Web 2.0 Attacks - Next Generation Threats on the Rise

Shreeraj Shah
Founder & Director, Blueinfy
shreeraj@blueinfy.com
+91-9879027018

Shreeraj Shah
Founder & Director, Blueinfy
shreeraj@blueinfy.com
+91-9879027018

The OWASP Foundation
http://www.owasp.org

Shreeraj Shah

- **Founder & Director**
  - Blueinfy Solutions Pvt. Ltd. (Brief)
  - SecurityExposure.com

- **Past experience**
  - Net Square, Chase, IBM & Foundstone

- **Interest**
  - Web security research

- **Published research**
  - Articles / Papers - Securityfocus, O’erilly, DevX, InformIT etc.
  - Tools - wsScanner, scanweb2.0, AppMap, AppCodeScan, AppPrint etc.
  - Advisories - .Net, Java servers etc.

- **Books (Author)**
  - Web 2.0 Security - Defending Ajax, RIA and SOA
  - Hacking Web Services
  - Web Hacking

http://shreeraj.blogspot.com
shreeraj@blueinfy.com
http://www.blueinfy.com
Web 2.0 Case - Eye Opening Findings

- Web 2.0 Portal – Buy / Sell
- Technologies & Components - Dojo, Ajax, XML Services, Blog, Widgets
- Scan with tools/products **failed**
- Security issues and hacks
  - SQL injection over XML
  - Ajax driven XSS
  - Several XSS with Blog component
  - Several information leaks through JSON fuzzing
  - CSRF on both XML and JS-Array

» HACKED
» DEFENSE

Agenda

- Web 2.0 – What is Going On?
- Web 2.0 Security & Impact
- Web 2.0 Attacks
- Defending Web 2.0
- Web 2.0 Methodology & Attacks
- Web Services & SOA Threats & Scan
Questions for Next Generation Applications?

- Where is your business logic resides? - Client side...
- Your Web 2.0 framework running on JavaScript, Flash/Flex or Silverlight is secure or not?
- Why are you moving from 1.0 to 2.0?
- Your feeds are secure or not?
- How much data you are using in your app from un-trusted sources?
- Are your end user secure against attacks on your application?

Where are we moving?

- 80% of companies are investing in Web Services as part of their Web 2.0 initiative (McKinsey 2007 Global Survey)
- By the end of 2007, 30 percent of large companies have some kind of Web 2.0-based business initiative up and running (Gartner)
- 2008. Web Services or Service-Oriented Architecture (SOA) would surge ahead. (Gartner)
- Several applications are moving towards Web 2.0 and India is not an exception.
- Off shore development for Web 2.0 on the rise and secure coding around Web 2.0 application is very important.
Web 2.0 Technology Perspective

- It is combination of technology - Ajax, RIA, SOA, REST etc...
- Internet - Network of Networks
- Web 2.0 - Application of Applications
- Internet itself is becoming a platform and applications are emerging as objects residing on it and building a large distributed framework
- Google, Yahoo, eBay, Amazon etc. are providing APIs for access - boosting
Web 2.0 Architecture

Web 2.0 Components
Web 2.0 Samples

Enterprise 2.0

- Web 2.0 is not restricted to just social platform
- Penetrating into corporate
- Known as Enterprise 2.0
- Old generation applications are changing across companies
Enterprise 2.0

- Enterprise mashup
  - Enterprise runs services in its own mashup
  - Web-based enterprise solutions
  - $700 million industry (2013) - Forrester
  - Databases get converted to RSS
  - Emerging strategies around 2.0
  - SOA mashups
  - Etrade, IBM, Wells Fargo – examples

Agenda

- Web 2.0 – What is Going On?
- Web 2.0 Security & Impact
- Web 2.0 Attacks
- Defending Web 2.0
- Web 2.0 Methodology & Attacks
- Web Services & SOA Threats & Scan
Impact of Web 2.0

Application Infrastructure

<table>
<thead>
<tr>
<th>Vector</th>
<th>Web 1.0</th>
<th>Web 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocols</td>
<td>HTTP &amp; HTTPS</td>
<td>SOAP, XML-RPC, REST etc. over HTTP &amp; HTTPS</td>
</tr>
<tr>
<td>Information structures</td>
<td>HTML transfer</td>
<td>XML, JSON, JS Objects etc.</td>
</tr>
<tr>
<td>Communication methods</td>
<td>Synchronous</td>
<td>Asynchronous &amp; Cross-domains (proxy)</td>
</tr>
<tr>
<td></td>
<td>Refresh and Redirect</td>
<td></td>
</tr>
<tr>
<td>Information sharing</td>
<td>Single place information (No</td>
<td>Multiple sources (Urge for integrated information platform)</td>
</tr>
<tr>
<td></td>
<td>urge for integration)</td>
<td></td>
</tr>
</tbody>
</table>
Impact of Web 2.0

Security Threats

<table>
<thead>
<tr>
<th>Vector</th>
<th>Web 1.0</th>
<th>Web 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry points</td>
<td>Structured</td>
<td>Scattered and multiple</td>
</tr>
<tr>
<td>Dependencies</td>
<td>Limited</td>
<td>Multiple technologies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Information sources</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Protocols</td>
</tr>
<tr>
<td>Vulnerabilities</td>
<td>Server side [Typical injections]</td>
<td>Web services [Payloads]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Client side [XSS &amp; XSRF]</td>
</tr>
<tr>
<td>Exploitation</td>
<td>Server side exploitation</td>
<td>Both server and client side exploitation</td>
</tr>
</tbody>
</table>

Methodology

<table>
<thead>
<tr>
<th>Vector</th>
<th>Web 1.0</th>
<th>Web 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Footprinting</td>
<td>Typical with &quot;Host&quot; and DNS</td>
<td>Empowered with search</td>
</tr>
<tr>
<td>Discovery</td>
<td>Simple</td>
<td>Difficult with hidden calls</td>
</tr>
<tr>
<td>Enumeration</td>
<td>Structured</td>
<td>Several streams</td>
</tr>
<tr>
<td>Scanning</td>
<td>Structured and simple</td>
<td>Difficult with extensive Ajax</td>
</tr>
<tr>
<td>Automated attacks</td>
<td>Easy after discovery</td>
<td>Difficult with Ajax and web services</td>
</tr>
<tr>
<td>Reverse engineering</td>
<td>On the server-side [Difficult]</td>
<td>Client-side with Ajax &amp; Flash</td>
</tr>
<tr>
<td>Code reviews</td>
<td>Focus on server-side only</td>
<td>Client-side analysis needed</td>
</tr>
</tbody>
</table>
### Impact of Web 2.0

- **Countermeasure**

<table>
<thead>
<tr>
<th>Vector</th>
<th>Web 1.0</th>
<th>Web 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner of information</td>
<td>Single place</td>
<td>Multiple places [Mashups &amp; RSS]</td>
</tr>
<tr>
<td>Browser security</td>
<td>Simple DOM usage</td>
<td>Complex DOM usage</td>
</tr>
<tr>
<td>Validations</td>
<td>Server side</td>
<td>Client side [incoming content]</td>
</tr>
<tr>
<td>Logic shift</td>
<td>Only on server</td>
<td>Client side shift</td>
</tr>
<tr>
<td>Secure coding</td>
<td>Structured and single place</td>
<td>Multiple places and scattered</td>
</tr>
</tbody>
</table>

### Web 2.0 Security

- Complex architecture and confusion with technologies
- Web 2.0 worms and viruses – Sammy, Yammaner & Spaceflash
- Ajax and JavaScripts - Client side attacks are on the rise
- Web Services attacks and exploitation
- Flash clients are running with risks
Web 2.0 Security

- Mashup and un-trusted sources
- RSS feeds manipulation and its integration
- Single Sign On and information convergence at one point
- Widgets and third-party components are bringing security concerns
- Old attacks with new carriers

Agenda

- Web 2.0 – What is Going On?
- Web 2.0 Security & Impact
- Web 2.0 Attacks
- Defending Web 2.0
- Web 2.0 Methodology & Attacks
- Web Services & SOA Threats & Scan
Fingerprinting

- Application fingerprinting - identifying web and application servers
- Ajax and RIA framework fingerprints
- Getting hold on to technologies - WebLogic or Tomcat, Atlas or Dojo
- Helps in assessment and mapping publicly known vulnerabilities
**Discovery**

- Ajax running with various different structures
- Developers are adding various different calls and methods for it
  - JSON, Array, JS-Object etc.
- JavaScript can talk with back-end sources
- Mashups application talking with various sources
- It has significant security impact
- Identifying and Discovery of structures

**Crawling & Enumeration for Web 2.0**

- Dynamic page creation through JavaScript using Ajax
- DOM events are managing the application layer
- DOM is having clear context
- Protocol driven crawling is not possible without loading page in the browser
Cross Site Scripting (XSS) – 2.0 Style

- What is different?
  - Ajax calls get the stream
  - Inject into current DOM using eval() or any other means
  - May rewrite content using document.write or innerHTML calls
  - Source of stream can be un-trusted
  - Cross Domain calls are very common

Addressing Cross Domain Calls

- Cross Domain calls are very important for Web 2.0 applications.
  - Proxy to talk with cross domain
  - Callback implementation to fetch them
  - Flash via crossdomain.xml

- These are types of bypass and can have security implications
- Source of the information – key!
**Scenario**

- **Blog** → **JSON feed**
- **Web Client** → **Web Server**
- **Vulnerable stream coming through proxy**
- **Hijack**
- **Posting to the site** [Malicious code]

**DDoS**

**XSS with RIA**

- Applications running with Flash components
- `getURL` - injection is possible
- SWFIntruder
- Flasm/Flare

(implode('缚', (http://www.nowrap.de/))

**Attack Configuration Window**

- Function to steal data
- `getURL` injection
- SWFIntruder
- Flasm/Flare

(iframe 'http://www.nowrap.de/')
Scanning for XSS

- Scanning Ajax components
- Retrieving all JS include files
  - Part of `<SCRIPT SRC=.....>`
- Identifying XHR calls
- Grabbing function
- Mapping function to DOM event
- Scanning code for XSS - look for `eval()` and `document.write()`

Ajax serialization issues

- Ajax processing various information coming from server and third party sources. - XSS opportunities

```javascript
message = {
  from : "john@example.com",
  to : "jerry@victim.com",
  subject : "I am fine",
  body : "Long message here",
  showsubject :
    function(){document.write(this.subject)}
};
```

JSON issues

```javascript
{"bookmarks":{{"Link":"www.example.com","Desc":"Interesting link"}}}
```

```javascript
new Array("Laptop", "Thinkpad", "T60", "Used", "900$", "It is great and I have used it for 2 years")
```
Countermeasures

- Client side code audit is required
- XHR calls and DOM utilization needs to be analyzed
- Content from un-trusted information sources should be filtered out at proxy layer
- Cross Domain Callback - careful
- Browser side content validation before consuming into DOM

Cross Site Request Forgery (CSRF)

- Generic CSRF is with GET / POST
- Forcefully sending request to the target application with cookie replay
- Leveraging tags like
  - IMG
  - SCRIPT
  - IFRAME
- Not abide by SOP or Cross Domain is possible
Cross Site Request Forgery (CSRF)

- What is different with Web 2.0
  - Is it possible to do CSRF to XML stream
  - How?
  - It will be POST hitting the XML processing resources like Web Services
- JSON CSRF is also possible
- Interesting check to make against application and Web 2.0 resources

One Way CSRF Scenario
One Way CSRF Scenario
One-Way CSRF Scenario

One-Way CSRF

- Splitting XML stream in the form
- Possible through XForms as well
- Similar techniques is applicable to JSON as well
Two-Way CSRF

- One-Way - Just making forceful request.
- Two-Way
  - Reading the data coming from the target
  - May be getting hold onto important information - profile, statements, numbers etc.
  - Is it possible with JSON/XML
Two-Way CSRF

- Application is serving various streams like JSON, JS-Object, Array etc.

- Attacker page can make cross domain request using SCRIPT (firefox)

- Following code can overload the array stream.

```javascript
function Array()
    { var obj = this; var index = 0; for(j=0;j<4;++j) obj[index++]
setter = spoof; } } function spoof(x) { send(x.toString()); }
```

Countermeasure

- Server Side Checks
  - Check for client’s content-type
  - XHR calls – xml/application
  - Native calls – text/html
  - Filtering is possible on it

- Client Side Checks
  - Stream can be started and terminated by /* or any predefined characters
  - Client can remove them before injecting to DOM
Web 2.0 Components

There are various other components for Web 2.0 Applications
- RSS feeds
- Mashups
- Widgets
- Blogs
- Flash based components

RSS feeds

- RSS feeds coming into application from various un-trusted sources
- Feed readers are part of 2.0 Applications.
- Vulnerable to XSS
- Malicious code can be executed on the browser.
- Several vulnerabilities reported
Mashups

- API exposure for Mashup supplier application
- Cross Domain access by callback may cause a security breach
- Confidential information sharing with Mashup application handling needs to be checked - storing password and sending it across (SSL)
- Mashup application can be man in the middle so can’t trust or must be trusted one

Widgets/Gadgets

- DOM sharing model can cause many security issues
- One widget can change information on another widget - possible
- CSRF injection through widget code
- Event hijacking is possible - Common DOM
- IFrame - for widget is a MUST
Blogs

- Blogs are common to Web 2.0 applications
- Many applications are plugging third party blogs
- One needs to check these blogs - XSS is common with blogging applications
- Exceptions and Search are common XSS points

Agenda

- Web 2.0 – What is Going On?
- Web 2.0 Security & Impact
- Web 2.0 Methodology & Attacks
- Web 2.0 Attacks
- Defending Web 2.0
- Web Services & SOA Threats & Scan
SOA Stack

- Presentation Stack
  - XML, JSON, JS-*
- Security Stack
  - WS-Security
- Discovery Stack
  - UDDI, DISCO
- Access Stack
  - WSDL, SOAP, XML-RPC, REST
- Transport Stack
  - HTTP, HTTPS

Primary Discovery

- Crawling the application and mapping file extensions and directory structures, like ".asmx"
- Page scrubbing - scanning for paths and resources in the pages, like atlas back end call to Web Services
- Recording traffic while browsing and spidering, look for XML based traffic - leads to XML-RPC, REST, SOAP, JSON calls
Secondary Discovery

- Searching UDDI server for Web Services running on particular domain
  - Three tactics for it – business, services or tModel
- Running queries against search engines like Google or MSN with extra directives like “inurl” or “filetype”
  - Look for “asmx”

wsScanner – Discovery!

Enumerating and Profiling

- Fingerprinting .Net framework and Client side technologies – Dojo or Atlas ...
- Scanning WSDL
  - Looking for Methods
  - Collecting In/Out parameters
  - Security implementations
  - Binding points
  - Method signature mapping
**Scanning strategies**

- Manual invocation and response analysis
- Dynamic proxy creation and scanning
- Auto auditing for various vectors
- Fuzzing Web Services streams – XML or JSON
- Response analysis is the key
  - Look for fault code nodes
  - Enumerating fault strings
  - Dissecting XML message and finding bits
  - Hidden error messages in JSON

**Agenda**

- Web 2.0 – What is Going On?
- Web 2.0 Security & Impact
- Web 2.0 Methodology & Attacks
- Web Services & SOA Threats & Scan
- Defending Web 2.0
Code Analysis for Web 2.0

- Scanning the code base
- Identifying linkages
- Method signatures and inputs
- Looking for various patterns for SQL, LDAP, XPATH, File access etc.
- Checking validation on them
- Code walking and tracing the base - Key

Content filtering with 2.0

- Regular firewall will not work
- Content filtering on HTTP will not work either since it is SOAP/JSON over HTTP/HTTPS
- SOAP/JSON level filtering and monitoring would require
- ISAPI level filtering is essential
- SOAP/JSON content filtering through IHTTPModule
HTTP Stack for .Net (IIS6/7)

HTTP Stack for .Net (IIS6/7)

Conclusion

- Web 2.0 bringing new challenges
- Needs to adopt new methodologies for scanning
- Attacks and entry points are scattered and multiple
- Ajax and SOA are key components
- WAF and Code review are important aspects for Web 2.0 defense
Question & Thanks!

Web 2.0 – What is Going On?

Web 2.0 Security & Impact

Web 2.0 Attacks

Defending Web 2.0

Web 2.0 Methodology & Attacks

Web Services & SOA Threats & Scan

http://shreeraj.blogspot.com
shreeraj@blueinfy.com
http://www.blueinfy.com