Intro to Security Testing

Michael Gegick
North Carolina State University
mcgegick@ncsu.edu

Security Testing: Testing for What It’s NOT supposed to do
Security Testing

- Security vulnerabilities can have a larger impact than traditional software bugs
  - Litigation
  - Brand
  - Regulatory requirements
- Functional security testing
- Risk-based security testing
- Penetration testing

Functional Security Testing

- Test the security mechanisms
  - Do they behave properly?
  - Same techniques as functional testing
    - Based on requirements
  - Example requirements
    - Disable user after three failed login attempts
    - All network traffic must be encrypted
- Can be done early in software process
- White hat
  - Defensive
**Risk-Based Security Testing**

- Know your threats
  - Abuse/Misuse cases
  - Architectural risk analysis
  - Attack patterns
- Test the software to make sure it does not misbehave
  - Performs an attack
- Let the risk analysis drive testing
- Can be done early in software process
  - Unit-level
- Black hat oriented
  - Destructive

**Injection Attacks**

- Largest problem in software security today
- Attackers inject malicious input into *input vectors*
  - The path into a software system
**Developer-View of Software System**

- Feature oriented
- All input will be valid
- Input will only be injected in anticipated input fields

**Attacker-View of Software System**

- Exploit assumptions made by developers
- Inject malicious input into input fields
- Inject malicious input into fields the developers assumed would receive no input
Penetration Testing

Test the environment/configuration
- Usually occurs when code is complete and in its operational environment
- Some configuration problems are easy/quick to fix
- Performed at the system/integration-level
- Modern penetration testing covers injection attacks
- Primarily black hat oriented

Penetration Testing (2)

- Use tools to automate the testing
- SPIDynamics
  - WebInspect
- Disadvantages
  - Penetrate-and-patch method
- Mostly black box testing
  - some white box testing
Testing Results

- When you have exhausted your test cases, then are you secure?

References

Exercise

- Identify technical risks of iTrust
- Ranks your risks
- Develop a test plan to test the risks
- What types of testing should you use and why?
- Indicate how you want to incorporate those tests in the software process
- Use http://cwe.mitre.org for information on weaknesses