Secure Software Development
Penetration Testing

Objectives

- Compare and contrast the artifacts and goals for QA testing and security testing
- Compare and contrast the types of testing used for security testing
- Explain the purpose for penetration testing and the activities performed during penetration testing
- Compare and contrast the mindset of a hacker and the mindset of a penetration tester
- Explain how penetration testing can be improved
- Explain what aspects need to be considered when planning penetration tests

Teaser

Brain Teaser

Discussion

- What do we test and how?

Provide input to functions and verify it matches

matches w/ what?

Driven by use cases and requirements

⇒ Strong emphasis on features and functions.

Security is not a set of features.

∴ Security does not fit model.

How is security testing versus security functionality

Defects from a security standpoint usually not related to S. functionality or SW. Vulnerabilites - end to end. Come from unexpected misuses of system.

Functional testing => testing for positives.

Security testing => testing for negatives.

Types of Testing

- White Box

⇒ Have access to internals (design / code / etc.)

- Black Box

Testing w/ out knowledge of system.

Definitions

- Vulnerability (Security Flaw):
  - Specific failure of the system to guard against unauthorized access or actions. It can be procedures, technology (SW or HW), or management.
- Exploiting a vulnerability
  - Using the failure of the system to violate the site security policy is called exploiting the vulnerability
- Penetration Study
  - is a test for evaluating the strengths of all security controls on the computer system. It intends to find all possible security holes and provides suggestions for fixing them.
- Penetration Testing
  - An authorized attempt to violate specific constraints stated in the form of a security or integrity policy.
  - a testing technique for discovering, understanding, and documenting all the security holes that can be found in a system.

- What is the difference between penetration testing and hacking/intrusion?

Penetration Testing

- Most common of software security best practices

- Testing performed with a mindset of trying to break into a deployed system
  - Requires complete, deployed system
  - Uncovers vulnerabilities, but may not aid in quantifying risk
  - "Badnessometer"
    - It is not a proof techniques. It can never prove the absence of security flaws. It can only prove their presence.

Be like the enemy.

We can't start early.

Goals of penetration Testing

- Gaining of read or write access to specific objects, files, or accounts;

⇒ Break in

- Gaining of specific privileges

⇒ Become root on a Unix machine

- Disruption or denial of the availability of objects.

⇒ Take the system offline.

Success Dependencies

Skill, knowledge, and experience of testers.

⇒ State of the practice tends to be ad hoc.

Results may/may not be reproducible between teams.

Requires access to real actual system.

Better Penetration Testing

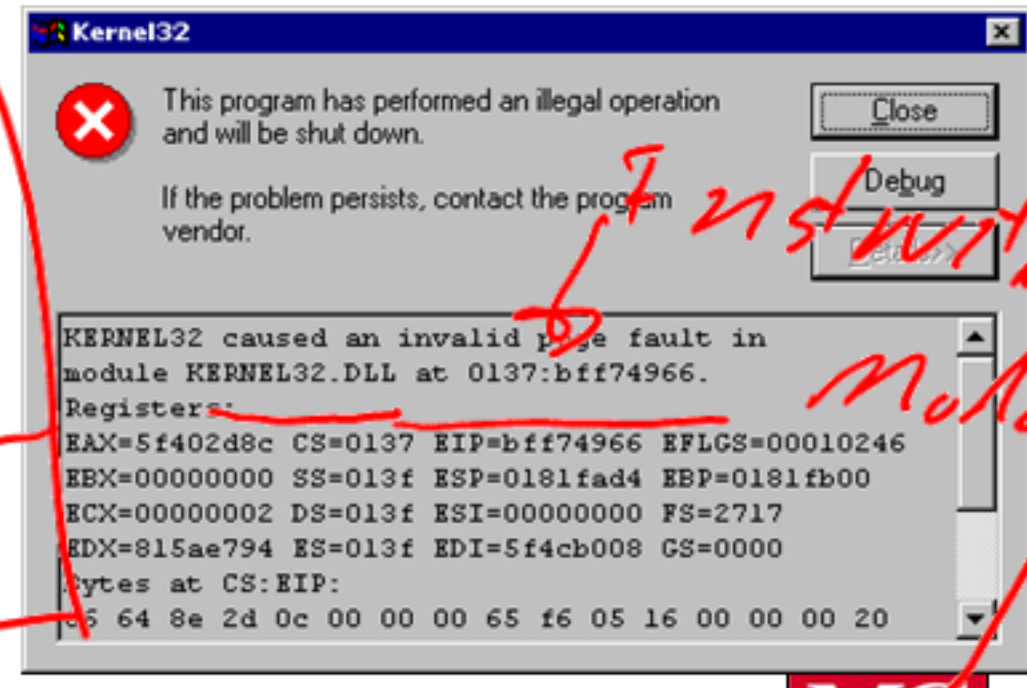
- Should emphasize issues discovered and tracked from the beginning of the software lifecycle
  - requirements
  - Architectural risk analysis
- Should be based on perceived risk

Better Penetration testing

- Disassemblers and compilers
- Control flow and coverage tools
- DLL Analyzers → LookQ Microsoft DLL's
- Debuggers
- Rootkits

Planning

- about verifying the absence of insecure functionality.
  - No software development artifacts for these behaviors.
- Staring points
  - interfaces between the software and its external environment
    - User interfaces
    - network interfaces
  - Error messages and user alert dialogs



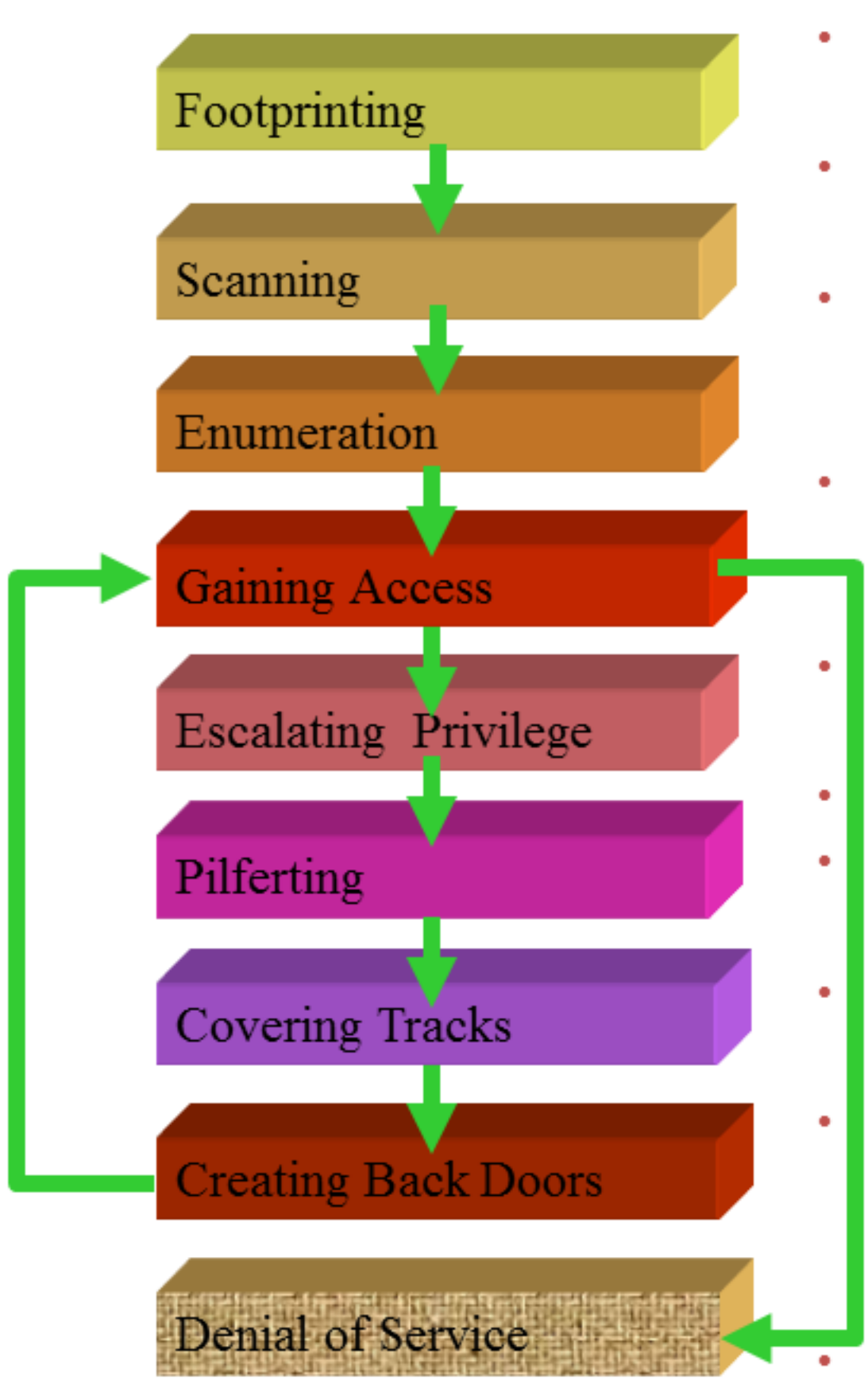
Layering of Tests

1. External attacker with no knowledge of the system.
2. External attacker with access to the system.
3. Internal attacker with access to the system.

Penetration Testing Types

- Environment Attacks
  - Scripts, plugins, other installed materials
  - Registry entries
- Input Attacks
  - network protocols and sockets
  - exposed remote functionality
    - DCOM, remote procedure calls (RPCs) and Web services,
  - data files (binary or text)
  - temporary files
  - UI controls allowing direct user input,
- Data and Logic Attacks
  - Denial of service attacks

Hacking Methodology



- whois, nslookup
- Nmap, fping
- dumpACL, showmount
  legion, rpcinfo
- Tcpdump, Lophtcrack
  NAT
- Johntheripper, getadmin
- Rhosts, userdata
- Config files, registry
- zap, rootkits
- Cron,at, startup folder
  netcat, keystroke logger
  remote desktop
- Synk4, ping of death
  tfn/stacheldraht

Fuzz testing

- Fuzz testing is a method of finding software security holes by feeding purposely invalid and ill-formed data as input to program interfaces
- *Inputs include:*
  - Files
  - Network ports
  - APIs

Advantages of Fuzz Testing

- High volume of testing
- Highly automated
- Finds many problems related to reliability
  - Many of which are potential security holes
  - Over 70% of security vulnerabilities Microsoft patched in 2006 were discovered by fuzzing
- Fuzz testing does not typically validate proper reaction to invalid data