Agenda

1. Problematic
2. Internals and results
3. A community project
4. Workshop
5. Conclusion
Agenda

1. Problematic
2. Internals and results
3. A community project
4. Workshop
5. Conclusion
<table>
<thead>
<tr>
<th>De: <a href="mailto:admin@chat-k.cat">admin@chat-k.cat</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>À: me</td>
</tr>
<tr>
<td>Sujet: Try this one !!!</td>
</tr>
<tr>
<td>&lt;3 cats</td>
</tr>
</tbody>
</table>

BestCatScreensaverEver.exe
Is BestCatScreensaverEver.exe clean?

Solution #1: scan it with your antivirus.
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #1: scan it with your antivirus.

+ easy
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #1: scan it with your antivirus.

+ easy
+ quick (well... often)
Is BestCatScreensaverEver.exe clean?

**Solution #1**: scan it with your antivirus.

+ easy
+ quick (well... often)
- all the security based on one vendor
Is BestCatScreensaverEver.exe clean?

Solution #1: scan it with your antivirus.

+ easy
+ quick (well... often)
- all the security based on one vendor

Good but not enough
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #2: send it to a website for scanning
Is BestCatScreensaverEver.exe clean?

**Solution #2**: send it to a website for scanning

+ many sites freely available:
  - virustotal.com
  - avcaesar.malware.lu
  - metascan.com
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #2: send it to a website for scanning

+ many sites freely available:
  virustotal.com
  avcaesar.malware.lu
  metascan.com
+ many antivirus supported
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #2: send it to a website for scanning

+ many sites freely available:
  virustotal.com
  avcaesar.malware.lu
  metascan.com

+ many antivirus supported

- one file at a time
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #2: send it to a website for scanning

+ many sites freely available:
  - virustotal.com
  - avcaesar.malware.lu
  - metascan.com
+ many antivirus supported
- one file at a time
- files are sent on the Internet
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #2: send it to a website for scanning

+ many sites freely available:
  - virustotal.com
  - avcaesar.malware.lu
  - metascan.com
+ many antivirus supported
- one file at a time
- files are sent on the Internet
- scan settings are unknown
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #2: send it to a website for scanning

+ many sites freely available:
  virustotal.com
  avcaesar.malware.lu
  metascan.com
+ many antivirus supported
- one file at a time
- files are sent on the Internet
- scan settings are unknown

Good but not enough
Is *BestCatScreensaverEver.exe* clean?

**Solution #3** : Open the file #YOLO
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #3: Open the file #YOLO
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #3: Open the file YOLO

+ opportunity to test your backup/restore procedures
Problematic

Is BestCatScreensaverEver.exe clean?

Solution #3: Open the file #YOLO

+ opportunity to test your backup/restore procedures

No comment
New threats → New tools

Companies and public CERT share the same analysis:

Use of a single antivirus is not enough, but antivirus cannot be avoided.

Antivirus are a source of information, among other ones, in the incident response process.
Companies and public CERT share the same analysis:

Use of a single antivirus is not enough, but antivirus cannot be avoided.

Antivirus are a source of information, among other ones, in the incident response process.

To handle all these sources and gather the most information, a modular, scalable tool which can rely on a community of users/contributors is needed.
IRMA

Incident Response & Malware Analysis

- Private file analysis platform
- Open source (Apache V2 license)
- Customisable
Key features

- **Private** platform: no data ever leaves your network
- **Analyze** files, and not only with antivirus
  (24 analyzers available)
- **Several** files simultaneously analyzed
- **Open source** (code hosted on GitHub)
- **Customizable** (API, plugins)
Analysis modules

**ANTIVIRUS**
- AVIRA
- GDATA
- MCAFEE
- SYMANTEC
- AVAST
- BITDEFENDER
- COMODO
- ESETNOD32
- FPROT
- MCAFEE
- AVG
- CLAMAV
- DRWEB
- ESCAN
- FSECURE
- SOPHOS
- VIRUSBLOKADA
- ZONER

**METADATA**
- PEiD
- YARA
- PE STATIC ANALYSIS

**DATABASE**
- NSRL

**EXTERNAL**
- VIRUSTOTAL
Other usage examples

• Web API
Other usage examples

- Web API
- Any client can access it
- New usages!
Other usage examples

Cleaning kiosk for USB keys
Other usage examples

- Cleaning kiosk for USB keys
- Filter for mail attachments
A few figures

- Project started in November 2013.
- 3 Quarkslab engineers.
- 1 Orange intern for 6 months.

Total: 680 days at the end of 2014 (3 man-years).
Agenda

1. Problematic
2. Internals and results
3. A community project
4. Workshop
5. Conclusion
Global architecture
Adding analysers

- Each analysis module is a plugin.
- Separated in two parts:
  - Interface, specific to IRMA
  - The processing part, which analyses the file. It is independant from IRMA and can be reused in another project.
- Plugins are automatically discovered when a probe is started.
Customizing the results

• Each analysis result can be independently filtered.
• Plugins are dynamically discovered when the \textit{frontend} is started.
• Results are kept in raw form in the database.
Customizing the results

- Each analysis result can be independently filtered.
- Plugins are dynamically discovered when the *frontend* is started.
- Results are kept in raw form in the database.
Demo
Agenda

1. Problematic
2. Internals and results
3. A community project
4. Workshop
5. Conclusion
Building a community

Creating an open source project is good

If the project has users, it is better.

If it has contributors, it is even better.
Creating an open source project is good

If the project has users, it is better.

If it has contributors, it is even better.

Need for a simple, deterministic installation system
Installation v1.0

ETA: Unknown
Installation v1.1.0

ETA: 5 minutes
**Installation v1.1.0**

Installing Vagrant:

```
https://www.vagrantup.com/downloads.html
```

Installing Ansible:

```
$ sudo pip install ansible
```

Installing IRMA:

```
$ git clone https://github.com/quarkslab/irma-ansible
$ cd irma-ansible
$ ansible-galaxy install -r ansible-requirements.yml
$ vagrant up
```
The birth of a community

2 contributors, 3 new probes:

- YARA
- GDATA for Windows
- AVIRA for Windows

HITB challenge:

- Outlook submitter (scan all attachments)
- ICAP probe
1. Problematic
2. Internals and results
3. A community project
4. Workshop
5. Conclusion
Workshop agenda

- PROBE - Create your own probe
- PROBE - Integrate it in IRMA
- FRONTEND - Add a formatter to customize its output
- FRONTEND - API 101
Workshop agenda

• PROBE - Create your own probe
  • PROBE - Integrate it in IRMA
  • FRONTEND - Add a formatter to customize its output
  • FRONTEND - API 101
Probe skeleton
Probe skeleton
Probe skeleton

IRMA plugin code
Probe skeleton

dependencies
Balbuzard - malware analysis tools to extract patterns of interest and crack obfuscation such as XOR

Author: Philippe Lagadec
Homepage: http://www.decalage.info/python/balbuzard
Balbuzard 101

```python
>> from balbuzard.balbuzard import patterns, Balbuzard
>> Bal = Balbuzard(patterns=patterns)
>> data = open("./attachment1.exe").read()
>> list(Bal.scan(data))

[(<balbuzard.balbuzard.Pattern at 0x7fd37cda23d0>, [(0, 'MZ'), (15320, 'MZ')]),
 (<balbuzard.balbuzard.Pattern at 0x7fd37cda2410>, [(232, 'PE'), (9541, 'PE'), (50172, 'PE'), (78332, 'PE')]),
 [...],
 (<balbuzard.balbuzard.Pattern at 0x7fd37cda2710>, [(27129, 'Pop')])]
```
Balbuzard probe – connect to VM

VM ADDRESS?

Credentials: vagrant/vagrant

SSH TIME

$ ssh vagrant@vm_address -i vagrant_insecure_private_key
vagrant@brain:~$
Create directory

- Copy Skeleton directory

```bash
$ sudo su deploy
$ cd /opt/irma/irma-probe/current/modules/metadata
$ git clone https://github.com/quarkslab/irma-probe-tutorial balbuzard_analyzer
$ cd balbuzard_analyzer
$ git checkout origin/balbuzard-level0
```
Update metadata

- Rename all Skeleton in Balbuzard
- Update Metadata
Update metadata

- Rename all Skeleton in Balbuzard
- Update Metadata

$ git diff origin/balbuzard-level1
$ git checkout -f origin/balbuzard-level1
Handle dependencies

- declare module dependencies
from balbuzard.balbuzard import patterns, Balbuzard

_plugin_dependencies_ = [
    ModuleDependency('balbuzard',
                     help='See requirements.txt for needed dependencies'),
]
>> from balbuzard.balbuzard import patterns, Balbuzard

```python
_plugin_dependencies_ = [
    ModuleDependency('balbuzard',
        help='See requirements.txt for needed dependencies'),
]
```

balbuzard>=0.19
from balbuzard.balbuzard import patterns, Balbuzard

_plugin_dependencies_ = [
    ModuleDependency('balbuzard',
    help='See requirements.txt for needed dependencies'),
]

balbuzard>=0.19

$ git diff origin/balbuzard-level2
$ git checkout -f origin/balbuzard-level2
Output results

- use analysis module to output interesting results
```python
>> Bal = Balbuzard(patterns=patterns)
>> data = open("./attachment1.exe").read()
>> list(Bal.scan(data))

def __init__(self):
    module = sys.modules['balbuzard.balbuzard']
    patterns = module.patterns
    self.Analyzer = module.Balbuzard(patterns=patterns)

def run(self, paths):
    [...]  # code snippet
    try:
        started = timestamp(datetime.utcnow())
        with open(paths, "rb") as f:
            data = f.read()
            res = list(self.Analyzer.scan(data))
            response.results = res
```
Balbuzard probe - processing - level3

```python
>> Bal = Balbuzard(patterns=patterns)
>> data = open("./attachment1.exe").read()
>> list(Bal.scan(data))
```

```python
def __init__(self):
    module = sys.modules['balbuzard.balbuzard']
    patterns = module.patterns
    self.Analyzer = module.Balbuzard(patterns=patterns)

def run(self, paths):
    [...]
    try:
        started = timestamp(datetime.utcnow())
        with open(paths, "rb") as f:
            data = f.read()
        res = list(self.Analyzer.scan(data))
        response.results = res
```

```
$ git diff origin/balbuzard-level3
$ git checkout -f origin/balbuzard-level3
```
vagrant@brain:~$ sudo su irma

irma@brain:~$ cd /opt/irma/irma-probe/current

irma@brain:~$ venv/bin/python -m tools.run_module

irma@brain:~$ venv/bin/python -m tools.run_module Balbuzard /bin/ls

[...]
{'duration': 0.03014206886291504, 'error': None, 'name': 'Balbuzard', 'type': 'metadata', 'version': None}
Workshop agenda

• PROBE - Create your own probe
• PROBE - Integrate it in IRMA
• FRONTEND - Add a formatter to customize its output
• FRONTEND - API 101
vagrant@brain:~$ sudo supervisorctl restart probe_app
probe_app: stopped
probe_app: started

vagrant@brain:~$ sudo supervisorctl tail probe_app
[...]
WARNING:probe.tasks: *** [metadata] Plugin Balbuzard successfully loaded
Job done!

Metadata

Balbuzard
Responded in 0.07s

{"EXE: section name": [480, "text"], [500, "data"], [520, "rdata"], [600, "rsrp"], "Executable filename": [631}
Workshop agenda

• PROBE - Create your own probe
• PROBE - Integrate it in IRMA
• FRONTEND - Add a formatter to customize its output
• FRONTEND - API 101
Formatter files
Balbuzard probe – level 0

Empty formatter

- Create empty formatter directory
- Apply only current formatter to balbuzard probe

```bash
$ sudo su deploy
$ cd /opt/irma/irma-frontend/current/frontend/helpers/formatters
$ git clone https://github.com/quarkslab/irma-formatter-tutorial balbuzard
$ cd balbuzard
$ git checkout origin/balbuzard-level0
```
vagrant@brain:~$ sudo supervisorctl restart frontend_api
frontend_api: stopped
frontend_api: started
Balbuzard probe – level 1

First shot

- return something

```bash
$ git diff origin/balbuzard-level1
$ git checkout -f origin/balbuzard-level1
```
Exception handling

- catch exceptions in format

$ git diff origin/balbuzard-level2
$ git checkout -f origin/balbuzard-level2
Pretty output

- iterate through results items to pretty print it

$ git diff origin/balbuzard-level3
$ git checkout -f origin/balbuzard-level3
vagrant@brain:~$ sudo supervisorctl restart frontend_api
frontend_api: stopped
frontend_api: started
Goal

- PROBE - Create your own probe
- PROBE - Integrate it in IRMA
- FRONTEND - Add a formatter to customize its output
- FRONTEND - API 101
Swagger documentation

visit http://<vm_address>/swagger
**IRMA API**

### Implementation Notes
When retrieving a scan, you'll get a results property containing the total count of scan results items. With this URL you can retrieve the full paginated list of items.

### Response Class (Status 200)

<table>
<thead>
<tr>
<th>Model</th>
<th>Model Schema</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Agenda

1. Problematic
2. Internals and results
3. A community project
4. Workshop
5. Conclusion
Modular solution to face malware infections

- File analysis framework.
- Private, customisable.
- Central brick for incident response.
- Various usages.
Contact

http://irma.quarkslab.com  -  contact@quarkslab.com

GitHub

https://github.com/quarkslab/irma

@qb_irma

#irc

#qb_irma@freenode

http://irma.quarkslab.com  -  contact@quarkslab.com