Fiddling with Fiddler
Testing Web Applications

Ron Woerner

NEbraskaCERT Conference 2006

The Weakest Link (technically)

*"When an organization puts up a web application, they invite the world to send them HTTP requests. Attacks buried in these requests sail past firewalls, filters, platform hardening, and intrusion detection systems without notice… **This means that your web application code is part of your security perimeter.** As the number, size and complexity of your web applications increases, so does your perimeter exposure."* (OWASP Top Ten, 2004).

The Weakest Link (technically)

- 8 out of 10 Web sites (applications) have at least one serious vulnerability
- Today, over 70% of attacks against a company's network come at the "Application Layer," not the Network or System Layer. [The Gartner Group]
- Web App customization causes greatest vulnerabilities.

Web App Vulnerabilities

- Three Categories
  - Coding Errors
  - Design Flaws
  - Application Configuration

OWASP Top 10 Web Programming Mistakes

1. Unvalidated Input
2. Broken Access Control
3. Broken Authentication & Session Management
4. Cross-Site Scripting (XSS)
5. Buffer Overflow
6. Command Injection Flaws
7. Improper Error Handling
8. Insecure Storage
9. Application Denial of Service
10. Insecure Configuration Management

http://www.owasp.org/index.php/OWASP_Top_Ten_Project

Web Testing Tools

There are three main classes of software security testing tools:

2. application scanning tools,
3. proxy-based tools, and
4. automated penetration-testing tools.

"Fancy tools aren't enough. Automated testing tools can't replace smart QA people. Just as attackers use tools and their own expertise, you need to combine tools and expertise to fight them." [Forrester View]

http://www.expresscomputeronline.com/20060306/management02.shtml

Web Testing Tools

- Fiddler
- WebScarab
- Tamper IE
- Bayden IEToys
- Internet Explorer Developer Toolbar
- Sandboxes / Playgrounds
  - HTTP Sandbox (http://www.bayden.com/sandbox/)
  - HTTPS Sandbox (https://www.fiddlertool.com/sandbox/)
  - WebGoat
- Manual Techniques

Fiddler

- “Unofficial” Microsoft
- Available at http://www.fiddlertool.com/fiddler/
- Fiddler is a HTTP Debugging Proxy which logs all HTTP traffic between your computer and the Internet. Fiddler allows you to inspect all HTTP Traffic, set breakpoints, and "fiddle" with incoming or outgoing data.

WebScarab

- Available from OWASP
  http://www.owasp.org/index.php/OWASP_WebScarab_Project
- Features
  - Proxy - observes traffic between the browser and the web server.
  - Manual intercept - allows the user to modify HTTP and HTTPS requests and responses on the fly, before they reach the server or browser.
  - Reveal hidden fields - changes all hidden fields found in HTML pages to text fields, making them visible, and editable
  - Manual request - Allows editing and replay of previous requests, or creation of entirely new requests

Tamper IE

- Bayden Systems (http://www.bayden.com/other/)
- An IE plugin that allows to intercept POSTs and GETs before they occur and gives you a chance to tweak it (change/add POST data)

Bayden IEToys

- Bayden Systems (http://www.bayden.com/other/)
- Many toys
  - Cleanup
  - Dictionary, Encyclopedia, & Google lookup
  - HMTL Source
  - IE7 Clear Tracks
  - Linkify

Internet Explorer Developer Toolbar

- Available from Microsoft (http://www.microsoft.com/downloads/details.aspx?familyid=e59c3964-672d-4511-bb3e-2d5e1db91038&displaylang=en)

- Locate and select specific elements on a Web page through a variety of techniques.
- Selectively disable Internet Explorer settings.
- View HTML object class names, ID's, and details such as link paths, tab index values, and access keys.
- Outline tables, table cells, images, or selected tags.
- Validate HTML, CSS, WAI, and RSS Web feed links.
- Display image dimensions, file sizes, path information, and alternate (ALT) text.
- Immediately resize the browser window to a new resolution.
- Selectively clear the browser cache and saved cookies.

Messing Around / Manual techniques

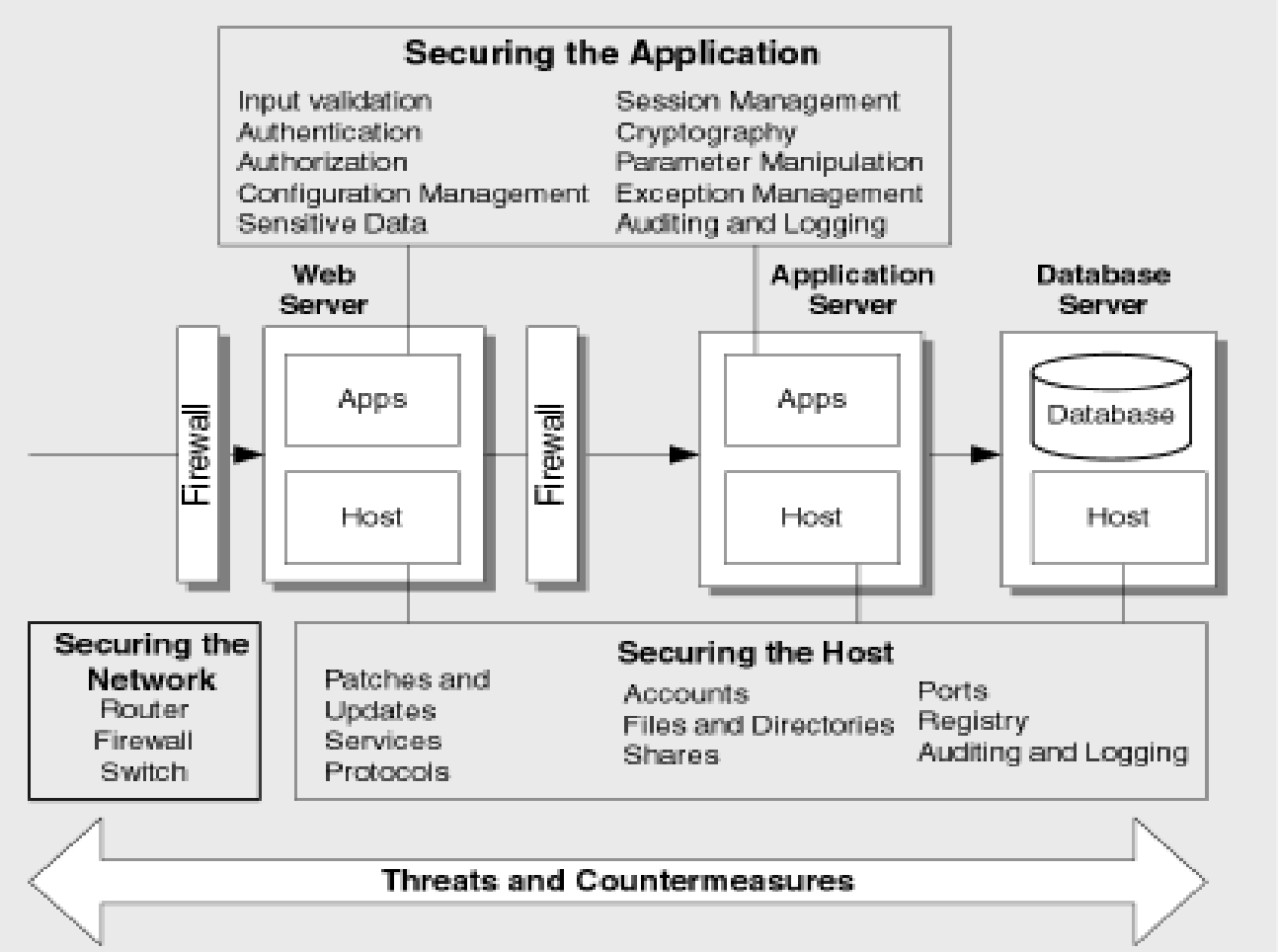
- URL manipulation
  - Directory Traversal
  - Predictable Resource Location
- Login tampering
- Error message handling
- Cookies and cached pages

Automated Tools

Application scanning tools include:

- The open source Nikto project (http://www.cirt.net/code/nikto.shtml),
- Application Security's AppDetective,
- Cenzic's Hailstorm,
- SPI Dynamics' WebInspect,
- Watchfire's AppScan,
- Metasploit

Stop the Insanity



Stop the Insanity

- Educate / Coach Developers
  - NEVER TRUST CLIENT-SIDE DATA!
  - Sanity check all input for information you are expecting to receive
  - Escape all input special characters
  - Don’t use hidden fields unless it’s absolutely necessary
  - Protect user passwords
  - Error handling
  - Perform internal code reviews or "buddy checks"
- Security **must** be in the SDLC –
  Up Front & Early and throughout the lifecycle

Stop the Insanity

- Web Application Firewalls

"An intermediary device, sitting between a web-client and a web server, analyzing OSI Layer-7 messages for violations in the programmed security policy. A web application firewall is used as a security device protecting the web server from attack."
- Web Application Security Consortium Glossary

Web Application Firewall Evaluation Criteria, Web Application Security Consortium, January 14, 2006 (http://www.webappsec.org/projects/waf_evaluation/)

Resources

- Microsoft: Improving Web Application Security: Threats and Countermeasures (http://msdn.microsoft.com/library/en-us/dnnetsec/html/ThreatCounter.asp)
- OWASP (http://www.owasp.org)
- Web Application Security Consortium (http://www.webappsec.org/)
- Secure Programming.com (http://www.secureprogramming.com/)
- NoticeBored SDLC Resources (http://www.noticebored.com/html/SDLC.html)