Key terms:

* Afc2 – Increases functionality of “Apple File Conduit”, Apple’s USB file transfer.
* APT – Command line package manager used primarily on Linux systems, used by Cydia to keep track of installed packages.
* Blobs – Your “ticket” from Apple verifying a restore. You can use a handful of different tools to save these and back them up. Needed to downgrade your device.
* Cydia – A GUI tool that manages packages and repositories, automates the installation process of DEB files
* Daemon – A program that runs in the background of your device. They quietly handle a variety of tasks. System Daemons include: backboardd, wifid, launchd. Third party daemons include: f.lux, OpenSSH
* DEB – A file used to deliver tweak contents and their information to the filesystem
* Downgrade – Process of restoring to an unsigned iOS version
* DPKG – Command line package installer, unlike APT, it does not handle dependencies automatically
* Dropbear – SSH server on iOS
* Dylib – A “Dynamic Library”, all tweaks are compiled as this. There are the programs the Substrate (or alternatives) load to enable tweaks.
* Exploit – The process of taking advantage of a fault in the operating system and using it to gain elevated privileges.
* Filesystem – The structure and layout of files within your iOS device.
* Impactor – A tool developed by Saurik allowing access to XCode’s app signature feature across other operating systems.
* IPA – Filetype used to deliver jailed apps to device. These apps are sandboxed and have limited permissions.
* IPSW – iPhone Software, the compressed version of what is delivered to your device during a restore. Contains full filesystem, which can be decrypted on MacOS.
* Jailbreak – Gaining root access to an iOS device, typically includes installation of Cydia and Cydia Substrate
* KPP – “Kernel Patch Protection”, software that prevents code from editing the boot process, protects the kernel
* KTRR – Hardware-based KPP
* Nonce – This is a boot variable that forces your device to accept SHSH blobs.
* NoSubstrateMode – Jailbreak without substrate enabled. Root file access, and (most) tweaks are disabled.
* OpenSSH – SSH Server on iOS
* Repository – Servers hosting packages that Cydia can call to retrieve information about packages and install them.
* Respring – Ending springboard and letting it restart. Allows newly installed tweaks to be loaded without a reboot.
* Restore – reinstalling the Operating System, removing all user files
* Root – user with full access to the iOS filesystem and services
* Safe mode – A failsafe built into Cydia Substrate to guard users if springboard crashes
* Semi-tethered – A jailbreak that boots the phone normally, sometimes breaking some system apps. Requires re-jailbreaking from a computer after every boot.
* Semi-Untethred – A jailbreak that can be enabled from an app on your device. Typically requires resigning the application, unless it exploits a vulnerability in a stock iOS app.
* SEP – “Secure Enclave Processor”, manages TouchID and FaceID. FingerPrint/FaceRecognition data is stored in the encrypted processor. A compatible version of it’s operating system must be signed for a successful downgrade.
* SFTP – “SSH File Transfer Protocol”, compatible with both OpenSSH and DropBear, allows sharing files between iOS device and other devices via SSH
* SHSH Blobs - Your “ticket” from Apple verifying a restore. You can use a handful of different tools to save these and back them up. Needed to downgrade your device.
* Source - Servers hosting packages that Cydia can call to retrieve information about packages and install them.
* SpringBoard – The base GUI of iOS. Most processes are built off this.
* SSH – Allows remote terminal usage. Control command line functions from other computers.
* Substrate – Allows code to hook into SpringBoard and other processes, the vast majority of tweaks rely on this or an alternative.
* Terminal – Allows command line
* Tethered – Jailbreak type that requires a connection to a computer to boot the device. The device will fail to boot up unless the tethered boot is run from a computer, and will boot into a jailbroken state.
* Theme – Used with the tweaks WinterBoard or Anemone, these are used to change the UI and other aspects of your device. They can be used to modify: icon images, status bar elements, UISounds, and much more.
* Theos – A tool used to develop tweaks for jailbroken iOS devices. Currently available on Windows, Linux, MacOS, and iOS.
* Tweak – code that hooks into running processes on your device and can be used to customize the look and feel of your device.
* Untethered – A jailbreak that is enabled at boot, requires a computer only to install the jailbreak, not to enable it.

List of Jailbreaks:

* 11.0-11.1.2, 64-bit: Electra, LiberiOS [Semi-Untethered]
* 10.x, 64-bit: Meridian [Semi-Untethered]
* 10.x, 32-bit: H3lix [Semi-Untethered]
* 10.3.x, 64-bit: G0blin [Semi-Untethered]
* 10-10.2, 64-bit (Excluding iP7): Yalu102 [Semi-Untethered]
* 10-10.1.1, 64-bit (Exclusively iP7): Extra\_recipe+YaluX [Semi-Untethered]
* 10.2.1, 64-bit: Saïgon [Semi-Untethered]
* 9.3.5, 32-bit: Phœnix [Semi-Untethered]
* 9.2-9.3.3, 64-bit: PanGu9 [Semi-Untethered]
* 9.1-9.3.4, 32-bit: JailbreakMe, HomeDepot [Semi-Untethered]
* 8.0-8.4, 32/64-bit: TaiG/PPJailbreak [Untethered]
* 8.0-8.1, 32/64-bit: PanGu8 [Untethered]
* 7.1.x, 32/64-bit: PanGu [Untethered]
* 7.0.x, 32,64-bit: evasi0n7 [Untethered]
* 6.1.3-6.1.6, 32-bit: P0sixspwn [Untethered]
* 6.0-6.1.2, 32-bit: Evasi0n [Untethered]
* 5.1.1, 32-bit: Absinthe 2 [Untethered]
* 5.0-5.0.1, 32-bit: Absinthe [Tethered]
* 3.0-6.1.6, 32-bit: Redsn0w [Tethered

Community contributors:

 George Hotz (GeoHot): The "original jailbreaker". Service unlocked the first iPhone. Has since moved on to developing a self-driving car toolkit.

 Jay Freeman (Saurik): Saurik is the sole developer of Cydia and Cydia Substrate. He has shaped the format of modern jailbreaks and provides tools to ideally make the jailbreak process more seamless.

 CoolStar: The developer of the Electra jailbreak, took matters into his own hands in making the iOS 11 jailbreak. He used Comex's "Substitute" (A Cydia Substrate alternative) to build his own code-injection system for iOS 11.

 PanGu Team: Developer of PanGu jailbreaks for iOS 7, 8, and 9. Also ported TaiG's Jailbreak to Mac OS X.

 Evasi0n Team: Developers of Evasi0n jailbreak for iOS 6 and Evasi0n7 For iOS 7.

ToDo:

* Tutorials
  + How to solve cydia errors
  + How to downgrade both 32-bit devices, and 64-bit devices
  + Managing installed packages and sources
  + How to jailbreak (with each jailbreaking tool)
  + Restoring while keeping your jailbreak
  + How to unjailbreak
  + SSH/SFTP basics
* More people added to Contributors section
* Add more tools to list of jailbreaks
* Add more definitions

Copy of responses to survey:

Use this to add information that you feel would be helpful

|  |  |  |
| --- | --- | --- |
| What tutorials should be added to Tips+? | What terms should be defined in the dictionary section? | Who should be included in the major contributors section? |
| Terminal / ssh commands or how to use to its full extent for jailbreak issues | Cydia, Tweaks, ssh and ssl, terminal, safe mode, non substrate mode, themes, substrate, saurik | GeoHot, Saurik , Comex, Ih8snOw, Dev-team, Chronic dev, i0n1c |
| "essentials" for the jailbreak. Not like tweaks, but how to keep it stable and be prepared for when you fuck up | Springboard, resorting, SSH, tweaks, Cydia, packages, repos, etc. | Jay, big boss owner, maybe even laughingquoll |
| How to fix common cydia errors, troubleshooting guides. | Cydia substrate, jailbreak, repo | Saurik, coolstar |
| recovery/wiping cydia and phone related stuff while jailbroken. Those are the most troublesome for new users as no one tells you not to erase all. | blobs, kpp, certs, tweaks, substrate, ipa, libs, terminal, shsh, afc2, anything people on reddit throw out there like its common knowledge but isn't to the new user | repo owners/devs |
| Data And Tools i.e: Snandbox etc | Terms related to Code Language | Jailbreak and SSH Terminal |
| Guides that involve preparing the escape plan and guides that explain how to fix major problems that come from doing things like installing incompatible tweaks and erasing all content and settings. Also how to fix major cydia errors (ie. dpkg locked) that can arise on various jailbreaks. Also the various errors and pitfalls that seem to accompany specific jailbreaks. | .deb, .ipa, .ipsw, shsh blobs, various areas of the root file system. | Basic tools and guides that everyone should be familiar with who are involved in jailbreaking. |
| Something about SSH, recent events have proved it’s a good knowledge to have. | Things like KPP, KPP-less, tethered/untethered (because every time there’s a new jailbreak everyone spams the reddit with questions), blobs (the fuck is a blob), shsh2... | Saurik, Ian Beer, all the big people who found the iOS 11 exploits... |
| Definitely jailbreak tutorials, but also simplifications on how to do things that’s might come handy to beginners but are more tools for advanced users. | Exploit, Substrate, SSH, different file extensions | I haven’t been on enough recently to help you with this |
| Installing tweaks, uninstalling tweaks, general information, removing jailbreak | Tweak, theme, substrate, ssh, filesystem, daemon, semi-tethered, tethered, untethered, KPP, dropbear, springboard. Respring, restore. | Saurik, CoolStar, pangu team, evasion team, redsnow, angelwix for appsync, Ian beer, |
| Some noob tutorials so we dont get as many posts on the r/jailbreak discord or subreddit, along with some tutorials on how to SSH or SFTP, and some tutorials on how to fix common issues when jailbreaking | Blobs, SSH, SFTP, Cydia, all the jailbreak's that are released (Electra, G0blin, ect) | Everyone who has made a jailbreak toolkit (Coolstar, CheesecakeUFO, ect) and maybe some large tweak devs (Such as LaughingQuoll) and, of course, Saurik. |
| How to ssh+ were to download anything like .deb files | No piracy | All, not just devs there’s to far of a gap between devs and joe public there should be someone here all the time to help out |
| Futurerestore, Cydia Eraser. | SHSH blobs, | Saurik, Tihmstar |
| Downgrading, upgrading, and restoring using blobs | Semi untethered, tethered, untethered, kpp, ktrr, kpp bypass, kppless | New jailbreaks/exploits, and tweaks like ext3nd3r that make jailbreaks better. |
| Blob saving, downgrading, advanced ssh usage, basic to advanced iOS security research, | Blobs, different parts of the iOS environment, package formats and directories (per iOS version, ie app folder, jb dylib location, etc) | Larger names in the community and people directly involved, and one full list of contributors somewhere else for the curious. |
| Jail breaking, ssh, theming, theme creation, sideloading apps | Tbh, all of them. | I think at this point, the only go to person would be Saurik. For everyone else, an argument can be made for or against having them in. |

Topics to consider:

These are very good, I’d like to squeeze them in.

|  |
| --- |
| Piracy, DMCA and jailbreaking regulations |
| Beginners guide, FAQ, popular repos |
| repo list and/or a tweak compatibility lists with repos in them. The hardest part about finding tweaks is first finding a working repo for it. |
| In-depth terminal commands. Guided tour of the root file system with emphasis on important areas and a tutorial on how and why it is structured the way it is. |
| A nice big guide on different iOS versions and the progress on jailbreaks for them |
| Known information on future jailbreaks and tweak developments |
| Risks?, Unjailbreak?, warranty? Troubleshooting |
| A troubleshooting section for people who have accidentally broken something, and maybe a FAQ. A section for original files that people might modify and break would be nice too. |
| Old FW jailbreaks and we’re is the trusted place to download them with out all the scammers |
| How to not make a developer leave the community. |
| History of jb’ing, including historical events (ie last untethered jb, first open source jb, etc) preferably in a timeline.  Additionally a tweak maker’s guide would be simply amazing. I have a bit of experience in programming, but can’t find a true way to adapt to the iOS environment. |
| Technical write-ups/presentations on current state of jailbreak. |