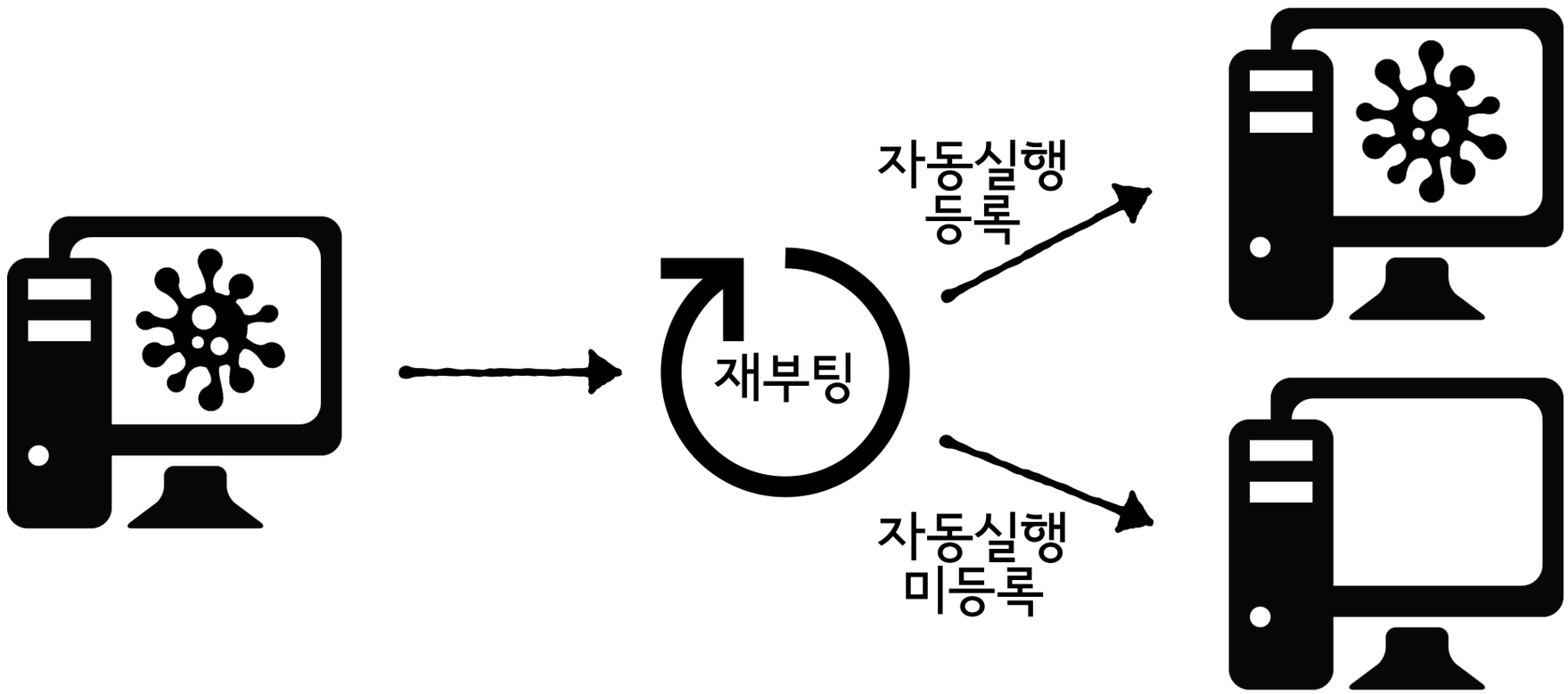
macOS 자동실행 아티팩트

forensic.nofate.com

macOS Sierra

- Siri
- Universal Clipboard
- iCloud Drive
- Auto Unlock with Apple Watch
- Apple Pay
- APFS, Optimized Storage
- Messages

자동실행



- 시스템이 재부팅 후, 지속적 임무 수행 가능
- 첩보형 악성코드가 주로 사용하는 아티팩트

macOS 자동실행

유닉스 아티팩트

- Run Control Script
- Cron Jobs
- Periodic Scripts

맥 아티팩트

- Startup Items
- LoginItem
- Launch Daemon/Agent
- Scripting Additions
- Log Hook
- Authorized Plugins
- Override

시작 스크립트 (Run Control)

```
$ cat /etc/rc.common
##
# Common setup for startup scripts.
##
# Copyright 1998-2002 Apple Computer, Inc.
##

[..SNIP..]

##
# Set command search path
##
PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/libexec:/System/Library/CoreServices; export PATH
```

- 부팅 시점에 실행하는 스크립트
- macOS는 ‘/etc/rc.common’에 정의

작업 예약 명령(Cron Jobs)

```
MacBook-Pro:tabs root# pwd
/usr/lib/cron/tabs
MacBook-Pro:tabs root# cat n0fate
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.pGEOrq43kf installed on Tue Sep 20 16:54:15 2016)
# (Cron version -- $FreeBSD: src/usr.sbin/cron/crontab/crontab.c,v 1.24 2006/09/03 17:52:19 ru Exp $)
0 12 * * * cd ~ && touch test1.txt
MacBook-Pro:tabs root#
```

```
* * * * *  command to execute
| | | | |
| | | | └─── day of week (0 - 6) (0 to 6 are Sunday to Saturday, or use names; 7 is Sunday, the same as 0)
| | | └───── month (1 - 12)
| | └─────── day of month (1 - 31)
| └───────── hour (0 - 23)
└─────────── min (0 - 59)
```

- 유닉스, 리눅스에서 사용하는 작업 스케쥴러
- crontab -e 로 추가 가능
- /usr/lib/cron/tabs 에 계정 별 저장

정기적인 스크립트 실행 (Periodic Scripts)

```
n0fate@n0fates-Mac-mini:/etc/periodic$ ls -R
daily monthly    weekly

./daily:
110.clean-tmps        199.clean-fax         420.status-network
130.clean-msgs        310.accounting        430.status-rwho
140.clean-rwho        400.status-disks 999.local

./monthly:
199.rotate-fax   200.accounting   999.local

./weekly:
320.whatis 999.local
n0fate@n0fates-Mac-mini:/etc/periodic$
```

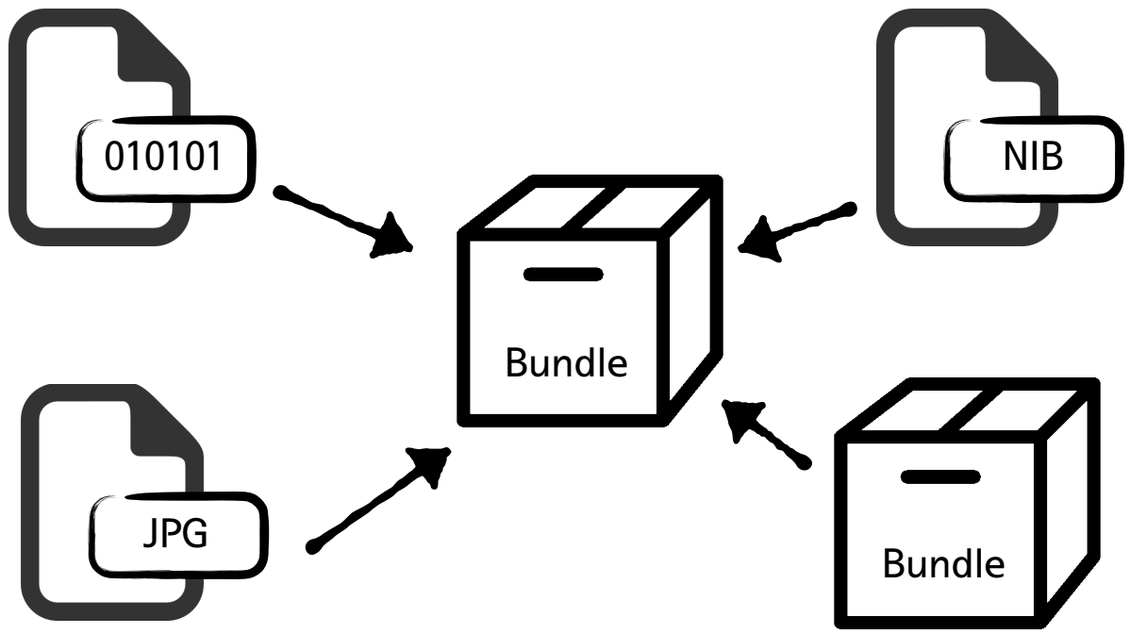
/var/log/daily.out

/var/log/monthly.out

/var/log/weekly.out

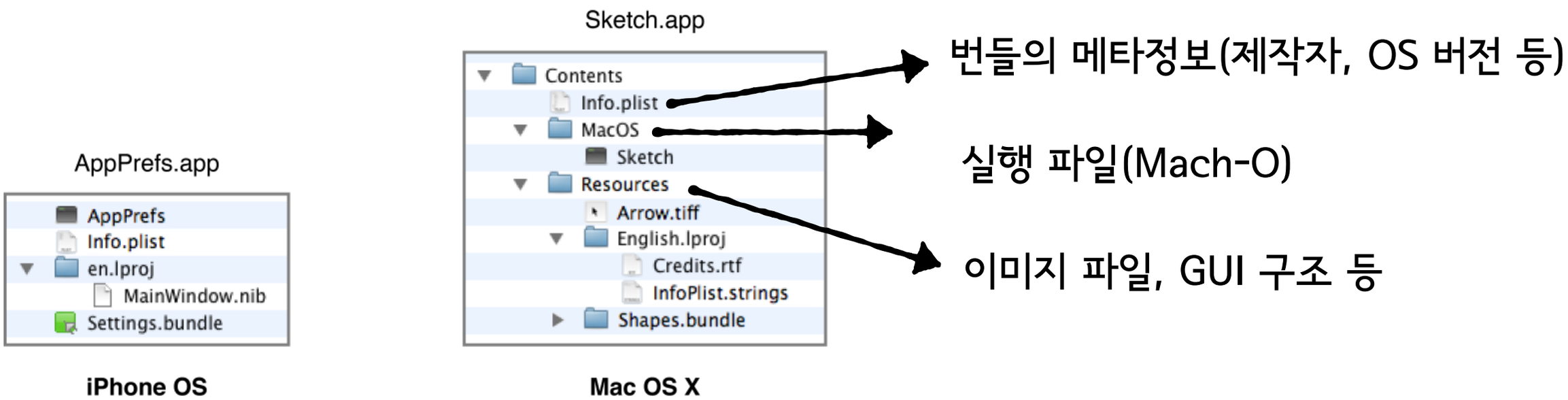
- 일정 주기로 쉘 스크립트 실행 후 결과 저장
- 주기 : daily, weekly, monthly

번들(Bundle)



- 하나의 역할을 수행하기 위해 사용되는 코드와 자원의 묶음

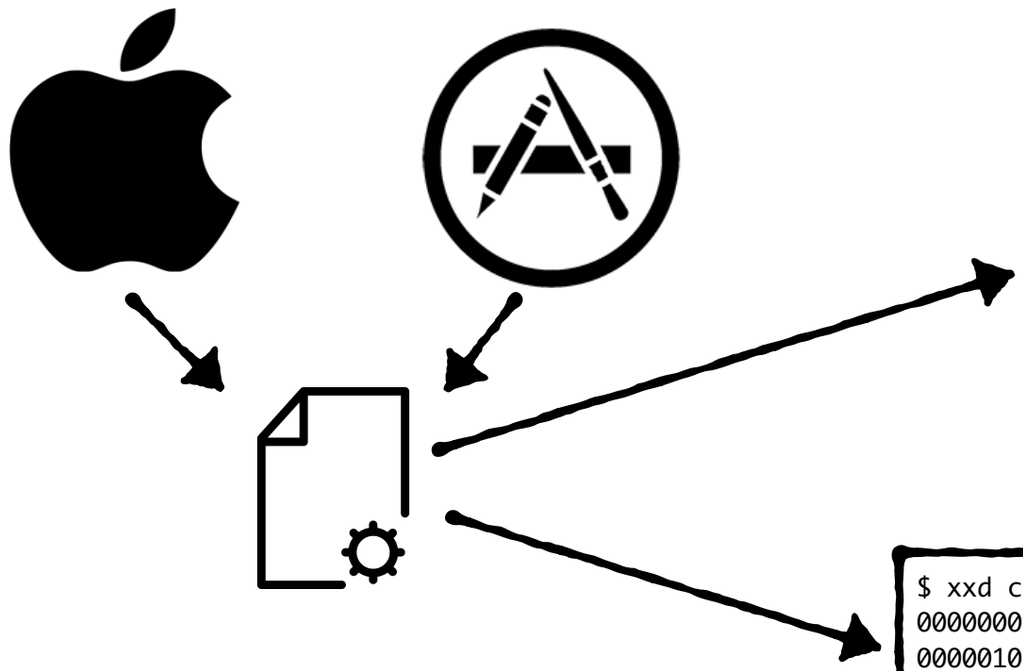
번들(Bundle)



Ref : developer.apple.com

- 애플리케이션, 프레임워크, 라이브러리 등
- 각 파일 포맷에 맞는 도구 이용

프로퍼티 리스트
(Property List)



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
        <key>Label</key>
        <string>com.synology.CloudStation</string>
        <key>ProgramArguments</key>
        <array>
                <string>/usr/bin/open</string>
                <string>/Applications/Synology Cloud
Station.app</string>
        </array>
        <key>RunAtLoad</key>
        <true/>
</dict>
</plist>
```

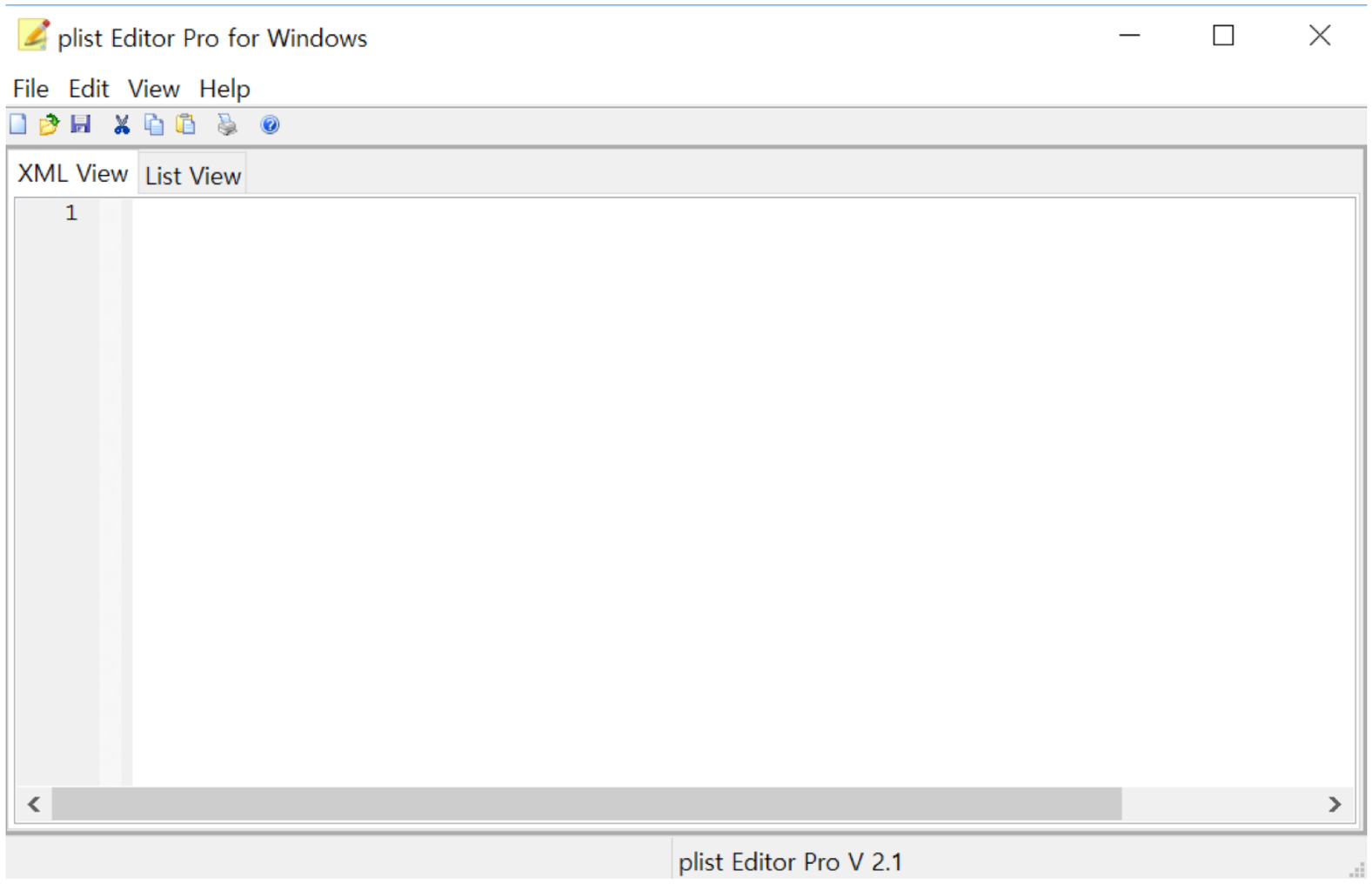
plist

```
$ xxd com.apple.nsservicescache.plist | more
0000000: 6270 6c69 7374 3030 d700 0100 0200 0300  bplist00........
0000010: 0400 0500 0600 0700 0800 2300 4d00 4e04  ..........#.M.N.
0000020: 6804 b904 ba5f 1020 4e53 5365 7276 6963  h...._. NSServic
0000030: 6573 4578 706c 6963 6974 4469 7265 6374  esExplicitDirect
0000040: 6f72 7943 6163 6865 5f10 114e 5353 6572  oryCache_..NSSer
0000050: 7669 6365 734c 5350 6174 6873 5f10 164e  vicesLSPaths_..N
```

bplist

- 설정 정보 저장 포맷
- XML 형태 또는 바이너리 형태

프로퍼티 리스트
(Property List)



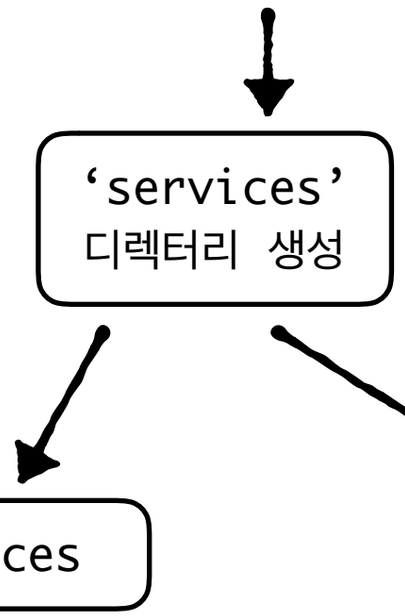
- Plist Editor for Windows

시작 항목(StartupItems)

```
#!/bin/sh
. /etc/rc.common

# The start subroutine
StartService() {
   [..SNIP..]
}

# The stop subroutine
StopService() {
   [..SNIP..]
}

# The restart subroutine
RestartService() {
   [..SNIP..]
}

RunService "$1"
```

자동 실행 경로

```
/Library/StartupItems/
/System/Library/StartupItems/
```

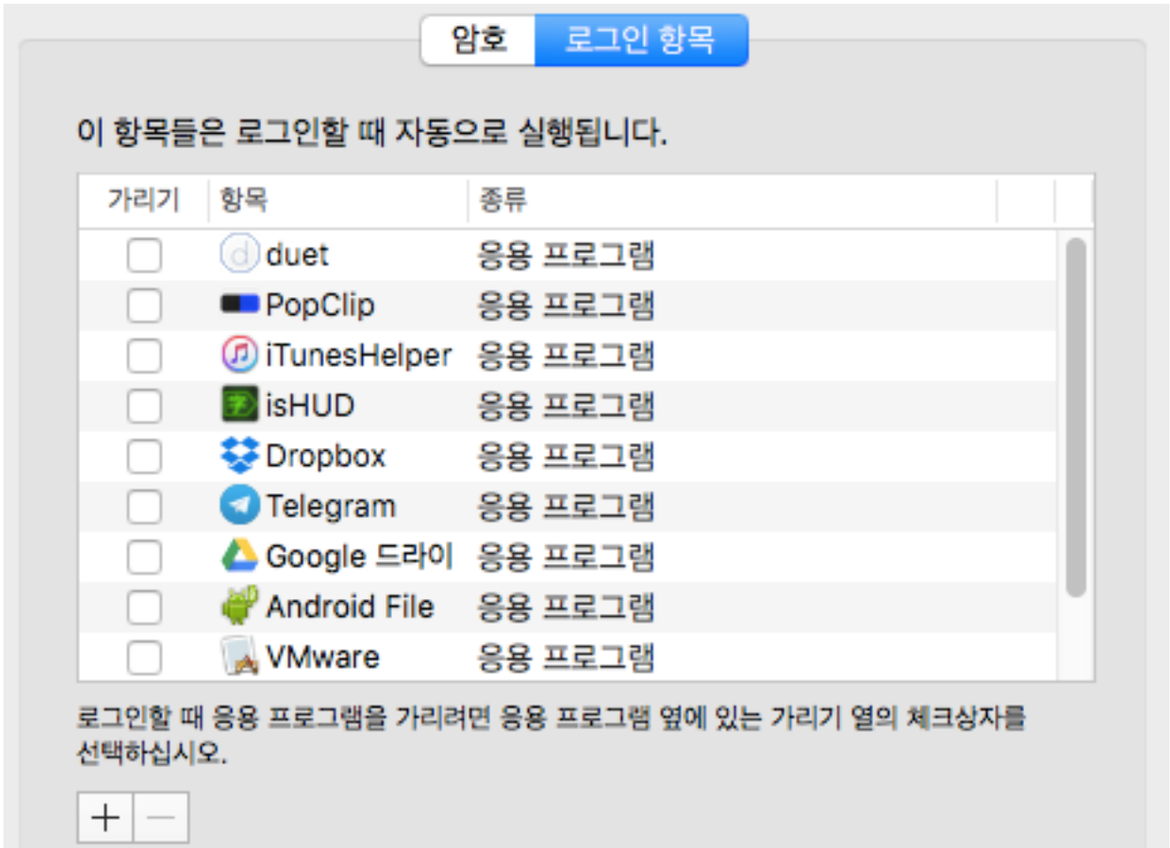



```
{
  Description     = "Software Update service";
  Provides        = ("SoftwareUpdateServer");
  Requires        = ("Network");
  Uses            = ("Network");
  OrderPreference = "Late";
  Messages =
  {
    start = "Starting Software Update service";
    stop  = "Stopping Software Update service";
  };
}
```

StartupParameters.plist

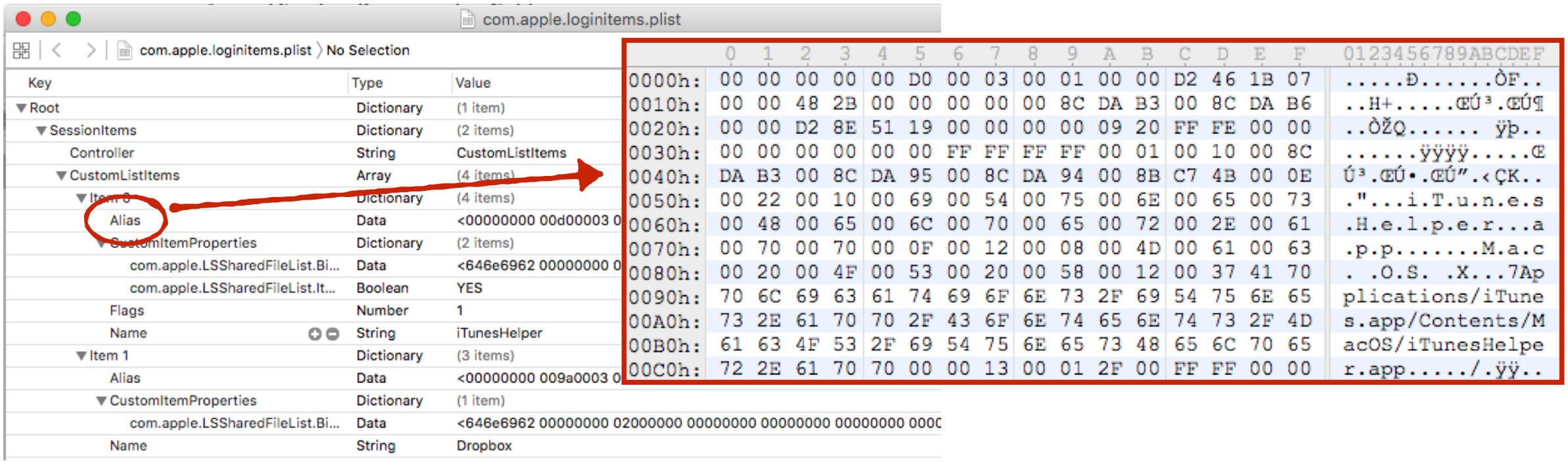
- 부팅 시점에 바이너리/스크립트를 실행
- 루트 권한 실행(사용자 로그인 필요 없음)

로그인 항목(LoginItems)



- 로그인 시점에 자동 실행할 앱을 GUI로 추가
- 가리기 체크 시 독에 나타나지 않음

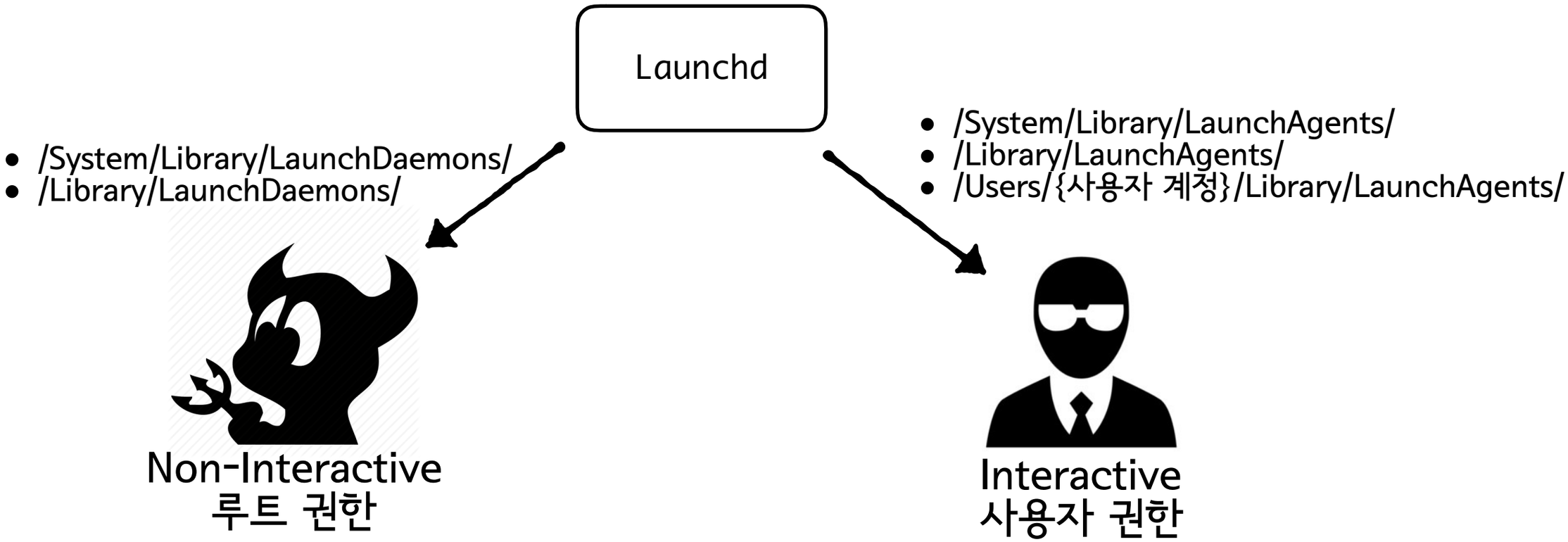
로그인 항목(LoginItems)



경로 : /Users/{사용자 계정}/Library/Preferences/com.apple.loginitems.plist

- 바이너리 프로퍼티 리스트로 저장
- 항목별로 ‘Item X’에 저장

데몬과 에이전트



- 악성코드가 가장 많이 사용하는 아티팩트
- 프로세스 하나 당 하나의 프로퍼티 리스트

데몬과 에이전트

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
     <key>Label</key>
     <string>com.apple.mdworker</string>
     <key>LimitLoadToSessionType</key>
     <string>Aqua</string>
     <key>OnDemand</key>
     <false/>
     <key>ProgramArguments</key>
     <array>
          <string>/Users/administrator/Library/Preferences/jlc3V7we.app/IZsROY7X.-MP</string>
     </array>
     <key>StandardErrorPath</key>
     <string>/Users/administrator/Library/Preferences/jlc3V7we.app/ji33</string>
     <key>StandardOutPath</key>
     <string>/Users/administrator/Library/Preferences/jlc3V7we.app/ji34</string>
</dict>
</plist>
```

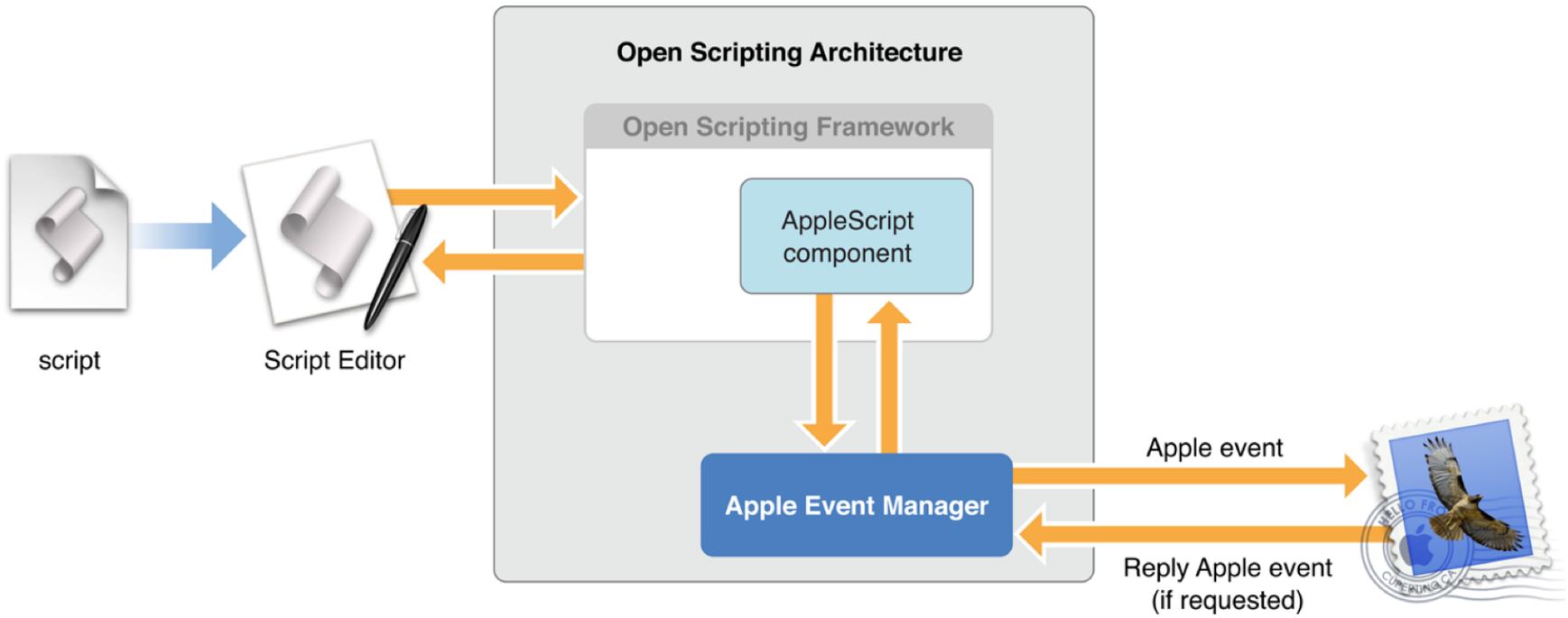
실행할 프로그램의 이름

실행할 프로그램과 인자(배열)

표준 에러와 출력을 저장할 파일 경로

- 프로퍼티 리스트를 분석하여 데몬 식별 가능

추가 스크립팅



- /Users/{사용자 계정}/Library/ScriptingAdditions/
- /Library/ScriptingAdditions/
- /System/Library/ScriptingAdditions/

- 애플 스크립트에 새로운 기능을 추가

추가 스크립팅

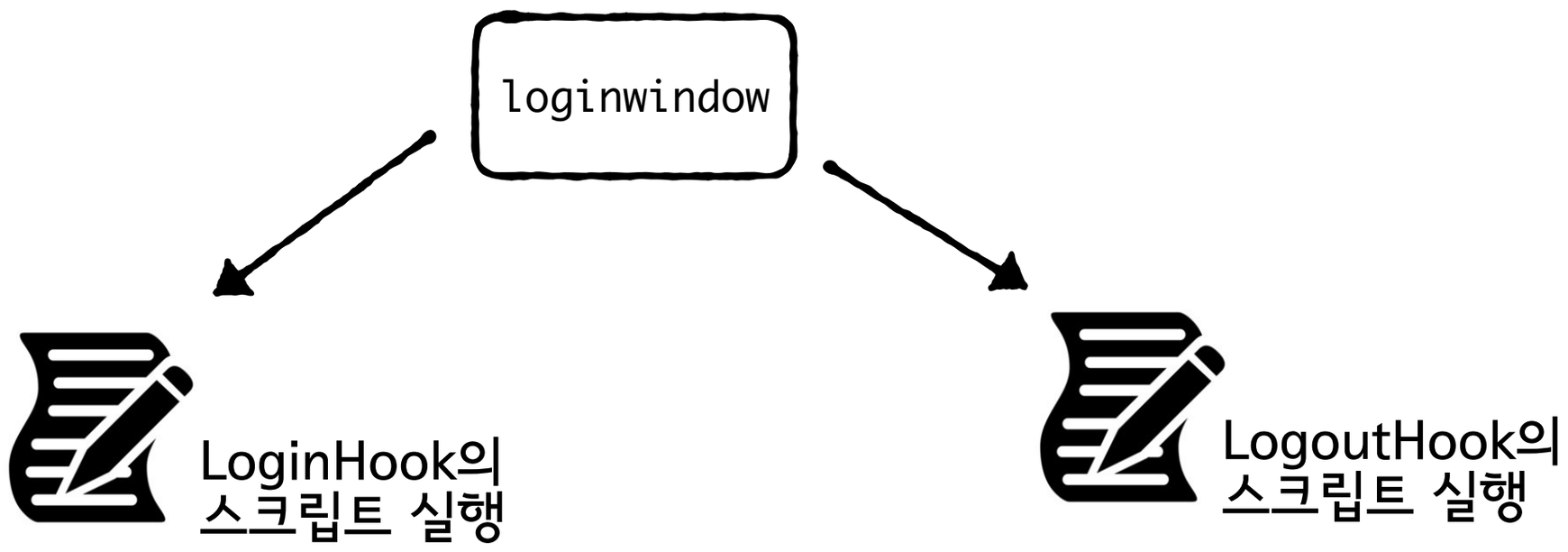
```
[..SNIP..]
      <key>OSAXHandlers</key>
      <dict>
            <key>Events</key>
            <dict>
                  <key>RCSeload</key>
                  <dict>
                        <key>Context</key>
                        <string>Process</string>
                        <key>Handler</key>
                        <string>InjectEventHandler</string>
                        <key>ThreadSafe</key>
                        <false/>
                  </dict>
            </dict>
      </dict>
[..SNIP..]
```

프로세스가 생성될 때

InjectEventHandler 함수 실행

- ‘Info.plist’로 확인 가능
- ‘appleOsax.r’ 에서 핸들러의 기능 정의

로그 훅(Log Hook)



- ‘loginwindow’ 는 로그인/아웃을 처리
- LogHook으로 스크립트 실행 가능
- 해당 기능은 제거될 예정(‘제한적’으로 가능)

로그 훅(Log Hook)

- 콘솔에서 한줄로 등록 가능(루트만 가능)

```
$ sudo defaults write com.apple.loginwindow LoginHook {스크립트 절대 경로}
```

- 등록된 스크립트는 다음 경로에서 분석 가능

```
/private/var/root/Library/Preferences/com.apple.loginwindow.plist
```

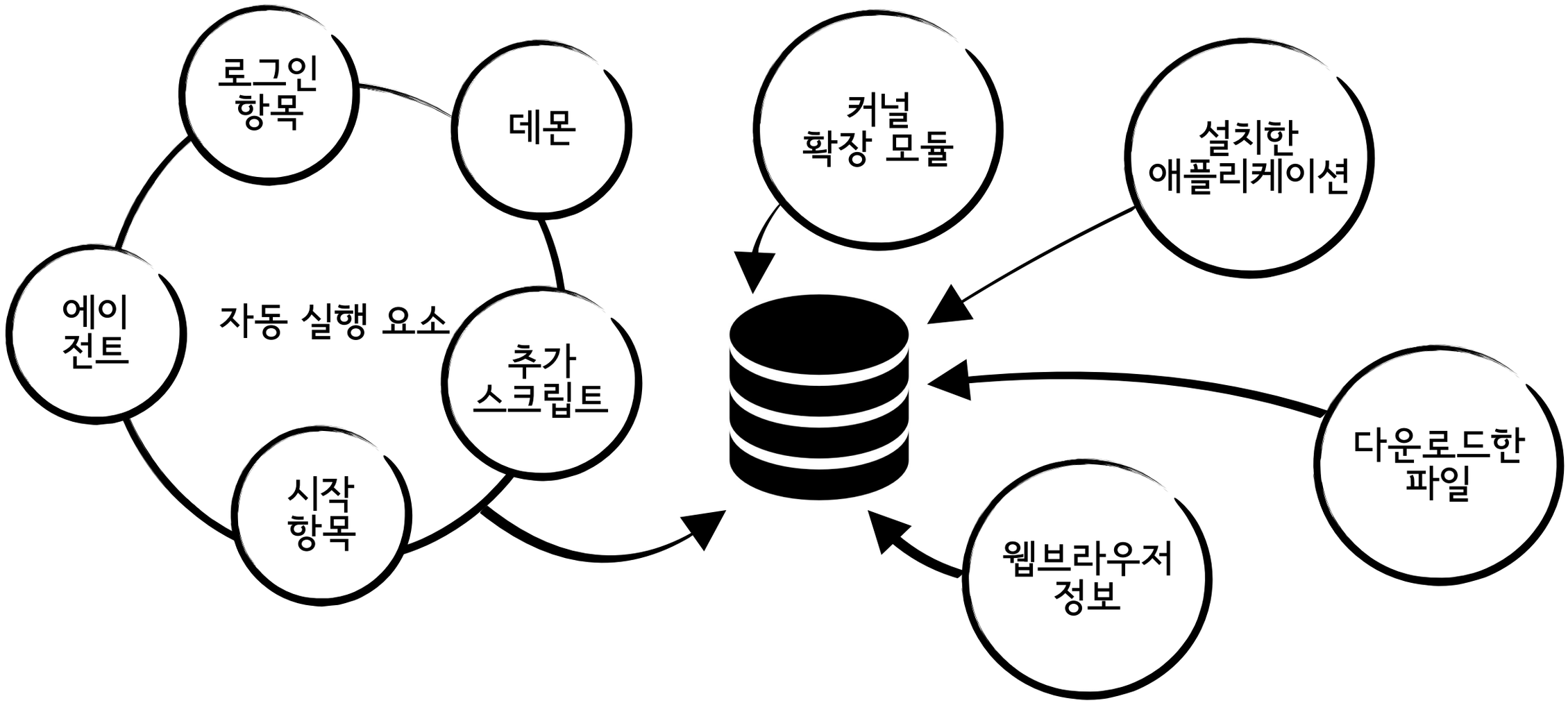
- 바이너리 프로퍼티 리스트 구조

```
[..SNIP..]
    <key>LoginHook</key>
    <string>/Users/n0fate/test.sh</string>
    <key>TALLogoutReason</key>
    <string>Restart</string>
    <key>TALLogoutSavesState</key>
    <false/>
[..SNIP..]
```

자동실행 정리

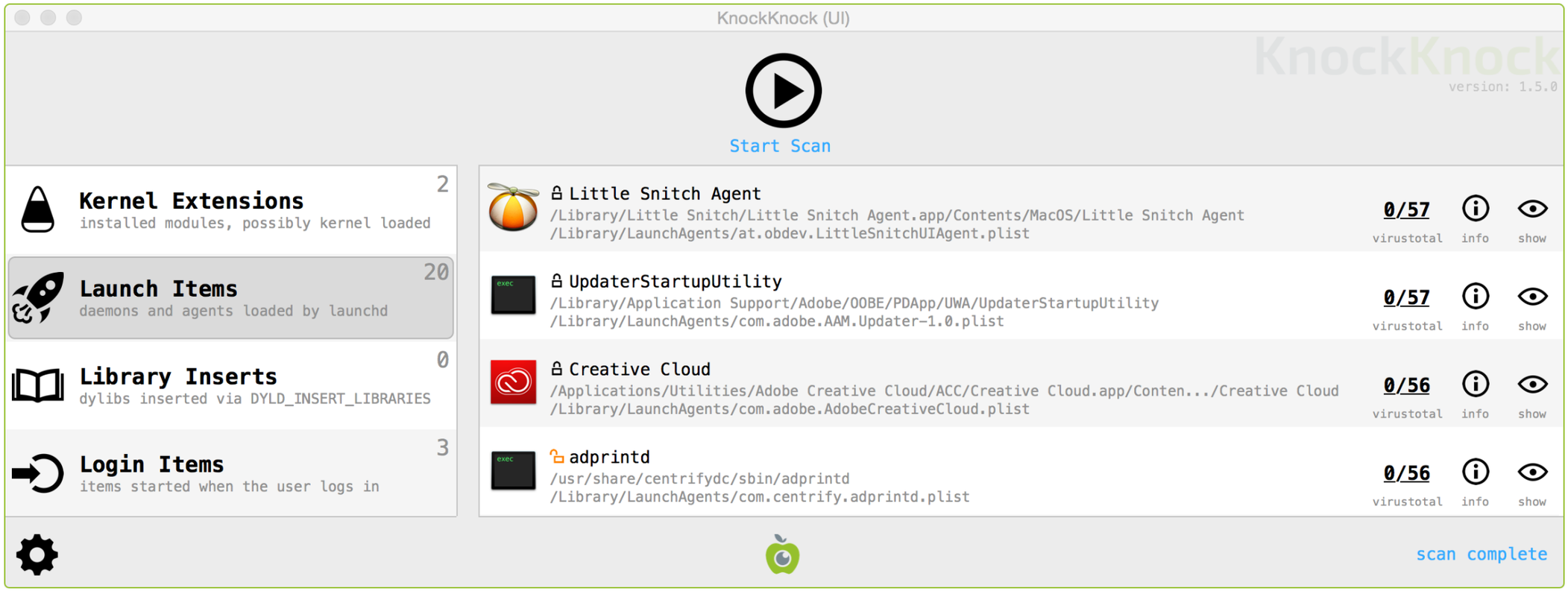
이름	권한	경로	비고
시작 스크립트	루트	/etc/rc.common	쉘 스크립트
작업 예약 명령	루트/사용자	/usr/lib/cron/tabs/	-
정기적 스크립트 실행	루트	/etc/periodic/{daily|weekly|monthly}/	쉘 스크립트
시작 항목	루트	/System/Library/StartupItems /Library/StartupItems	삭제 예정
로그인 항목	사용자	/Users/{사용자 계정}/Library/Preferences/com.apple.loginitems.plist	프로퍼티 리스트
데몬	루트	/System/Library/LaunchDaemons/ /Library/LaunchDaemons/	프로퍼티 리스트
에이전트	사용자	/System/Library/LaunchAgents/ /Library/LaunchAgents/ /Users/{사용자 계정}/Library/LaunchAgents/	프로퍼티 리스트
추가 스크립팅	루트/사용자	/Users/{사용자 계정}/Library/ScriptingAdditions/ /Library/ScriptingAdditions/ /System/Library/ScriptingAdditions/	번들 구조
Log Hook	루트	/private/var/root/Library/Preferences/com.apple.loginwindow.plist	삭제 예정

분석 도구



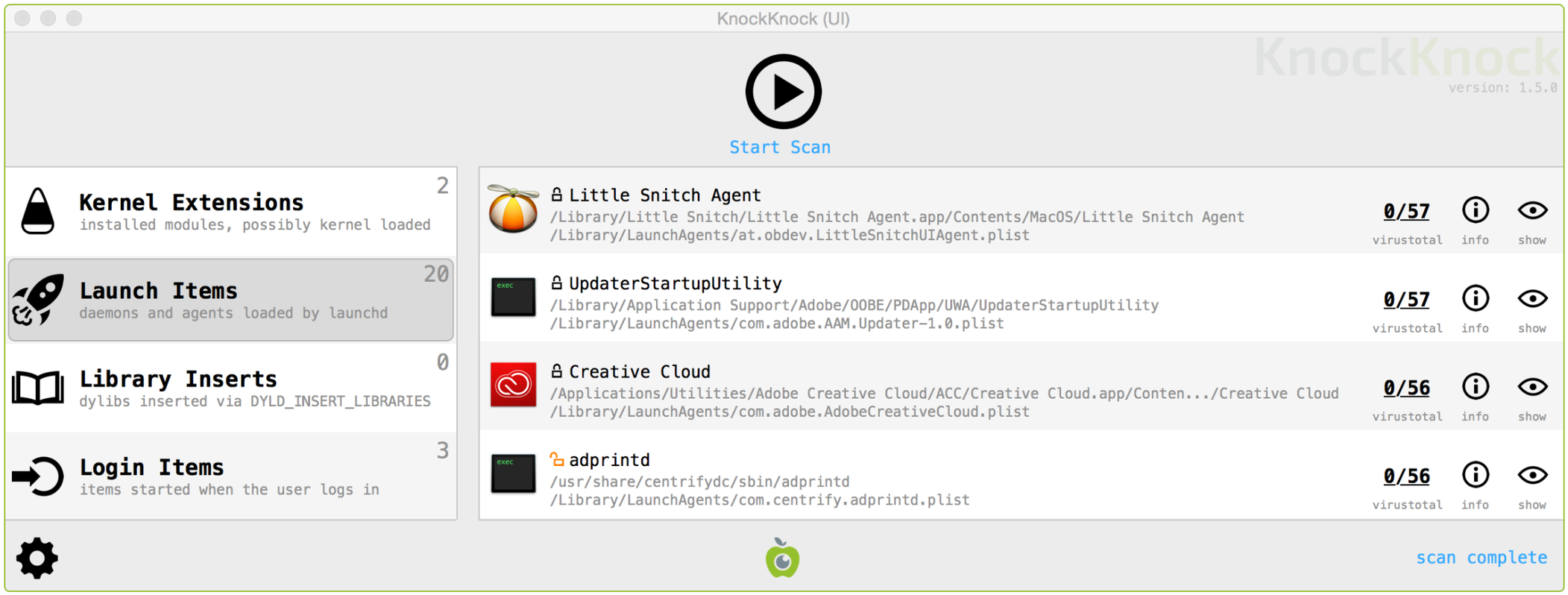
- OSXCollector - free Mac OS X computer forensic tool

분석 도구



- KnockKnock : Persistent 감지 도구
- 오픈소스, 알려진 모든 자동실행 요소 탐지

분석 도구



- Authorization Plugins
- Browser Extensions
- Cron Jobs
- Extensions and Widgets
- Kernel Extensions
- Launch Items
- Library Inserts
- Login Items
- Log Hook
- Periodic Scripts
- Spotlight Importers
- Startup Scripts(Run Control)

Q & A

nofate@nofate.com