在 iOS 设备中辨认出后门、攻击点和监视机制

JONATHAN ZDZIARSKI
JONATHAN@ZDZIARSKI.COM

@JZDZIARSKI

翻译 by TaoKY

不为翻译内容、排版负责，不代表本人支持文本内容。

PS：翻译只能保证你能差不多看懂。

谁是 NerveGas

- 在很多早期越狱中作为开发者团队成员工作，直到 iOS 4 左右。
- 是 5 本与 iOS 有关的，O'Reilly 出版的书籍作者，包括「破解及加固 iOS 应用」。
- 设计了所有现在在法律实施和商业产品中的 iOS 法医技术。
- 紧密与联邦及本地法律实施代理和美国为高级档案信息以及犯罪案件军用。
- 在全球训练法律实施在 iOS 法医和注入技能。

iOS 操作系统

- 主题：在法医、法律实施和犯罪团体中的有趣事情。
- 根据 Der Spiegel （德国杂志）的泄漏，iOS 是 NSA 用来目标收集的目标。
- 之后在更多 C&C 能力在 DROPOUTJEEP 泄漏通过**靠近访问方法**中发现更多证据。
- 攻击任何东西从国家安全案件到有些著名的人的 XX 图片。
- 存在许多法医技术来获取数据。

这次演讲是什么

- 概述许多未在文档中记录的运行在每台 iOS 设备中的高价值法医服务。
  - 它们是如何逐步形成的。
  - 它们提供怎样的信息。
- 法医人工制品的例子获取不应该不经用户许可离开设备的信息。
- 越过个人安全的监视机制（企业想要的），但制造可能的目标
- iOS 中可疑的设计纰漏，使信息收集更简单。

这次演讲不是什么

- 一次关于有趣的零日漏洞的演讲以及我们如何会在接下来几天中获得短期的快乐。
  - 在这里讨论的内容已经存在很多年时间，以及是低级操作系统零件。
  - Apple 知道这些零件，并已经因为一些未知原因清楚地更新支持它们。
  - 我已给 Tim Cook 和 Steve Jobs 发过多次邮件来问他们提供这些服务的解释，援引它们作为「后门」，并且没有收到回复。
  - 我*已经*收到来自 Tim Cook 的回复关于 Apple 糟糕的保修回复，所以我知道他得到了我的邮箱地址。

集中控制

- Apple 努力工作使 iOS 设备明智地保持安全对抗典型攻击者。
- Apple 努力工作来保证其能在最终用户设备上访问数据，为了法律实施。
- 对他们的赞扬，iPhone 5* + iOS 7 更加对任何人安全，除了 Apple 和政府。
- Apple 的法律实施过程手册：
  - https://www.apple.com/legal/more-resources/lawenforcement/

法律实施过程

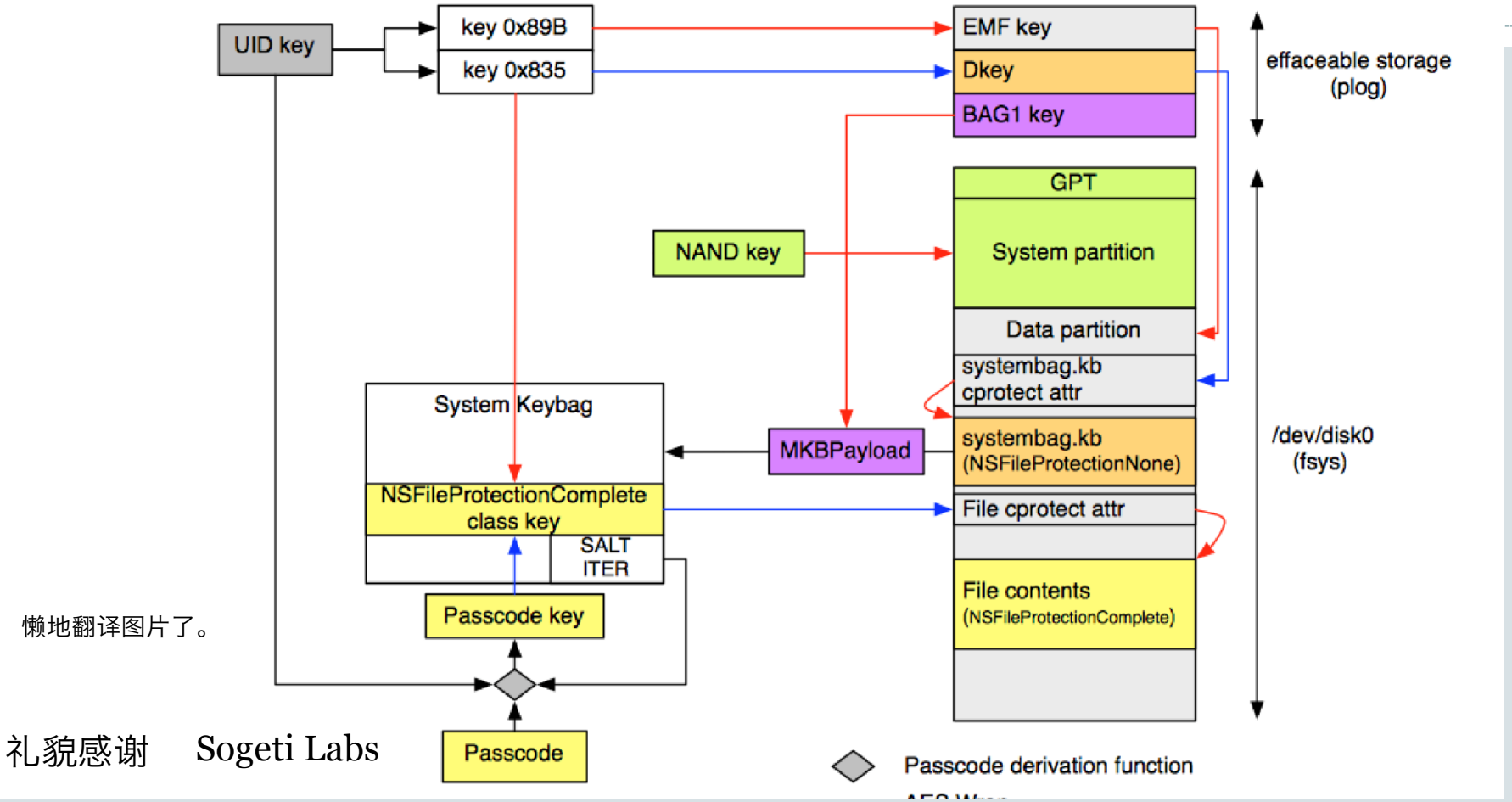
- 需要一个对 iCloud、iTunes 或是设备自身实际内容的担保。
- 一张传票对「元数据」来讲再好不过。
- 最近的改变将会提示所有消费者，除非包括机密性要求；所以大多代理现在为机密性要求担保。
- 当提供物理设备时，Apple 将会取回，并且从加密设备中返回 NSProtectionNone 数据；关于 PIN 麻烦的力量的谣言正广为流传，但我被告知这种练习在 iOS 5 左右停止了。
- 以一般水平看，现在的过程需要 4 个月， 约花费 1000 美元，所以 LE 正在寻找高效率／低花费工具以收集证据。

Apple 法律实施过程

从加密 iOS 设备中提取信息

在收到有效担保后，**Apple 可以从加密 iOS 设备中收集特定分类的活跃信息**。特别地，在包含 Apple 原生应用的 iOS 设备的用户创建的活跃文件的**数据不会用密码加密** 。这些文件可以被提取并用外在媒介提供给法律实施。Apple 可以进行数据提取实施在运行 iOS 4 以及更新版本 iOS 的 iOS 设备上。请注意只有分类为用户创建的活跃文件才能被提供给法律实施，根据一个有效的搜索担保，文件包括：**信息、照片、视频、通讯录、语音备忘录和通话记录**。Apple 不能提供：邮件、日历或是任何第三方应用程序的信息。

iOS 4 存储加密概述



懒地翻译图片了。

礼貌感谢　Sogeti Labs

iOS 7 中的加密：没有太大改变

- 几乎所有原生应用／操作系统数据被一个**与密码无关**的密钥加密，但是密钥是**硬件推断出来的（NSProtectionNone）**。
- 作为 iOS 7 的一部分，第三方文档被加密，**但是资源库和缓存文件夹没有加密。**
- 一旦设备在重启后**第一次解锁**，大多被保护的加密数据可以被访问，直到设备关机。
  - 锁屏 != 加密
- 未在文档中记录的、在每台 iOS 设备上运行的服务使这些成为可能。
- 你的设备大多一直处在溢出**所有**数据的风险之中，由于这大多一直被证明是真的，即使锁屏之时。

法律实施技术

- 最新商业法医工具利用这些服务进行深度提取。
- 此领域中的平板法医工具可以获得一个设备当常规交通停止时，或是被逮捕时——在设备可以被关机之前（使编码解锁）。
- 联邦代理一直对黑包技术感兴趣（妥协扩充口、闹钟之类）。
- 斯诺登文件：计算机渗透被使用。

未在文档中记录的服务

- 通过 *lockdownd* 访问，要求配对验证。（解释配对）
- MACTANS 演讲证明了用 Juice 劫持来建立配对是多么简单。
  - iOS 7 信任对话框有帮助，但是第三方附件使人们再次犯傻……而且人们也自然地犯傻。
- 法律实施代理在领域中转向平板设备来配对和获得；USB 拇指硬盘来扫描电脑获得配对记录。
- Der Spiegel 概述黑包技术来访问拥有配对记录的目标计算机。

Der Spiegel

- 「这些文档说明 NSA 有可能窃听大部分在这些智能手机上的敏感信息，包括**通话记录、SMS 通信、备忘录**和**定位信息**关于用户是谁。在内部文档中，专家炫耀成功读取 iPhone 信息的情况当 NSA 有能力**渗透一个人用的计算机来同步他们的 iPhone**。迷你程序，就是所谓的『脚本』，然后启用额外访问对至少 38 中 iPhone **特性**。」

未在文档中记录的服务

- 越过给用户的「备份加密」机制。
- 可以通过 USB 和无线（Wi-Fi，也可能是蜂窝移动网络）；网络可以被扫描得到一个具体目标。
- 如果设备在最后用户输入 PIN 之后还未重启，可以访问所有用**数据保护**加密的数据（第三方应用之类）。
- 其它（更加合法）的服务启用软件安装、APN 安装（添加代理服务器）来继续监控。

未在文档中记录的服务

- 大多服务没有被任何已知 Apple 软件提及（我们看着呢）。
- 数据原始格式使它不可能被放回手机，使它对天才吧或是运营商的技术目的（cpio.gz 之类）无用。
- 数据的个人自然使它非常不象调试机制。
- 越过备份加密是欺骗的。
- 可用的服务**不需要**开发者模式，排除它们作为开发者工具的目的。

DROPOUTJEEP

- DROPOUTJEEP 描述技术，大多可能与 Apple 的未在文档中记录的服务有关。
- 短信表明越狱或是基带代码。

DROPOUTJEEP

(TS//SI//REL) DROPOUTJEEP 是一个 STRAITBIZARRE 基于为 Apple iPhone 操作系统的软件移植，并使用 CHIMNEYPOOL 框架。

(TS//SI//REL) DROPOUTJEEP 是一个为 Apple iPhone，利用组合任务软件来提供实用详细的信号情报的软件移植。实用的东西包括**远程从设备上推送／拉取文件、信息检索、通讯录检索、语音信箱、地理位置、**最新录音、相机捕获、**基站定位**之类。命令、控制和数据渗出可以通过短信或者 **GPRS 数据通信**。所有移植通信将会被转换、加密。

(TS//SI//REL) 一开始释放的 DROPOUTJEEP 将会集中在通过靠近访问方法安装移植。一个远程安装能力将会追查将来的释放。

启动服务

- 连接 lockdownd (tcp:62078) 通过 usbmux 或 TCP。
- 证明拦截／生成配对记录。
- 呼叫「StartService」命令，伴随想要启动的服务名
- 受益*

- * 许多商业法律实施法医生产商已经启动窃听以下服务:
  - Cellebrite
  - AccessData (Mobile Phone Examiner)
  - Elcomsoft

开源!

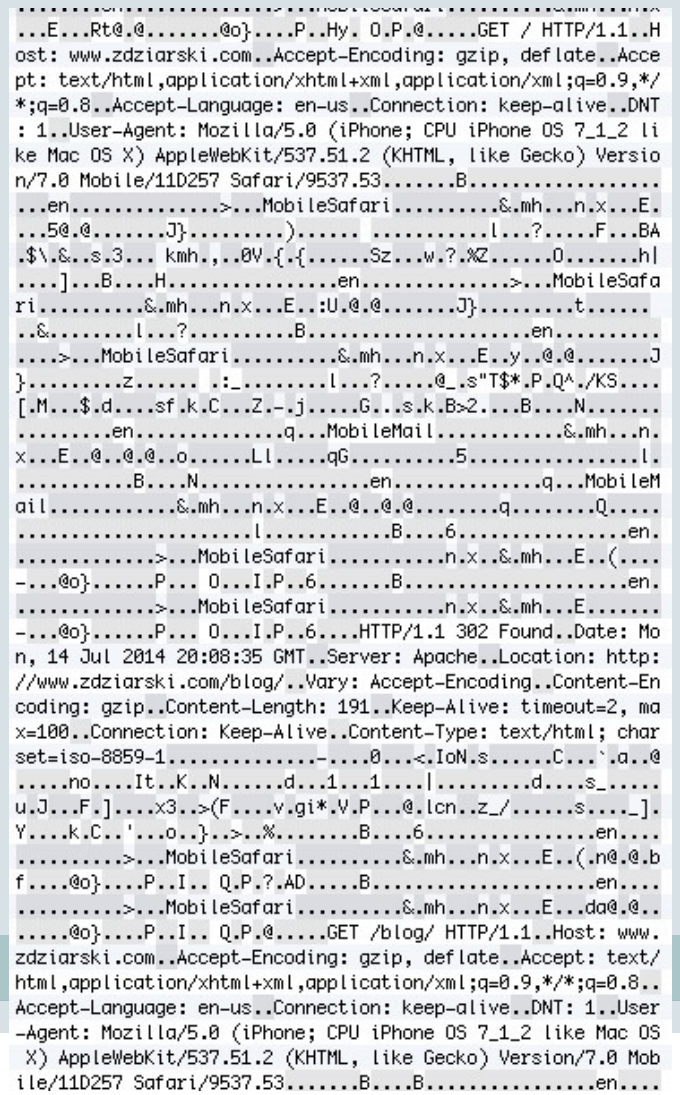
- 几乎所有 *lockdownd* 协议已经在 libimobiledevice 计划(libimobiledevice.org)中被写入文档.
- 自从 2009 年起就有了，但是从那时起，许多服务还未被重新检查；起初有益。
- 许多私人工具和代码都超出了这些服务的益处。

com.apple.pcapd

- 立即在设备上启动 libpcap。
- 转存网络交通和进出设备的 HTTP 请求／答复数据
- 不依赖 开发者模式，在每个 iOS 设备商可用。
- 可以通过 Wi-Fi 作为目标来远程监控。
- 对用户而言，没有视觉迹象表明数据包嗅探器正在运行。

为什么我们需要一个在 6oo 百万个人 iOS 设备上运行着的数据包嗅探器?

com.apple.pcapd



```
...E...Rt@.@.......@o}....P..Hy. O.P.@.....GET / HTTP/1.1..H
ost: www.zdziarski.com..Accept-Encoding: gzip, deflate..Acce
pt: text/html,application/xhtml+xml,application/xml;q=0.9,*/
*;q=0.8..Accept-Language: en-us..Connection: keep-alive..DNT
: 1..User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 7_1_2 li
ke Mac OS X) AppleWebKit/537.51.2 (KHTML, like Gecko) Versio
n/7.0 Mobile/11D257 Safari/9537.53.......B.................
...en.............>...MobileSafari.........&.mh...n.x...E.
...5@.@.......J}.........)...... ..........l...?.....F...BA
.$\.&..s.3... kmh.,..0V.{.{......Sz...w.?.%Z......O.......h|
....]...B....H..............en.............>...MobileSafa
ri.........&.mh...n.x...E..:U.@.@.......J}.........t......
..&........l...?..........B.....................en.........
....>...MobileSafari.........&.mh...n.x...E..y..@.@.......J
}.........z...... .:_........l...?.....@_.s"T$*.P.Q^./KS....
[.M...$.d....sf.k.C...Z.-.j.....G...s.k.B>2....B....N.......
.........en.............q...MobileMail...........&.mh...n.
x...E..@..@.@..o......Ll.....qG.........5.................l.
...........B....N..............en.............q...MobileM
ail...........&.mh...n.x...E..@..@.@........q........Q.....
.....................l...........B....6..............en.
.............>...MobileSafari..........n.x..&.mh...E..(....
-...@o}......P... O...I.P..6.......B.....................en.
.............>...MobileSafari..........n.x..&.mh...E.......
-...@o}......P... O...I.P..6....HTTP/1.1 302 Found..Date: Mo
n, 14 Jul 2014 20:08:35 GMT..Server: Apache..Location: http:
//www.zdziarski.com/blog/..Vary: Accept-Encoding..Content-En
coding: gzip..Content-Length: 191..Keep-Alive: timeout=2, ma
x=100..Connection: Keep-Alive..Content-Type: text/html; char
set=iso-8859-1.............-....0...<.IoN.s......C...`.a..@
.....no....It..K..N......d...1...1....|.........d....s_.....
u.J...F.]....x3..>(F....v.gi*.V.P...@.lcn..z_/.......s...._].
Y....k.C..'...o..}..>..%........B....6..............en....
..........>...MobileSafari.........&.mh...n.x...E..(.n@.@.b
f....@o}....P..I.. Q.P.?.AD.....B.....................en....
..........>...MobileSafari.........&.mh...n.x...E...da@.@..
.....@o}....P..I.. Q.P.@.....GET /blog/ HTTP/1.1..Host: www.
zdziarski.com..Accept-Encoding: gzip, deflate..Accept: text/
html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8..
Accept-Language: en-us..Connection: keep-alive..DNT: 1..User
-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 7_1_2 like Mac OS
 X) AppleWebKit/537.51.2 (KHTML, like Gecko) Version/7.0 Mob
ile/11D257 Safari/9537.53.......B....B..............en....
```

来自 iOS 7.1.2 的例子。

没有启动开发者模式。

数据包嗅探器在 600 百万 iOS 设备上可用。☺☺

com.apple.mobile.file_relay

- 设备上最大的法医情报的财宝。
- 在设备上被发现位于 /usr/libexec/mobile_file_relay。
- 提供*物理*显证 vs. *逻辑* （数据库；删除的记录可被恢复。）
- 传播大长条原始数据信息在一个压缩后的 cpio 压缩包中，基于数据源码要求。
- 彻底**越过 Apple** 为最终用户安全的**备份加密**。
- 曾被认为有益，已逐步发展出相当多的，甚至在 iOS 7 中，来揭露很多个人数据。
- 非常有意地放置和计划来从设备中根据要求转存数据

com.apple.mobile.file_relay

- 重新铺设的源码在 iOS v2:

  AppleSupport
  Network
  WiFi
  UserDatabases
  CrashReporter
  SystemConfiguration

com.apple.mobile.file_relay

- 重新铺设的源码在 iOS 7（下页）：

com.apple.mobile.file_relay

Accounts
AddressBook
AppleSupport
AppleTV
Baseband
Bluetooth
CrashReporter
CLTM
Caches
CoreLocation
DataAccess
DataMigrator
demod
Device-o-Matic
EmbeddedSocial
FindMyiPhone
GameKitLogs
itunesstored
IORegUSBDevice
HFSMeta
Keyboard
Lockdown
MapsLogs
MobileAsset
MobileBackup
MobileCal
MobileDelete
MobileInstallation
MobileMusicPlayer
MobileNotes
NANDDebugInfo
Network
Photos
SafeHarbor
SystemConfiguration
tmp
Ubiquity
UserDatabases
VARFS
VPN
Voicemail
WiFi
WirelessAutomation

com.apple.mobile.file_relay

- Accounts 一个列表关于邮件、推特、iCloud、脸书之类的在设备上设置的账号。
- AddressBook 一个用户通讯录拷贝的 SQLite 数据库；删除的数据可恢复。
- Caches 用户缓存文件夹：暂时的屏幕截图（你在看的最后一样东西）、分享的图片、离线内容、剪贴板、地图图片、键盘输入缓存、其他个人数据。

com.apple.mobile.file_relay

- CoreLocation GPS 日志; 定位缓存常去间隔 (com.apple.routined)
  - fileslockCache_encryptedA.db 和 cache_encryptedA.db
  - 类似于老的 consolidated.db 数据库在 iOS 4 中
  - 时间戳持续约 60 天在我的手机上

com.apple.mobile.file_relay

- HFSMeta （iOS 7 新增!）一个全部的元数据硬盘，iOS 文件系统的 sparseimage，除了实际内容。
  - 时间戳、文件名、大小、**所有**文件创建时间
  - 设备上次被激活／擦除时间
  - 所有安装在设备上的应用和所有文档的文件名（比如说 Dropbox 文档之类）
  - 所有邮件附件的文件名
  - 所有在设备上设置的邮件账户
  - 主机 ID 和所有设备配对的时间戳
  - 电话号码和每个人存储信息草稿的时间戳被保存
  - 基于时间戳的活动时间轴

com.apple.mobile.file_relay

- Keyboard 一份键盘自动纠错的缓存拷贝。
  - DynamicDictionary-4: 一开始一半含有所有最近输入所有应用的内容，合并和输入顺序
  - DynamicDictionary-5: 被改进，只含有词语和词语计数
- MobileCal, MobileNotes 全部用户日历、闹钟和备忘录的数据库图像，保存为 SQLite 格式（被删除数据可恢复）
- Photos 全部转存用户储存在设备上的相册（不只是相机胶卷）

com.apple.mobile.file_relay

- UserDatabases （从 v2 开始存在）转存通讯录、日历、通话记录、信息数据库、邮件元信息（信封收件箱）、SQLite 数据库（被删除信息可恢复）
- VARFS （HFSMeta 的前任）虚拟文件系统元数据转存为 statvfs 格式
- Voicemail 用户语音信箱数据库和音频文件（AMR 格式）的拷贝

com.apple.mobile.house_arrest

- 起初用来允许 iTunes 来复制文档来去第三方应用。
- 即使 iTunes 不通过 GUI 许可，服务允许访问资源库、缓存、Cookies，同时还有设置文件夹。
- 这些文件夹提供高度敏感账户储存、社交／脸书缓存、图片和其它数据保存在「保险箱」中，还有更多。

例子：推特

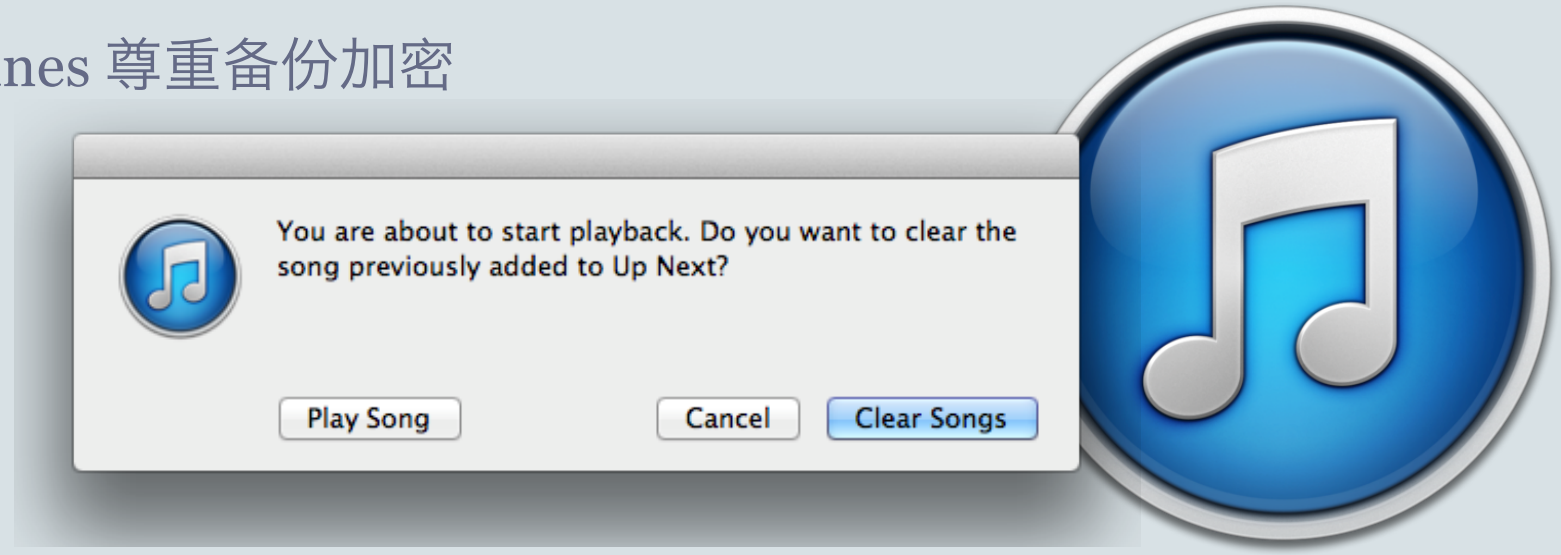
- 我的流中的最近图片
- 大多最近时间线
- 个人信息数据库；许多被删除的消息被恢复
- 最后使用推特的屏幕截图
- OAuth 令牌（合并了用户密钥／秘密，可以用来未来进行远程间谍活动）

例子：图片保险箱

- 保险箱正在「保护」的实际图片
- 包括 PIN 的配置文件，或是 PIN 的散列
- 偶尔情况下，开发者确实加密了图片
- 有时候加密密钥或者 PIN 转存到了系统日志中

理论

- 可能是 iTunes 或者 Xcode 使用它们？不。
  - iTunes 使用 com.apple.mobilesync、 backup2，和其它设施，但是没有使用 file relay 或者 pcap
  - iTunes 使用 house_arrest，但只是用来访问文档；没有必要允许访问资源库、缓存或者其它特权文件夹
  - iTunes 尊重备份加密

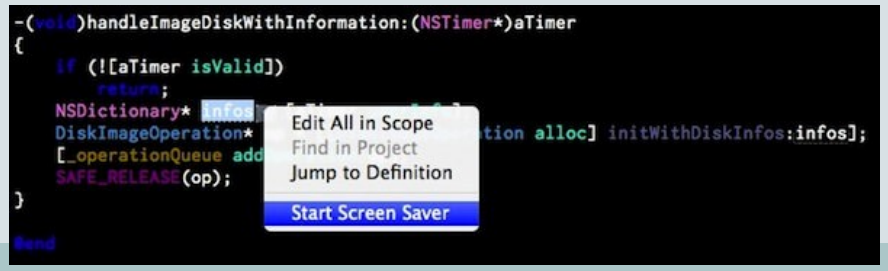


理论

- 可能是用来给天才吧或者 Apple 支持？不。
  - 对技术支持来说，数据太原始了。
  - 无法放回手机
  - 技术支持不应该要求绕过备份加密
  - 数据对技术支持来讲过于私人。

理论

- 可能是给开发者调试的？不。
  - 实际上开发者工具是在开发者图像中的，并且只在开发者模式启用时有效
  - Xcode 不为开发者提供数据包嗅探器界面
  - 开发者不需要绕过备份加密
  - 开发者不需要访问如此敏感内容
  - Apple 想让开发者用 SDK APIs 来得到数据
  - 没有文档告诉开发者有这些「特性」

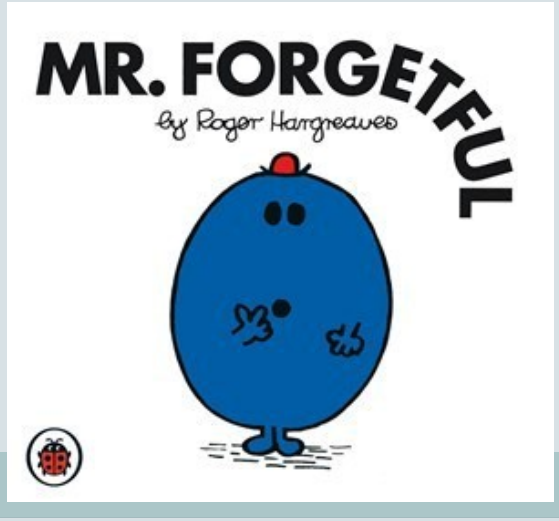


理论

- 可能是用来设计／调试的？不。
  - 不是所有 600 百万设备都需要调试
  - 为防止本地连接，Apple 必然知道这些服务会被恶意软件滥用
  - 你还是不需要绕过备份加密
  - 设计不需要读取如此个人的数据

理论

- 可能是忘掉的旧调试代码？不
  - Apple 已经维持提高这些代码，甚至在 iOS 7 中；他们知道代码在这
  - 已给 Apple CEO 发邮件但无回复
  - 它没有被埋葬；它在 Services.plist 中被列出
  - house_arrest 安全问题可能是「漏洞」，但 file relay 和 pcap 大多不是



更有益的服务

- 当更有益时，以下服务是人工法医的好攻击目标:
- com.apple.iosdiagnostics.relay 提供每个应用详细网络使用，根据每日情况
- com.apple.mobile.installation_proxy 提供企业证书，可以用它加载自定义软件到设备上（可隐秘后台运行）
- com.apple.syslog_relay: 系统日志，提供许多设备活动细节，经常从第三方程序泄漏用户资格通过 NSLog()

隐秘恶意软件

- 在 iOS 7 中在后台隐秘安装软件还是很简单
- Apple 在 iOS 7 中做了一个至关重要的安全升级: 防止套接字连接到本地机器／本地 IP
  - 在此之前，我有间谍软件隐秘运行于后台，转存通话，并在任意地点远程传送内容。（不会因为已知原因发布）
- 这件事阻止了大量个人在行迹中使用间谍软件 ； 它们无法连接到 localhost:62078
- 未来的间谍软件：网络通话攻击其他手机（令人惊叹的僵尸）

隐秘恶意软件

- Info.plist:

```
<key>SBAppTags</key>
<array>
<string>hidden</string>
</array>

<key>UIBackgroundModes</key>
<array>
<string>voip</string>
</array>
```

后台恶意软件

```
[ [ UIApplication sharedApplication ]
setKeepAliveTimeout: 600 handler:^(void)
{
    /* 在后台干点坏事 */
}
```

在 iOS 7 中，你还是可以夺取:

- 所有套接字连接（netstat 数据）
- 进程信息（ps 数据）
- 设备上的许多个人信息
- 启动一些非常靠近核心的用户态漏洞利用

但是等一下。我付了 600 美金为了指纹阅读器

- 指纹阅读器：不添加任何超越基本 PIN 的额外加密
- 已经显示可以被正确的设备欺骗
- 允许 GUI 访问，因此允许配对，因此允许法医转存

- 哦，还有……再说可以绕过，选择配对

配对绕过

- 添加管理设备来使其易获得（比如说，雇员的去世、在坏期限离开、犯罪侦查）
- 设备会尝试拨通家中当第一次设置来自动下载配置概述（经常用来大范围 MDM 首次发布）
- 一个电子选择来阻断可以导致使用模仿 Apple 证书和配置／配对设备开箱即用部署
- 或者通过渗入目标组织，管理者记录可以被用来配对并访问任何管理设备

MCCloudConfiguration

- 拒绝所有配对
- 允许配对，但通知用户
- 允许配对，不通知用户（当锁定时）
- 允许用邀请／答复配对

Pairing Bypass

```
                    ; Check -[ MCProfileConnection hostMayPairWithOptions:challenge: ]
__text:0001938E     LDR.W       R0, [R8,#0xC]
__text:00019392     BL          sub_5754
__text:00019396     CMP         R0, #0
__text:00019398     BNE.W       loc_19AA8
__text:0001939C     LDR.W       R1, [R8,#0x1C]
__text:000193A0     ADD         R2, SP, #0x7E8+var_420
__text:000193A2     ADD         R3, SP, #0x7E8+out
__text:000193A4     MOV         R0, R4
__text:000193A6     BL          sub_1F100

                    ; Pairing is explicitly forbidden by MC
__text:000193AA     CMP         R0, #0
__text:000193AC     BEQ.W       loc_19AB0

                    ; Pairing is allowed by MC, but with challenge/response
__text:000193B0     LDRB.W      R0, [SP,#0x7E8+out]
__text:000193B4     CMP         R0, #0
__text:000193B6     BNE.W       loc_19AC2


                    ; Pairing is allowed by MC while locked / untrusted without
                    ; any challenge/response (pairing security is bypassed)


__text:000193BA     LDRB.W      R0, [SP,#0x7E8+var_420] <- Profit
__text:000193BE     CMP         R0, #0
__text:000193C0     BNE.W       loc_19B06

                    ; Pairing is allowed while locked / untrusted if the device
                    ; doesn't support it
__text:000193C4     MOV         R0, #(cfstr_Hasspringboa_1 - 0x193D0) ; "HasSpringBoard"
__text:000193CC     ADD         R0, PC  ; "HasSpringBoard"
__text:000193CE     BLX         _MGGetBoolAnswer
__text:000193D2     CMP         R0, #1
__text:000193D4     BNE.W       loc_19B06

                    ; Actual pairing security routines (check device lock, whether
                    ; user has pressed "Trust", and so on)

__text:000193D8     MOVS        R0, #0 __text:000193DA      BLX
_MKBGetDeviceLockState
```

在 Pseudocode 中

```
if (mc_allows_pairing_while_locked || device_has_no_springboard_gui)
{
    goto skip_device_lock_and_trust_checks; /* Skip security */
}

/* Pairing Security */

if (device_is_locked == true) {     if
(setup_has_completed) {         if
(user_never_pushed_trust) {
            error(PasswordProtected);
        }
    }
}
```

呼叫家中

- 在安装时，teslad 连接 https://iprofiles.apple.com
  - /resource/certificate.cer
  - /session and /profile
  - 能下载 MCCloudConfiguration
- 可以被电子阻断，有技术或者秘密 FISA 要求
- MCCloudConfiguration 影响配对绕过
- 内置机制绕过 SSL 认证，去你的。
  - MCTeslaConfigurationFetcher 检查 MCCloudConfigAcceptAnyHTTPSCertificate

呼叫家中

- 当被设置时，一个新云设置会被周期检查下载
- -[MCProfileConnection retrieveCloudConfiguration FromURL:username:password:anchorCertificates: completionBlock:]
  - 极大攻击表面如果你能通过 SSL
  - 如果你有 FISA 要求，那就不需要了

给 Apple 的问题

- 为什么会有数据包嗅探器运行在 600 百万个人 iOS 设备上，替代了开发者加载？
- 为什么会有未在文档中记录的服务绕过用户备份加密来从手机中转存极大量个人数据？
- 为什么大部分我的用户数据*还是*没有被用 PIN 或者密码加密，使你有机会侵入我的个人隐私？
- 为什么还是没有机制来检查我的 iPhone 配对的设备，来使我删除不属于我的设备？

配对锁定

```
                    ; Check -[ MCProfileConnection hostMayPairWithOptions:challenge: ]
__text:0001938E                 LDR.W           R0, [R8,#0xC]
__text:00019392                 BL              sub_5754
__text:00019396                 CMP             R0, #0
__text:00019398                 BNE.W           loc_19AA8
__text:0001939C                 LDR.W           R1, [R8,#0x1C]
__text:000193A0                 ADD             R2, SP, #0x7E8+var_420
__text:000193A2                 ADD             R3, SP, #0x7E8+out
__text:000193A4                 MOV             R0, R4
__text:000193A6                 BL              sub_1F100

                    ; Pairing is explicitly forbidden by MC

__text:000193AA                 CMP             R0, #0
__text:000193AC                 BEQ.W           loc_19AB0

                    ; Pairing is allowed by MC, but with challenge/response
__text:000193B0                 LDRB.W          R0, [SP,#0x7E8+out]
__text:000193B4                 CMP             R0, #0
__text:000193B6                 BNE.W           loc_19AC2


                    ; Pairing is allowed by MC while locked / untrusted without
                    ; any challenge/response (pairing security is bypassed)
__text:000193BA                 LDRB.W          R0, [SP,#0x7E8+var_420]
__text:000193BE                 CMP             R0, #0
__text:000193C0                 BNE.W           loc_19B06

                    ; Pairing is allowed while locked / untrusted if the device
                    ; doesn't support it
__text:000193C4                 MOV             R0, #(cfstr_Hasspringboa_1 - 0x193D0) ; "HasSpringBoard"
__text:000193CC                 ADD             R0, PC  ; "HasSpringBoard"
__text:000193CE                 BLX             _MGGetBoolAnswer
__text:000193D2                 CMP             R0, #1
__text:000193D4                 BNE.W           loc_19B06

                    ; Actual pairing security routines (check device lock, whether
                    ; user has pressed "Trust", and so on)

__text:000193D8                 MOVS            R0, #0 __text:000193DA                BLX
_MKBGetDeviceLockState
```

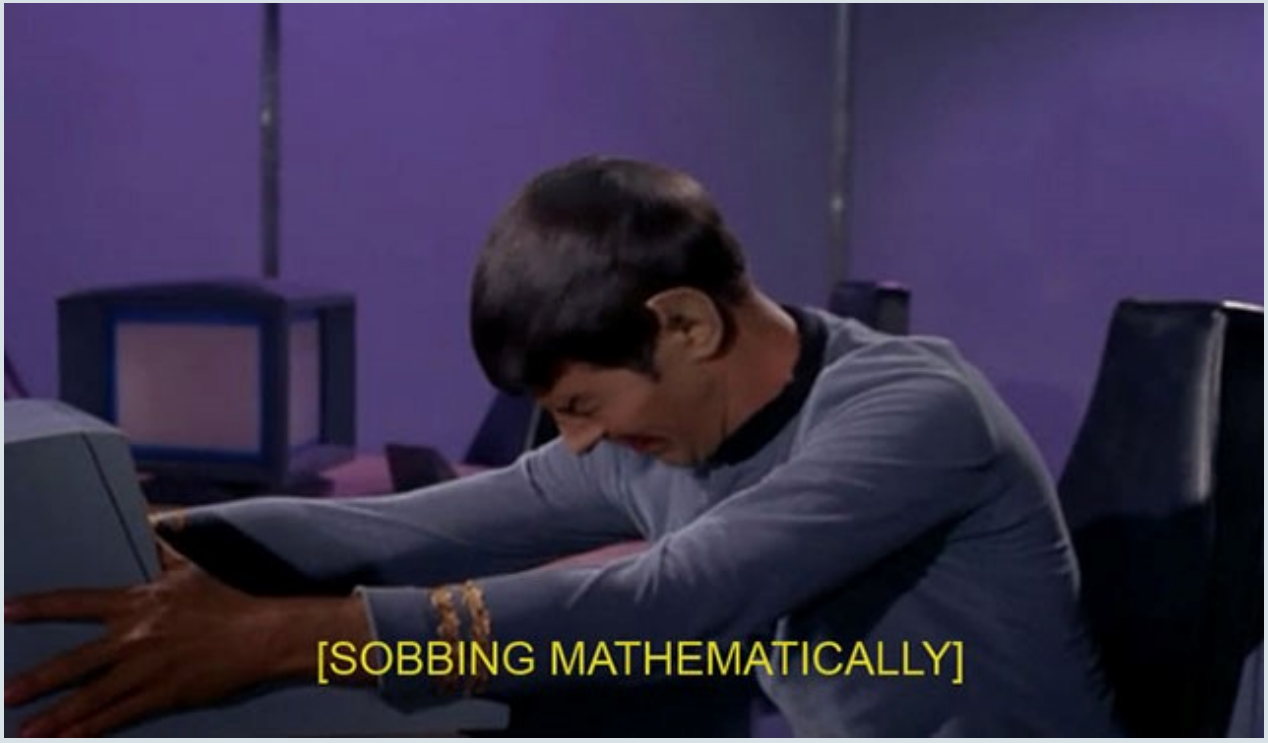
<- HOW DO WE MAKE THIS WORK?

Apple Configurator

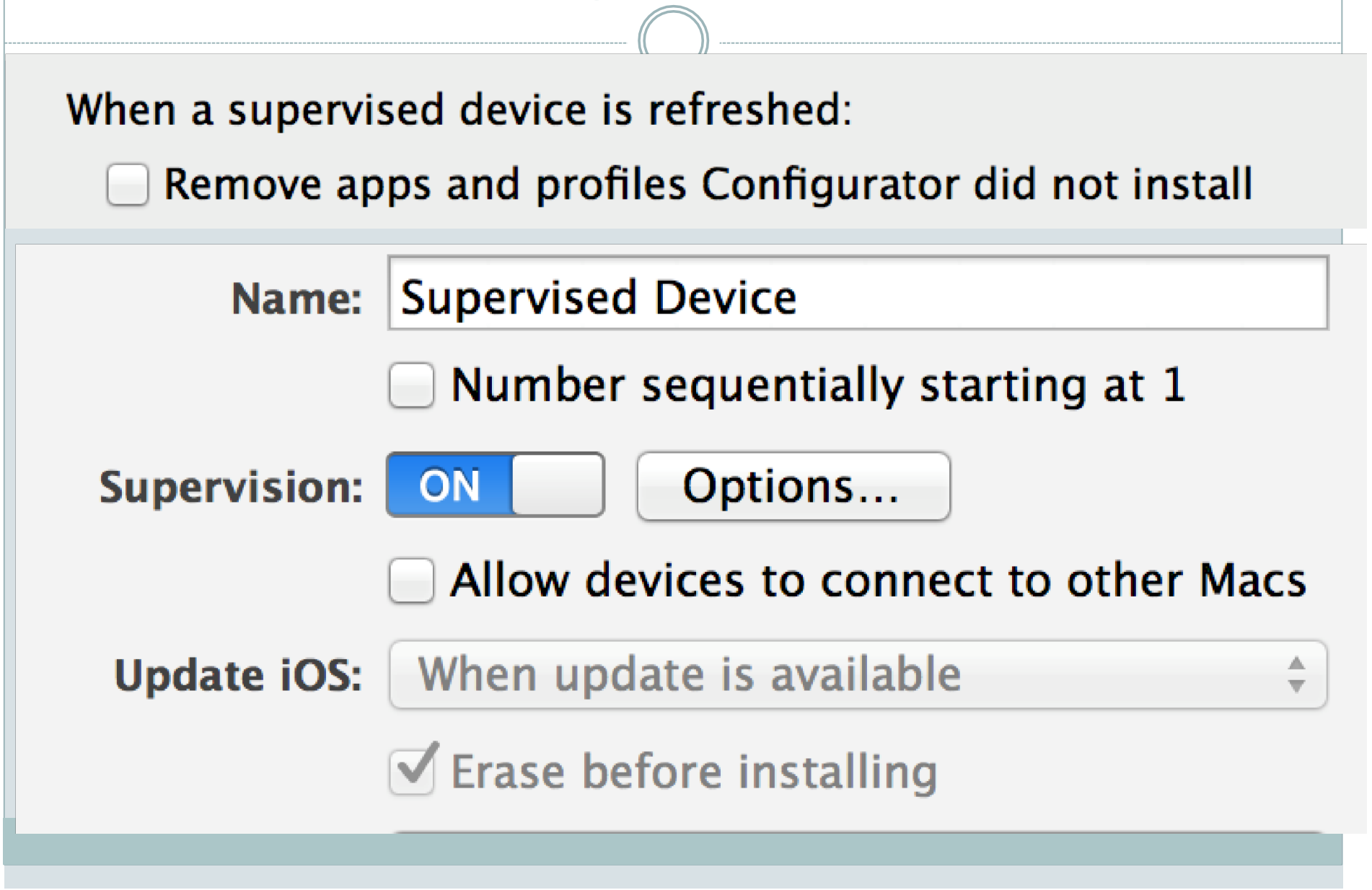
- 在 Mac App Store 中免费
- 允许你设置企业 MDM 限制在你的设备上
- 可用来阻止配对，**即使解锁**
- 桌面上配对一次，然后再也不用……或者（如果你有妄想症）删除所有配对记录并阻止任何通信
- 如果设备被送到 Apple，这没有用；应该还是用复杂密码
- 如果主意变了，一会儿可以删掉它

法医工具

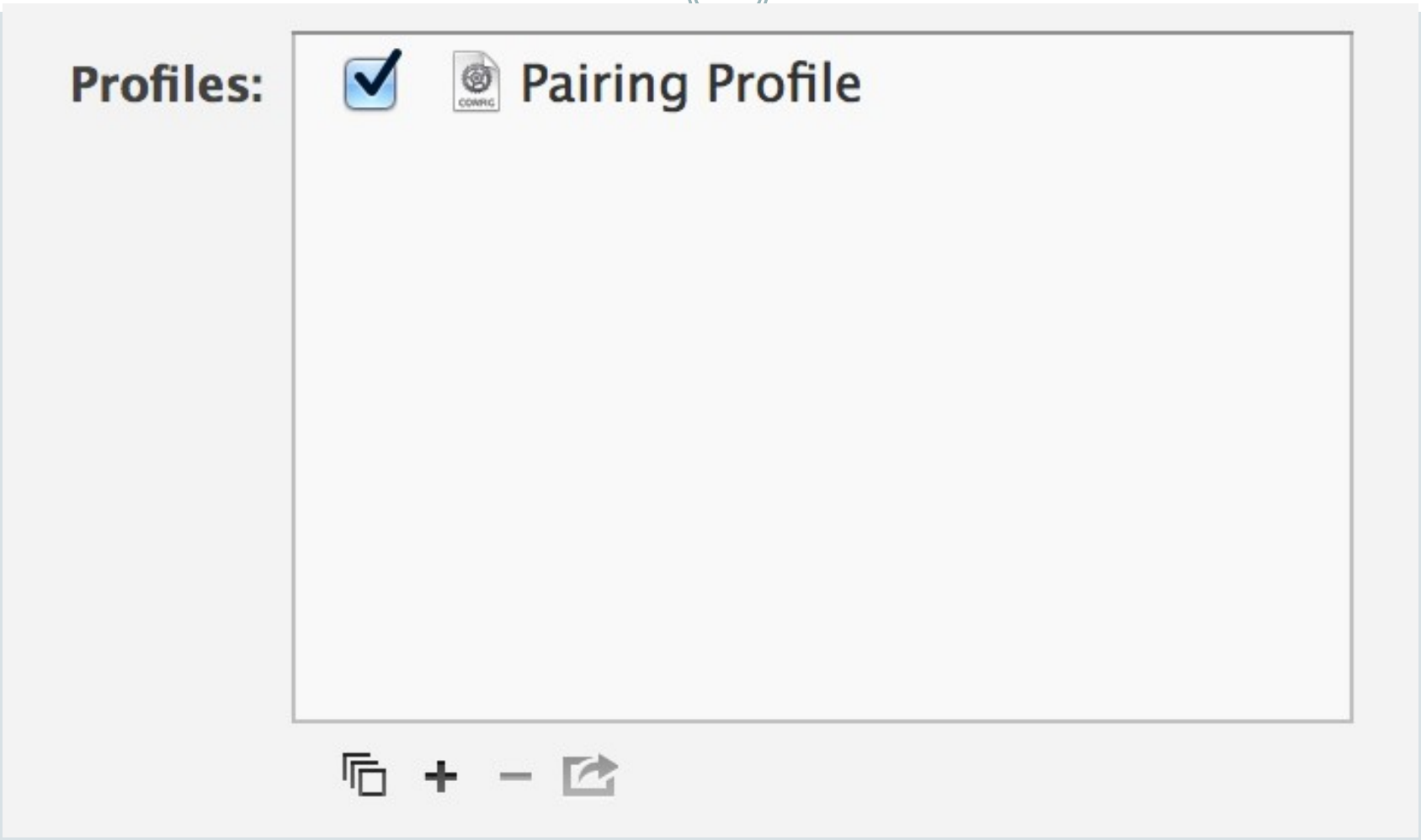
- 每个商业法医工具，在用 Configurator 配对锁定之后：



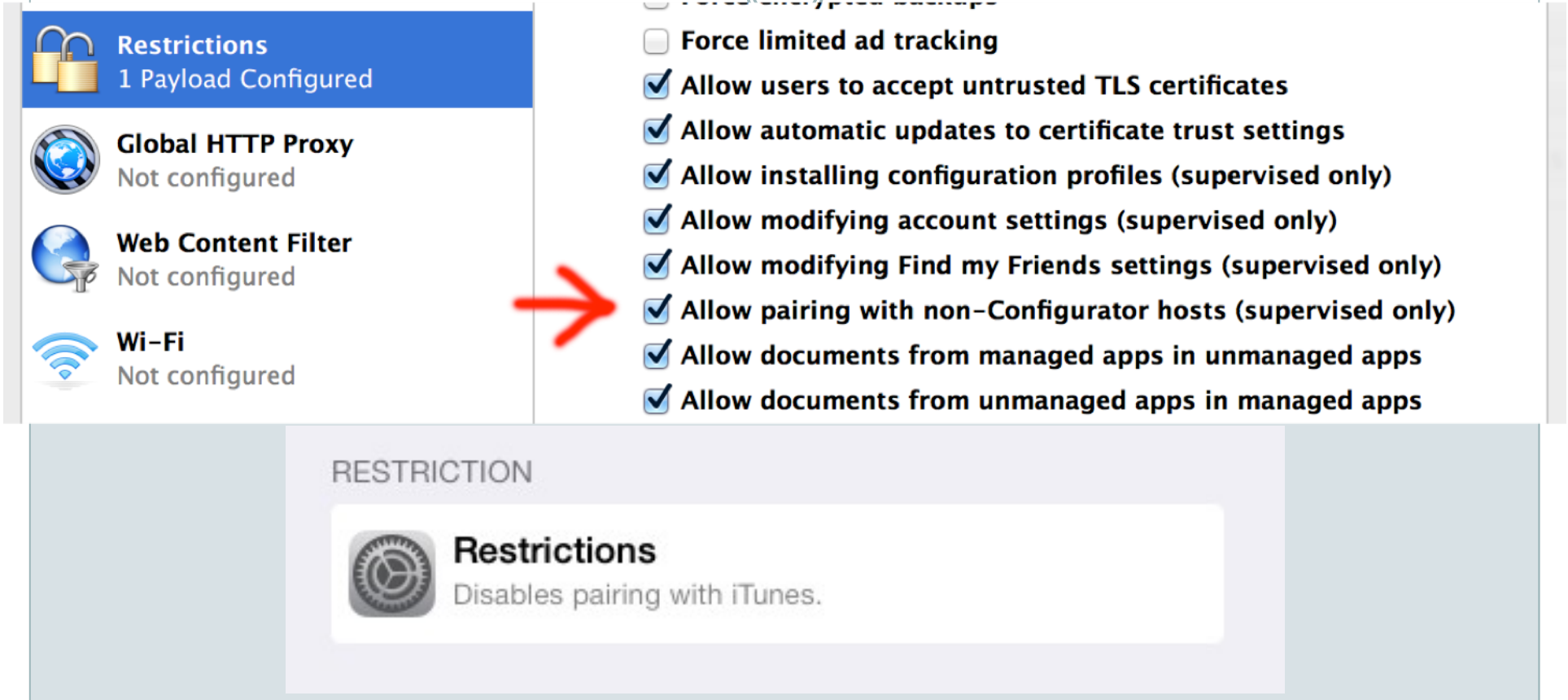
使用 Configurator 配对锁定



使用 Configurator 配对锁定



使用 Configurator 配对锁定



设计建议

- 不对称密码算法来允许加密信息、照片之类，不需要解密
- 文件系统相等于「会话密钥」来内存居民处理（CommCenter）来唯一解密相应数据的影子拷贝（AddressBook）
- 加入启动密码以包含存在 FS 加密；用更强更复杂的密码，减小不方便
- 配对时加密所有密钥和 EscrowBag 从使用备份密码的手机中发送，所以不会在不知情时被使用

总结

- Apple 在我们背后分发了许多数据
- 绕过备份加密是对用户的违规行为
- 没有有效借口在不经过用户知识与许可时泄漏个人数据或允许数据包嗅探器
- 许多数据就是不应离开手机，即使备份之中
- Apple 为给政府和犯罪分子提供诱人攻击服务的企业提供了不少方便之处
- 总而言之，其他情况下的 iOS 的重要安全妥协了，被 Apple，被设计

谢谢

有问题?

找 @JZdziarski

（译者不为翻译内容负责，不代表认同翻译内容）