Ajax Security Concerns

OWASP KC Meeting

December 6, 2006

Rohini Sulatycki

What is Ajax?

- Ajax is an approach
- Grouping of technologies
  - JavaScript
  - XML
  - XMLHttpRequest
  - http://java.sun.com/developer/technicalArticles/J2EE/AJAX/
  - http://ajax.asp.net/Default.aspx

Is Ajax Insecure?

- Ajax in of itself is neither secure or insecure

Ajax Security Concerns

Increased complexity

- Need to understand several technologies
- Backend developers unfamiliar with client side coding and vice versa
- Difficult to address security in this complex technological mix

Increased attack area

- Points of inputs and outputs
- Ajax does not in of itself increase the attack area
- Ajax does make the surface area more difficult to secure
  - Use of automated tools to crawl sites more difficult

Client code more vulnerable

- Client code can be viewed
- Client code can be easily modified by an attacker using injection
- Developers need to understand JavaScript and DOM

Validation

- Typical Ajax implementations have significantly more client side validation
- Client code make calls to server business service layer
- Easy to by-pass client side validation
  - Use HHTP debugging proxy such as Fiddler
  - Modify request

Denial of Service

- Many small requests between client and server
- Heavy loads will exponentially increase number of requests to the server

Cross Site Scripting

- Still require attacker to inject malicious script
- XSS attacks more stealthy
  - No visual clues that session has been hijacked
- Ajax worm
  - Intercepts all user activity on a website
  - http://myappsecurity.blogspot.com/

Mashing

- Aggregate content from multiple domains
- Ajax security does not allow this
- Can use JSON (JavaScript object notation)
  - Serialized representation of a JS object
  - Browser makes call to JS function
  - JS function modifies head to do JS include
  - JS gets executed and makes a cross domain call with callback function
  - The callback function executes and updates the page

How to secure Ajax sites?

Validation

- Validate ALL inputs
- All client side validation **must** be backed up by server side validation
- Don’t implement business logic validation client side
- Implement whitelist validation
  - Identify valid data and reject everything else
- Encode all outputs

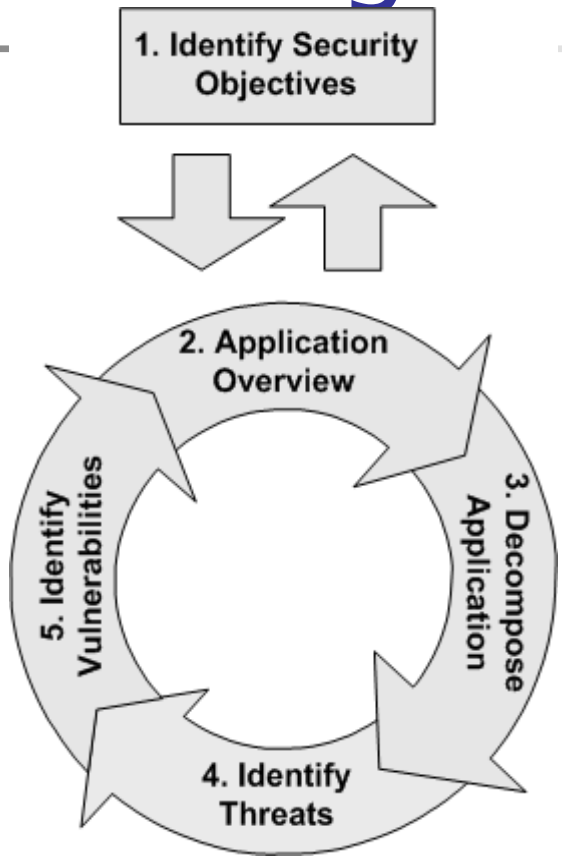
Use secure libraries

- Don’t re-invent the wheel
- Use tried and tested components such as Microsoft Atlas
  - Provide client scripts, sever controls and bridge

Integrate security in SDLC

- Requirements
  - Data classification
  - Functional boundaries
- Design
  - Misuse cases
  - Threat Modeling
  - Trust Boundaries
  - Session management
  - Cryptography
  - Exception handling
  - Auditing and Logging
- Implementation
- QA
  - Load testing

Threat Modeling



http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnpag2/html/tmwa.asp

If best practices are being followed then there is a good chance that you have a secure Ajax application