
iOS/ARM Security and Exploitation

Jailbreaking

- History of Jailbreaking
- Why do people Jailbreak?
- Types of Jailbreaks
- Requirements for Jailbreaking and Development
- Famous Jailbreak Tools and the latest ones
- Package Managers
- Jailbreak Detection
- Checkm8
- checkra1n

History Of Jailbreaking

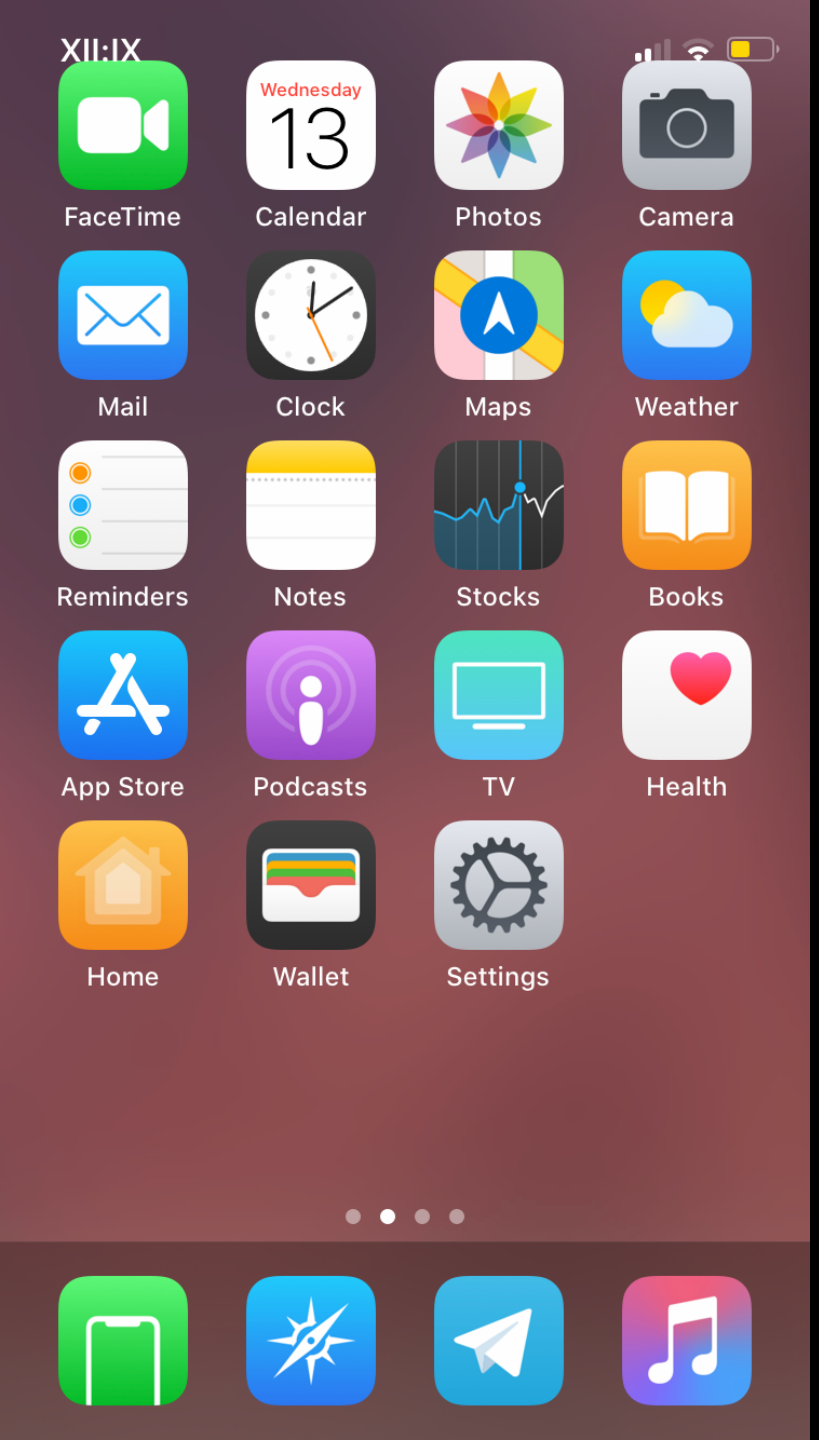
- Jailbreaking began shortly after the release of the iPhone
- Carrier compatibility on the original iPhone (2G)
- First Jailbreak by a 17 year old man named George Hotz
- iPhone Dev Team (2007) released JailbreakMe, PwnageTool (2008)
  - Jay Freeman @Saurik developed Cydia

Why Do People Jailbreak?

- The ability to fully customize your device
- To get full root access
- To develop jailbreak tweaks

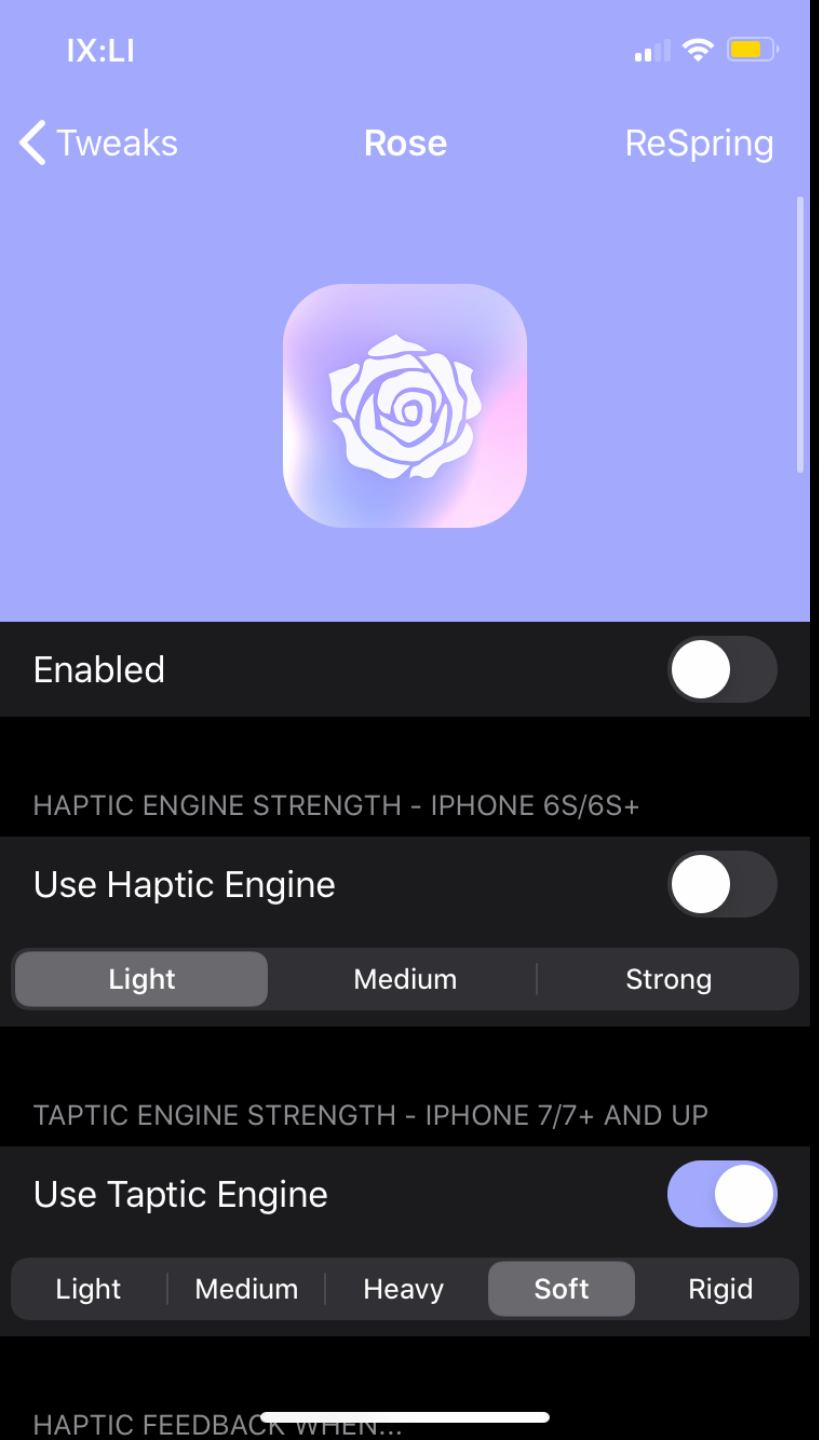
Tweaks

- Tweaks can be used to:
  - Customize/Theme your device
  - Install new features to your device
  - Install 3rd party applications
- Tweak Development:
  - Objective-C knowledge
  - Mac (only mac supports compiling for A12 devices) or a PC with theos



Hello from ShortLook!

SNAPCHAT




```
void triggerFeedback() {

    int hapticStrength = [hapticLevel intValue];
    int tapticStrength = [tapticLevel intValue];

    if (enableHapticEngineSwitch) {
        if (hapticStrength == 0) {
            AudioServicesPlaySystemSound(1519);

        }

        else if (hapticStrength == 1) {
            AudioServicesPlaySystemSound(1520);

        }

        else if (hapticStrength == 2) {
            AudioServicesPlaySystemSound(1521);

        }

    }

    if (enableTapticEngineSwitch) {
        if (tapticStrength == 0) {
            gen = [[UIImpactFeedbackGenerator alloc] initWithStyle:UIImpactFeedbackStyleLight];

        } else if (tapticStrength == 1) {
            gen = [[UIImpactFeedbackGenerator alloc] initWithStyle:UIImpactFeedbackStyleMedium];

        } else if (tapticStrength == 2) {
            gen = [[UIImpactFeedbackGenerator alloc] initWithStyle:UIImpactFeedbackStyleHeavy];

        } else if (tapticStrength == 3) {
            gen = [[UIImpactFeedbackGenerator alloc] initWithStyle:UIImpactFeedbackStyleSoft];

        } else if (tapticStrength == 4) {
            gen = [[UIImpactFeedbackGenerator alloc] initWithStyle:UIImpactFeedbackStyleRigid];

        }
```

Types of Jailbreaks

- Untethered
- Tethered
- Semi-Untethered
- Semi-Tethered

Untethered

- This Type of Jailbreak can be achieved with a BootROM exploit
- No Mac or PC needed to boot the device
- The phone will still be jailbroken after boot
- Rarest Jailbreak to be achieved

Tethered

- This Type of Jailbreak can be achieved with a BootROM exploit
- A Mac or PC is required to boot the device

Semi-Untethered

- This Type of Jailbreak needs to patch the kernel, an App is used for this
- An App is used to boot into jailbroken mode again
- Most common Jailbreak

Semi-Tethered

- This Type of Jailbreak can be achieved with a BootROM exploit
- A Mac or PC is needed to boot Jailbroken but not for normal booting

Requirements for Jailbreaking and Development

- For Jailbreaking:
  - A compatible iOS Device
  - A compatible iOS Version

For Jailbreak Development:

- Objective-C knowledge
- Knowledge about ARM/iOS Internals
- A macOS running computer (especially for >A12 support if developing tweaks)
- theos for tweak development

Famous Jailbreak tools and the latest ones

- The old OG Jailbreak tools:
  - Redsn0w – untethered for iOS 3-6
  - Pangu – untethered for iOS 7-9
  - TaiG – untethered for iOS 8

The latest Jailbreak tools:

- unc0ver for iOS 11.0-12.4 (excluding 12.3 and 12.3.1) with full fledged A12 Support
- Chimera for iOS 12.0-12.4 (excluding 12.3 and 12.3.1) and iOS 12.0-12.2 A12 Support
- checkra1n for iOS 12.3 and up for all A5 – A11 devices

Package Managers

- Cydia was the first Package Manager, released by the iPhone Dev Team in 2008
- Usually used to install tweaks
- APT – the little robot which unpacks Debian packages and puts them in the right place

Package Managers List

- The old ones:
  - Cydia
  - Icy
  - Installer 2-3

- The new modern ones:
  - Cydia
  - Sileo
  - Installer 5
  - Zebra
  - Saily



Jailbreak Detection

- Many apps have a built-in Jailbreak Detection to make the app unusable for the user
- Some apps just warn the user to not use the app while jailbroken
- Jailbreaking could lead to security leaks in the app or Cheaters in games
- Jailbreak Detections can be bypassed with tweaks or with patched/cracked versions of the app

```objc
#import "ViewController.h"

@interface ViewController ()

@end

@implementation ViewController

- (void)viewDidLoad {
    [super viewDidLoad];
    // Do any additional setup after loading the view.

    [self jailbreakDetection];

}

- (void)jailbreakDetection {

    NSFileManager *fileManager = [NSFileManager defaultManager];
    NSString *pathForCydia = @"/Applications/Cydia.app";

    if ([fileManager fileExistsAtPath:pathForCydia]) {
        exit(0);

    }

}


@end

```

Checkm8

- Checkm8 is a BootRom exploit which can’t be patched with Software updates by Apple and gives us the ability to:
    - Jailbreak A5-A11 devices for life
    - Boot CFW (custom firmware)
    - Verbose Boot
    - Dual Boot – multiple iOS’s, Linux, Android, Windows ARM

checkra1n

- Based on checkm8 exploit
- Checkra1n is the most powerful Jailbreak for all A5 – A11 devices
- Semi-Tethered