Quick Overview of Leaked LockBit 3.0 (Black) builder program

By S2W

Published: 2022-09-23 · Archived: 2026-04-10 02:25:27 UTC

6 min read

Sep 22, 2022

Author: HuiSeong, Yang & Hyunsik, Jeong | S2W TALON

> ***Last Modified*** *: Sep 22, 2022*

Press enter or click to view image in full size

Photo by Niranjan Photographs on Unsplash

Executive Summary

- According to a tweet from 3xp0rt, Ali Qushji was able to infiltrate LockBit's server and acquire the builder for the ransomware
- According to vx-underground, Proton, one of the programmers for the LockBit ransomware group, mentioned that the builder was leaked, **but the tweet has now been deleted.**

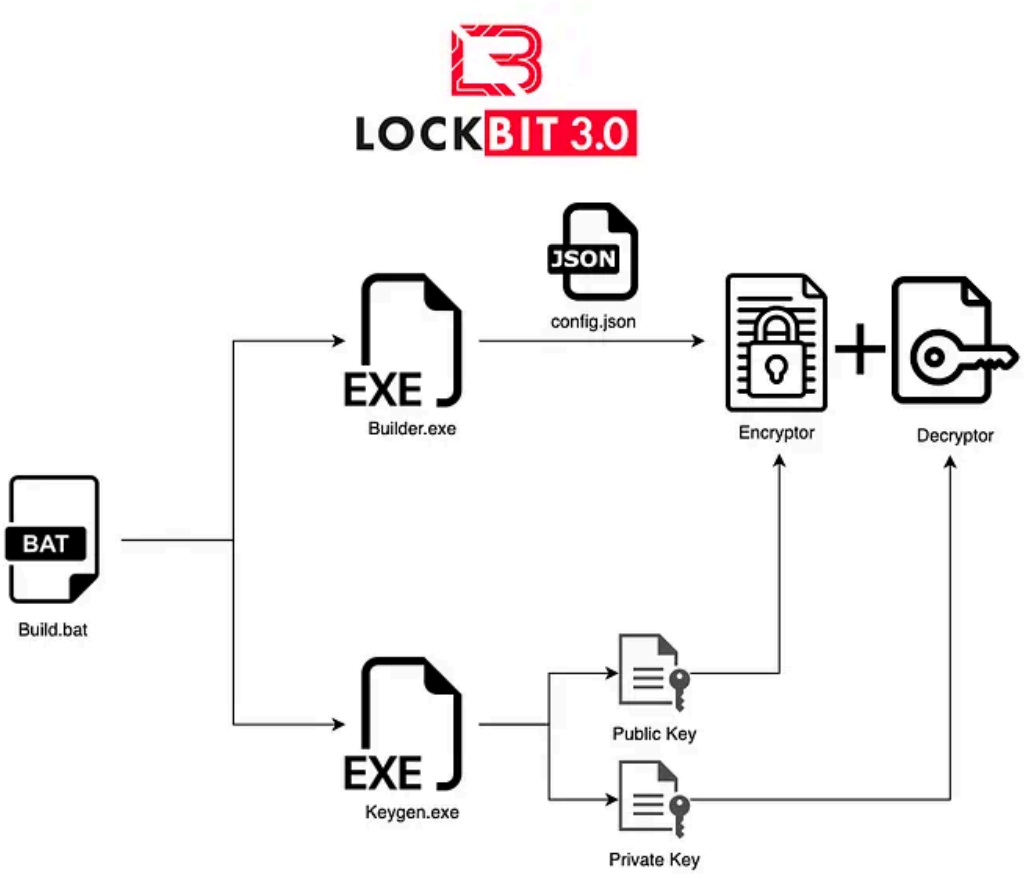
Press enter or click to view image in full size

```
Lockbit ransomware group was not hacked. September 10th,
an individual named "Proton" contacted us and gave us the
Lockbit 3.0 Builder. "Proton" said it was a leak.

We reviewed the data and confirmed its legitimacy. The
builder is a new release, it has the previous
vulnerability repaired, and new features not typically
described in Lockbit 3.0. It encrypts and decrypts
flawlessly

We reached out to Lockbit ransomware group regarding this
and discovered this leaker was a programmer employed by
Lockbit ransomware group. They were upset with Lockbit
leadership and leaked the builder.
```

- The ransomware group indirectly admitted that the allegations above are true, saying that nothing has been hacked and that they have fired the coder.
- LockBit 3.0 Builder leaked by Ali Kushii and Proton are both shared on 3xp0rt's Github.

Detailed Analysis

Press enter or click to view image in full size



LockBit builder flowchart

1. Build.bat

`Build.bat` creates an RSA public/private key pair by executing `Keygen.exe`, and `Builder.exe` that generates a LockBit 3.0 ransomware using the generated key pair.

```
ERASE /F /Q %cd%\Build\*.*
keygen -path %cd%\Build -pubkey pub.key -privkey priv.key
builder -type dec -privkey %cd%\Build\priv.key -config config.json -ofile %cd%\Build\LB3Decryptor.ex
builder -type enc -exe -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build\LB3.exe
builder -type enc -exe -pass -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build\LB3_pa
builder -type enc -dll -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build\LB3_Rundll32
builder -type enc -dll -pass -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build\LB3_Ru
builder -type enc -ref -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build\LB3_Reflecti
exit
```

Press enter or click to view image in full size

Command	Description
ERASE /F /Q %cd%₩Build₩.	Delete all files in the Build folder
keygen -path %cd%₩Build -pubkey pub.key -privkey priv.key	Generate public key and private key in the given path using Keygen.exe
builder -type dec -privkey %cd%₩Build₩priv.key -config config.json -ofile %cd%₩Build₩LB3Decryptor.exe	Generate Decryptor with Builder.exe using private key and Config.json
builder -type enc -exe -pubkey %cd%₩Build₩pub.key -config config.json -ofile %cd%₩Build₩LB3.exe	Generate Encryptor in EXE file format with Builder.exe using public key and Config.json
builder -type enc -exe -pass -pubkey %cd%₩Build₩pub.key -config config.json -ofile %cd%₩Build₩LB3_pass.exe	Generate Encryptor by specifying password in EXE file format with Builder.exe using public key and Config.json
builder -type enc -dll -pubkey %cd%₩Build₩pub.key -config config.json -ofile %cd%₩Build₩LB3_Rundll32.dll	Generate Encryptor in DLL file format with Builder.exe using public key and Config.json
builder -type enc -dll -pass -pubkey %cd%₩Build₩pub.key -config config.json -ofile %cd%₩Build₩LB3_Rundll32_pass.dll	Generate Encryptor by specifying password in DLL file format with Builder.exe using public key and Config.json
builder -type enc -ref -pubkey %cd%₩Build₩pub.key -config config.json -ofile %cd%₩Build₩LB3_ReflectiveDll_DllMain.dll	Generate Encryptor in refelctive-DLL file format with Builder.exe using public key and Config.json

Command line description

The list of files created after execution is as follows.

Files created after executing Build.bat

2. config.json

`config.json` is a JSON configuration file that contains the setting values to be used when generating LockBit 3.0 Encryptor and Decryptor.

- **bot**: Configuration about the bot feature stealing information from infected devices (Not used)
- **config**: Configuration values that determine the behaviors for the LockBit 3.0 ransomware

Press enter or click to view image in full size

Num	Flag	Role
1	encrypt_mode	Encryption mode for large size files
2	encrypt_filename	Whether the file name is encrypted
3	impersonation	Whether to log in using stored credentials
4	skip_hidden_folders	Whether to exclude hidden folder from encryption
5	language_check	Whether to check CIS countries
6	local_disks	Whether to encrypt the local disk after mounting
7	network_shares	Whether to encrypt network shared folders
8	kill_processes	Whether to terminate a specific process list
9	kill_services	Whether to terminate a specific service list
10	running_one	Whether to create a mutex
11	print_note	Whether to print ransom notes to a printer
12	set_wallpaper	Whether to change the wallpaper
13	set_icons	Whether to change the encrypted files' icon
14	send_report	Whether infection system information is stolen
15	self_destruct	Whether to perform self-deletion
16	kill_defender	Whether to kill specific AV software
17	wipe_freespace	Whether to wipe by itself when self-deleting
18	psexec_netspread	Whether network propagation using psexec
19	gpo_netspread	Whether network propagation using gpo
20	gpo_ps_update	Whether to update system group policy in all domains using powershell command
21	shutdown_system	Whether to reboot the system
22	delete_eventlogs	Whether to delete the event log
23	delete_gpo_delay	Whether to delete the gpo after executing

Configuration description

- **white_folders**: List of folders to exclude from encryption
- **white_files**: List of files to exclude from encryption
- **white_extens**: List of extensions to exclude from encryption
- **white_hosts**: List of hostnames to exclude from encryption
- **kill_processes**: List of processes to be terminated before encryption
- **kill_services**: List of services to be terminated before encryption
- **gate_urls**: List of URLs to be used as the C2 server
- **impers_accounts**: List of credentials to be used for logon
- **note**: Ransom note content

```
~~~ LockBit 3.0 the world's fastest ransomware since 2019~~~
>>>> Your data are stolen and encrypted
The data will be published on TOR website if you do not pay the ransom
Links for Tor Browser:http://lockbitapt2yfbt7lchxejug47kmqvqqxvvjpqkmevv4l3azl3gy6pyd.onionhttp://lo
>>>> AdvertisementWould you like to earn millions of dollars $$$ ?Our company acquire access to netw
If you want to contact us, write in jabber or tox.Tox ID LockBitSupp: 3085B89A0C515D2FB124D645906F5D
```

3. Builder.exe

`Builder.exe` is a tool to generate LockBit 3.0 Encryptor and Decryptor. Encryptor and Decryptor are embedded in the resource section.

- 100: LockBit 3.0 Decryptor (EXE)
- 101: LockBit 3.0 Encryptor (EXE)
- 103: LockBit 3.0 Encryptor (DLL)
- 106: LockBit 3.0 Encryptor (Reflective DLL)

The parameters used during execution are as follows.

-type

- enc: Generate Encryptor
- dec: Generate Decryptor

-config

- Configuration file path

-exe, -dll, -ref(reflectiveDLL)

- File type to be created

-pass

- When creating an Encryptor, the password required to execute the Encryptor
- Passwords required to execute Encryptor are stored in `Password_exe.txt` and `Password_dll.txt` respectively

-pubkey, -privkey

- Path of the key file to be used when creating Encryptor and Decryptor

-ofile

- File path to save

4. Keygen.exe

`Keygen.exe` is a tool that generates key pairs required for encryption. The parameters used during execution are as follows.

- -path : Folder path to save generated key pair file

- -pubkey : File name to use for Encryptor as public key (256 bytes)

— The first 128 bytes contain e value (fixed at 65537), and the last 128 bytes contain N value

- -privkey : File name to use for Encryptor as private key (256 bytes)

— The first 128 bytes contain d value and the last 128 bytes contain N value

Key generation is performed as follows.

- keygen.exe is written based on MIRACL.
- Generates an RSA-1024 key to encrypt the file encryption key, and the **e** value is fixed to 65537.
- When generating 512-bit prime numbers **p** and **q**, create a 256-byte seed with the **rdrand** x86 instruction.
- Then, pass the seed to the *strong_init* function of MIRACL to initialize the CSPRNG defined in *mrstrong.c*, and use the *strong_bigdig* function to get a 512-bit value, which will be used for generating a prime number.
- The keygen.exe uses a modified version of MIRACL, which uses **RIPEMD-160** instead of SHA-256 inside the CSPRNG from *mrstong.c*.

Afterward, a 16-byte Decryption ID is generated to identify the infected PC and stored in the DECRYPTION_ID.txt file.

File information

1. Build.bat

- MD5 : 4e46e28b2e61643f6af70a8b19e5cb1f
- SHA-1 : 804a1d0c4a280b18e778e4b97f85562fa6d5a4e6
- SHA-256 : 8e83a1727696ced618289f79674b97305d88beeeabf46bd25fc77ac53c1ae339
- FileType : BAT

2. config.json

- MD5 : a6ba7b662de10b45ebe5b6b7edaa62a9
- SHA-1 : f3ed67bdaef070cd5a213b89d53c5b8022d6f266
- SHA-256 : 3f7518d88aefd4b1e0a1d6f9748f9a9960c1271d679600e34f5065d8df8c9dc8
- FileType : json

3. Builder.exe

- MD5 : c2bc344f6dde0573ea9acdfb6698bf4c
- SHA-1 : d6ae7dc2462c8c35c4a074b0a62f07cfef873c77
- SHA-256 : a736269f5f3a9f2e11dd776e352e1801bc28bb699e47876784b8ef761e0062db
- CreationTime : 2022–09–14 08:31:18
- FileType : EXE

4. Keygen.exe

- MD5 : 71c3b2f765b04d0b7ea0328f6ce0c4e2
- SHA-1 : bf8ecb6519f16a4838ceb0a49097bcc3ef30f3c4
- SHA-256 : ea6d4dedd8c85e4a6bb60408a0dc1d56def1f4ad4f069c730dc5431b1c23da37
- CreationTime : 2022–09–09 08:58:31
- FileType : EXE
