

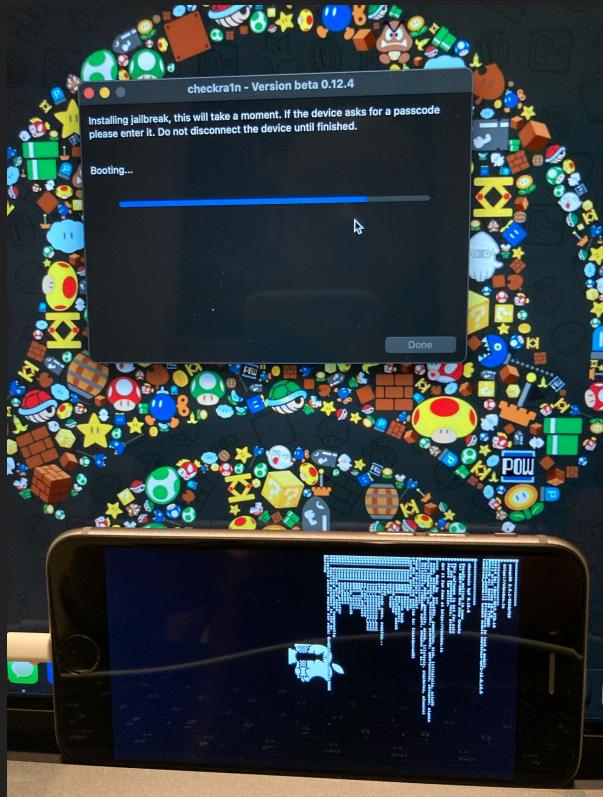
demo: iPhone bridging

demo: iPhone bridging – iPhone prep



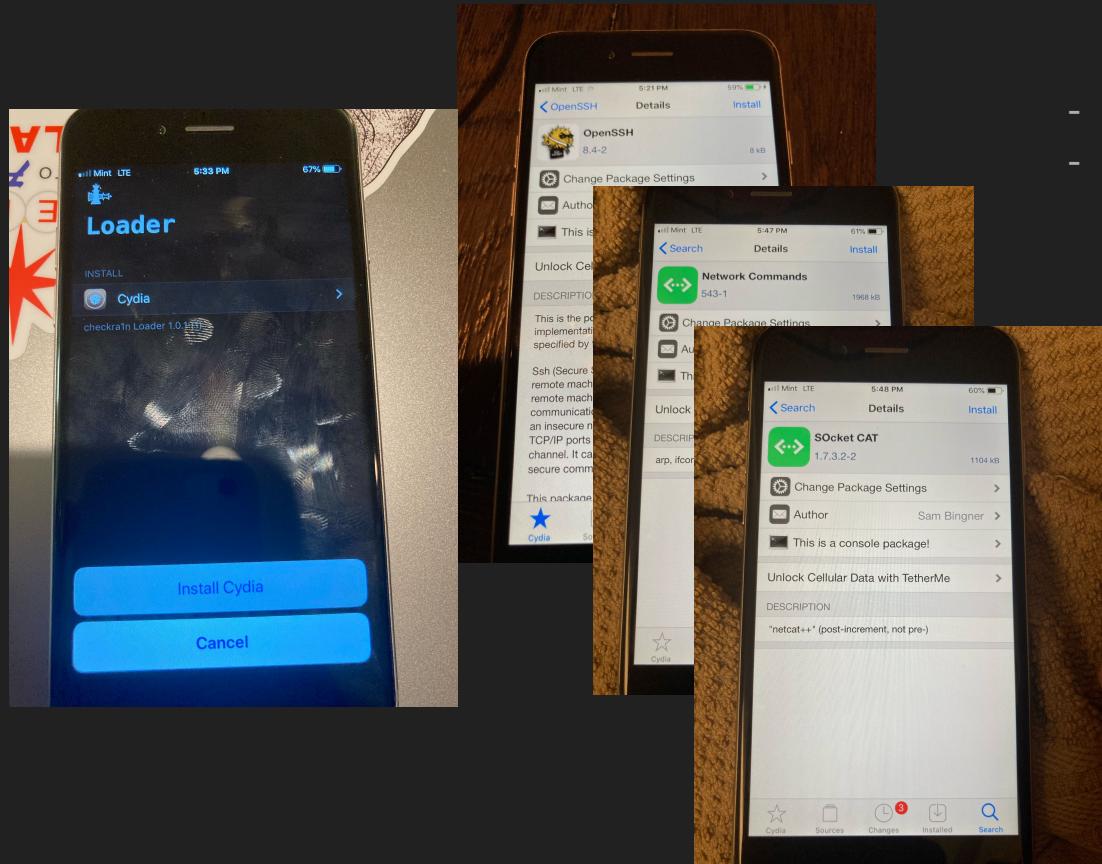
- Install WireGuard via app store

demo: iPhone bridging – iPhone prep



- Jailbreak (checkra1n pictured)

demo: iPhone bridging – iPhone prep



- Install Cydia via checkra1n app
- Install OpenSSH, network commands, socat and wifiutils

demo: iPhone bridging – iPhone prep

```
iPhone:- root# apt update
Ign:1 http://cydia.zodttd.com/repo/cydia stable
Ign:2 http://apt.modmyi.com/stable InRelease
Ign:3 http://apt.thebigboss.org/repofiles/cydia
Hit:4 http://cydia.zodttd.com/repo/cydia stable
Ign:5 https://repo.chariz.com/ InRelease
Hit:6 http://apt.modmyi.com/stable Release
Ign:7 https://repo.dynastic.co/ InRelease
Get:8 http://apt.thebigboss.org/repofiles/cydia
Get:9 http://cydia.zodttd.com/repo/cydia stable
Get:10 http://apt.modmyi.com/stable Release.gpg
Get:11 https://apt.bingner.com/ InRelease [63B]
Get:12 https://repo.chariz.com/ Release [115B]
Get:13 https://repo.dynastic.co/ Release [204B]
Get:14 https://apt.thebigboss.org/repofiles/cydia
Get:15 https://repo.chariz.com/ Release.gpg [1B]
Ign:16 https://repo.dynastic.co/ Release.gpg
Get:17 http://cydia.zodttd.com/repo/cydia stable
Ign:18 https://repo.dynastic.co/ Packages [28.3B]
Get:18 https://repo.dynastic.co/ Packages [28.3B]
Get:19 https://apt.bingner.com/ Packages [65.3B]
Get:20 http://apt.thebigboss.org/repofiles/cydia
Get:21 https://repo.chariz.com/ Packages [38.9B]
Fetched 2735 kB in 12s (221 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  org.thebigboss.wifiutil
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded
Need to get 21.0 kB of archives.
After this operation, 180 kB of additional disk space will be
Get:1 http://apt.thebigboss.org/repofiles/cydia stable/main iphoneos-arm 0.0.1-98 [21.0 kB]
Fetched 21.0 kB in 1s (27.7 kB/s)
Selecting previously unselected package org.thebigboss.wifiutil.
(Reading database ... 1775 files and directories currently installed)
Preparing to unpack .../org.thebigboss.wifiutil_0.0.1-98.ipa...
Unpacking org.thebigboss.wifiutil (0.0.1-98) ...
Setting up org.thebigboss.wifiutil (0.0.1-98) ...
iPhone:- root#
```

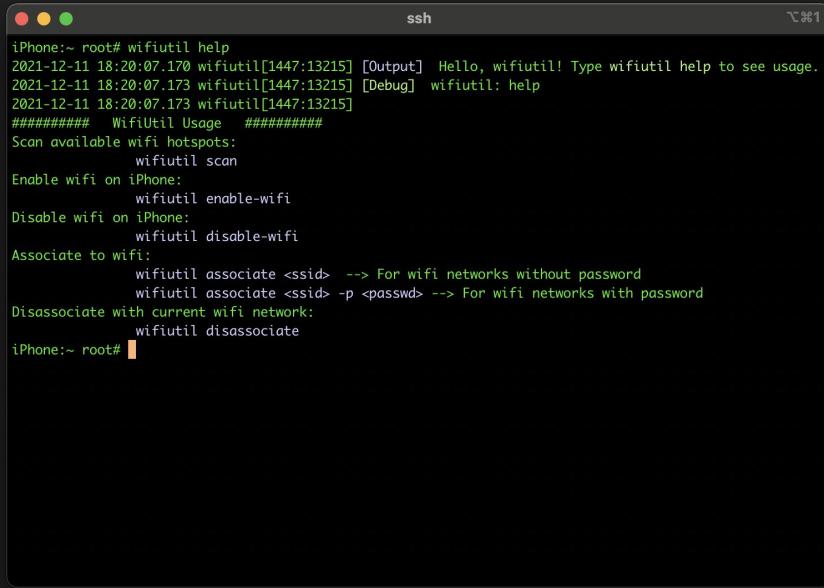
- Alternatively, you can install via SSH command line
- Pictured: apt update && apt install org.thebigboss.wifiutil

demo: iPhone bridging – Remote pentester

```
j-mpb:~ jesse$ ssh -l root 15.0.0.6
The authenticity of host '15.0.0.6 (15.0.0.6)' can't be established.
RSA key fingerprint is SHA256:z62d3w0xA6+a8aSsuheRC+MP2v8Q50+RK5gd6TVTKHQ.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '15.0.0.6' (RSA) to the list of known hosts.
root@15.0.0.6's password:
iPhone:~ root#
```

- You've now turned your phone into a pentesting tool.
- SSH into the iPhone

demo: iPhone bridging – Remote pentester

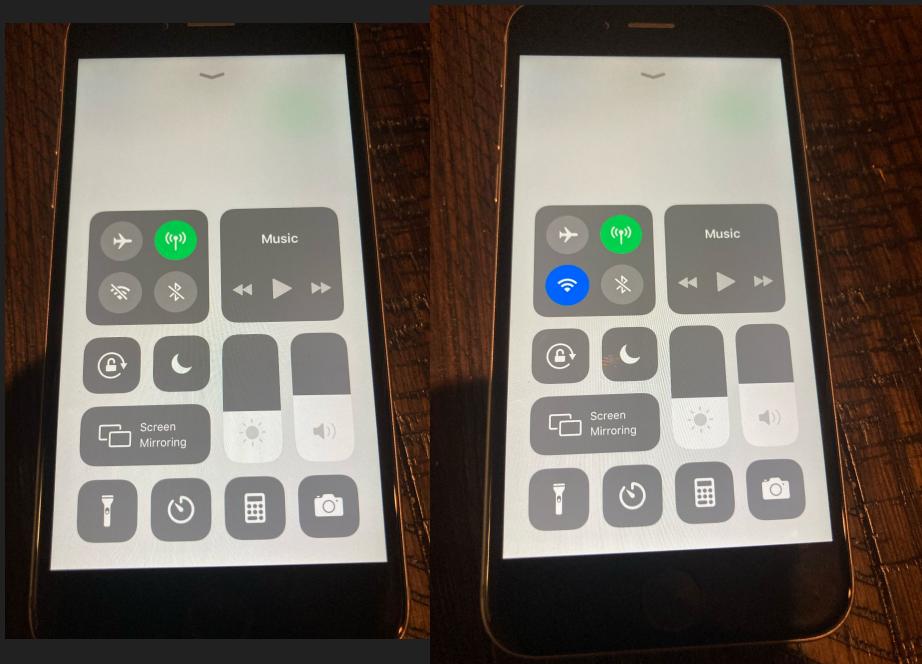


The screenshot shows an iPhone running iOS 14.7.1 in ssh mode. A terminal window titled "ssh" displays the output of the "wifiutil help" command. The output provides usage information for various wifiutil subcommands, including "scan", "enable-wifi", "disable-wifi", "associate", and "disassociate". The terminal window has a dark background with white text.

```
iPhone:~ root# wifiutil help
2021-12-11 18:20:07.170 wifiutil[1447:13215] [Output] Hello, wifiutil! Type wifiutil help to see usage.
2021-12-11 18:20:07.173 wifiutil[1447:13215] [Debug] wifiutil: help
2021-12-11 18:20:07.173 wifiutil[1447:13215]
#####
WifiUtil Usage #####
Scan available wifi hotspots:
    wifiutil scan
Enable wifi on iPhone:
    wifiutil enable-wifi
Disable wifi on iPhone:
    wifiutil disable-wifi
Associate to wifi:
    wifiutil associate <ssid> --> For wifi networks without password
    wifiutil associate <ssid> -p <passwd> --> For wifi networks with password
Disassociate with current wifi network:
    wifiutil disassociate
iPhone:~ root#
```

- Begin running wifiutil
- wifiutil scan – helps you determine when you're in proximity of target AP

demo: iPhone bridging – Remote pentester



- wifiutil handles toggling the WiFi on and off on the iPhone
- At no point does the physical pentester/tiger team need to remove the iPhone from their pocket

demo: iPhone bridging – Remote pentester

```
iPhone:~ root# wifiutil associate MIFI -p "████████"
2021-12-11 18:58:41.266 wifiutil[1514:17567] [Output] Hello, wifiutil! Type wifiutil help to see usage.
2021-12-11 18:58:41.269 wifiutil[1514:17567] [Debug] wifiutil: associate
2021-12-11 18:58:41.269 wifiutil[1514:17567] [Debug] Prepare to associate with network MIFI, passwd: ██████████
2021-12-11 18:58:44.187 wifiutil[1514:17567] [Output] Finished scanning! 47 networks:
    Fios-5VX8x | 3c:bd:c5:████████ | channel 6
    Fios-N23F1 | 20:c0:47:████████ | channel 11
    | fa:cc:c0:████████ | channel 36
```

- When you identify your target AP, run wifiutil associate command
- This is assuming you have captured password in a previous phase of your pentesting

```
2021-12-11 18:58:44.188 wifiutil[1514:17567] [Output] Scanning is successful :)
2021-12-11 18:58:44.189 wifiutil[1514:17567] [Output] Found network MIFI :)
2021-12-11 18:58:44.212 wifiutil[1514:17567] [Debug] Start associating
2021-12-11 18:58:44.476 wifiutil[1514:17567] [Output] Association is successful :)
```

demo: iPhone bridging – Remote pentester

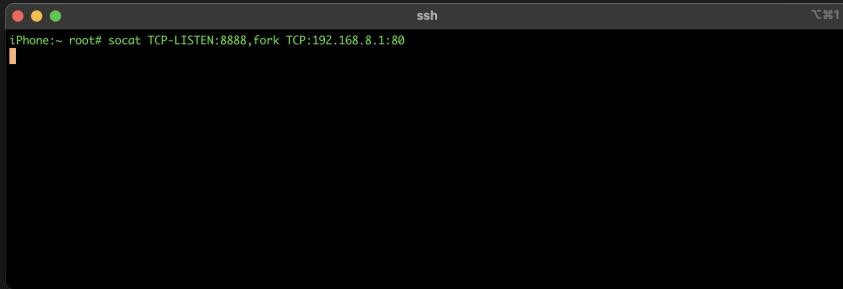
```
iPhone:~ root# ifconfig en0
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500 rref 0 index 9
  ether cc:25:ef:██████████ prefixlen 64 secured scopeid 0x9
  inet6 f██████████/64 brd f██████████ scopeid 0x9
    inet 192.168.8.232 netmask 0xffffffff broadcast 192.168.8.255
  netif: 39DB3111-9E30-4██████████
  multistack: 100EE1BD-F0E3-4██████████
  media: autoselect
  status: active
  nd6 options=201<PERFORMNUD,DAD>
  type: Wi-Fi
  agent domain:Skywalk type:FlowSwitch flags:0x83 desc:"MultiStack"
  link quality: 100 (good)
  state availability: 0 (true)
  qosmarking enabled: yes mode: none
iPhone:~ root#
```

- Discover your environment
- Identify your target IP (in this example, I am targeting the upstream router)

```
iPhone:~ root# arp -a
console.gl-inet.com (192.168.8.1) at 94:83:c4:██████████:0 ifscope [ethernet]
? (224.0.0.251) at 1:0:██████████:en0 ifscope permanent [ethernet]
iPhone:~ root#
```

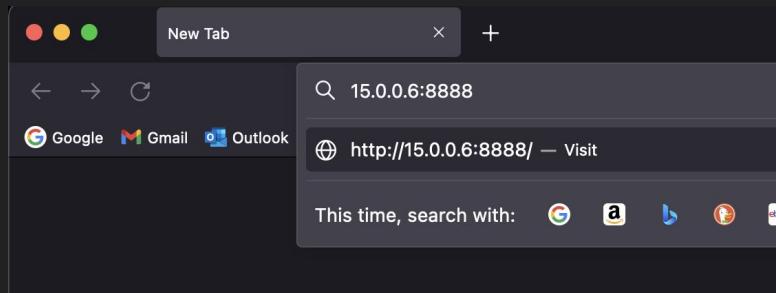
demo: iPhone bridging – Remote pentester

- Prepare your socat command

A screenshot of an iPhone's terminal application window titled "ssh". The window shows a single line of text: "iPhone:~ root# socat TCP-LISTEN:8888,fork TCP:192.168.8.1:80". The background of the slide is dark, and the terminal window has a light gray background.

- Your mileage may vary targeting different protocols and services

demo: iPhone bridging – Remote pentester



- Navigate to iPhone WireGuard IP and socat port

demo: iPhone bridging – Success!

```
jessie: ~ jesse$ curl 15.0.0.6:8888
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>GL.iNet Admin Panel</title>
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no">
    <meta name="googlebot">
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <script src="src/lib/client.js">
    <link rel="stylesheet" href="src/style.css">
<ng-app="glAdminPanel">
<ng-controller="ClientsController as clientsCtrl">
<ng-view>
<div id="app">
    <transition name="slide-left">
        <!-- cloudflare -->
        <!-- gl.i-loo -->
        <div index></div>
    </transition>
</div>
</body>

```

GL.iNet ADMIN PANEL

No Internet Connection! Find new networks to reconnect.

Brand	Name	IP	MAC	Block
2.4G Wireless Device	iPhone	192.168.8.232	CC:2E:XX:XX:XX:XX	

Copyright © 2021 GL.iNet. All Rights Reserved.



- Interact with target as if you were connected directly to the wireless LAN