The Missing Piece – Sophisticated OS X Backdoor Discovered

SL securelist.com/blog/research/75990/the-missing-piece-sophisticated-os-x-backdoor-discovered/
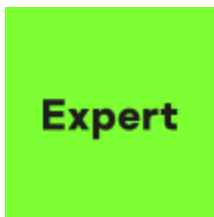

Malware descriptions

Malware descriptions

07 Sep 2016

minute read

Authors

Expert [Stefan Ortloff](#)

In a nutshell

- Backdoor.OSX.Mokes.a is the most recently discovered OS X variant of a cross-platform backdoor which is able to operate on all major operating systems (Windows,Linux,OS X). Please see also our [analysis on the Windows and Linux variants](#).
- This malware family is able to steal various types of data from the victim's machine (Screenshots, Audio-/Video-Captures, Office-Documents, Keystrokes)
- The backdoor is also able to execute arbitrary commands on the victim's computer
- To communicate it's using strong AES-256-CBC encryption

Background

Back in January this year [we found a new family of cross-platform backdoors for desktop environments](#). After the discovery of the binaries for Linux and Windows systems, we have now finally come across the OS X version of Mokes.A. It is written in C++ using Qt, a cross-platform application framework, and is statically linked to OpenSSL. This leads to a filesize of approx. 14MB. Let's have a look into this very fresh sample.

"Unpacked" Backdoor.OSX.Mokes.a

Its filename was "unpacked" when we got our hands on it, but we're assuming that in-the-wild it comes packed, just like its Linux variant.

```
$ file unpacked
unpacked: Mach-O 64-bit x86_64 executable
```

Startup

When executed for the first time, the malware copies itself to the first available of the following locations, in this order:

- $HOME/Library/App Store/storeuserd
- $HOME/Library/com.apple.spotlight/SpotlightHelper
- $HOME/Library/Dock/com.apple.dock.cache
- $HOME/Library/Skype/SkypeHelper
- $HOME/Library/Dropbox/DropboxCache
- $HOME/Library/Google/Chrome/nacld
- $HOME/Library/Firefox/Profiles/profiled

Corresponding to that location, it creates a plist-file to achieve persistence on the system:

```
        -Mac-Pro-2:          root# cat /Users/         /Library/LaunchAgents/storeuserd.plist
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
        <key>Label</key>
        <string>storeuserd</string>
        <key>ProgramArguments</key>
        <array>
                <string>/Users/         /Library/App Store/storeuserd</string>
        </array>
        <key>RunAtLoad</key>
        <true/>
        <key>KeepAlive</key>
        <true/>
</dict>
</plist>
```

After that it's time to establish a first connection with its C&C server using HTTP on TCP port 80:

```
GET /v1 HTTP/1.1
Connection: Close
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_3) AppleWebKit/537.75.14 (KHTML,
 like Gecko) Version/7.0.3 Safari/7046A194A
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,*
Host: 158.69.241.141

HTTP/1.1 200 OK
Server: nginx/1.6.2
Date:
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: close
Content-Encoding: gzip
```

The User-Agent string is hardcoded in the binary and the server replies to this "heartbeat" request with "text/html" content of 208 bytes in length. Then the binary establishes an encrypted connection on TCP port 443 using the AES-256-CBC algorithm.

```
__text:0000000100016EA1        mov     rdx, r13
__text:0000000100016EA4        sub     rdx, 0FFFFFFFFFFFFFF80h ; key
__text:0000000100016EA8        mov     esi, 256               ; bits
__text:0000000100016EAD        mov     rdi, rbx               ; userKey
__text:0000000100016EB0        call    _AES_set_encrypt_key
```

Backdoor functionality

Its next task is to setup the backdoor features:

Function name	Segment	Start
𝑓 EkomsUserActivity::EkomsUserActivity(void)	__text	000000010000CDB0
𝑓 EkomsUserActivity::service(void)	__text	000000010000CDD0
𝑓 EkomsUserActivity::~EkomsUserActivity()	__text	000000010000CEB0
𝑓 EkomsUserActivity::~EkomsUserActivity()	__text	000000010000CEC0

- **Capturing Audio**

Function name	Segment	Start
𝑓 AudioCaptureSession::record(void)	__text	00000001000DCB50
𝑓 AudioCaptureSession::setCaptureDevice(QString const&)	__text	00000001000DE080
𝑓 AudioCaptureSession::setContainerFormat(QString const&)	__text	00000001000DC8A0
𝑓 AudioCaptureSession::setFormat(QAudioFormat const&)	__text	00000001000DC890
𝑓 AudioCaptureSession::setOutputLocation(QUrl const&)	__text	00000001000DC930
𝑓 AudioCaptureSession::setState(QMediaRecorder::State)	__text	00000001000DC7E0
𝑓 AudioCaptureSession::state(void)	__text	00000001000DD080
𝑓 AudioCaptureSession::stateChanged(QMediaRecorder::State)	__text	00000001000DF010
𝑓 AudioCaptureSession::status(void)	__text	00000001000DD090
𝑓 AudioCaptureSession::statusChanged(QMediaRecorder::Sta...	__text	00000001000DF060
𝑓 AudioCaptureSession::stop(void)	__text	00000001000DCA40
𝑓 AudioCaptureSession::~AudioCaptureSession()	__text	00000001000DC6C0

- **Monitoring Removable Storage**

Function name	Segment	Start
ƒ AbRemovableStorageMonitorService::AbRemovableStorage...	__text	0000000100014770
ƒ AbRemovableStorageMonitorService::onStartService(void)	__text	0000000100014820
ƒ AbRemovableStorageMonitorService::onStopService(void)	__text	0000000100014830
ƒ AbRemovableStorageMonitorService::~AbRemovableStorag...	__text	0000000100014790
ƒ AbRemovableStorageMonitorService::~AbRemovableStorag...	__text	00000001000147D0

- **Capturing Screen (every 30 sec.)**

```
; Attributes: bp-based frame

; __int64 __fastcall AbScreenCapture::start(AbScreenCapture *__hidden this)
                public __ZN15AbScreenCapture5startEv
__ZN15AbScreenCapture5startEv proc near
                push    rbp
                mov     rbp, rsp
                mov     dword ptr [rdi+28h], 0
                and     byte ptr [rdi+24h], 0FEh
                add     rdi, 8 ; this
                mov     esi, 30000 ; int
                pop     rbp
                jmp     __ZN6QTimer5startEi ; QTimer::start(int)
__ZN15AbScreenCapture5startEv endp
```

- **Scanning the file system for Office documents (xls, xlsx, doc, docx)**

```
100818510  3A 00 2F 00 66 00 69 00  6C 00 65 00 2D 00 73 00   :./.f.i.l.e.-.s.
100818520  65 00 61 00 72 00 63 00  68 00 00 00 00 00 00 00   e.a.r.c.h.......
100818530  FF FF FF FF 01 00 00 00  00 00 00 00 00 00 00 00   ................
100818540  18 00 00 00 00 00 00 00  2D 00 00 00 00 00 00 00   ........-.......
100818550  FF FF FF FF 01 00 00 00  00 00 00 00 00 00 00 00   ................
100818560  18 00 00 00 00 00 00 00  2F 00 00 00 00 00 00 00   ........./......
100818570  FF FF FF FF 06 00 00 00  00 00 00 00 00 00 00 00   ................
100818580  18 00 00 00 00 00 00 00  2A 00 2E 00 78 00 6C 00   ........*...x.l.
100818590  73 00 78 00 00 00 00 00  FF FF FF FF 05 00 00 00   s.x.............
1008185A0  00 00 00 00 00 00 00 00  18 00 00 00 00 00 00 00   ................
1008185B0  2A 00 2E 00 78 00 6C 00  73 00 00 00 00 00 00 00   *...x.l.s.......
1008185C0  FF FF FF FF 06 00 00 00  00 00 00 00 00 00 00 00   ................
1008185D0  18 00 00 00 00 00 00 00  2A 00 2E 00 64 00 6F 00   ........*...d.o.
1008185E0  63 00 78 00 00 00 00 00  FF FF FF FF 05 00 00 00   c.x.............
1008185F0  00 00 00 00 00 00 00 00  18 00 00 00 00 00 00 00   ................
100818600  2A 00 2E 00 64 00 6F 00  63 00 00 00 00 00 00 00   *...d.o.c.......
100818610  31 39 41 46 69 6C 65 53  65 61 72 63 68 43 61 6C   19AFileSearchCal
100818620  6C 62 61 63 6B 00 00 00  00 00 00 00 00 00 00 00   lback...........
100818630  31 38 41 75 74 6F 46 69  6C 65 53 65 61 72 63 68   18AutoFileSearch
100818640  54 61 73 6B 00 00 00 00  FF FF FF FF 10 00 00 00   Task............
```

The attacker controlling the C&C server is also able to define own file filters to enhance the monitoring of the file system as well as executing arbitrary commands on the system.

Just like on other platforms, the malware creates several temporary files containing the collected data if the C&C server is not available.

- $TMPDIR/ss0-DDMMyy-HHmmss-nnn.sst (Screenshots)
- $TMPDIR/aa0-DDMMyy-HHmmss-nnn.aat (Audiocaptures)
- $TMPDIR/kk0-DDMMyy-HHmmss-nnn.kkt (Keylogs)
- $TMPDIR/dd0-DDMMyy-HHmmss-nnn.ddt (Arbitrary Data)

DDMMyy = date: 070916 = 2016-09-07
HHmmss = time: 154411 = 15:44:11
nnn = milliseconds

If the environment variable $TMPDIR is not defined, "/tmp/" is used as the location (http://doc.qt.io/qt-4.8/qdir.html#tempPath).

Hints from the author

The author of this malware again left some references to the corresponding source files:

Detection

We detect this type of malware as **HEUR:Backdoor.OSX.Mokes.a**

IOCs

Hash:
664e0a048f61a76145b55d1f1a5714606953d69edccec5228017eb546049dc8c

Files:
$HOME/LibraryApp Store/storeuserd
$HOME/Library/com.apple.spotlight/SpotlightHelper
$HOME/Library/Dock/com.apple.dock.cache
$HOME/Library/Skype/SkypeHelper
$HOME/Library/Dropbox/DropboxCache
$HOME/Library/Google/Chrome/nacld
$HOME/Library/Firefox/Profiles/profiled
$HOME/Library/LaunchAgents/$filename.plist
$TMPDIR/ss*-$date-$time-$ms.sst
$TMPDIR/aa*-$date-$time-$ms.aat
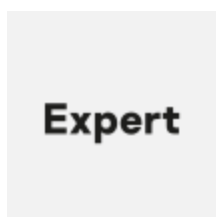$TMPDIR/kk*-$date-$time-$ms.kkt
$TMPDIR/dd*-$date-$time-$ms.ddt

Hosts:
158.69.241[.]141
jikenick12and67[.]com
cameforcameand33212[.]com

User-Agent:
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_3) AppleWebKit/537.75.14 (KHTML, like Gecko) Version/7.0.3 Safari/7046A194A

- Apple MacOS
- Backdoor
- Linux
- Microsoft Windows
- Non-Windows Malware

Authors

Stefan Ortloff

The Missing Piece – Sophisticated OS X Backdoor Discovered

Your email address will not be published. Required fields are marked *

GReAT webinars

From the same authors

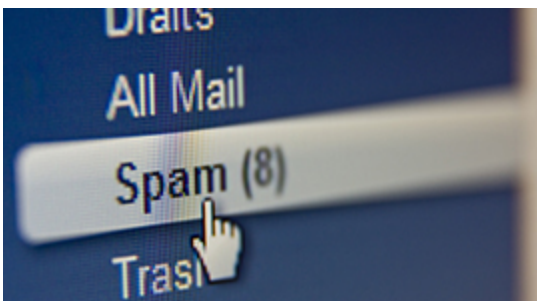## New wave of Mirai attacking home routers

Beware of Backdoored Linux Mint ISOs

From Linux to Windows – New Family of Cross-Platform Desktop Backdoors Discovered
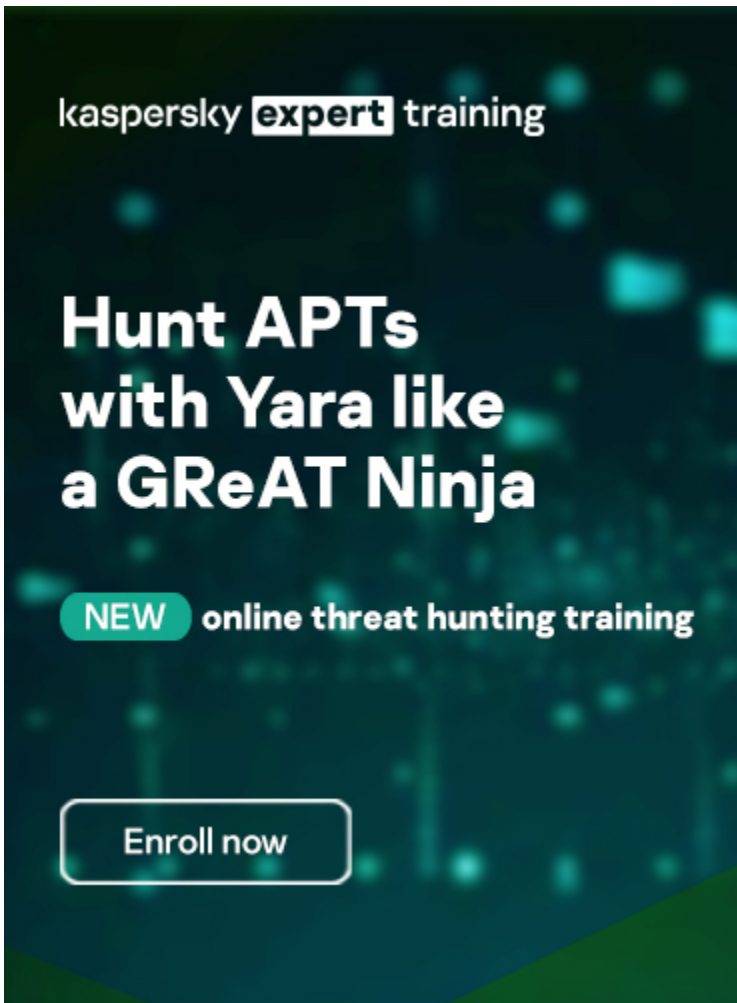

Shellshock and its early adopters

Spammy Facebook friends from the neighborhood

Subscribe to our weekly e-mails

The hottest research right in your inbox

-
-
-

-