iOS Mobile Application Cheatsheet:

# Iphone and mac setup

* iPhone 6 — iOS 11.2.6

1. Jailbreaking
   1. 2 type: tethered vs. untethered (i)
   2. <https://canijailbreak.com/>
   3. Using Impactor to install the app
2. Installing tools
   1. Cydia
      * Network Commands
      * Arp (ii)
      * ifconfig (iii)
      * make
      * nano
      * unzip
      * OpenSSH
      * OpenSSL
      * Git
      * AppList
      * Scp
      * Ssh
      * Dpkg
      * Chmod
      * Mv
      * Cp
      * Debian Packager
      * Cydia Substrate
      * PreferenceLoader
      * Clutch 🡪 extract ipa file
      * Liberty 🡪 bypass jb detection
      * Xcon 🡪 bypass jb detection
3. Change default password for root/alpine
4. Installing SSL KILL Switch 2
   1. <https://github.com/nabla-c0d3/ssl-kill-switch2/releases>
   2. scp com.nablac0d3.sslkillswitch2\_0.12.deb [root@192.168.2.41:/var/root](mailto:root@192.168.2.41:/var/root)
   3. ssh [root@192.168.2.41](mailto:root@192.168.2.41)
   4. dpkg -i com.nablac0d3.sslkillswitch2\_0.12.deb
   5. go to cydia, install again the:
      * Debian Packager
      * Cydia Substrate
      * PreferenceLoader
5. Installing ipainstaller from cydia
   1. To list out all application
      * ipainstaller -l
   2. To extract the ip
      * ipainstaller -b <app\_bundle>
6. Installing Clutch
   1. git clone <https://github.com/KJCracks/Clutch.git>
   2. cd Clutch
   3. Disable SDK code signing requirement
      * killall Xcode
      * cp /Applications/Xcode.app/Contents/Developer/Platforms/iPhoneOS.platform/Developer/SDKs/iPhoneOS.sdk/SDKSettings.plist ~/
      * sudo /usr/libexec/PlistBuddy -c "Set :DefaultProperties:CODE\_SIGNING\_REQUIRED NO" /Applications/Xcode.app/Contents/Developer/Platforms/iPhoneOS.platform/Developer/SDKs/iPhoneOS.sdk/SDKSettings.plist
      * sudo /usr/libexec/PlistBuddy -c "Set :DefaultProperties:AD\_HOC\_CODE\_SIGNING\_ALLOWED YES" /Applications/Xcode.app/Contents/Developer/Platforms/iPhoneOS.platform/Developer/SDKs/iPhoneOS.sdk/SDKSettings.plist
   4. Compiling
      * xcodebuild clean build
   5. scp ./build/Clutch [root@<your.device.ip>:/usr/bin/Clutch](mailto:root@%3cyour.device.ip%3e:/usr/bin/Clutch)
7. Installing Frida
   1. Iphone
      * Refer here:
        1. <https://www.frida.re/docs/ios/#with-jailbreak>
        2. <https://build.frida.re/frida/ios/lib/>
8. Installing needle
   1. Kinfly refer link:
      * <https://github.com/mwrlabs/needle>
      * <https://github.com/mwrlabs/needle/wiki/Installation-Guide>
      * <https://labs.mwrinfosecurity.com/assets/BlogFiles/Needle-Finding-Issues-within-iOS-Applications.pdf>
      * <https://medium.com/@drag0n/needle-analysis-of-ios-mobile-applications-cfd9e407c0d9>
9. Installing inalyzer
   1. Kindly refer link:
      * <https://resources.infosecinstitute.com/part-15-static-analysis-of-ios-apps-using-inalyzer/#gref>
      * <https://appsec-labs.com/iNalyzer/>
      * <https://www.youtube.com/watch?v=zbeFlkvCxIA>
      * <https://appsec-labs.com/wp-content/uploads/2014/10/Attacking-iPhone-Applications.pdf>
10. Installing iRET(ios 6&7 only)
    1. ssh root@ip-device
    2. wget –no-check-certificate <https://www.veracode.com/sites/default/files/Resources/Tools/iRETTool.zip>
    3. unzip iRETTool.zip
    4. cd iRET-Tool/
    5. ls
    6. dpkg -i iRET.deb
    7. Reboot the device then check for iRET icon
    8. More information can refer <https://resources.infosecinstitute.com/ios-application-security-part-32-automating-tasks-ios-reverse-engineering-toolkit-iret/#gref>
    9. https://www.veracode.com/blog/2014/03/introducing-the-ios-reverse-engineering-toolkit
11. Installing Objection
    1. Kindly refer:
       * https://github.com/sensepost/objection
       * https://www.slideshare.net/wojdwo/testing-ios-apps-without-jailbreak-in-2018

* MacBook Pro — 10.13.6

1. iExplorer
   1. <https://macroplant.com/downloads>
2. Cydia Impactor
   1. <http://www.cydiaimpactor.com/>
3. Burpsuite
4. Hopper
   1. <https://www.hopperapp.com/>
5. IDA
   1. <https://www.hex-rays.com/products/ida/>

# Reverse Engineering and Static Analysis:

* MobSF
* Unzip <file.ipa>
* Read Info.plist using plist reader
* Check property.plist and other info.plist files inside the other folders

# Dynamic Analysis:

* Installing Frida
* Installing burp.cer

Reverse

* Kindly see the picture.

A screenshot of a cell phone

Description automatically generated

## References:

1. <https://www.iphonefaq.org/archives/971144>
2. <https://www.computerhope.com/unix/arp.htm>
3. <https://www.computerhope.com/unix/uifconfi.htm>
4. <https://medium.com/@ehsahil/basic-ios-apps-security-testing-lab-1-2bf37c2a7d15>
5. <https://resources.infosecinstitute.com/top-6-iphone-hacking-tools-mobile-penetration-testers/#gref>
6. <https://github.com/ivRodriguezCA/RE-iOS-Apps>
7. <https://github.com/kif-framework/KIF>
8. <https://techblog.mediaservice.net/2018/04/brida-a-step-by-step-user-guide/>
9. <https://cydia.angelxwind.net/?page/net.angelxwind.appsyncunified>
10. <https://github.com/BishopFox/bfinject>
11. <http://www.3u.com/news/articles/6030/liberty-lite-hide-jailbreak-status-on-ios-11-with-electra-jailbreak>
12. <https://resources.infosecinstitute.com/ios-application-security-part-44-bypassing-jailbreak-detection-using-xcon/#gref>
13. <https://techblog.mediaservice.net/2018/04/brida-a-step-by-step-user-guide/>
14. <https://github.com/kif-framework/KIF>
15. <https://techblog.mediaservice.net/2017/07/brida-advanced-mobile-application-penetration-testing-with-frida/>
16. <https://techblog.mediaservice.net/2017/07/universal-android-ssl-pinning-bypass-with-frida/>
17. <https://techblog.mediaservice.net/2017/09/tracing-arbitrary-methods-and-function-calls-on-android-and-ios/>
18. <http://www.iphonehacks.com/2018/02/how-to-restore-unsigned-ios-firmwares.html>

## Secure Programming reference:

1. <https://github.com/OWASP/owasp-mstg/blob/1.1.0/Document/0x06j-Testing-Resiliency-Against-Reverse-Engineering.md#device-binding>