macOS Hacking Tricks

NullByte

$Whoami

Ricardo L0gan

Security Specialist with over 15 years of experience, malware research enthusiastic, pentest and reverse engineering. I’ve a solid knowledge on topics like network security, hardening and tuning across multiple platforms such as Windows, Linux, OS X and Cisco. Beginner in programming languages as Python, C and Assembly.

In Brazil, I contribute with some security conferences organizations such as SlackShow Community, bSides SP and Hackers to Hackers Conference (H2HC).

Long live Open Source - Use Linux (Slackware)

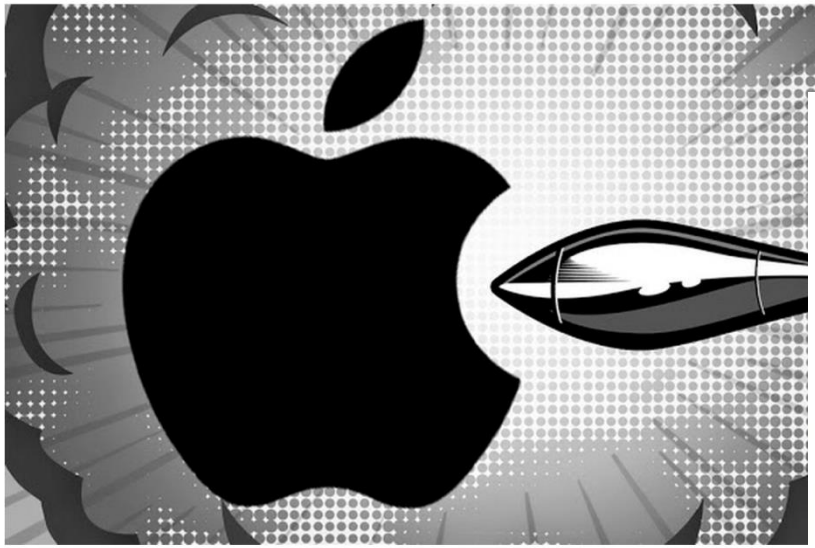
Agenda

0x00 Motivation of Research
0x01 macOS Security Characteristics
0x02 Hacking macOS target
0x03 Hacking macOS target + Demo
0x04 macOS Tools
0x05 Reference
0x06 Conclusion

0x00 Motivation of Research

Windows? NO, Linux and Mac OS X Most Vulnerable Operating System In 2014

Tuesday, February 24, 2015 Swati Khandelwal



Source: http://thehackernews.com/2015/02/vulnerable-operating-system.html

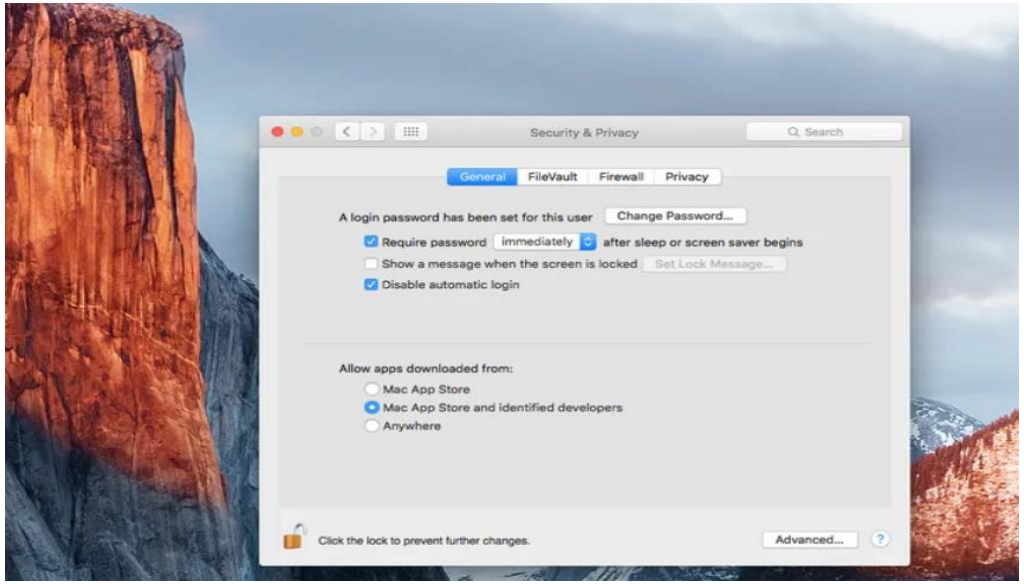
Revoltado com a Apple, hacker do mundo jailbreak publica detalhes de uma falha de segurança do OS X

Rafael Fischmann 23/07/2015 às 09:31

0x00 Motivation of Research

Researcher exposes vulnerability in macOS Gatekeeper security mechanism

Chance Miller - May. 25th 2019 9:52 am PT @ChanceHMiller



0x00 Motivation of Research

OSX/CrescentCore 2019

OSX/Linker 2019

LoundMiner 2019

OSX/MaMi 2018

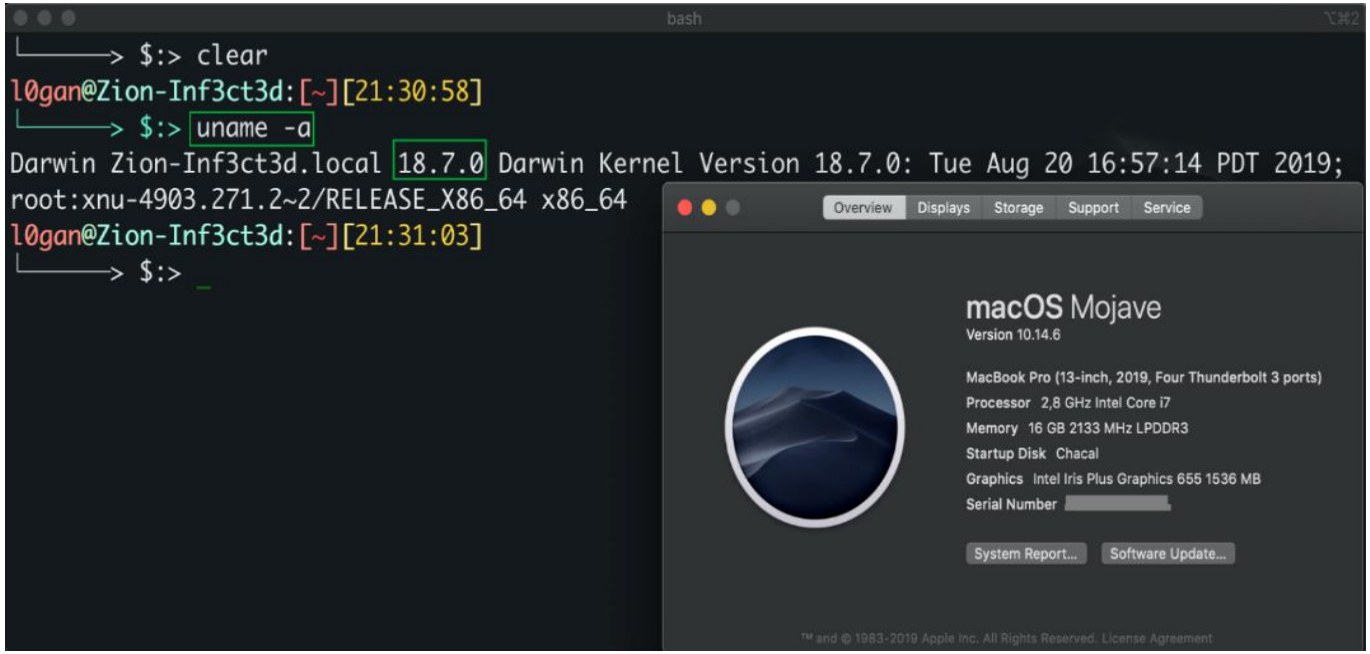
Crossrider (OSX/Shalyer) 2018

Mac Auto Fixer 2018

Mshelper 2018

0x01 Security Characteristics

- A modern Operating system (macOS fanBoy LOL) Unix based
- Kernel XNU is based on micro-kernel of NeXTSTEP (Mach) and kernel of BSD (FreeBSD)
- Lots of userland applications
- Mac OS X has grown significantly in market share.



0x01 Security Characteristics

SIP (System Integrity Protection)

Xprotect

FileVault

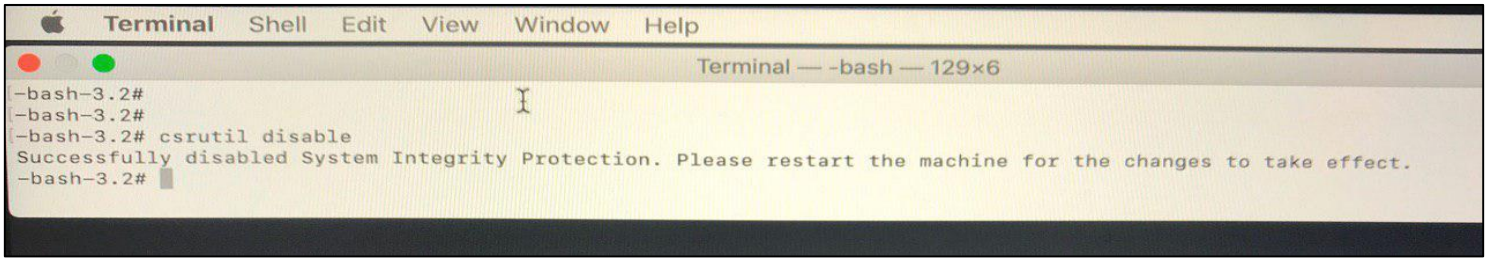
Gatekeeper

Secure Boot

0x01 Security Characteristics

SIP

Originally introduced with **OS X** El Capitan, System Integrity Protection, usually referred to as **SIP**, is a security feature built into the Mac operating system that's designed to protect most system locations, system processes, and Kernel extensions from being written to, modified, or replaced.



0x01 Security Characteristics

XProtect

```
bash
l0gan@Zion-Inf3ct3d:[/System/Library/CoreServices/CoreTypes.bundle/Contents/Resources][19:12:15]
└──────> $:> pwd
/System/Library/CoreServices/CoreTypes.bundle/Contents/Resources
l0gan@Zion-Inf3ct3d:[/System/Library/C
└──────> $:> ls XProtect.*
XProtect.meta.plist XProtect.plist
l0gan@Zion-Inf3ct3d:[/System/Library/C
└──────> $:> _
```

```
<dict>
  <key>Description</key>
  <string>OSX.KeRanger.A</string>
  <key>LaunchServices</key>
  <dict>
    <key>LSItemContentType</key>
    <string>com.apple.application-bundle</string>
  </dict>
  <key>Matches</key>
  <array>
    <dict>
      <key>MatchFile</key>
      <dict>
        <key>NSURLTypeIdentifierKey</key>
        <string>public.unix-executable</string>
      </dict>
      <key>MatchType</key>
      <string>Match</string>
      <key>Pattern</key>
      <string>488DBDD0EFFFFFBE00000000BA0004000031C049
</string>
    </dict>
  </array>
</dict>
```

Anti-virus product is internally referred to as XProtect. Implemented within the CoreServicesUIAgent.

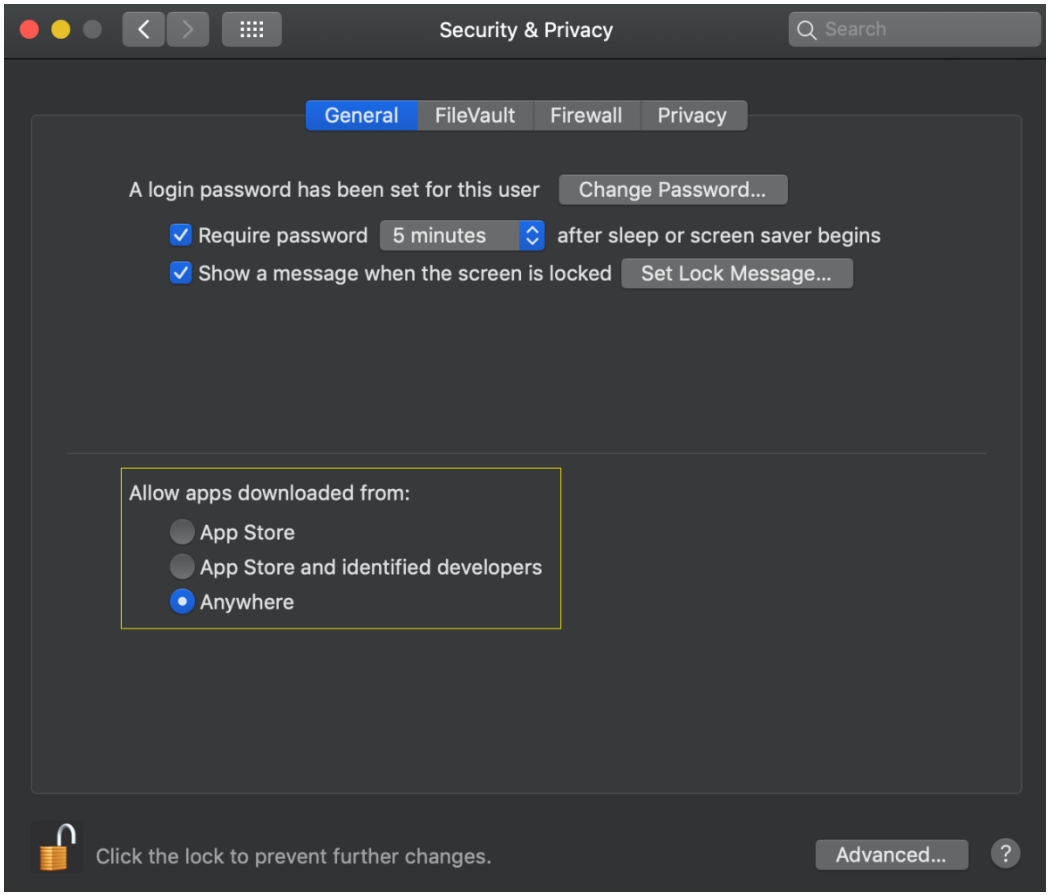
0x01 Security Characteristics

Filevault

Apple implementation of encrypting your data on **macOS** and Mac hardware. It will encrypt all of your data on your startup disk (although you can also encrypt your Time Machine backups as well) and once enabled, it will encrypt your data on the fly and will work seamlessly in the background

0x01 Security Characteristics

Gatekeeper

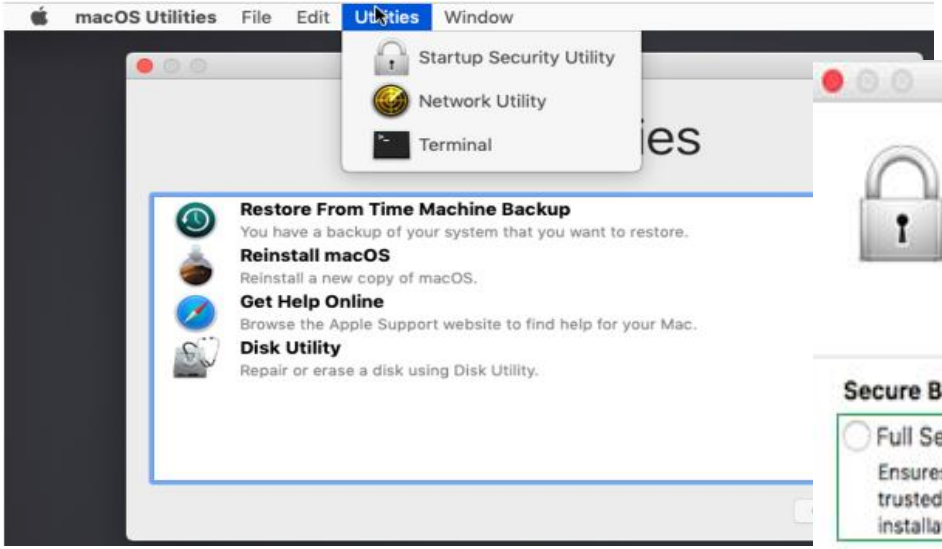


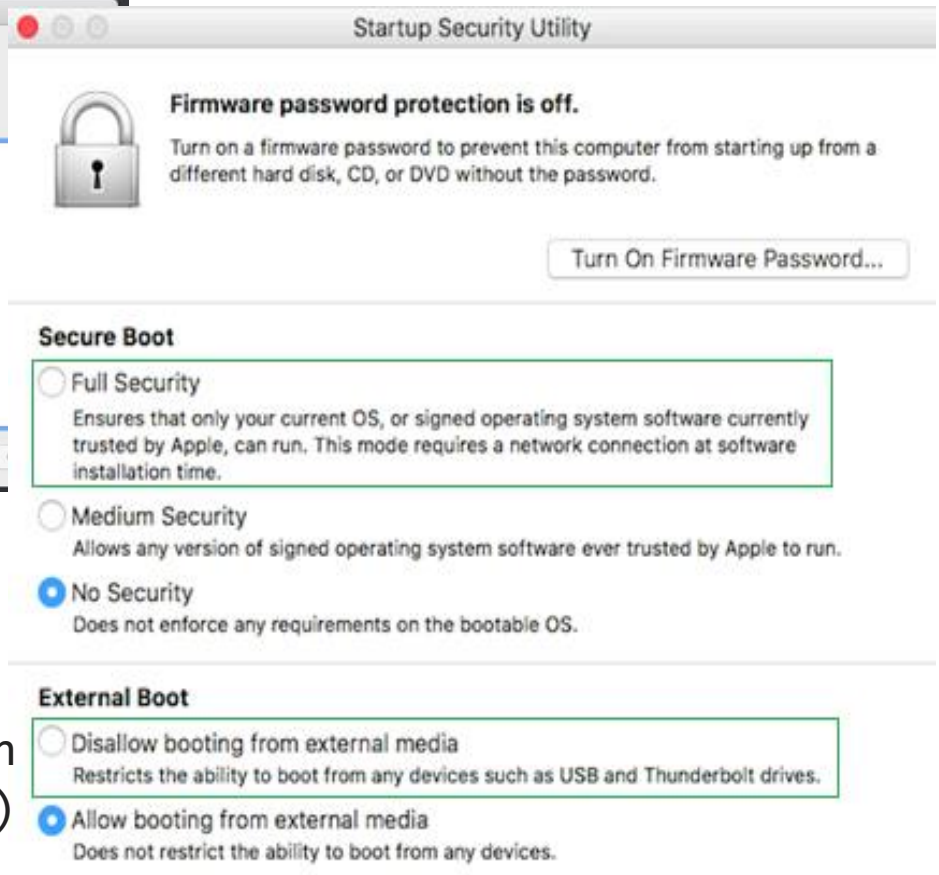
Security feature of the macOS operating system by Apple. It enforces code signing and verifies downloaded applications before allowing them to run, thereby reducing the likelihood of inadvertently executing malware.

```
$ sudo spctl –master-disable
```

0x01 Security Characteristics

SecureBoot

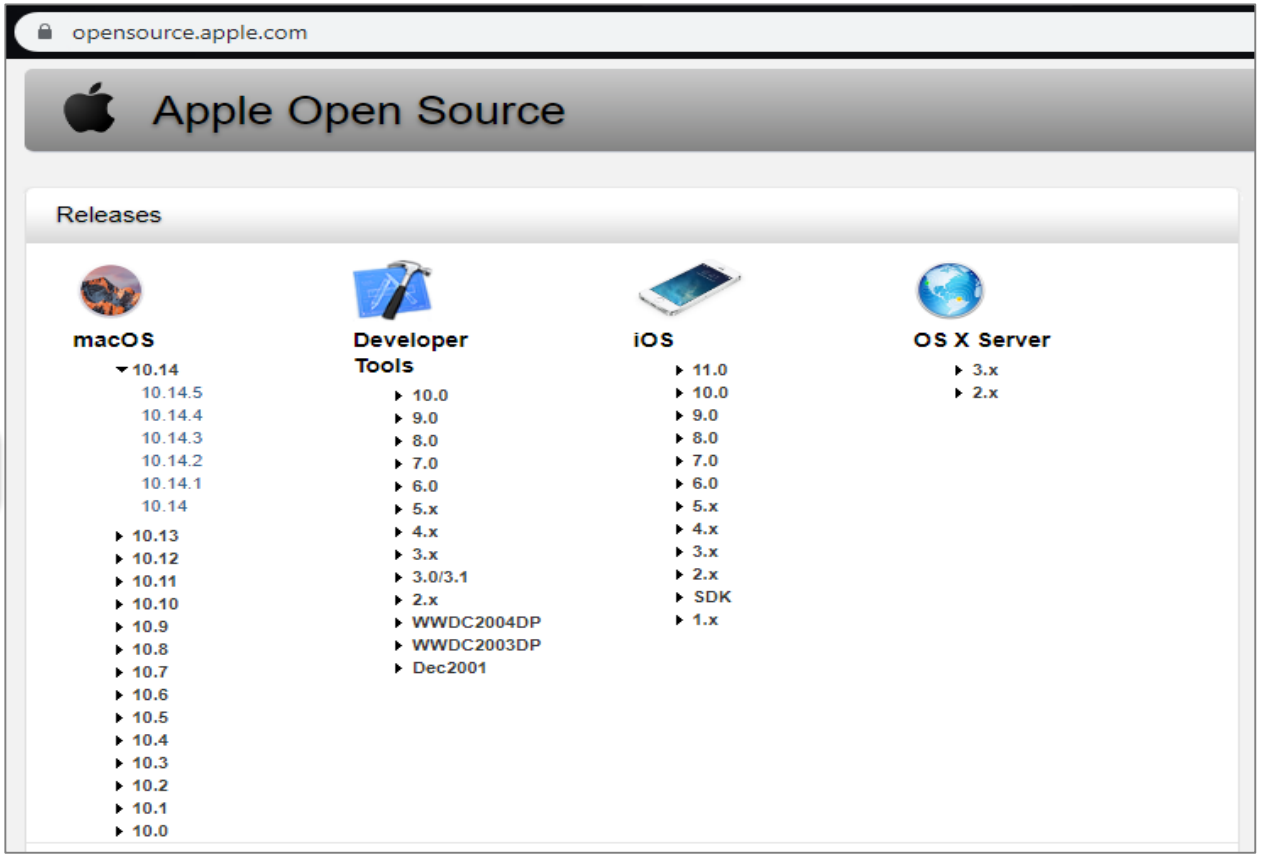




Process where the firmware validates the bootloader prior to loading. It is at the start of the chain of trust that ensures that code that gets run (drivers, kernel, applications) is known and validated

0x01 Security Characteristics

XNU



Source: https://opensource.apple.com/source/xnu/

0x01 Security Characteristics

```
l0gan@Zion-Inf3ct3d:[~][20:58:07]
└──────> $:> sudo sqlite3 /var/db/SystemPolicyConfiguration/KextPolicy
SQLite version 3.24.0 2018-06-04 14:10:15
Enter ".help" for usage hints.
sqlite> SELECT * from kext_policy;
EG7KH642X6|com.vmware.kext.vmci|1|VMware, Inc.|1
EG7KH642X6|com.vmware.kext.vmnet|1|VMware, Inc.|1
EG7KH642X6|com.vmware.kext.vmx86|1|VMware, Inc.|1
EG7KH642X6|com.vmware.kext.vmioplug.18.1.2|1|VMware, Inc.|1
2Y8XE5CQ94|com.kaspersky.kext.klif|1|Kaspersky Lab UK Limited|1
2Y8XE5CQ94|com.kaspersky.nke|1|Kaspersky Lab UK Limited|1
sqlite> _
```

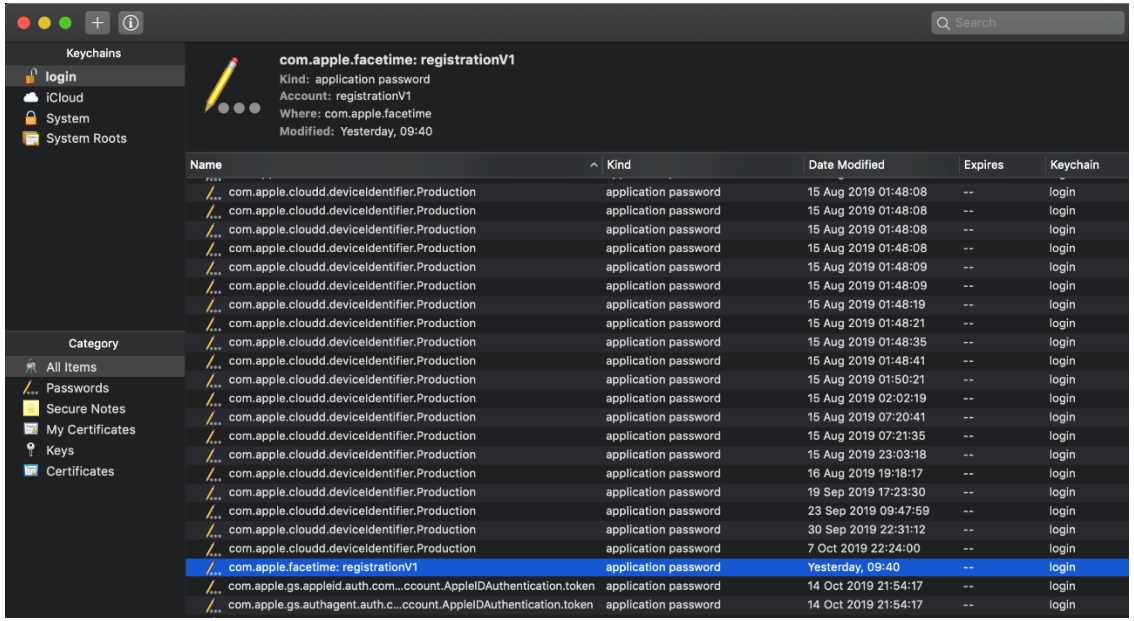
Kext files are essentially drivers for macOS. "Kext" stands for Kernel Extension; kext files "extend" Mac OS X's kernel, the core part of the operating system, by providing additional code to be loaded when your computer boots.

0x01 Security Characteristics

```
bash
l0gan@Zion-Inf3ct3d:[/System/Library/Extensions][22:59:00]
└──────> $:> pwd
/System/Library/Extensions
l0gan@Zion-Inf3ct3d:[/System/Library/Extensions][22:59:05]
└──────> $:> ls -l
total 0
drwxr-xr-x@ 3 root  wheel   96 Jun 20 22:51 ALF.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMD10000Controller.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMD7000Controller.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMD8000Controller.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMD9000Controller.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMD9500Controller.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDFramebuffer.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDMTLBronzeDriver.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonVADriver.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonVADriver2.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX4000.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX4000GLDriver.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX4000HWServices.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX5000.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX5000GLDriver.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX5000HWServices.kext
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX5000MTLDriver.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDRadeonX5000Shared.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDShared.bundle
drwxr-xr-x@ 3 root  wheel   96 Jul  3 01:51 AMDSupport.kext
drwxr-xr-x@ 3 root  wheel   96 Jun 20 22:57 Apple16X50Serial.kext
drwxr-xr-x@ 3 root  wheel   96 Jun 24 01:33 AppleACPIPlatform.kext
drwxr-xr-x@ 3 root  wheel   96 Jun 24 01:30 AppleAHCIPort.kext
drwxr-xr-x@ 3 root  wheel   96 Jun 20 22:57 AppleAPIC.kext
drwxr-xr-x@ 3 root  wheel   96 Jun 20 23:02 AppleAVEBridge.kext
drwxr-xr-x@ 3 root  wheel   96 Jul 25 23:07 AppleActuatorDriver.kext
drwxr-xr-x@ 3 root  wheel   96 Jun 24 01:37 AppleBCMWLANBusInterfacePCIe.kext
drwxr-xr-x@ 3 root  wheel   96 Jun 24 01:37 AppleBCMWLANCore.kext
```

0x01 Security Characteristics

Keychain Access



Keychain file stores secrets data like:
Safari passwords, WIFI keys, Skype username/password, Google username/password (contact, Picasa), Exchange username/password

0x01 Security Characteristics

```
bash
sh                                                    ⌘1
l0gan@Zion-Inf3ct3d:[~/Library/Keychains][22:04:29]
└──────> $:> pwd
/Users/l0gan/Library/Keychains
l0gan@Zion-Inf3ct3d:[~/Library/Keychains][22:04:31]
└──────> $:> ls -l
total 976
drwx------  10 l0gan  staff     320 Oct 17 16:10 8DEA70A8-20E3-5FB9-BBC4-132BC9525660
-rw-r--r--@  1 l0gan  staff  474568 Oct 19 19:49 login.keychain-db
-rw-------   1 l0gan  staff   23804 Oct  8 09:52 metadata.keychain-db
l0gan@Zion-Inf3ct3d:[~/Library/Keychains][22:04:34]
└──────> $:> _
```

```
l0gan@Zion-Inf3ct3d:[~][16:15:17]
└──────> $:> security list-keychains
    "/Users/l0gan/Library/Keychains/login.keychain-db"
    "/Library/Keychains/System.keychain"
l0gan@Zion-Inf3ct3d:[~][16:15:26]
```

```
l0gan@Zion-Inf3ct3d:[~/Library/Keychains][16:18:14]
└──────> $:> file login.keychain-db
login.keychain-db: Mac OS X Keychain File
```

0x01 Security Characteristics

```
bash
l0gan@Zion-Inf3ct3d:[~/Library/LaunchAgents][23:01:10]
└──────> $:> pwd
/Users/l0gan/Library/LaunchAgents
l0gan@Zion-Inf3ct3d:[
└──────> $:> ls -l
total 24
-rw-r--r--  1 l0gan
-rw-r--r--@ 1 l0gan
-rw-r--r--@ 1 l0gan
l0gan@Zion-Inf3ct3d:[
└──────> $:> _
```

```
bash
l0gan@Zion-Inf3ct3d:[/Library/LaunchAgents][23:01:47]
└──────> $:> pwd
/Library/LaunchAgents
l0gan@Zion-Inf3ct3
└──────> $:> ls -l
total 24
-rw-r--r--  1 root
-rw-r--r--  1 root
-r-xr-xr-x  1 root
l0gan@Zion-Inf3ct3
└──────> $:> _
```

```
bash
l0gan@Zion-Inf3ct3d:[/Library/LaunchDaemons][23:02:23]
└──────> $:> pwd
/Library/LaunchDaemons
l0gan@Zion-Inf3ct3d:[/Library/LaunchDaemons][23:02:24]
└──────> $:> ls -l
total 48
-rw-r--r--  1 root  wheel   632 Sep 19 14:55 com.apple.installer.osmessagetracing.plist
-rw-r--r--  1 root  wheel   584 Oct  7 22:25 com.bjango.istatmenus.daemon.plist
-rw-r--r--  1 root  wheel   557 Oct  7 22:25 com.bjango.istatmenus.fans.plist
-rw-r--r--  1 root  wheel   608 Aug 15 21:58 com.bjango.istatmenus.installerhelper.plist
-r-xr-xr-x  1 root  wheel  1080 Jun 25 14:24 com.kaspersky.kav.plist
-rw-r--r--  1 root  wheel   382 Aug 15 07:31 org.wireshark.ChmodBPF.plist
l0gan@Zion-Inf3ct3d:[/Library/LaunchDaemons][23:02:26]
└──────> $:> _
```

```
$ launchctl load arquivo.plist
```

PLIST file is a settings file, also known as a "properties file," used by macOS applications.

It contains properties and configuration settings for various programs. PLIST files are formatted in XML and based on Apple's Core Foundation DTD.

0x02 Hacking macOS target

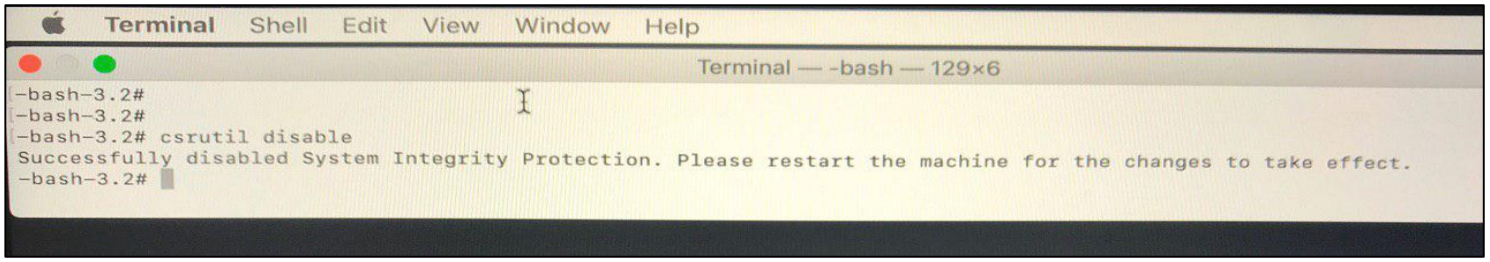
Obtain system user access

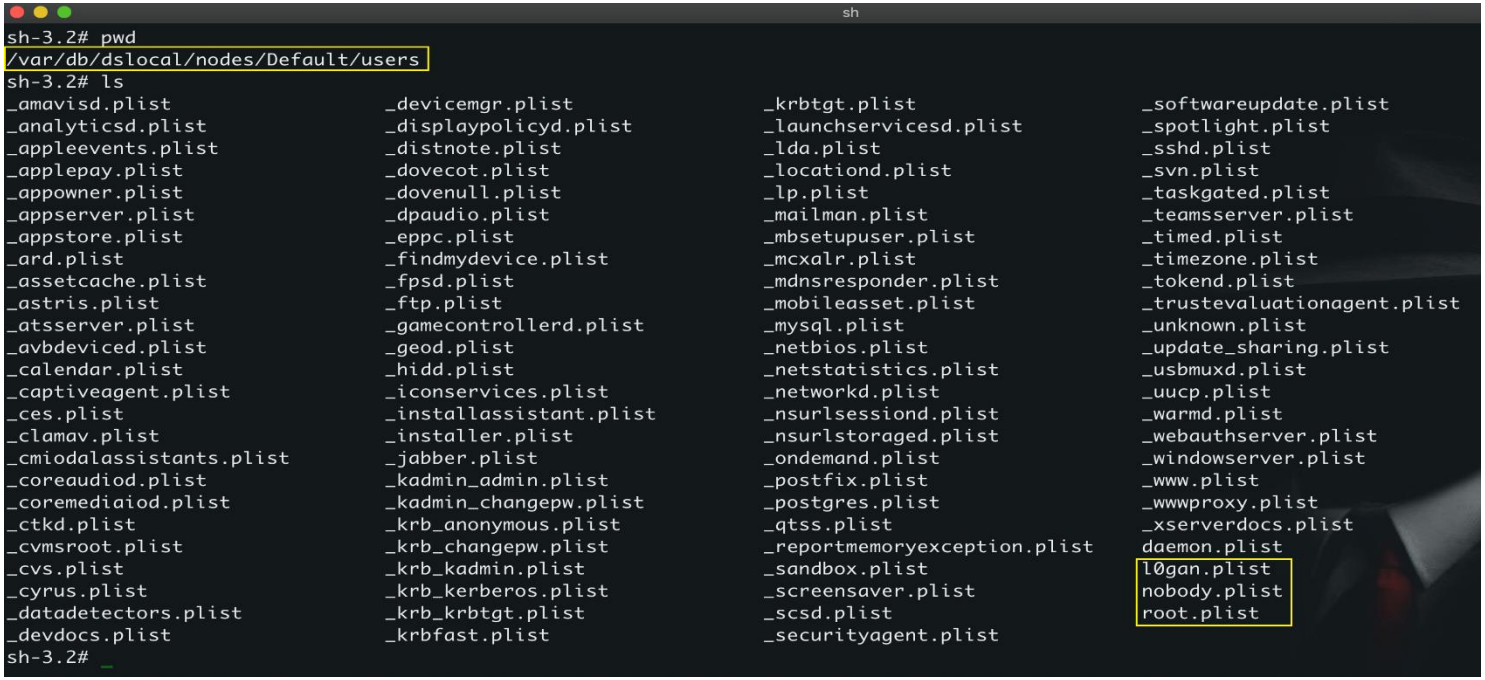
From remote access:

- By common “server side” vulnerabilities like SMB, SSH, WEB, ...

- By “client side” vulnerabilities of Safari, iTunes, iChat, Quicktime, Skype, ..

0x02 Hacking macOS target





Hashdump Python Script + Crack the Hash (Hashcat)

0x02 Hacking macOS target

Exploit-db

```
root@Zion-Hell:[/opt/tools/exploit-db][22:05:33]
└> #:> searchsploit macOS
--------------------------------------------------------------------------------------------------------------------------------------------------- -----------------------------------
 Exploit Title                                                                                                                                      |  Path
---------------------------
Apple Intel GPU Driver
Apple Safari 10.0.3 - '
Apple Safari 10.1 - Spr
Apple WebKit 10.0.2 - '
Apple iOS/macOS - Kerne
Apple iOS/macOS - Sandb
Apple iOS/macOS - Sandb
Apple macOS - '32-bit s
Apple macOS - 'getrusag
Apple macOS - 'necp_get
Apple macOS - 'stacksho
Apple macOS - Disk Arbi
Apple macOS - IOHIDSyst
Apple macOS - Kernel Co
Apple macOS - Lack of B
Apple macOS 10.12 - 'ta
Apple macOS 10.12 - Dou
Apple macOS 10.12 16A32
Apple macOS 10.12.1 / i
Apple macOS 10.12.1 / i
Apple macOS 10.12.1 / i
Apple macOS 10.12.1 / i
Apple macOS 10.12.1 / i
Apple macOS 10.12.1 Ker
Apple macOS 10.12.3 / i
Apple macOS 10.13 - 'wo
Apple macOS 10.13.1 (Hi
Apple macOS 10.13.1 (Hi
```

```
Apple macOS 10.13.1 (High Sierra) - Insecure Cron System Local Privilege Escalation                                | exploits/macos/local/43247.md
Apple macOS 10.13.2 - Double mach_port_deallocate in kextd due to Failure to Comply with MIG Ownership Rules        | exploits/macos/dos/44561.txt
Apple macOS 10.13.4 - Denial of Service (PoC)                                                                      | exploits/macos/dos/45391.py
Apple macOS < 10.12.2 / iOS < 10.2 - '_kernelrpc_mach_port_insert_right_trap' Kernel Reference Count Leak / Use-Afte | exploits/macos/local/40956.c
Apple macOS < 10.12.2 / iOS < 10.2 - Broken Kernel Mach Port Name uref Handling Privileged Port Name Replacement Pri | exploits/macos/local/40957.c
Apple macOS < 10.
Apple macOS HelpVi
Apple macOS High S
Apple macOS Kernel
Apple macOS Kernel
Apple macOS Kernel
Apple macOS Kernel
Apple macOS Kernel
Apple macOS Sierra
Apple macOS Sierra
Apple macOS Sierra
Apple macOS XNU Ke
Apple macOS/IOS 10
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS -
Apple macOS/iOS Ke
Apple macOS/iOS Ke
Apple macOS/iOS Ke
Apple macOS/iOS Ke
Apple macOS/iOS Ke
Apple macOS/iOS Ke
Apple macOS/iOS Ke
Apple macOS/iOS Ke
```

```
Arq 5.10 - Local Privilege Escalation (2)                                                                   | exploits/macos/local/43926.sh
Arq 5.9.6 - Local Privilege Escalation                                                                      | exploits/macos/local/43218.sh
Arq 5.9.7 - Local Privilege Escalation                                                                      | exploits/macos/local/43216.rb
Charles Proxy 4.2 - Local Privilege Escalation                                                              | exploits/macos/local/45107.txt
CuteFTP Mac 3.1 - Denial of Service (PoC)                                                                   | exploits/macos/dos/45823.py
FaceTime - 'VCPDecompressionDecodeFrame' Memory Corruption                                                  | exploits/macos/dos/45788.txt
FaceTime - 'readSPSandGetDecoderParams' Stack Corruption                                                    | exploits/macos/dos/45787.txt
Google Software Updater macOS - Unsafe use of Distributed Objects Privilege Escalation                      | exploits/macos/local/44307.m
Hashicorp vagrant-vmware-fusion 4.0.23 - Local Privilege Escalation                                         | exploits/macos/local/43224.sh
Hashicorp vagrant-vmware-fusion 4.0.24 - Local Privilege Escalation                                         | exploits/macos/local/43223.sh
Hashicorp vagrant-vmware-fusion 5.0.0 - Local Privilege Escalation                                          | exploits/macos/local/43222.sh
Hashicorp vagrant-vmware-fusion 5.0.1 - Local Privilege Escalation                                          | exploits/macos/local/43220.sh
Hashicorp vagrant-vmware-fusion 5.0.3 - Local Privilege Escalation                                          | exploits/macos/local/43219.sh
Hashicorp vagrant-vmware-fusion < 4.0.20 - Local Privilege Escalation                                       | exploits/macos/local/42334.txt
HideMyAss Pro VPN Client for macOS 3.x - Local Privilege Escalation                                         | exploits/macos/local/41952.txt
LiquidVPN 1.36 / 1.37 - Privilege Escalation                                                                | exploits/macos/local/45782.c
Mac OS X - libxpc MITM Privilege Escalation (Metasploit)                                                    | exploits/macos/local/45916.rb
MacOSXLabs RsyncX 2.1 - Insecure Temporary File Creation                                                    | exploits/osx/local/24609.txt
MacOSXLabs RsyncX 2.1 - Local Privilege Escalation                                                          | exploits/osx/local/24608.txt
Microsoft Remote Desktop 10.2.4(134) - Denial of Service (PoC)                                              | exploits/macos/dos/46236.py
Murus 1.4.11 - Local Privilege Escalation                                                                   | exploits/macos/local/43217.sh
Power On Software On Guard for MacOS 3.2 - Emergency Password                                               | exploits/osx/local/19439.txt
Proxifier for Mac 2.17/2.18 - Privesc Escalation                                                            | exploits/macos/local/41854.txt
Proxifier for Mac 2.18 - Multiple Vulnerabilities                                                           | exploits/macos/local/41853.txt
Proxifier for Mac 2.19 - Local Privilege Escalation                                                         | exploits/macos/local/43225.sh
Safari - Proxy Object Type Confusion (Metasploit)                                                           | exploits/macos/remote/45998.rb
Sera 1.2 - Local Privilege Escalation / Password Disclosure                                                 | exploits/macos/local/43221.sh
SwitchVPN for macOS 2.1012.03 - Privilege Escalation                                                        | exploits/macos/local/45854.txt
Xamarin Studio for Mac 6.2.1 (build 3) / 6.3 (build 863) - Local Privilege Escalation                       | exploits/macos/local/42454.txt
Yosoro 1.0.4 - Remote Code Execution                                                                        | exploits/macos/webapps/44803.txt
iOS/macOS - 'task_swap_mach_voucher()' Use-After-Free                                                       | exploits/multiple/dos/46248.c
macOS - 'process_policy' Stack Leak Through Uninitialized Field                                             | exploits/macos/dos/43521.c
macOS - 'sysctl_vfs_generic_conf' Stack Leak Through Struct Padding                                         | exploits/macos/dos/43923.c
macOS 10.13 (17A365) - Kernel Memory Disclosure due to Lack of Bounds Checking in 'AppleIntelCapriController::getDis | exploits/macos/dos/43780.c
macOS Kernel - Use-After-Free Due to Lack of Locking in 'AppleEmbeddedOSSupportHostClient::registerNotificationPort' | exploits/macos/dos/44007.c
macOS/iOS - JavaScript Injection Bug in OfficeImporter                                                      | exploits/multiple/dos/45032.txt
```

0x03 Hacking macOS target Demo

Demos 01

Service: RAE (Remote Apple Events)

Detail: AppleScript and Objects
Port TCP/UDP 3031 = eppc

0x03 Hacking macOS target Demo

Demo 02

XNU: copy-on-write behavior bypass via mount of user-owned filesystem

Autor: Jann Horn (Google Project Zero)
https://bugs.chromium.org/p/project-zero/issues/detail?id=1726&q=

CVE-2019-6208
Corrigido = macOS Mojave 10.14.3
https://support.apple.com/pt-br/HT209446

- buggycow.c
- mod.c
- pressure.c

0x03 Hacking macOS target Demo

Demo 03

“local phishing”

Detail: OSASCRIPT (OSA Open Scripting Architecture Language Script)

Reference: https://ss64.com/osx/osascript.html

Bônus:

How to Create a Fake PDF Trojan with AppleScript, Part 1 (Creating the Stager)
https://null-byte.wonderhowto.com/how-to/hacking-macos-create-fake-pdf-trojan-with-applescript-part-1-creating-stager-0184692/

How to Create a Fake PDF Trojan with AppleScript, Part 2 (Disguising the Script)
https://null-byte.wonderhowto.com/how-to/hacking-macos-create-fake-pdf-trojan-with-applescript-part-2-disguising-script-0184706/

0x04 macOS Tools

```
Lipo        -> create or operate on universal files
otool       -> object file displaying tool like a objdump and ldd
nm          -> display name list (symbol table)
codesign    -> create and manipulate code signatures
machOView   -> visual Mach-O file browser
class-dump -> utility for examining the Objective-C runtime information stored in
Mach-O files.
dtrace      -> generic front-end to the DTrace facility
fs_usage    -> report system calls and page faults related to filesystem activity in
real-time
xattr       -> display and manipulate extended attributes
Xcode       -> xcode is an (IDE) containing a suite of software development.
hopper      -> tool used for disassemble, and decompile your 32/64bits mach-o file.
lldb        -> debugger
fseventer   -> disk activity tool with a good graphical representation and solid
filter tool.
```

0x04 macOS Tools

```
launchctl-> Manage and Inspect daemons, agents and XPC Services (PLIST)
sysctl    -> get or set kernel state
nettop    -> Display updated information about the network
lsmp      -> list port used by process
ndisasm   -> The Netwide Disassembler, an 80x86 binary file disassembler
spctl     -> SecAssessment system policy security (Gatekeeper)
dscl      -> Directory Service command line utility
csrutil   -> Configure system security policies (SIP)
open snoop -> snoop file opens as they occur. Uses DTrace.
activity Monitor -> tool to help you keep your system in good shape.
procoxp     -> It's a simple tool like a top get information accessible by proc_info
lsock       -> based on PF_SYSTEM provider, you can get real time notifications of
socket activity like TCPView from SysInternals.
little Snitch -> network traffic monitoring and control.
```

0x05 Reference

Hacking is a way of life

Reference:

Kernel Architecture Overview
https://developer.apple.com/library/archive/documentation/Darwin/Conceptual/KernelProgramming/Architecture/Architecture.html#//apple_ref/doc/uid/TP30000905-CH1g-TPXREF101

Apple Developer
https://developer.apple.com/

macOS Kernel Debugging
https://blog.quarkslab.com/an-overview-of-macos-kernel-debugging.html

Building XNU for macOS
https://kernelshaman.blogspot.com/2018/01/building-xnu-for-macos-high-sierra-1013.html

0x06 Conclusion

Thanks a Lot

Any Questions ?

ricardologanbr@gmail.com @l0ganbr

https://www.slideshare.net/l0ganbr/nullbyte-6ed-2019