“WMAP: Metasploイト goes Web”
ET
Introduction

- No agenda in this presentation
- No Sun Wu Tzu “The Art of War” Stuff In this presentation either.
  - #nomorefree-art-of-war-stuff-in-security-presentations.
- No history of web scanners
• Efrain Torres
  – 2*5+ years enjoying IT security
  – Metasploit team
  – et [at] metasploit.com
- Web assessment as Metasploit auxiliary modules
  - Run modules by hand or automated
- Still early stages
  - blame it to the crisis
- Metasploit Prime (SecTor 08)
  - “Expect a big announcement soon!”, HD
    - This is it.
• Why?
  – Struggle with tools that find vulnerabilities while browsing an application.
    • Easy way to get detected by IDS/IPS
      – Crawl as a user.
      – Attack like a ninja later.
  – Suffering by tools that can only be run on windows environments.
    • Tied to a specific browser
    • IE Control. (You are assuming the target is good and no evil)
• Why?
  – Too much focus on whistles and bells.
    • When was the last time a pie chart was useful for you.
  – No more crap regarding vulnerabilities classification and risks.
    • Sometimes a High is just a Low
      – (I'm not talking about FP's)
    • And a couple of Lows can get you High
    • What the hell is a High, Medium, Low?
  – Real Impact?
• Objectives
  – A way to tie testing methods with exploitation methods
  – Make something useful to help in the assessment of anything related to HTTP/S
  – WMAP may be used as a scanner but it should be treated as an extension of the Metasploit framework.
• The more web assessment/scanners/etc tools the better.
  – Each tool has its own limitations, pros and cons.
  – Choose the damn tool you like (it’s just a tool)
  – “A Poor Workman Blames His Tools”
• W3AF is awesome.
• SQLmap is awesome.
• Multiple Clients
• Any Proxy
• Metasploit DB
• Simple modules
• Ruby
• db_wmap
  – Identify possible targets
  – Build target website structure
  – Run the modules in a specific order
  – Reporting

msf > load db_wmap
[ * ] = [ WMAP v0.5 - ET LoWNOISE
[ * ] Successfully loaded plugin: db_wmap

msf > db_create /path/to_db
msf > db_connect /path/to_db
db_wmap commands

msf > help

Wmap Database Backend Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>wmap_reports</td>
<td>List all reported results</td>
</tr>
<tr>
<td>wmap_run</td>
<td>Automatically test/exploit everything</td>
</tr>
<tr>
<td>wmap_targets</td>
<td>List all targets in the database</td>
</tr>
<tr>
<td>wmap_website</td>
<td>List website structure</td>
</tr>
</tbody>
</table>
- /modules/auxiliary/scanner/http
  - Discovery / Information gathering
  - Files and Directories
  - SQL Injection
  - XPATH Injection
  - Webservices
  - Webdav
  - ...
• WMAPModules Types
  – /lib/msf/core/auxiliary/wmapmodule.rb
    • Module Types and Basic supporting methods
  – WMAPScanServer
  – WMAPScanFile
  – WMAPScanDir
  – WMAPScanQuery
  – WMAPScanUniqueQuery
  – WMAPScanBody
  – WMAPScanHeaders
  – WMAPScanGeneric
    • wmap_generic_email_extract.rb
• How to build a WMAP module
  – 1) Build/Take a Metasploit module
    – Not only auxiliary modules
    – Not only HTTP
  – 2) Include the mixin type.
  – 3) Done!

```ruby
class Metasploit3 < Msf::Auxiliary
  include Msf::Exploit::Remote::HttpClient
  include Msf::Auxiliary::WMAPScanType
  include Msf::Auxiliary::Scanner

  ...

end
```
WMAP provides new building blocks for your cross-protocols attacks.
- Be creative

Remember each module behaves as a scanner
- Set RHOSTS 192.168.0.0/24
• SQL Injection
  – No reinventing the wheel
    • Wmap_sqlmap.rb
      – SQLmap by Bernardo Damele A. G
    – wmap_blind_sql_query.rb
      • Basic detect sql injections.
  – lucky_punch.rb
    • Same technique massive sql injection attacks in april/2008
    • Awesome companion for browser autopwn
• XPATH Injection
  – Xphat.rb
  – HTTP Blind XPATH 1.0 Injector
    • Simple search
    • substring() function = k
  – XPATH 2.0
    • Fast binary search
    • string-to-codepoints(string) function
    • and string-to-codepoints(substring()) < k

• SOAP
  – HTTP SOAP Verb/Noun Brute Force Scanner
  – Brute force SOAP/XML requests to uncover hidden methods
  – By patrick
• Files and Directories
  – wmap_backup_file.rb
  – wmap_brute_dirs.rb
  – wmap_copy_of_file.rb
  – wmap_dir_listing.rb
  – wmap_dir.Scanner.rb
  – wmap_files_dir.rb
  – wmap_file_same_name_dir.rb
  – wmap_prev_dir_same_name_file.rb
  – wmap_replace_ext.rb
  – writeable.rb (Put a nice metasploit payload)
• **Additional Server Modules**
  – `wmap_verb_auth_bypass.rb`
    • The old trick of bypassing authentication modifying the HTTP method.
  – `wmap_vhost.Scanner.rb`
    • Brute force vhost
      – NOTE: set VHOST vhost.target.com
  – `wmap_ssl.rb`
    • Easy way to pull vhost from a server(s)
    • Grab info from certificate.
Why use it

• Easy way to jump from web testing to exploitation methods.
  – Examples
    • Find files and directories that other scanners are not build to find
    • Jump from sql injection to XSS and back
      – Use a lucky punch with browser_autopwn
    • Use MSF payloads/file exploits to upload to web directory
      – MS09-XXX?

• If it runs metasploit, it runs wmap
Why use it

• Take results and feed them back to scan engine.
  – Use the proxy…
  – No more 1 round testing.
• Grab information from results and use them for other attacks
• The results are in the database, the database is the report
  – Wmap_report
    • -x xml
• It’s Metasploit.
• DEMO
Thanks

• Special thanks to HD, the Metasploit team and contributors.
• Questions?

et [at] metasploit.com