DarkUniverse – the mysterious APT framework #27

Authors

- Expert Andrey Dolgushev
- Expert Vasily Berdnikov
- Expert Alexander Fedotov
- Expert GReAT

In April 2017, ShadowBrokers published their well-known ‘Lost in Translation’ leak, which, among other things, contained an interesting script that checked for traces of other APTs in the compromised system.

```
def find_27():
    search_set = set(('qtlib.sqt', 'zl4vq.sqt', 'dfrgntfs5.sqt', 'msvcrt58.sqt'))
    if (not datastore.SYSTEMROOT_FILE_SET.isdisjoint(search_set)):
        return True
    return False
```

In 2018, we found an APT described as the 27th function of this script, which we call ‘DarkUniverse’. This APT was active for at least eight years, from 2009 until 2017. We assess with medium confidence that DarkUniverse is a part of the ItaDuke set of activities due to unique code overlaps. ItaDuke is an actor known since 2013. It used PDF exploits for dropping malware and Twitter accounts to store C2 server urls.

Technical details

Infection vector

Spear phishing was used to spread the malware. A letter was prepared separately for each victim to grab their attention and prompt them to open an attached malicious Microsoft Office document.

Each malware sample was compiled immediately before being sent and included the latest available version of the malware executable. Since the framework evolved from 2009 to 2017, the last releases are totally different from the first ones, so the current report details only the latest available version of the malware used until 2017.

The executable file embedded in the documents extracts two malicious files from itself, updater.mod and glue30.dll, and saves them in the working directory of the malware – %USERPROFILE%\AppData\Roaming\Microsoft\Windows\Reorder.

After that, it copies the legitimate rundll32.exe executable into the same directory and uses it to run the updater.mod library.

The updater.mod module

This module is implemented as a dynamic-link library with only one exported function, called callme@16. This module is responsible for such tasks as providing communication with the C2 server, providing the malware integrity and persistence mechanism and managing other malware modules.

The persistence mechanism is provided by a link file, which is placed by updater.mod into the startup folder, ensuring malware execution after a reboot. If the link file becomes corrupted, the updater.mod module restores it.

Communication with C2

In this campaign the C2 servers were mostly based on cloud storage at mydrive.ch. For every victim, the operators created a new account there and uploaded additional malware modules and a configuration file with commands to execute it. Once executed, the updater.mod module connected to the C2 and performed the following actions:

- downloaded the command file to the working directory;
- uploaded files collected and prepared by additional malicious modules (if any) to the C2. These files were located in a directory called ‘queue’ or ‘ntfsrecover’ in the working directory. Files in this directory could have one of two extensions: .d or .upd depending on whether they had already been uploaded to the server or not.
- downloaded additional malware modules:
    - dfrgntfs5.sqt – a module for executing commands from the C2;
    - msvcrt58.sqt – a module for stealing mail credentials and emails;
    - zl4vq.sqt – legitimate zlib library used by dfrgntfs5;
    - %victim_ID%.upe – optional plug-in for dfrgntfs5. Unfortunately, we were unable to obtain this file.

All malware modules are encrypted with a custom algorithm:

```
43        for ( i = 0; i < filesize; ++i )
44        {
45          key1 = key4[5] * key1 % key4[4];
46          xor = (unsigned __int8)(key1 ^ BYTE1(key1));
47          key2 = key4[3] * key2 % key4[2];
48          xor ^= (unsigned __int8)(key2 ^ BYTE1(key2));
49          key3 = key4[1] * key3 % key4[0];
50          xor ^= (unsigned __int8)key3;
51          data[i] ^= xor;
52        }
```

The credentials for the C2 account are stored in the configuration that is placed in the registry, but the updater.mod module also stores a copy as an encrypted string in the executable file. Also, the configuration specifies how often updater.mod polls the C2, supporting both an active mode and a partly active mode.

Malware configuration in the registry

The malware configuration is stored in the registry in the SOFTWARE\AppDataLow\GUI\LegacyP entry. Different values are detailed in the following table:

Value name	Description
C1	C2 domain.
C2	C2 domain path.
C3	C2 credential username.
C4	C2 credential password.
install	1 if malware is installed.
TL1	DESACTIVAR | HABILITAR – specifies whether msvcrt58 and glue libraries are active.
TL2, TL3	If TL1 is not NULL, it specifies time bounds when TL1 option is applied.
“kl”	If 1, updater.mod should download msvcrt58.sqt from C2 again.
“re”	If 1, updater.mod should download dfrgntfs5.sqt from C2 again.
“de”	If not 0, framework should uninstall itself.
“cafe”	REDBULL | SLOWCOW specifies how often updater.mod polls C2.
“path”	Path to the folder from which files are being sent to C2.

Modules glue30.dll and msvcrt58.sqt

The glue30.dll malware module provides keylogging functionality. The updater.mod module uses the Win API function SetWindowsHookExW to install hooks for the keyboard and to inject glue30.dll into processes that get keyboard input. After that, glue30.dll loads and begins intercepting input in the context of each hooked process.

The msvcrt58.sqt module intercepts unencrypted POP3 traffic to collect email conversations and victims’ credentials. This module looks for traffic from the following processes:

- outlook.exe;
- winmail.exe;
- msimn.exe;

- nlnotes.exe;
- eudora.exe;
- thunderbird.exe;
- thunde~1.exe;
- msmsgs.exe;
- msnmsgr.exe.

The malware parses intercepted POP3 traffic and sends the result to the main module (updater.mod) for uploading to the C2. This is done by hooking the following network-related Win API functions:

- ws2_32.connect;
- ws2_32.send;
- ws2_32.recv;
- ws2_32.WSARecv;
- ws2_32.closesocket.

The dfrgntfs5.sqt module

This is the most functional component of the DarkUniverse framework. It processes an impressive list of commands from the C2, which are listed in the following table.

Command	Description
VER	Sends malware version to server.
DESINSTALAR	Uninstalls itself.
PANTALLA	Takes screenshot of the full screen and saves it to the \queue folder.
CAN_TCP, CAN_HTTP, CAN_HTTPS	Injects a shellcode into IE that establishes a direct connection with the C2, downloads additional code, sends info about the download results to the C2 and executes the downloaded code.
MET_TCP, MET_HTTPS	Also injects a shellcode into IE. The only difference with the previous command set is that in this case the shellcode doesn't send any additional info to the C2 – it only establishes the connection, downloads additional code and executes it.
CAN_HTTP_LSASS	Injects the same shellcode as in the case of CAN_HTTP into the LSASS.exe process.
SCAN/STOPSCAN	Starts/stops network scan. Collects lots of different info about the local network.
CREDSCAN	Brute-forces IP range with specified username and password.
ACTUALIZAR	Updates dfrgntfs5.sqt.
ACTUALIZARK	Updates msvcrt58.sqt.
SYSINFO	Collects full system info.
REDBULL	Sets cafe flag to 1 – active.
SLOWCOW	Sets cafe flag to 0 – slow mode.
X	Runs specified process and logs its output, then prepares this output log for uploading to the C2.
T	Obtains list of files from a specific directory.
TAUTH	Obtains list of files of remote server if specified credentials are valid.
G	Sends a file to the C2.
GAUTH	Downloads a particular file from a shared resource if specified credentials

	are valid.
SPLIT	Splits file into 400 KB parts and uploads them to the C2.
FLUSH	Sends file with the data collected by all components that day and deletes it.
C1 – C4	Sets the C2 in its configuration in the registry (C1-C4).
TL1 – TL3	Sets the active state in its configuration in the registry (T1-T3).
ONSTART	Sets process to be started every malware startup.
CLEARONSTART	Undoes previous ONSTART command.
ARP	Runs unavailable ARP module (uncparse.dll – unavailable). This module stores data in a file internally named arpSniff.pcap.
AUTO	Automatically looks for updates of predefined files.
MANUAL	Files in the specified directory are searched using the * .upd pattern, all found files are deleted.
REGDUMP	Collects information from the registry.
PWDDUMP	Collects and decrypts credentials from Outlook Express, Outlook, Internet Explorer, Windows Mail and Windows Live Mail, Windows Live Messenger, and also Internet Cache;
LOGHASH	Injects process into lsass.exe and starts collecting password hashes in the file checksums.bk.
SENDLOGHASH	Sends collected lsass.exe process password hashes to the C2.
PROXYINFO	Checks if credentials for proxy are valid.
DHCP	Sets DHCP settings for local machine.
DNS	Sets DNS settings for local machine.
FAKESSL	Provides basic MITM functionality.

Victimology

We recorded around 20 victims geolocated in Syria, Iran, Afghanistan, Tanzania, Ethiopia, Sudan, Russia, Belarus and the United Arab Emirates. The victims included both civilian and military organizations. We believe the number of victims during the main period of activity between 2009 and 2017 was much greater.

Conclusions

DarkUniverse is an interesting example of a full cyber-espionage framework used for at least eight years. The malware contains all the necessary modules for collecting all kinds of information about the user and the infected system and appears to be fully developed from scratch. Due to unique code overlaps, we assume with medium confidence that DarkUniverse's creators were connected with the ItaDuke set of activities. The attackers were resourceful and kept updating their malware during the full lifecycle of their operations, so the observed samples from 2017 are totally different from the initial samples from 2009. The suspension of its operations may be related to the publishing of the 'Lost in Translation' leak, or the attackers may simply have decided to switch to more modern approaches and start using more widely available artefacts for their operations.

Appendix I – Indicators of Compromise

MD5 Hashes

- 1addee050504ba999eb9f9b1ee5b9f04
- 4b71ec0b2d23204e560481f138833371
- 4e24b26d76a37e493bb35b1a8c8be0f6
- 405ef35506dc864301fada6f5f1d0711
- 764a4582a02cc54eb1d5460d723ae3a5
- c2edda7e766553a04b87f2816a83f563
- 71d36436fe26fe570b876ad3441ea73c

A full set of IOCs, including YARA rules, is available to customers of the Kaspersky Intelligence Reporting service. For more information, contact intelreports@kaspersky.com