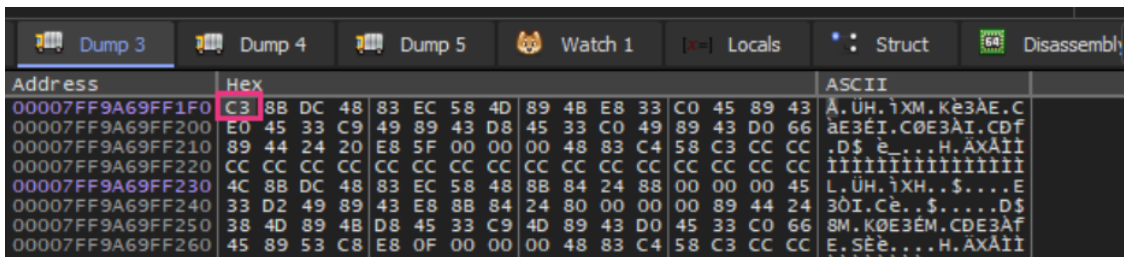


Figure 10. After patching EtwEventWrite, the malware shows a C3 byte forcing it to immediately return.

It compares all the services if they run the system by hashing the service name and comparing it with the hardcoded list. Service names that match are then terminated.

FEF56F15, BEC3470B, 9757464D, 88CE6B8E, 826AC445, 83143F70, 8685D050, 493AEE1F, 35BE2F4E, 23FA53E4, FEF56F16, 10D06066, 1370CEA3, E11A285C, DBECA3C2, BEC3470C, C347B317, CA6C4394, 732AA0BF, 60B29D13, 493AEE20, 5E5F1954, 5EF504FC, A49FA5E2, 9757464E, 9A768D62, A1816235, 41278146, 35BE2F4F, 369D7114, 3B6794E3, E7AA4056, E11A285D, E5C9CC93, E66A2C63, 7B39B584, 732AA0C0, 739BF272, 7ABD1404, 88CE6B8F, 9439954E, 9655130F, 23FA53E5, 26336765, 2C1F8E5F, DBECA3C3, DCF04E8C, DEED0E56, 60B29D14, 62C32884, 6337AD82, A49FA5E3, A8F16BAB, BD071334, 41278147, 4292EDD8, 47F1286A, E7AA4057, E7BF305D, F82A288D, 7B39B585, 7F480CF7, 7DD43601

```
if ( ((__int64 (__fastcall *) (__int64, __int64, _QWORD, _QWORD))v0.m128i_i64[0])(v290, *v179, 0LL, 0LL) )// EvtClearLog
{
    ++dword_1401257D0;
    v149 = v268;
}
```

Figure 11. Event log clearing using EvtClearLog API

Consistent with previous versions, LockBit 5.0 includes geopolitical safeguards, terminating execution when detecting Russian language settings or Russian geolocation. This is a common practice among Eastern European ransomware groups.

```
v1245.m128i_i64[0] = v11.m128i_i64[0];
if ( ((unsigned __int16 (*)(void))v11.m128i_i64[0])() == 1049 )// GetUserDefaultUILanguage
{
}
```

Figure 12. This code terminates if the language is Russian

```
if ( ((__int64 (__fastcall *) (__int64))v1245.m128i_i64[0])(16LL) == 0xC9 )// GetUserGeoID
{
    // Retrieves information about the geographical location of the user.
    v39 = 1;
}
```

Figure 13. This code terminates if the geolocation is Russia

LockBit 5.0 Linux analysis

The 5.0 Linux variant has similar features with its Windows counterpart, demonstrating LockBit's commitment to cross-platform capabilities. The command-line interface mirrors the Windows version's formatting and functionality, providing attackers with the same operational flexibility across both platforms.

```
ubuntu@ubuntu:~/Desktop$ ./lockbit -h
LOCKBIT5.0  LINUX Locker v1.01  Linux amd64

USAGE
./lockbit [options]

BASIC OPTIONS
-h          Show this help
-d <list>   Specific directories to encrypt (semicolon-separated, default: /)
-l          Enable logging to file

OPERATION MODES
-b          Run in background mode (no console output)
-m <mode>   Note storage mode (0: none, 1: all dirs, 2: root only)
-k          Do not self-destruct
-q          Quiet mode (don't change extensions, no notes)

ENCRYPTION SETTINGS
-r <10-90>  Encryption percentage per file
-w          Wipe free disk space

FILTERING
-s <list>   Skip directories (semicolon-separated)
-t <seconds> Set timeout before starting. Implies background mode (-b)

EXAMPLES
./lockbit
  Encrypt entire system with default settings

./lockbit -b -d "/home"
  Encrypt /home directory in background mode

./lockbit -l -d "/path1;/path2"
  Encrypt multiple directories with logging enabled

./lockbit -r 50 -s "/tmp;/var/log"
  Encrypt 50% of each file, skip /tmp and /var/log

./lockbit -t 300
  Wait 5 minutes before starting encryption in background
```

Figure 14. The LockBit 5.0 Linux version shows similar formatting of help options

During execution, the Linux variant provides detailed logging of its activities, displaying files targeted for encryption and folders designated for exclusion. This transparency in operation logs suggests the variant can be used in testing environments or by affiliates requiring detailed execution feedback.

```
ubuntu@ubuntu:~/Desktop$ ./lockbit -l -d "./test_files/"
[INFO] Logging enabled
[INFO] Free space wiping enabled

[OK] Starting encryption in path(s): ./test_files/

[+] ::: ./test_files/6.html | 264.6 kB
[+] ::: ./test_files/9.mp4 | 264.6 kB
[+] ::: ./test_files/7.json | 264.6 kB
[+] ::: ./test_files/3.png | 264.6 kB
[+] ::: ./test_files/2.txt | 264.6 kB
[+] ::: ./test_files/4.jpg | 264.6 kB
[+] ::: ./test_files/5.pdf | 264.6 kB
[+] ::: ./test_files/8.mp3 | 264.6 kB
[+] ::: ./test_files/docs/10.mpeg | 264.6 kB
[+] ::: ./test_files/docs/6.html | 264.6 kB
[+] ::: ./test_files/docs/9.mp4 | 264.6 kB
[+] ::: ./test_files/docs/11.webp | 264.6 kB
[+] ::: ./test_files/docs/7.json | 264.6 kB
[+] ::: ./test_files/docs/3.png | 264.6 kB
[+] ::: ./test_files/docs/2.txt | 264.6 kB
[+] ::: ./test_files/docs/4.jpg | 264.6 kB
[+] ::: ./test_files/docs/8.mp3 | 264.6 kB
[+] ::: ./test_files/docs/5.pdf | 264.6 kB
[+] ::: ./test_files/docs/docs/6.html | 264.6 kB
[+] ::: ./test_files/docs/docs/10.mpeg | 264.6 kB
[+] ::: ./test_files/docs/docs/9.mp4 | 264.6 kB
[+] ::: ./test_files/docs/docs/11.webp | 264.6 kB
[+] ::: ./test_files/docs/docs/3.png | 264.6 kB
[+] ::: ./test_files/docs/docs/7.json | 264.6 kB
[+] ::: ./test_files/docs/docs/2.txt | 264.6 kB
[+] ::: ./test_files/docs/docs/4.jpg | 264.6 kB
[+] ::: ./test_files/docs/docs/8.mp3 | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/6.html | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/5.pdf | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/9.mp4 | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/7.json | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/3.png | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/2.txt | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/8.mp3 | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/5.pdf | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/4.jpg | 264.6 kB
[+] ::: ./test_files/docs/docs/docs/1.doc | 264.6 kB
[+] ::: ./test_files/docs/docs/1.doc | 264.6 kB
[+] ::: ./test_files/1.doc | 264.6 kB
[+] ::: ./test_files/docs/1.doc | 264.6 kB
```

Figure 15. Logging activity shows the files to be encrypted

```
[INFO] Check if all files were encrypted ... Wait...
[i] Starting free space wiping process...
[i] Skipping wipe for /sys
[i] Skipping wipe for /proc
[i] Skipping wipe for /dev
[i] Skipping wipe for /dev/pts
[i] Skipping wipe for /run
[i] Wiping /...
[i] Skipping wipe for /sys/kernel/security
[-] create_secure_temp_file failed for /
[i] Skipping wipe for /dev/shm
[i] Skipping wipe for /run/lock
[i] Skipping wipe for /sys/fs/cgroup
[i] Skipping wipe for /sys/fs/pstore
[i] Skipping wipe for /sys/fs/bpf
[i] Skipping wipe for /proc/sys/fs/binfmt_misc
[i] Skipping wipe for /dev/hugepages
[i] Skipping wipe for /dev/mqueue
[i] Skipping wipe for /sys/kernel/debug
[i] Skipping wipe for /sys/kernel/tracing
[i] Skipping wipe for /sys/kernel/config
[i] Skipping wipe for /sys/fs/fuse/connections
[i] Skipping wipe for /run/vmblock-fuse
[i] Wiping /snap/bare/5...
[i] Wiping /snap/core22/2045...
[i] Wiping /snap/firefox/6565...
[-] create_secure_temp_file failed for /snap/bare/5
[-] create_secure_temp_file failed for /snap/core22/2045
[-] create_secure_temp_file failed for /snap/firefox/6565
[i] Wiping /snap/firmware-updater/167...
[i] Wiping /snap/gnome-42-2204/202...
[i] Wiping /snap/snap-store/1270...
[-] create_secure_temp_file failed for /snap/firmware-updater/167
[-] create_secure_temp_file failed for /snap/gnome-42-2204/202
[i] Wiping /snap/gtk-common-themes/1535...
[-] create_secure_temp_file failed for /snap/snap-store/1270
[i] Wiping /snap/snapd/24792...
[-] create_secure_temp_file failed for /snap/gtk-common-themes/1535
```

Figure 16. Logs show the list of folders to be skipped on wiping

Upon completion, the ransomware generates a comprehensive summary showing the total number of files encrypted and their cumulative size. Like the Windows version, it applies randomized extensions to encrypted files, maintaining consistency in post-encryption file handling across platforms.

```
[ - ] create_secure_temp_file failed for /snap/snapd-desktop-integration/315
[ i ] Skipping wipe for /run/snapd/ns/snapd-desktop-integration.mnt
[ i ] Skipping wipe for /run/user/1000
[ i ] Skipping wipe for /run/user/1000/doc
[ i ] Skipping wipe for /run/user/1000/gvfs
[ i ] Wiping /media/ubuntu/CDROM...
[ i ] Wiping /media/ubuntu/Ubuntu\04024.04.3\040LTS\040amd64...
[ - ] create_secure_temp_file failed for /media/ubuntu/CDROM
[ - ] create_secure_temp_file failed for /media/ubuntu/Ubuntu\04024.04.3\040LTS\040amd64
[ + ] All wiping threads have completed.
```

LOCKBIT 5.0 **LINUX Locker v1.01** **Linux amd64**

```
Files processed : 40
Files skipped  : 0
Total files     : 48
Files size     : 10.3 MB
Encrypted data : 10.3 MB
Execution time : 1 s
```

Figure 17. A summary shows the total of number of files and size encrypted

Name ^	Size	Modified
docs	13 items	Today 11:16
1.doc.e63d39c82977a027	271.2 kB	Today 11:16
2.txt.7affa9bd92a6f676	271.2 kB	Today 11:16
3.png.ff3583571d7262e3	271.2 kB	Today 11:16
4.jpg.bd3d1839a209d958	271.2 kB	Today 11:16
5.pdf.bcd2841ff64cf7d	271.2 kB	Today 11:16
6.html.1440eae15c3444c4	271.2 kB	Today 11:16
7.json.abcc78e1fd2fdf5f	271.2 kB	Today 11:16
8.mp3.36721463305e6eee	271.2 kB	Today 11:16
9.mp4.49faa9445d9cac2f	271.2 kB	Today 11:16
ReadMeForDecrypt.txt	4.4 kB	Today 11:16

Figure 18. A list of files encrypted that have random extensions

LockBit 5.0 ESXi analysis

Further investigation revealed a dedicated ESXi variant of LockBit 5.0, specifically targeting VMware virtualization infrastructure. This variant represents a critical escalation in LockBit's capabilities, as ESXi servers typically host multiple virtual machines, allowing attackers to encrypt entire virtualized environments with a single payload execution.

The ESXi variant maintains the same command-line interface structure as its Windows and Linux counterparts, ensuring operational consistency for attackers across all platforms. The help menu reveals ESXi-specific parameters optimized for virtual machine encryption, including options to target specific directories and VM configuration files.

```
LOCKBIT5.0  LINUX Locker v1.01  ESXi x64

USAGE
./90b06f07eb75045ea3d4ba6577afc9b58078eafeb2cdd417e2a88d7ccf0c0273 [options]

BASIC OPTIONS
-h          Show this help
-d <list>   Specific directories to encrypt (semicolon-separated, default: /)
-l          Enable logging to file

OPERATION MODES
-b          Run in background mode (no console output)
-m <mode>   Note storage mode (0: none, 1: all dirs, 2: root only)
-k          Do not self-destruct

ENCRYPTION SETTINGS
-r <10-90>  Encryption percentage per file
-w          Wipe free disk space

FILTERING
-t <seconds> Set timeout before starting. Implies background mode (-b)

VIRTUALIZATION
-o          Don't try to automatically kill VMs
-n <ID;ID>  Skip VMs by ID (semicolon-separated)
           To list VM IDs: /bin/vim-cmd vmsvc/getallvms

EXAMPLES

./90b06f07eb75045ea3d4ba6577afc9b58078eafeb2cdd417e2a88d7ccf0c0273
  Encrypt entire system with default settings

./90b06f07eb75045ea3d4ba6577afc9b58078eafeb2cdd417e2a88d7ccf0c0273 -b -d "/home"
  Encrypt /home directory in background mode

./90b06f07eb75045ea3d4ba6577afc9b58078eafeb2cdd417e2a88d7ccf0c0273 -l -d "/path1;/path2"
  Encrypt multiple directories with logging enabled

./90b06f07eb75045ea3d4ba6577afc9b58078eafeb2cdd417e2a88d7ccf0c0273 -r 50 -s "/tmp;/var/log"
  Encrypt 50% of each file, skip /tmp and /var/log

./90b06f07eb75045ea3d4ba6577afc9b58078eafeb2cdd417e2a88d7ccf0c0273 -t 300
  Wait 5 minutes before starting encryption in background

./90b06f07eb75045ea3d4ba6577afc9b58078eafeb2cdd417e2a88d7ccf0c0273 -n "100;101"
  Skip VMs with ID 100 and 101

/bin/vim-cmd vmsvc/getallvms
  List all VMs to get their IDs
```

Figure 19. ESXi variant help command showing virtualization-specific parameters

This ESXi variant demonstrates LockBit's strategic focus on maximizing impact through virtualization infrastructure, where a single compromised ESXi host can result in dozens or hundreds of encrypted virtual machines, significantly amplifying the attack's business disruption potential.

LockBit 4.0 versus LockBit 5.0

A comparative analysis between LockBit 4.0 and 5.0 reveals significant code reuse and evolutionary development rather than a complete rewrite. Both versions share identical hashing algorithms for string operations, a critical component for API resolution, and service identification. The code structure for dynamic API resolution remains remarkably similar between versions, suggesting the developers built upon the existing LockBit 4.0 codebase. The screenshots on the left in figures 20 and 21 are from [chuong dong blog](#).

```

index = 0;
result_hash = 0x148D;
while ( 1 )
{
    curr_char = data[index];
    if ( !data[index] )
        break;
    lower_curr_char = curr_char + 0x20;
    if ( (curr_char - 0x41) >= 0x1Au )
        lower_curr_char = data[index]; // to lower
    v14 = (lower_curr_char ^ result_hash) + lower_curr_char * (index + 0x148D);
    v24 = index ^ 0x148D;
    if ( !index )
        v24 = 0;
    result_hash = lower_curr_char + v14 * v24;
    ++index;
}

v99 = *v98;
v100 = 0xB97A;
if ( *v98 )
{
    LODWORD(v101) = 0;
    do
    {
        v102 = v99 + 32;
        if ( (unsigned __int8)(v99 - 0x41) >= 0x1Au )
            v102 = v99;
        v103 = (v102 ^ v100) + v102 * (v101 + 0xB97A);
        v104 = v101 ^ 0xB97A;
        if ( !(_DWORD)v101 )
            v104 = 0;
        v100 = v102 + v103 * v104;
        v101 = (unsigned int)(v101 + 1);
        v99 = v98[v101];
    }
    while ( v99 );
}
    
```

Figure 20. Similarities of hashing algorithm of string of LockBit 4.0 (left – screenshot from [chuong dong blog](#)) and LockBit 5.0 (right)

```

if ( !HASH_SETUP_FLAG ) // if hash is not set up
{
    INITIAL_HASH = _mm_load_si128(&dword_41CB00);
    HASH_SETUP_FLAG = 1; // set_up_flag = TRUE
}
API_hash = initial_hash_processing();
v100 = query_API_from_storage(API_hash);
if ( !v100 )
{
    kernel32_dll_base = retrieve_DLL(0x28649624); // kernel32.dll
    v100 = retrieve_API(kernel32_dll_base, API_hash);
    store_API(API_hash, v100);
}

Hash = _LoadHash(GetUserDefaultUILanguage); // GetUserDefaultUILanguage
if ( !Hash )
{
    strcpy(v1244.m128i_i8, "kernel32.dll");
    DLLHash = _GetDLLHash(v1244.m128i_i8);
    if ( DLLHash && (APIHash = _GetAPIHash(DLLHash, 333476660LL)) != 0 )
    {
        v944 = APIHash;
        Hash = ((__int64 (__fastcall *) (__int64))WrapAPI)(APIHash);
        if ( !Hash )
            Hash = v944;
        _StoreAPIHashAddress(333476660, Hash);
    }
}
    
```

Figure 21. Dynamic API resolution of LockBit 4.0(left – screenshot from [blog](#)) and LockBit 5.0 (right)

Trend Research believes that these similarities are a clear indication that LockBit 5.0 represents a continuation of the LockBit ransomware family and is not an imitation or rebrand by different threat actors. The preservation of core functionalities while adding new evasion techniques demonstrates the group's strategy of incremental improvement to their ransomware platform.

Conclusion

The existence of Windows, Linux, and ESXi variants confirms LockBit's continued cross-platform strategy. This enables simultaneous attacks across entire enterprise networks, from workstations to critical servers hosting databases and virtualization platforms, with the ESXi variant designed to cripple entire virtual infrastructures. Heavy obfuscation across these new variants significantly delays detection signature development, while technical improvements including removed infection markers, faster encryption, and enhanced evasion make LockBit 5.0 significantly more dangerous than its predecessors.

LockBit is among the most notorious ransomware-as-a-service (RaaS) groups that consistently stayed ahead of its competitors with an aggressive evolution of its techniques and tactics. Despite Operation Cronos, the criminals behind the group exhibit resilience with all three variants of version 5.0 now confirmed. Organizations must ensure comprehensive cross-platform defenses are in place, with particular attention to protecting virtualization

infrastructure. LockBit 5.0's Windows, Linux, and ESXi variants reinforce that no operating system or platform can be considered safe from modern ransomware campaigns.

Mitigating risk from LockBit 5.0

Organizations are highly encouraged to evaluate and enhance their security posture by proactively conducting threat hunting activities tailored to group-specific tools, tactics, and procedures. It is essential to reinforce both endpoint and network protections, as well as early detection of defense evasion techniques aimed at compromising security solutions.

Proactive security with Trend Vision One™

[Trend Vision One™one-platform](#) is the only AI-powered enterprise cybersecurity platform that centralizes cyber risk exposure management, security operations, and robust layered protection. This holistic approach helps enterprises predict and prevent threats, accelerating proactive security outcomes across their respective digital estate. With Trend Vision One, you're enabled to eliminate security blind spots, focus on what matters most, and elevate security into a strategic partner for innovation.

Trend Vision One™ Threat Intelligence

To stay ahead of evolving threats, Trend customers can access [Trend Vision One™ Threat Insightsproducts](#), which provides the latest insights from Trend Research on emerging threats and threat actors.

Trend Vision One Threat Insights

- Emerging Threats: [LockBit Strikes Again: Updates in Version 5.0](#)

Trend Vision One Intelligence Reports (IOC Sweeping)

[LockBit Strikes Again: Updates in Version 5.0](#)

Hunting Queries

Trend Vision One Search App

Trend Vision One customers can use the Search App to match or hunt the malicious indicators mentioned in this blog post with data in their environment.

LockBit File Renaming with 16-Character Extension

```
eventSubId: 106 AND objectFilePath: /\. [a-f0-9]{16}$/ AND NOT srcFilePath: /\. [a-f0-9]{16}$/
```

LockBit 5 Ransom Note — ReadMeForDecrypt.txt

```
eventSubId: 101 AND objectFilePath: ReadMeForDecrypt.txt
```

More hunting queries are available for Trend Vision One customers with [Threat Insights Entitlement enabledone-platform](#).

Indicators of Compromise

Indicators of compromise can be found [here](#).

Tags

Source: https://www.trendmicro.com/en_us/research/25/i/lockbit-5-targets-windows-linux-esxi.html