AMIRA
Automated Malware Incident Response and Analysis

Jakub (Kuba) Sendor
@jsendor
Dramatis personæ

Yelp Employee

HelpDesk IT Engineer

Security Team Malware Analyst
● Is this machine infected?

● How'd that malware get there?

● How can I prevent and detect further infection?
Malware incident response process

- Initial triage // Malware Analyst
- Forensics collection // IT Engineer
- Forensics analysis // Malware Analyst
- Final remediation // IT Engineer
A forensic evidence collection & analysis toolkit for OS X http://yelp.github.io/osxcollector

- 235 commits
- 29 branches
- 0 releases
- 11 contributors

Merge pull request #91 from Yelp/issue_90

- osxcollector
  - Fix bad boolean compare that strips most browser lines
  - Deleting never_write_api_cache
  - Add OpenDNSFilter
  - Update pre-commit hooks
  - Add comment explaining why the language in .travis is obj-c
  - Install precommit hooks and do lots of guided cleanup

latest commit e5ae922678

Issues
- Pull requests
- Pulse
- Graphs

HTTPS clone URL
https://github.com/Yelp/o

You can clone with HTTPS or Subversion.

Clone in Desktop
Forensics Collected by OSXCollector

OS System Info  Applications  Browser History
Kernel Extensions  Quarantines  Email Info
Downloads  Startup Items  Groups & Accounts
OSXCollector output

path, hashes, timestamps, signature chain, ...

{
    "file_path": "/System/Library/Extensions/Apple_iSight.kext/Contents/MacOS/Apple_iSight",
    "sha2": "19b7b85eaed17d9565dce872f0d1ea8fc0761f508f28bedcc8606b828cbf614",
    "sha1": "99005b68295c202fd359b46cd1411acea96b2469",
    "md5": "b8cc164b6546e4b13768d8353820b216",
    "ctime": "2016-03-31 16:50:39",
    "mtime": "2016-03-30 00:16:50",
    "osxcollector_section": "kext",
    "osxcollector_incident_id": "DelayedHedgehog-2016_08_01-12_35_11",
    "osxcollector_plist_path": "/System/Library/Extensions/Apple_iSight.kext/Contents/Info.plist",
    "osxcollector_bundle_id": "com.apple.driver.Apple_iSight",
    "signature_chain": [
        "Software Signing",
        "Apple Code Signing Certification Authority",
        "Apple Root CA"
    ]
}
Manual analysis with **grep** and **jq** works pretty well

grep a time window

```bash
$ cat bheurope.json | grep '2016-11-03 14:1[2-8]'`
```

only urls in a time window

```bash
$ cat bheurope.json | grep '2016-11-03 14:1[2-8]' | jq 'select(has("url"))\.url'
```

grep a single user

```bash
$ cat bheurope.json | jq 'select(.osxcollector_username=="kuba") | .'
```
$ sudo osxcollector.py
Wrote 35394 lines.
Output in osxcollect-2016_11_03-14_12_39.tar.gz
OSXCollector Output Filters

1. Domain extraction
2. Internal blacklists
3. Threat intel APIs
4. Related files
5. Browser history filter

JSON in -> Domain extraction -> Internal blacklists -> Threat intel APIs -> Related files -> Browser history filter -> JSON out
Automated Malware Incident Response and Analysis
Uploading OSXCollector output

SlickApocalypse_2016-08-03_10:23:13.tar.gz
S3 Event Notifications

- **SlickApocalypse_2016-08-03_10:23:13.tar.gz**
  - Event: ObjectCreated
  - Notification: AmiraS3EventNotifications
S3 Event Notifications

SlickApocalypse_2016-08-03_10:23:13.tar.gz

ObjectCreated

AmiraS3EventNotifications

AMIRA
S3 Event Notifications

SlickApocalypse_2016-08-03_10:23:13.tar.gz

AMIRA

SlickApocalypse_2016-08-03_10:23:13.tar.gz
Automated analysis

- Internal blacklists
- Domain extraction
- Threat intel APIs
- Related files
- Browser history filter
Uploading the results

SlickApocalypse_2016-08-03_10:23:13.json
SlickApocalypse_2016-08-03_10:23:13.html
Analysis results

- Domains and hashes found on the blacklist.
- Threat intel APIs hits for domains and file hashes.
- Blacklist suggestions.
Running AMIRA

Prerequisites:

- SQS queue for the S3 event notifications.
- S3 bucket configured to send S3 event notifications.
- *(optional)* Another S3 bucket for the analysis results.
from amira.amira import AMIRA

amira = AMIRA('us-west-1', 'AmiraS3EventNotifications')

# register results uploader

from amira.s3 import S3ResultsUploader

s3_results_uploader = S3ResultsUploader('amira-results-bucket')
amira.register_results_uploader(s3_results_uploader)

# Ready, set, GO!
amira.run()
Automated forensics collection

1. Trigger OSXCollector via inventory management system.

2. Upload the results to an S3 bucket*.

3. Profit!

*hint: use create-only rights in the S3 bucket policy
#!/bin/bash

file="$1"
bucket="$2"

echo "Uploading file $file to bucket $bucket"

resource="/${bucket}/${file}"
contentType="application/x-compressed-tar"
dateValue=`date -u +"%a, %d %b %Y %T GMT"`
stringToSign="PUT\n\n${contentType}\n${dateValue}\n${resource}"
s3Key="$3"
s3Secret="$4"

signature=`echo -en ${stringToSign} | openssl sha1 -hmac ${s3Secret} -binary | base64`

curl -X PUT -T "$1" \
   -H "Host: $bucket.s3.amazonaws.com" \
   -H "Date: ${dateValue}" \
   -H "Content-Type: $contentType" \
   -H "Authorization: AWS $s3Key:$signature" \
   "https://$bucket.s3.amazonaws.com/$file" | cat
Automation FTW

Before
1-2 days

Now
Several minutes, up to a couple of hours in the worst case
Malware incident response process

Initial triage // Malware Analyst
Forensics collection // AMIRA
Forensics analysis // AMIRA/Malware Analyst
Final remediation // IT Engineer
Takeaways

- Automated process equal less human errors
- No need for physical collection.
- More proactive forensics collection.
Try it out!

https://github.com/Yelp/amira

@jsendor

kuba@yelp.com
@YelpEngineering
fb.com/YelpEngineers
@YelpEngineering
engineeringblog.yelp.com
github.com/yelp