

Msft.guy@googlewave.com:

Working iPhone recovery ramdisk with SSH (Public wave)

This wave is read-only, please use the writable copy for comments:**>>> Current discussion HERE! <<<**

Discussion archive:

- [Discussion wave 1 \(slow, read-only\)](#)
- [Slow Discussion wave 2 \(WAS: Working iPhone recovery ramdisk with SSH\)](#)

Requirements: iPod or iPhone with fw 3.1.2 and intact iBoot (not a DFU-only brick), OR with saved SHSH hashes for 3.1.2.

If your iPhone does not boot and you are too lazy to reinstall everything/have some data that needs to be recovered, this may just work for you. Allows you to copy full disk images among other things.

Video (untethered JB case):

Note: video uses iRecovery utility, new instructions are different..

Update3: Experimental support for 3GS iPhones with 3.1.2 SHSH on file, even with new bootrom (tethered).**Ramdisk prep tool (currently Windows version only, needs .NET Framework 4):**http://code.google.com/p/iphonetunnel-usbmuxconnectbyport/downloads/detail?name=RecoveryRamdiskBuilder_rev_2.zip*Note that you still need a *pwned* kernelcache (from a pwnageTool generated IPSW)!Now we are going to boot the ramdisk using itunnel_mux utility (currently needs **iTunes 9.1** or newer version).http://code.google.com/p/iphonetunnel-usbmuxconnectbyport/downloads/detail?name=itunnel_mux_r61.exe*** Put your iPhone in recovery mode!**

If you have iPhone 3G or iPhone 2G, please use the iReb utility (on a computer with iTunes 9.0) to get a white screen, then connect the phone back to the computer with iTunes 9.2

Now run this command:

```
itunnel_mux_r61 --ibec iBEC_file_from_custom_FW --ramdisk created_ramdisk.dmg.ssh --devicetree DevicetreeXXX.img3 --kernelcache  
kernelcache_file_from_custom_FW --ramdisk-delay 5
```

Only if you have a 3G model iPhone, add this option (7 zeroes after '9'):

```
--ramdisk-command "ramdisk 0x90000000"
```

Only if you have an EDGE (2G) model iPhone, add this option (6 zeroes after '9'):

```
--ramdisk-command "ramdisk 0x9000000"
```

At this point, you should see an Apple logo with a progress bar on the iPhone screen.



DSC_0489

Now run this command:

```
itunnel_mux_r61 --lport 22
```

Then connect using SSH: `ssh root@localhost -p 22`

Now the iPhone display should change to this, with a rotating progress indicator:



DSC_0491

• Common errors:

iTunnel output:

```
USBmuxConnectByPort = ?, handle=ffffff
```

Error: Device Service

Cause:

You ARE NOT USING kernelcache file from an ipsw made with PwnageTool. Please read instructions carefully!



Msft.guy@googlewave.com: **Tethered support:**

Aug 5 ▼

Advanced skills and OS X recommended.

If you have iPhone 3GS with 3.1.2 SHSH on file and new bootrom:

- Replace [gs.apple.com](#) with Saurik's server or your local tinyTss (duh!)
- Start a **DFU mode** restore.
- **!IMPORTANT!** Unplug the USB right after the screen turns white. This happens after iTunes message 'preparing iPhone for restore' which loads iBSS.
- Copy your personalized iBSS file from %TEMP% - sort by modification date and search for file named 'ibss*' in the newest directory.
- Assemble the iBSS payload - here are the instructions: [iBSS payload: 3.1.2 3GS](#).
- Now add extra options to the itunnel_mux command line: `--ibss ibss_personalized_312 --exploit exploit`. 'exploit' file is the file you got from the previous step.

If you don't have SHSH for 3.1.2 saved BUT still have a working iBoot 636.66,

HERE are the payload assembly instructions: [iPhone 3GS 3.1.2 iBoot pwn payload + instructions](#)

In this case, just use additional `--exploit exploit` option on the command line of itunnel_mux tool.



Msft.guy@googlewave.com: **Useful SSH commands:**

Jul 15 ▼

```
mount /                ;# make ramdisk readwrite, I have no idea how/why it works )
fsck_hfs /dev/disk0s1   ;# check system volume
mount_hfs /dev/disk0s1 /mnt1 ;# mount the system volume
fsck_hfs /dev/disk0s2s1 ;# check user (data) volume (3GS)
fsck_hfs /dev/disk0s2   ;# check user volume (3G and 2G models)
mount_hfs /dev/disk0s2s1 /mnt2 ;# mount user volume (3GS)
mount_hfs /dev/disk0s2 /mnt2 ;# mount user volume (3G and 2G models)
```

Advanced:

```
export PATH=$PATH:/mnt1/bin:/mnt1/sbin:/mnt2/stash/bin: ;# more stuff to run
export DYLD_LIBRARY_PATH=/mnt1/usr/lib ;# to run stuff without having to copy/symlink the libs
kill 1 ;# reboot, since we replaced /sbin/reboot with sshd
```



Msft.guy@googlewave.com: **Tech details (how this works)**

May 20 ▼

restored daemon enables USB MUX kernel module to accept connections, after which we can use standard MobileDevice framework functions for port forwarding. Now we just need to start **sshd**.

By replacing `/sbin/reboot` with `sshd` and issuing a reboot command to *restored* we make *restored* launch `sshd` and hang waiting for reboot. Now we just need to make sure the restore dmg has required libraries and `/bin/sh` (this is the login shell for root user specified in passwd file). Password is *alpine*, as usual ;-)



Msft.guy@googlewave.com: **If your iBoot behaves weird (like, reboots after you issue the ramdisk command)**

May 20 ▼

Use the patched iBEC to get a better iBoot env:


Use iBEC from the **pwned** ipsw ("Firmware\dfu\iBEC.n88ap.RELEASE.dfu").

```
iRecovery -f iBEC.n88ap.RELEASE.dfu
```


```
iRecovery -c go
```


now the screen turns white, connect iRecovery and repeat the usual steps (ramdisk, kernelcache) but with iBEC..


let me know what happens.

 Msft.guy@googlewave.com: **Build an IPSW:** Jul 28 ▼
Add patched ramdisk to the pwned ipsw package and you can use iTunes to load this. Faster and less buggy than iRecovery...
Remove root fs dmg, iBoot, LLB and BuildManifest.plist from ipsw, modify manifest file in all flash.* production folder.
This is to guarantee that the restore won't modify anything.
Not sure this is the case - restore can still repartition the disk before finding out rootfs is missing. Not recommended (unless you are using upgrade ramdisk).

 Msft.guy@googlewave.com: Notes on FW 4.0: Since 4.0 now uses a journaled FS for user data, you need to create 4.0-based ramdisk to access the data on 4.0 install. May 25 ▼

 Msft.guy@googlewave.com: Non-jailbroken older models: iPhone Edge and 3G and iPod Touch 1G and 2G: Jun 30 ▼
Use [iReb utility](#) to get to a pwned iBoot environment from DFU.
<http://twitter.com/iH8sn0w/status/17459935366> - iReb apparently needs iTunes 9.0

 Msft.guy@googlewave.com: **Copy disk image** Jul 3 ▼
OS X: Use CyberDuck
Windows: Use [PsFTP](#)
to copy copy /dev/disk, where **disk** is disk0s2s1 on 3gs and 3g iPod, disk0s2 on older devices.

 Msft.guy@googlewave.com: **Advanced2: Manual ramdisk preparation steps:** Jul 14 ▼

- Jailbreak the firmware using PwnageTool, we need pwned kernelcache.
- Rename pwned ipsw to zip and unzip it.
- Unpack the pwned restore ramdisk dmg: `xpwnntool 018-6051-014.dmg 018-6051-014.unpacked.dmg -iv .. -k ..` (Google "firmware_build phone_model ramdisk" for iv/key, *firmware_build* is a string like **7D11** for 3.1.2 fw. Example: '3GS ramdisk 7D11')
- Mount the dmg. If it does not mount, you found wrong key, recheck that the key is for your device.
- Download ramdisk preparation tool package (http://www.google.com/url?sa=D&q=http%3A%2F%2Fcode.google.com%2Fp%2Fiphonetunnel-usbmuxconnectbyport%2Fdownloads%2Fdetail%3Fname%3DRecoveryRamdiskBuilder_rev_2.zip).
- Unpack the package, unpack the ssh.tar archive.
- Delete /Volumes/ramdisk/usr/local/standalone/firmware/* to free some space (baseband flash files there)
- Put the contents of the unpacked ssh.tar archive inside the mounted dmg so that reboot file at the root of the ramdisk gets replaced.
- If you end up with an 'ssh' directory at the root of the ramdisk, you did previous step incorrectly.
- Pack the dmg back: `xpwnntool 018-6051-014.unpacked.dmg 018-6051-014.dmg.ssh -t 018-6051-014.dmg -iv .. -k ..` ;Use same iv/key as for unpacking.

Prebuilt xpwnntool for OS X:



Adv-users:

Hint: you can use **otool -L** to figure out required libraries..

As a reference, my personal list of added/changed files verified to work:

```
./bin/bash
./bin/sh <-symlink to bash
./etc/ssh/moduli
./etc/ssh/ssh_config
./etc/ssh/ssh_host_dsa_key
./etc/ssh/ssh_host_dsa_key.pub
./etc/ssh/ssh_host_key
./etc/ssh/ssh_host_key.pub
./etc/ssh/ssh_host_rsa_key
./etc/ssh/ssh_host_rsa_key.pub
./etc/ssh/sshd_config
./sbin/reboot <- this is actual sshd binary
./usr/bin/scp
./usr/lib/libcrypto.0.9.8.dylib
./usr/lib/libhistory.6.0.dylib
./usr/lib/libncurses.5.dylib
./usr/lib/libreadline.6.0.dylib
./usr/lib/libexec/sftp-server
./var/root <- homedir, just in case..
```

 Msft.guy@googlewave.com: **OS X steps described in detail for 3G iPhone - thanks to straughn.chuck@** Aug 5 ▼
iPhone is 100%! Exact steps I took from Mac OS X (for a 3G jailbroken phone)

[iPhone to 3GS](#), [latest steps](#), [rooting with the 3GS](#), [rooting a 3GS with the iPhone](#)

You might want to find a PC and use the builder tool to create the ramdisk, though..

Tags: [iPhone](#) [SSH](#) [iRecovery](#) [ramdisk](#) [payload](#) [recovery](#) [3GS](#) 

Images ▾

Next unread **5**