Earth Lusca Uses KTLVdoor Backdoor for Multiplatform Intrusion

: 9/4/2024

Malware

While monitoring Earth Lusca, we discovered the threat group's use of KTLVdoor, a highly obfuscated multiplatform backdoor, as part of a large-scale attack campaign.

By: Cedric Pernet, Jaromir Horejsi September 04, 2024 Read time: 6 min (1736 words)

Summary

- During our monitoring of the Chinese-speaking threat actor Earth Lusca, we discovered a new multiplatform backdoor written in Golang, named KTLVdoor, which has both Microsoft Windows and Linux versions.
- KTLVdoor is a highly obfuscated malware that masquerades as different system utilities, allowing attackers to carry out a variety of tasks including file manipulation, command execution, and remote port scanning.
- The malware's configuration and communication involve sophisticated encryption and obfuscation techniques to hinder malware analysis.
- The scale of the attack campaign is significant, with over 50 C&C servers found hosted at a China-based company; it remains unclear whether the entire infrastructure is exclusive to Earth Lusca or shared with other threat actors.

We discovered a new multiplatform backdoor written in Golang that we named KTLVdoor while monitoring Earth Lusca, a Chinese-speaking threat actor we had previously covered. Our investigation also uncovered both Microsoft Windows and Linux versions of this new malware family.

This previously unreported malware is more complex than the usual tools used by the threat actor. It is highly obfuscated and is being spread in the wild impersonating various system utilities names or similar tools, such as sshd, java, sqlite, bash, edr-agent, and more. The backdoor (agent) is usually distributed as a dynamic library (DLL, SO). The malware features enable the attackers to fully control the environment: run commands, manipulate files, provide system and network information, using proxies, download/upload files, scan remote ports and more.

The scale of the attack campaign is surprising, as we were able to find more than 50 C&C servers, all hosted at Alibaba in China, communicating with variants of the malware family. While some of those malware samples are tied to Earth Lusca with high confidence, we cannot be sure that the whole infrastructure is used solely by this threat actor. The infrastructure might be shared with other Chinese-speaking threat actors.

We could only find one target of the operation for the moment, a trading company based in China. It is not the first time a Chinese-speaking threat actor has targeted a Chinese company; groups like Iron Tiger and Void Arachne have likewise used tools aimed specifically at Chinese-language speakers.

KTLVdoor malware analysis

Highly obfuscated

Most of the samples discovered in this campaign are obfuscated: embedded strings are not directly readable, symbols are stripped and most of the functions and packages were renamed to random Base64-like looking strings, in an obvious effort from the developers to slow down the malware analysis (Figure 1).
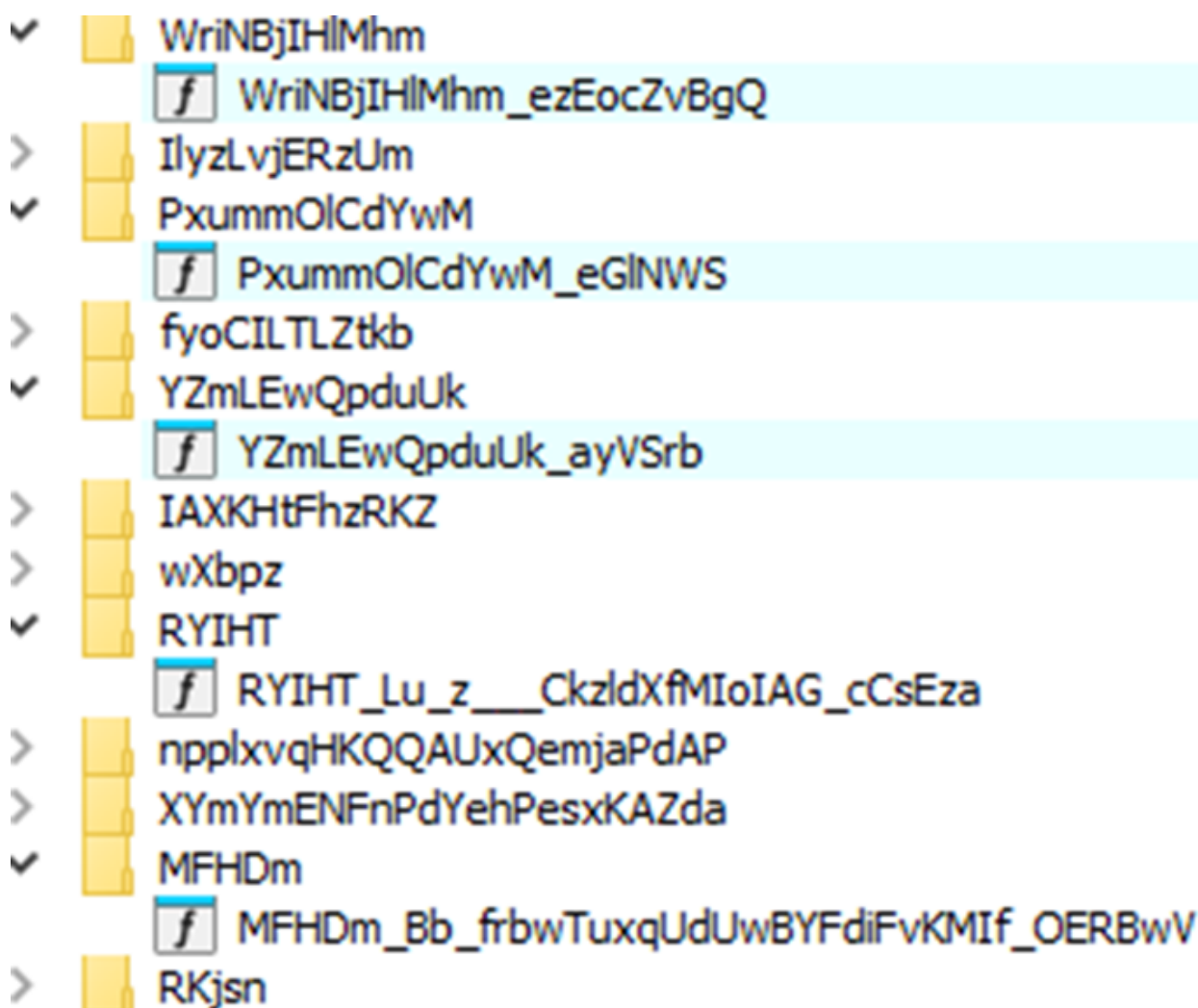

Figure 1. Obfuscated function names as shown in decompiler

Configuration

The first step is the initialization of the agent's configuration parameters. The initialization values are XOR-encrypted and Base64-encoded in the agent's binary (Figure 2).

```
KTLV·proto.·    http·host.     ·long_connection_boundary·Ê2        ·jitter·2
    .conn_timeout··'       ·auth_param.|.  KTLV·http_header.R.  Accept: text
/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,ima
ge/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7.User-Agent: Mozill
a/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/112.0.0.0 Safari/537.36.Accept-Language: zh-CN,zh;q=0.9.Content-Type
: application/octet-stream·uri..   /favicon.ico·proxy.    ·tls··.connect.¾
  8rgC18NQUUYvaBPEk2oqnxjppETZRhSi2N2oCjdCab9zsPYCbk+O/hx7u+Grew==:8rgC18NQ
UUYvaBPEk2kinxjjoETZRRSi0NCrBDdFIg90MmoI0wKBJ7ezi5lUni8=:8rgC18NQUUYvaBPEk2
otnxjuuFnbWxK51d+mCjAWbUPhEiN0py/HWfqdWevU.client_id.$   7a10025f-e332-4512
-b56a-3f38c6522e0d·debug· ·conn_max_retry··        ·domain.    ·sleep·0u
  ·secret..   Ou0LSMYJx9M9fComjFV2SEffhcMs8PZ·max_read_limit··         ·short
 connection wait time·ô.
```

Figure 2. Example of the decrypted configuration

The configuration's file format is a custom TLV-like (length-type-length-value) format. The "KTLV" marker is usually prepended with four-byte length of the structure and behaves like a marker of the beginning of the structure. Then, a list of parameters and their values follows. From Figure 3, you can notice a "proto" parameter, which is five bytes long (notice 05 followed by **proto** string, followed by type 02 (= string), followed by length 04 (= 4 bytes), followed by string **http**, which is the protocol parameter value.

```
4B 54 4C 56   05 70 72 6F   74 6F 02 04   00 00 00 68    KTLV·proto ·      h
74 74 70 04   68 6F 73 74   02 00 00 00   00 18 6C 6F    ttp·host      ·lo
```

Figure 3. Parameter proto, type string (02), value http, as stored in configuration file

```
06 64 6F 6D   61 69 6E 02   00 00 00 00   05 73 6C 65    ·domain       ·sle
65 70 08 30   75 00 00 00   00 00 00 06   73 65 63 72    ep·0u       ·secr
```

Figure 4. Parameter sleep, type long long (08), value 0x7530 = 30 000 milliseconds = 30 seconds, as stored in configuration file

The supported type formats are shown in Table 1:

Value	Type
01	structure/iteration (followed by KTLV marker)
02	string
03	boolean (1byte)
08	long long (8 bytes)
09	integer (4 bytes)
0B	byte (1 byte)

Table 1. Supported value type formats

The configuration file may contain the following parameters, as shown in Table 2:

Parameter name	Type	Description/comment
listen	string	active mode (default) / passive mode
connect	string	Encrypted C&C servers
duplex	boolean	Simplex or duplex delivery
conn_timeout	long long	
max_read_limit	long long	
conn_max_retry	long long	
proxy	string	

proto	string	http, tcp, dns, icmp
domain	string	
host	string	
secret	string	To decrypt value(s) of "connect"
tls	boolean	Enabled or disabled
stls	boolean	Enabled or disabled
sleep	long long	
jitter	long long	Sleep time variation
silent	boolean	
long_connection_boundary	long long	
short_connection_wait_time	long long	
client_id	string	Hardcoded GUID of target
external_channel_enabled	boolean	Should get external IP or not
external_type	string	
auth_param	struct	Contains "http_header" and "uri"
http_header	string	
uri	string	Request's URL path
debug	boolean	Enabled or disabled

Table 2. Supported configuration parameters

The configuration is processed, and the internal configuration structure (Config) is updated. Part of the Config structure is the HostInfo structure, which also contains additional parameters about the currently infected machine (Table 3). These structures are updated based on the current machine environment.

Parameter name	Type	Description/comment
Session	string	Randomly generated GUID during each run
RealIP	string	Obtained from ipconfig / ifconfig
Username	string	
Hostname	string	
ProcessName	string	
Executable	string	
PID	uint32	Process ID
ParentProcessName	string	
PPID	uint32	Parent process ID
Arch	string	32 or 64 bit
OS	string	OS name
Platform	string	OS name + version
Disks	string	List of available disks
DiskDetails	string	List of available disks + their sizes
Uptime	string	
Feature	string	MachineGuid from HKLM\SOFTWARE\Microsoft\Cryptography
Protocol	string	Value from the config
Proxy	string	Value from the config
Domain	string	Value from the config
Host	string	Value from the config
TLSEnable	boolean	Value from the config
STLSEnable	boolean	Value from the config
ExternalIP	string	External IP address obtained via http://myip.ipip.net; only if

		external_channel_enabled set
SleepTime	uint64	Value from the config
Jitter	uint64	Value from the config
ReConnectTime	uint64	Value from the config
Env	string	Environment variables
ClientID	string	Value from the config
IsAdmin	boolean	True or false
Mode	string	Active or passive

Table 3. HostInfo parameters

Connection settings

The C&C server(s) are stored in the "connect" value. The value is AES-GCM-encrypted and Base64-encoded. The AES-GCM method uses a standard prepended 12-byte nonce and appended 16-byte tag. The AES-GCM key is derived from a "secret" value by computing the MD5 hash of it and using key padding (extending the key size) to 32-bytes by appending 16 zeroes (0x00 bytes) to it.

KTLVdoor malware communication

After the initialization steps are completed, the agent starts a communication loop with the C&C server. The communication is done by sending and receiving messages, which are GZIP-compressed and AES-GCM-encrypted. Based on the configuration settings, the message delivery can be either in simplex mode (one device on channel can only send, another device on the channel can only receive) or in duplex mode (both devices can simultaneously send and receive messages).

Each message contains a message header followed by the message data (msg).

Field name	Field type	Field value
sender	String	Session ID or admin
receiver	String	Session ID or admin
token	String	
route	String	
task_id	uint64	
task_status	uint8	
task_type	uint64	
sub_task_type	uint64	

Table 4. Message header fields

Figure 5. Message with OS info sent to the C&C server

Notice that the sender of this outgoing message (from the infected machine to the C&C server) has the session ID of the currently infected machine. The receiver is "admin", which is the C&C server. In the case of the incoming message (from the C&C server to the infected machine), the "sender" is "admin" and the "receiver" is the session ID. In the case of sending the HostInfo message to the C&C server, notice that the parameter name "msg" (containing message content) followed by "KTLV" marker (Figure 5), which contains all the fields from HostInfo structure (Table 4).

Receiving task

The agent implements several handlers for processing received tasks from the C&C server (Table 5).

Handler	Subtasks	Parameters	Description
Breakchain		shell flag cmd abs_path	Start terminal Run command Wait three seconds Kill terminal (SIGKILL)
Exit			Exit process
FileDownload		file_path section_size	Read file Upload it to C&C
FileMD5		file_path	Read file Compute MD5 hash
FileManager	01 - list all files 02 - Create dir 03 - Create file 04 - Delete file 05 - Copy file 06 - Rename 07 - Write file 08 - Read file 09 - Change access permissions	dirName OR file_path OR dst_path src_path OR file_path file_content OR mode	File and directory operations
FileUpload		file_path file_contents position	Write data from server to file on victim machine

GC			Run garbage collection
InteractiveShell		send data OR start OR stop OR recv	
NetStat	01 - list connections		
PortScan		gateway ips ports	
Process	01 - list 02 - kill	pid	
RefreshHostInfo			
Run		cmdn!	Run command
Sleep		sleep_time jitter	
TimeStomp		src_path dst_path time	
TaskCache	01 - list of tasks 02 - delete task 03 - clear task cache	task_id	
SoInject	04 - inject to library	plugin_task_type tmp_payload_cache params	Run shellcode, Linux platform
ReflectDllInject			Run shellcode, Windows platform
Socks	01 – start handler 02 – get task 04 – data via TCP 05 – close TCP 06 – data via UDP 08 – connect to address via UDP	seq addr username password OR task_id OR seq data	Socks proxy

Table 5. Handlers for processing tasks from C&C server

PortScan implements many scanning methods, including:

- ScanTCP
- ScanRDP
- ScanWinRM
- ScanSmb2
- RdpWithNTLM
- DialTLS
- DialTCP

- ScanPing
- ScanPing
- ScanMssql
- ScanBanner
- ScanWeb

Conclusion

We have been able to tie samples of KTLVdoor to the threat actor Earth Lusca with high confidence. However, we were not able to tie several other samples of this malware family to this threat actor. In addition, the size of the infrastructure we have been able to discover is very unusual. Seeing that all C&C servers were on IP addresses from China-based provider Alibaba, we wonder if the whole appearance of this new malware and the C&C server could not be some early stage of testing new tooling.

This new tool is used by Earth Lusca, but it might also be shared with other Chinese-speaking threat actors. While a lot of details on this campaign are not yet known, we will keep monitoring this activity and possibly give updates about it at a later time.

Trend solutions

Organizations looking to defend themselves from sophisticated attacks can consider powerful security technologies such as Trend Vision One™, which allows security teams to continuously identify attack surfaces, including both known and unknown, plus managed and unmanaged cyber assets. Vision One™ offers multilayered protection and behavior detection, helping block malicious tools and services before they can inflict damage on user machines and systems.

Indicators of Compromise (IOCs)

The full list of IOCs can be found here.