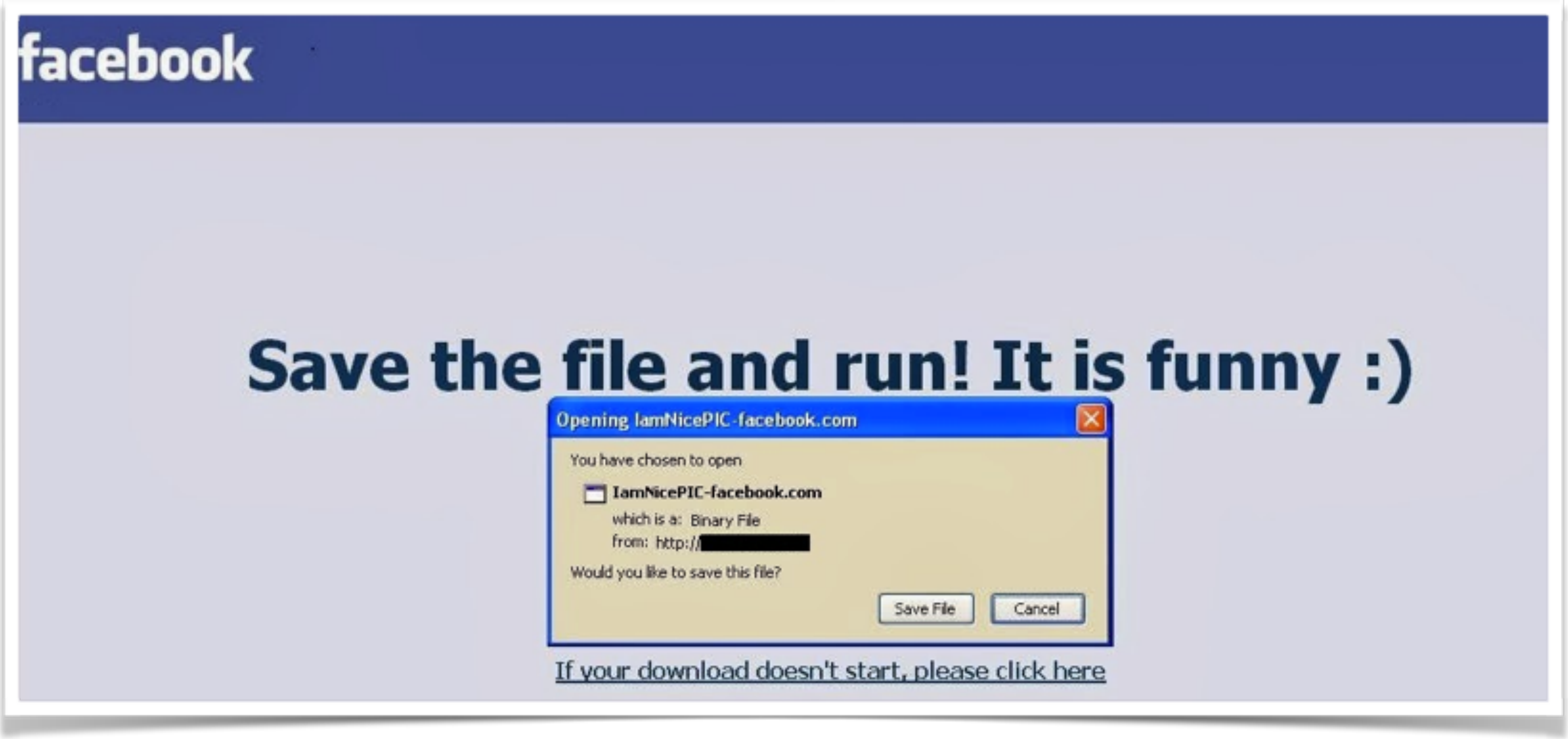
Malware Analysis Free Toolbox



Save the file and run! It is funny :)

Opening IamNicePIC-facebook.com

You have chosen to open

IamNicePIC-facebook.com

which is a: Binary File

from: http://[redacted]

Would you like to save this file?

If your download doesn't start, please click here

RMLL - Montpellier - July 2014 - Xavier Mertens

$ whoami

- Xavier Mertens (@xme)

- Consultant @ day

TRUESEC

- Blogger, Hacker @ night

/dev/random

Can't sleep, hackers will eat me!

- BruCON co-organizer

HACKING FOR B33R

BRUCON

$ cat ~/.profile

- I like (your) data
- Offensive / defensive security
- Security visualization
- I like to play!

$ cat disclaimer.txt

“The opinions expressed in this presentation are those of the speaker and do not necessarily reflect those of past, present employers, partners or customers.”

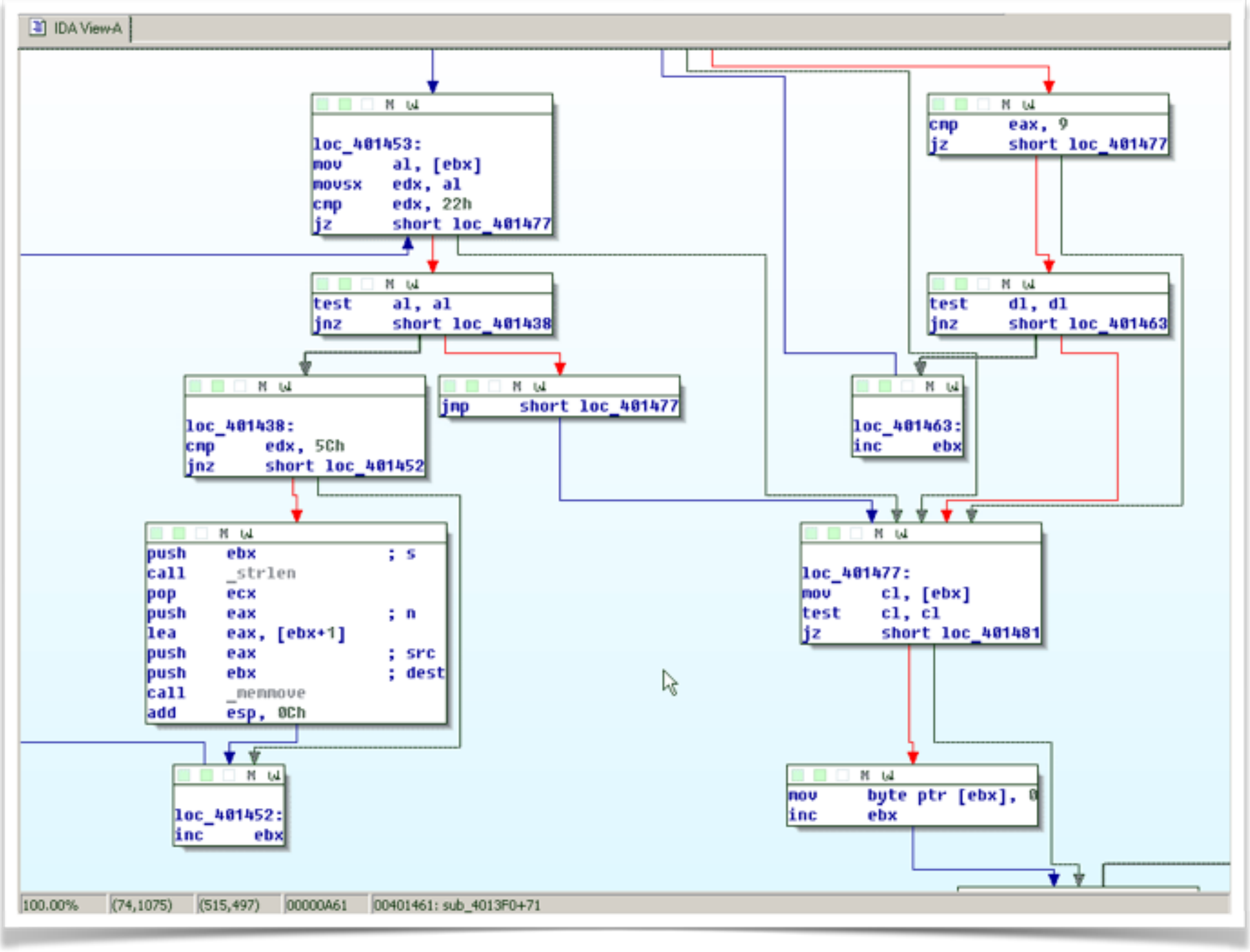
Agenda

- Introduction
- Build your lab
- Automate
- Conclusions

Why This Talk?

KEEP
CALM
AND
KNOW YOUR
ENEMY

Don’t expect this!



```
loc_401453:
mov     al, [ebx]
movsx   edx, al
cmp     edx, 22h
jz      short loc_401477
```

```
test    al, al
jnz     short loc_401438
```

```
loc_401438:
cmp     edx, 5Ch
jnz     short loc_401452
```

```
jmp     short loc_401477
```

```
push    ebx             ; s
call    _strlen
pop     ecx
push    eax             ; n
lea     eax, [ebx+1]
push    eax             ; src
push    ebx             ; dest
call    _memmove
add     esp, 0Ch
```

```
loc_401452:
inc     ebx
```

```
cmp     eax, 9
jz      short loc_401477
```

```
test    dl, dl
jnz     short loc_401463
```

```
loc_401463:
inc     ebx
```

```
loc_401477:
mov     cl, [ebx]
test    cl, cl
jz      short loc_401481
```

```
mov     byte ptr [ebx], 0
inc     ebx
```

Today’s Facts

29.122.849

unique malicious objects: scripts, web pages, exploits, executable files, etc.

81.736.783

unique URLs were recognized as malicious by web antivirus.

Q1 2014

(source: Kaspersky Security Network)

Sources

- My spam folder (rootshell.be has been registered in 2001)
- Torrents (Keygens)
- P0rn sites
- You & me!

Motivations

- Plenty of material
- To improve our security (integration with other tools)
- Because I’m lazy! (automation)
- Because it’s fun!

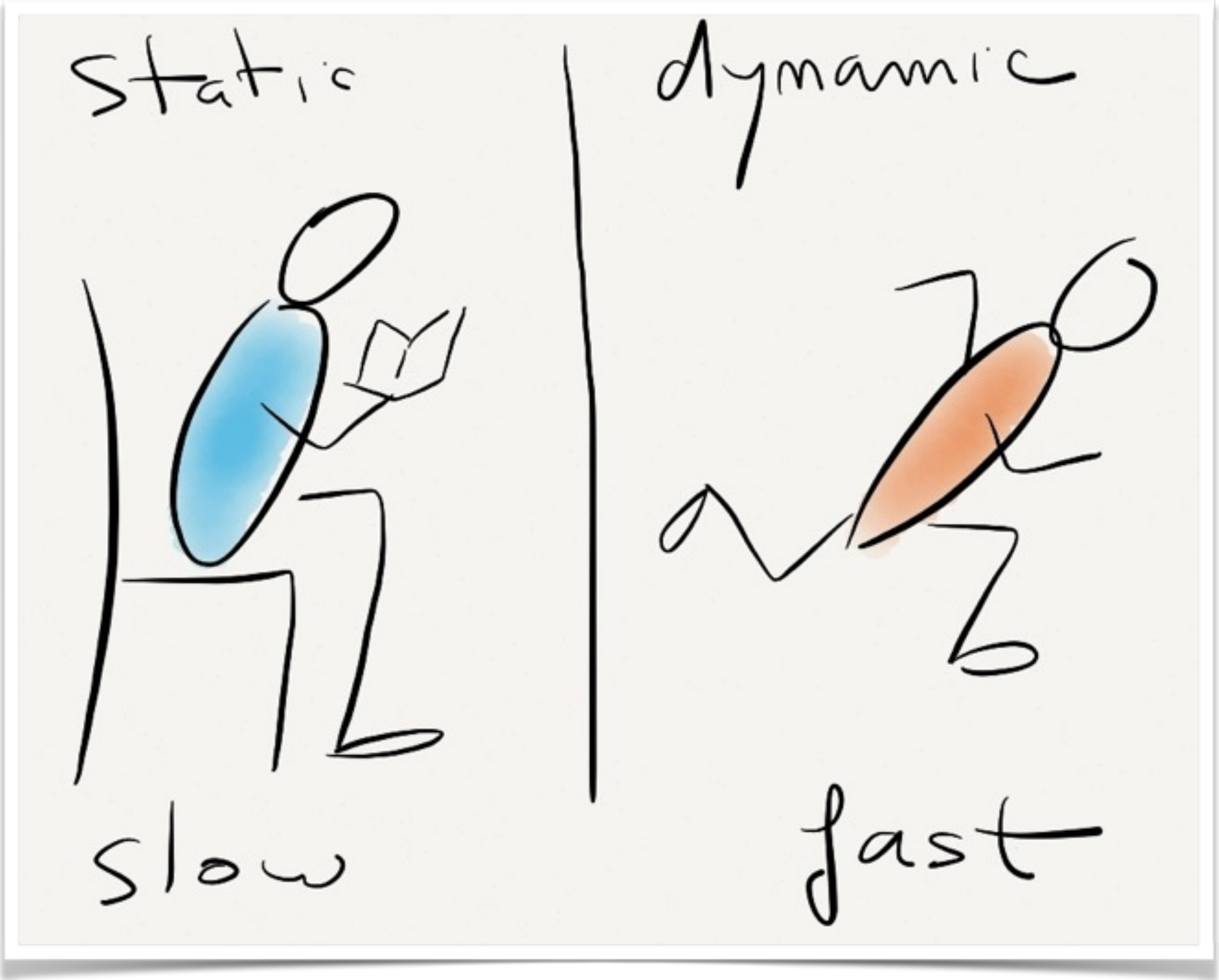
The Attack Vector

“APT”

vs

“BPT”

Analysis



Be Dynamic

- Execute the malware in a safe environment and watch what it does
- Goals
  - Understand how malwares work
  - Get IOC's

We Need “IOC”!

We Need "IOC"!

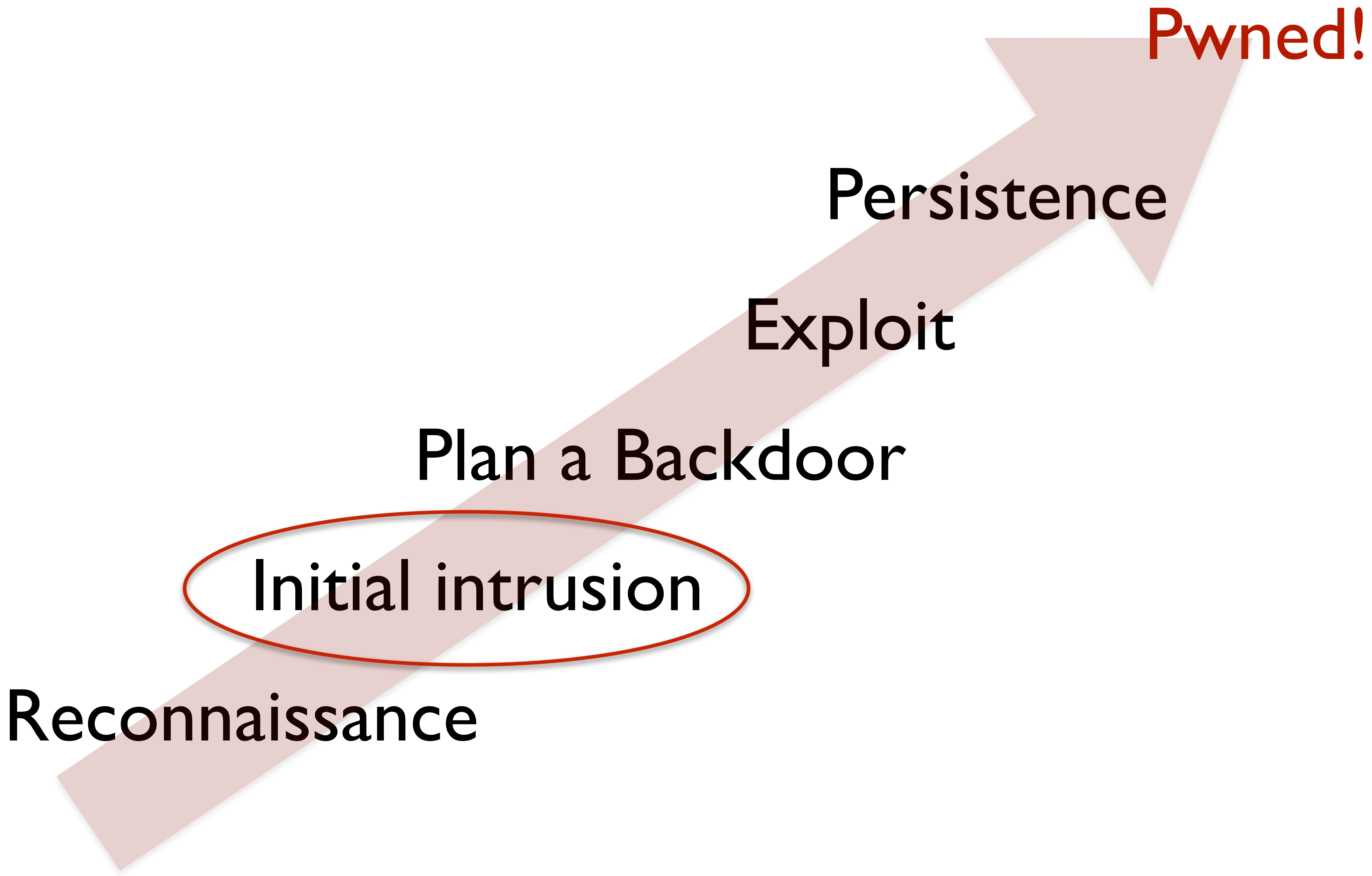
- Hashes
- IP addresses
- Domain names
- Files
- Registry keys
- URLs

Share!

Today's Market

- A niche market
- Big players
  (read: $$$)
- Integrated into an existing platform
  (Many 2.0 or NG firewalls)

An Attack in 5 Steps



Pwned!

Persistence

Exploit

Plan a Backdoor

Initial intrusion

Reconnaissance

The Patient "0"

The index case or primary case is the initial patient in the population of an epidemiological investigation

(Source: Wikipedia)

Agenda

- Introduction
- Build your lab
- Automate
- Conclusions

Requirements

- Free (because we are @ RMLL!)
- Virtualized (easy & snapshots)
- Open (to interconnect with other tools)
- Automatization

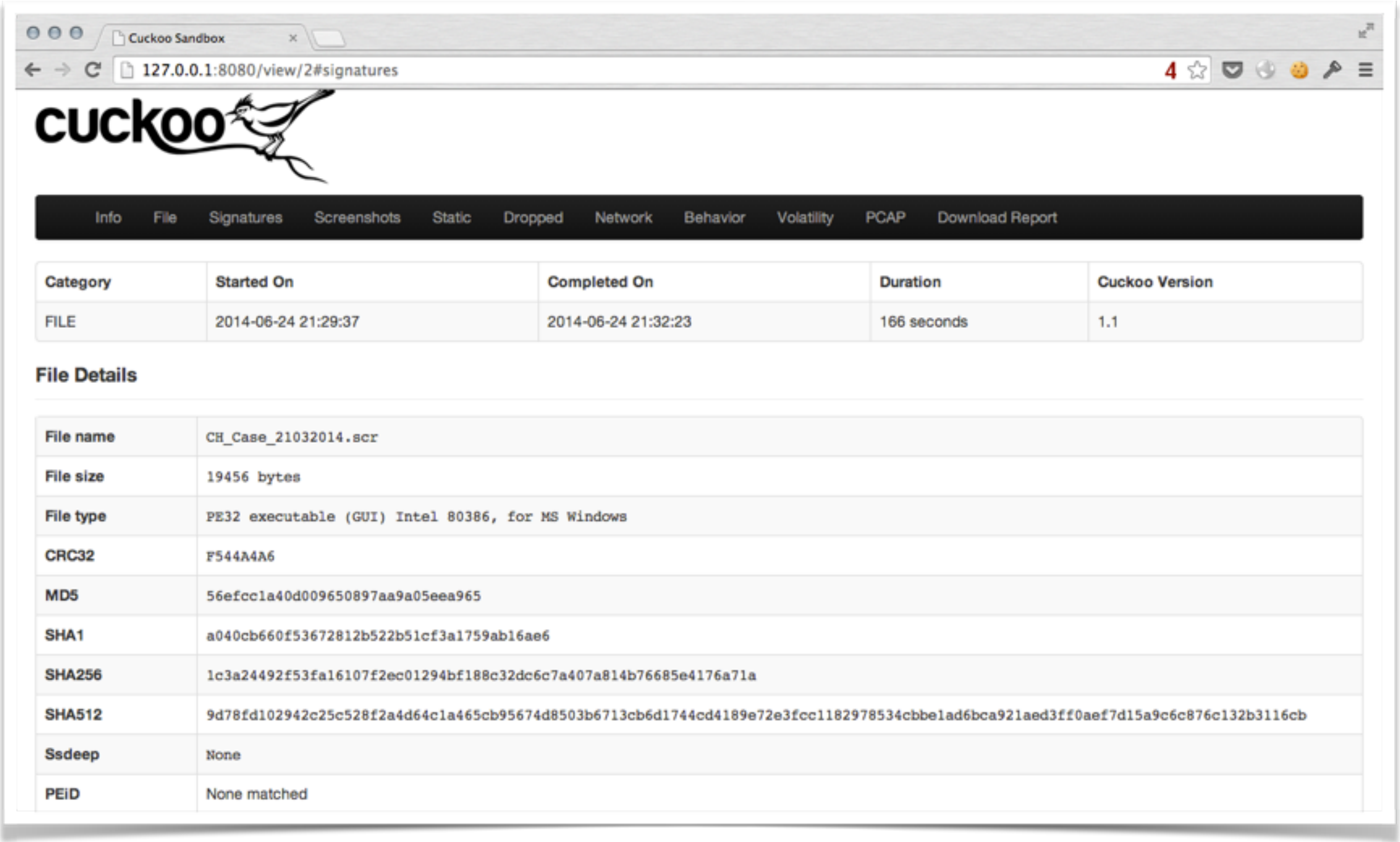
Cuckoo

- Dynamic code analysis framework developed in Python
- “Python” means “open, modular, easy to modify”
- Based on the classic “sandboxing” system

Features

- Automation
- Capture data
  - API calls
  - Network traffic
  - Screenshots
  - Filesystem / Registry operations
  - Memory dump
- Reporting in many formats

Cuckoo

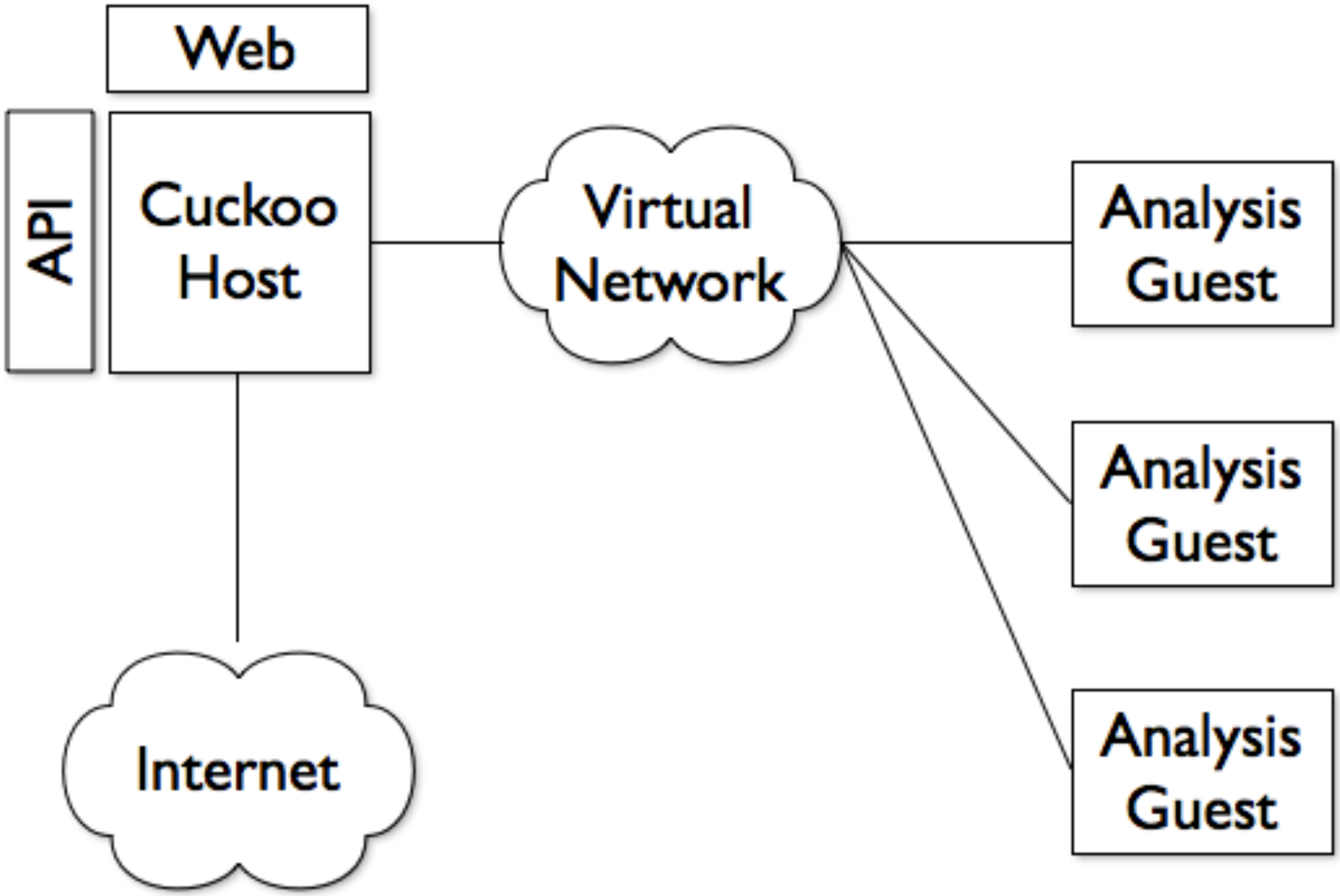


Category	Started On	Completed On	Duration	Cuckoo Version
FILE	2014-06-24 21:29:37	2014-06-24 21:32:23	166 seconds	1.1

File Details

File name	CH_Case_21032014.scr
File size	19456 bytes
File type	PE32 executable (GUI) Intel 80386, for MS Windows
CRC32	F544A4A6
MD5	56efcc1a40d009650897aa9a05eea965
SHA1	a040cb660f53672812b522b51cf3a1759ab16ae6
SHA256	1c3a24492f53fa16107f2ec01294bf188c32dc6c7a407a814b76685e4176a71a
SHA512	9d78fd102942c25c528f2a4d64c1a465cb95674d8503b6713cb6d1744cd4189e72e3fcc1182978534cbbe1ad6bca921aed3ff0aef7d15a9c6c876c132b3116cb
Ssdeep	None
PEiD	None matched

Architecture



Setup

Basic Installation

- VirtualBox (recommended)
- Lot of Python lib dependencies
- Recommended platform: Ubuntu
- Ninja mode: OSX

We Need Intertubes

- Use Host-only networking with Virtualbox
- Connect to the world

```
# sysctl -w net.ipv4.ip_forward=1
# iptables -A FORWARD -o eth0 -i vboxnet0 \
-s 192.168.1.0/24 -m conntrack -ctstate NEW \
-j ACCEPT
# iptables -A FORWARD -m conntrack \
-ctstate ESTABLISHED,RELATED -j ACCEPT
# iptables -A POSTROUTING -t nat -j MASQUERADE
```

OSX Ninja? Visit http://goo.gl/aEM7gO

“Your” Sandbox

- Windows XP SP3 or Windows 7 SP1 32bits
- Acrobat Reader, M$ Office, Browsers
- Generate some content (cookies, browsers history)
- Install the Cuckoo agent
- Disable all security features!

VM Hardening

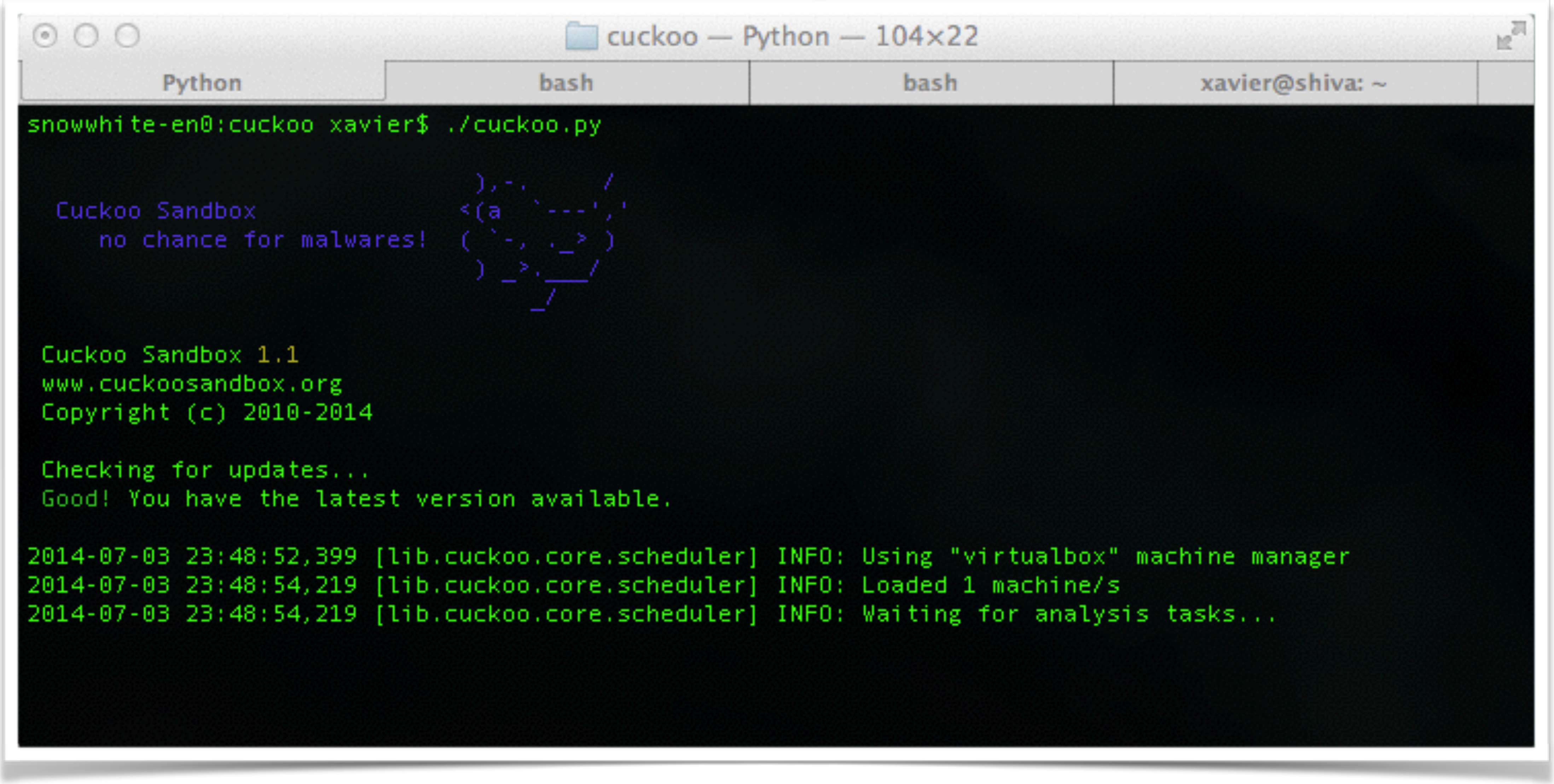
- VM must be “vulnerable” but hardened against anti-VM detection
- http://github.com/markedoe/cuckoo-sandbox
- https://github.com/a0rtega/pafish

Info

Running inside Virtual PC!

Attack of the Clones

Demo!



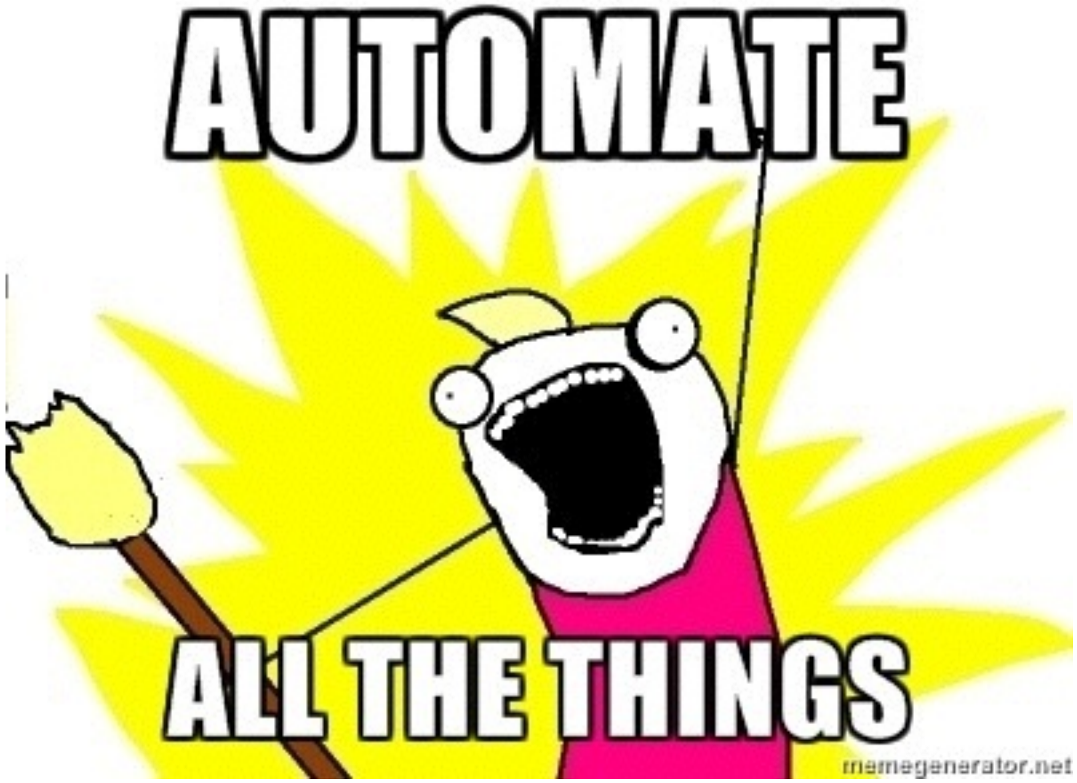
```
snowwhite-en0:cuckoo xavier$ ./cuckoo.py

                               ),-.     /
  Cuckoo Sandbox              <(a  `---','
     no chance for malwares!  ( `-, ._> )
                               ) _>.___/
                                  _/

 Cuckoo Sandbox 1.1
 www.cuckoosandbox.org
 Copyright (c) 2010-2014

 Checking for updates...
 Good! You have the latest version available.

2014-07-03 23:48:52,399 [lib.cuckoo.core.scheduler] INFO: Using "virtualbox" machine manager
2014-07-03 23:48:54,219 [lib.cuckoo.core.scheduler] INFO: Loaded 1 machine/s
2014-07-03 23:48:54,219 [lib.cuckoo.core.scheduler] INFO: Waiting for analysis tasks...
```

Agenda

- Introduction
- Build your lab
- Automate
- Conclusions

Automation

Cuckoo is a nice tool to analyse files on demand but some automation will be helpful to detect more suspicious stuff!



Bro IDS

- Bro is a powerful network analysis framework. Bro is not only a IDS
- Bro comes with analysers for many protocols which allow processing at layer-7
- http://bro.org

Bro Scripting

Bro has a simple and powerful scripting language. All the output generated by Bro is based on scripts!

Extract Those Files!

- Bro can extract files from network streams and save them on the file system
- There is an “extraction” analyzer to perform this task

Extract Those Files!

```
global ext_map: table[string] of string = {
    ["application/x-dosexec"] = "exe",
} &default ="";

event file_new(f: fa_file) {
    local ext = “data";

    if ( f?$mime_type )
        ext = ext_map[f$mime_type];

    local fname = fmt("%s-%s.%s",
                      f$source, f$id, ext);
    Files::add_analyzer(f, Files::ANALYZER_EXTRACT,
                        [$extract_filename=fname]);
}
```

Juicy Files

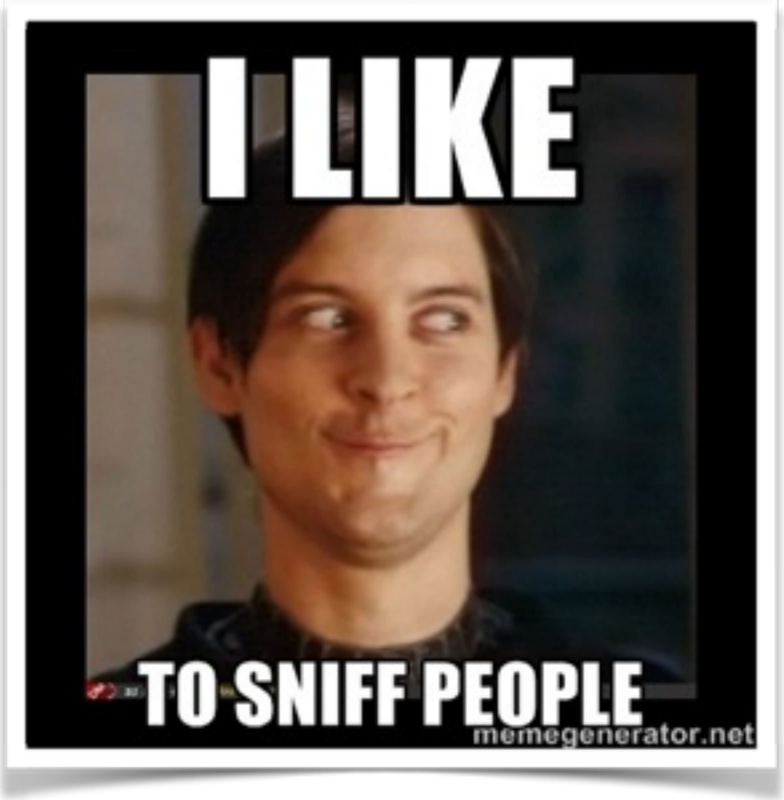
```
application/x-dosexec
application/vnc.ms-cab-compressed
application/pdf
application/x-shockware-flash
application/x-java-applet
application/jar
application/zip
```

And URLs?

- Extracting URLs from network?
- Flood! ("HTTP is the new TCP")
- Analysing one-time URLs may break some tools (think about password recovery)

Sniff!

```
# cd /tools/bro/logs
# vi extract.bro
# mkdir extract_files
# ../bin/bro -i eth1 extract.bro
listening on eth1, capture length 8192 bytes
```

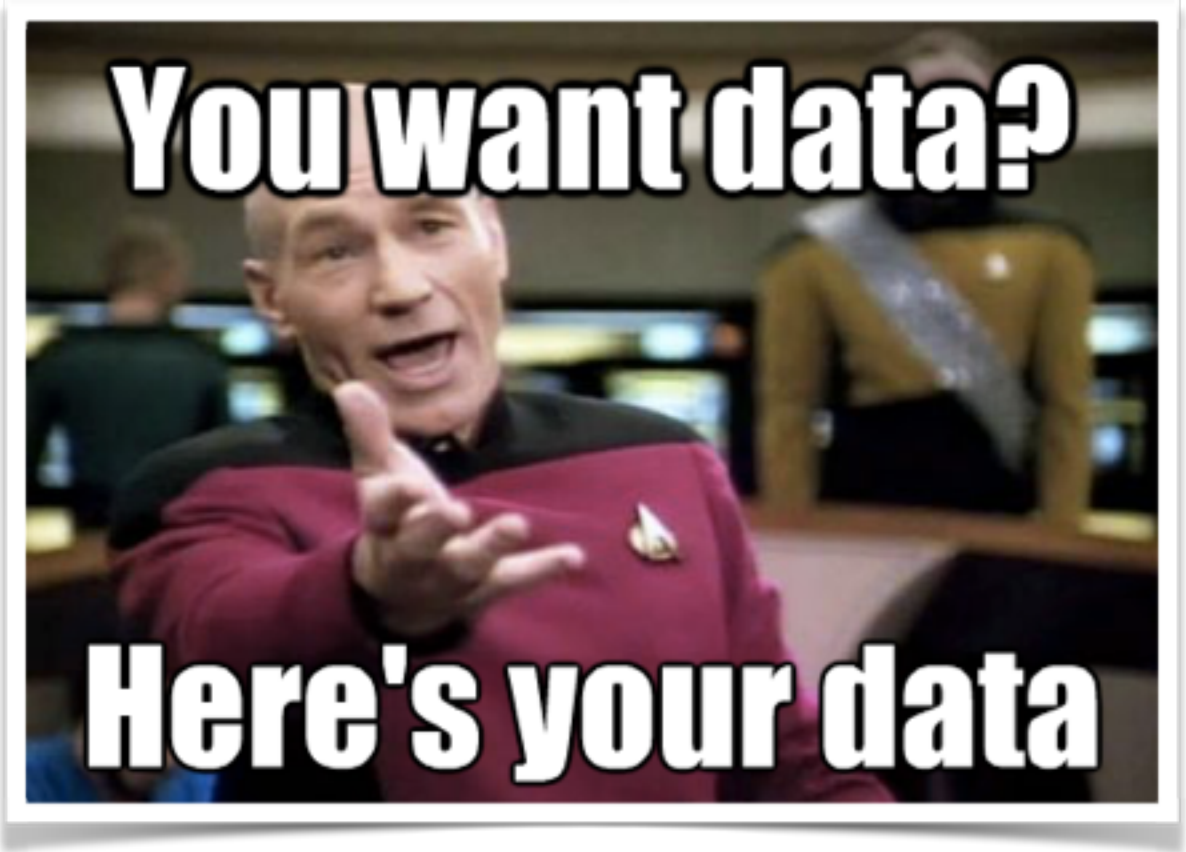


Feed Cuckoo!

```
# cd /tools/bro/logs/extract_files
# inotifywait -m -q -e create –format %f . |
while read F
do
  case “${F##*.}” in
    “zip|exe|doc|dll|jar|msi”)
      /tools/cuckoo/utils/submit.py $F
  esac
done
```

Want Data?

- Cuckoo has a REST API
- Useful to automate even more



Get results!

```
# curl http://localhost:8090/tasks/list
# curl http://localhost:8090/tasks/view/10
# curl http://localhost:8090/tasks/report/10
# curl http://localhost:8090/files/view/md5/xxxxxx
```

Extract IOC’s

```
#curl -s http://localhost:8090/tasks/report/2/json | \
python extract-domains.py
premiercrufinewine.co.uk 188.65.114.122
fidaintel.com 216.224.182.75
```

Feed OSSEC

- Create CDB lists (“active lists”)

```
<ossec_config>
  <rules>
    <list>lists/baddomains.cdb</list>
    <list>lists/badips.cdb</list>
  </rules>
</ossec_config>
```

- Populate them
- Re-generate them

```
/var/ossec/bin/ossec-makelists
```

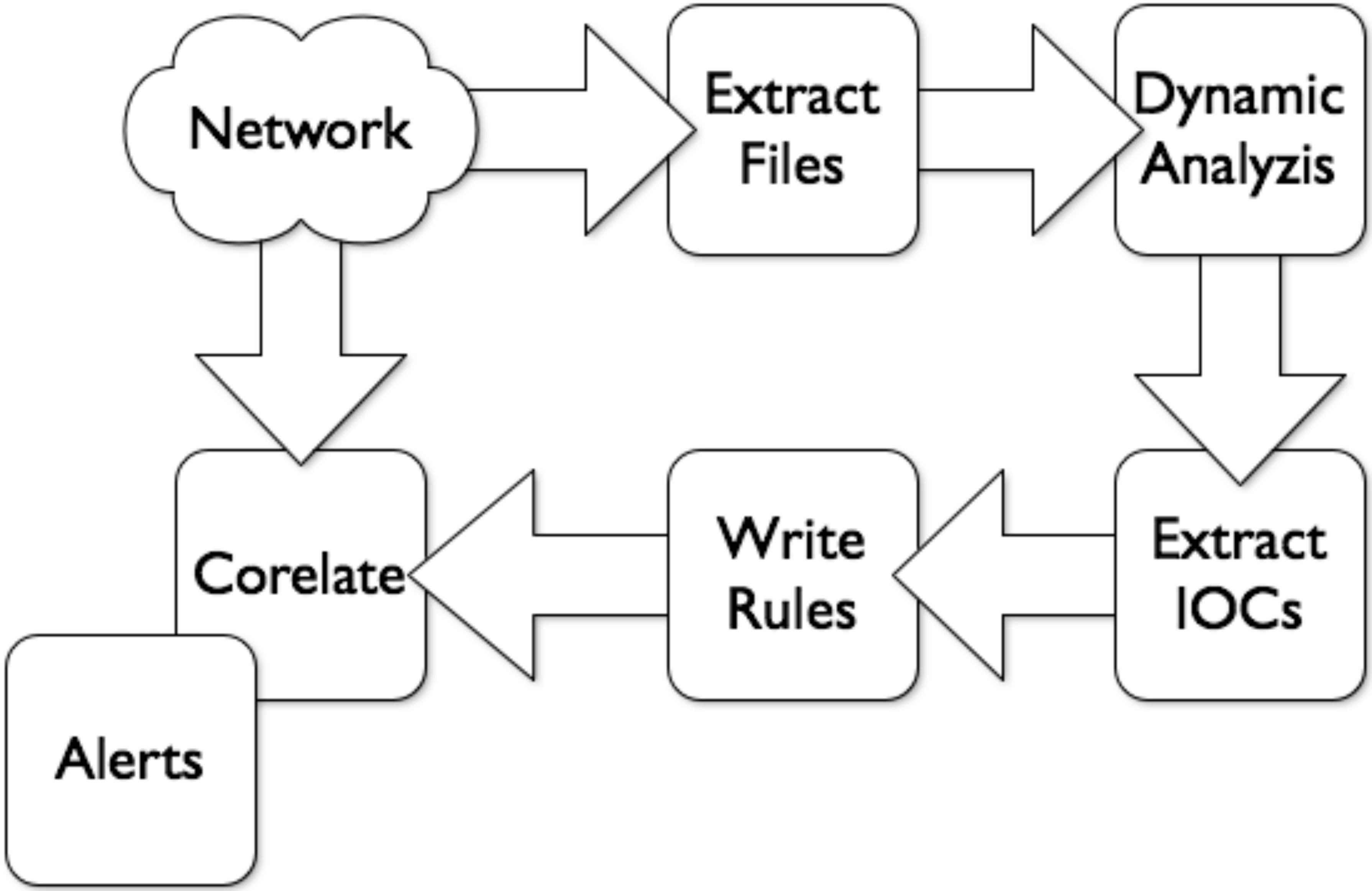
Correlate

```
<rule id=“99001” level=“10”>
  <decoded_as>bind9</decoded_as>
  <list field=“url”>lists/baddomains</list>
  <description>DNS query: malicious domain</description>
</rule>
```

Agenda

- Introduction
- Build your lab
- Automate
- Conclusions

Conclusions



Conclusions

- We don’t have time to handle such amount of data!
- Know your Enemy!
- Correlate your logs with external content

Thank you!

@xme

xavier@truesec.be

http://blog.rootshell.be

https://www.truesec.be