



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

**VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)**

## UNIT - 3 NOTES

# Managing a Penetration Test

Managing a penetration test involves careful planning, execution, communication, and reporting to ensure the assessment is effective, ethical, and aligned with organizational objectives.

---

## 1. Planning a Penetration Test

### Objectives

Define what the organization wants to achieve:

- Identify security vulnerabilities
- Validate security controls
- Meet compliance requirements
- Assess security posture

### Scope Definition

Determine:

- Systems to be tested
- Networks and applications involved
- Physical locations (if applicable)
- Testing limitations

### Rules of Engagement (RoE)

Establish:

- Testing schedule
- Authorized activities
- Emergency contacts
- Communication procedures
- Success criteria



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

### Risk Assessment

Evaluate potential business impacts and plan mitigation strategies.

#### Key Deliverable


 **Penetration Testing Plan**

## 2. Structuring a Penetration Test


### Typical Team Structure

 **Project Manager**

Coordinates activities and communications.

 **Lead Penetration Tester**

Oversees technical testing activities.

 **Security Analysts**

Perform assessments and document findings.

 **Client Representatives**

Provide access and approve testing activities.

---

## 3. Execution of a Penetration Test

### Phase 1: Information Gathering

Collect information about:

- Systems
- Applications
- Networks



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## **VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)**

- Security controls

## Phase 2: Vulnerability Assessment

Identify weaknesses such as:

- Misconfigurations
- Outdated software
- Weak authentication controls

## Phase 3: Validation Testing

Safely verify whether identified vulnerabilities are exploitable within the approved scope.

## Phase 4: Post-Assessment Analysis

Evaluate:

- Business impact
- Access levels
- Security control effectiveness

## Phase 5: Evidence Collection

Document findings with:

- Screenshots
- Logs
- Technical notes

---

# 4. Information Sharing During a Penetration Test

Effective communication is critical throughout the engagement.

## Communication Channels



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

### Regular Status Meetings

Provide progress updates.

### Incident Notifications

Report critical findings immediately.

### Progress Reports

Share testing status and observations.

### Escalation Procedures

Address unexpected issues or high-risk discoveries.

## Information Security Requirements

- Protect sensitive data
- Follow confidentiality agreements
- Limit access to assessment information
- Secure communication channels

## Benefits

- ✓ Improved coordination
- ✓ Faster response to critical findings
- ✓ Reduced operational risk

---

## 5. Reporting the Results of a Penetration Test

The final report is one of the most important deliverables.

### Executive Summary



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

Designed for management:

- Overall security posture
- Major risks identified
- Business impact

## Technical Findings

Detailed information about:

- Vulnerabilities discovered
- Evidence collected
- Risk ratings

## Risk Assessment

Common severity levels:

Critical

High

Medium

Low

## Recommendations

Provide:

- Remediation steps
- Security improvements
- Prioritized action plans

## Retesting Results

Verify whether vulnerabilities have been successfully remediated.

## Best Practices

✓ Obtain written authorization before testing

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

- ✓ Clearly define scope and objectives
- ✓ Maintain continuous communication
- ✓ Document all activities and findings
- ✓ Protect confidential information
- ✓ Provide actionable recommendations
- ✓ Conduct retesting after remediation

## MANAGING A PENETRATION TEST

A STRUCTURED APPROACH TO IDENTIFY RISKS, IMPROVE SECURITY, AND DELIVER ACTIONABLE RESULTS

### 1 PLANNING A PENETRATION TEST

OBJECTIVES	SCOPE DEFINITION	RULES OF ENGAGEMENT (RoE)	RISK ASSESSMENT	KEY DELIVERABLE
Define the goals of the test and what success looks like.	Identify systems, applications, networks and locations to be tested. Define in-scope and out-of-scope items.	Agree on test schedule, authorized activities, limitations, communication methods, and emergency contacts.	Identify potential risks and business impacts. Plan mitigations to minimize disruption.	Penetration Testing Plan and signed authorization.

THE PLAN SETS THE FOUNDATION

- ✓ Aligns the test with business goals
- ✓ Reduces risk and ensures clarity
- ✓ Helps the team prepare for success

### 2 STRUCTURING A PENETRATION TEST

TEST PHASES

1. Planning & Authorization
2. Information Gathering
3. Vulnerability Assessment
4. Exploitation Validation
5. Post-Exploitation Analysis
6. Reporting
7. Remediation & Retesting

TEAM STRUCTURE

<b>PROJECT MANAGER</b> Coordinates the project, communication and timelines.	<b>LEAD PENETRATION TESTER</b> Leads the technical assessment and strategy.	<b>SECURITY ANALYSTS</b> Perform testing, analyze findings and collect evidence.	<b>CLIENT REPRESENTATIVES</b> Provide access, information and business context. Review results.
---	--	---	--

### 3 EXECUTION OF A PENETRATION TEST

<b>PHASE 1: INFORMATION GATHERING</b>	Collect data about the target environment. • OSINT • Network discovery • Service & banner enumeration
<b>PHASE 2: VULNERABILITY ASSESSMENT</b>	Identify weaknesses and misconfigurations. • Automated scanning • Manual verification • Review of configurations and patches
<b>PHASE 3: VALIDATION TESTING</b>	Safely attempt to exploit identified vulnerabilities within the approved scope. • Proof of concept • Controlled exploitation
<b>PHASE 4: POST-ASSESSMENT ANALYSIS</b>	Evaluate the impact and access gained. • Privilege levels • Data access • Business impact
<b>PHASE 5: EVIDENCE COLLECTION</b>	Document all findings with clear evidence. • Screenshots • Logs & command output • Notes and timestamps

### 4 INFORMATION SHARING DURING A PENETRATION TEST

COMMUNICATION CHANNELS

<b>REGULAR STATUS MEETINGS</b> Provide updates on progress, challenges and key observations.	<b>INCIDENT NOTIFICATIONS</b> Immediately report critical or high-risk findings to the right contacts.	<b>PROGRESS REPORTS</b> Share ongoing results and testing status at agreed intervals.	<b>ESCALATION PROCEDURES</b> Follow defined processes for unexpected issues or high-risk events.
---	---	--	---

INFORMATION SECURITY REQUIREMENTS

Protect sensitive data and confidential information.	Limit access to assessment information.	Use secure communication channels.	Follow NDAs and organizational policies.
--	---	------------------------------------	--

BENEFITS

- ✓ Improved coordination
- ✓ Faster response to critical findings
- ✓ Reduced operational risk

### 5 REPORTING THE RESULTS OF A PENETRATION TEST

REPORT COMPONENTS

<b>EXECUTIVE SUMMARY</b> High-level overview of the assessment, key risks and business impact.	<b>SCOPE &amp; METHODOLOGY</b> What was tested, how it was tested and any limitations.	<b>FINDINGS &amp; EVIDENCE</b> Detailed vulnerabilities with supporting evidence.	<b>RISK RATINGS</b> Severity levels help prioritize remediation efforts.	<b>RECOMMENDATIONS</b> Actionable steps to remediate issues and strengthen security.	<b>RETESTING RESULTS</b> Validate that issues have been fixed and risks are reduced.
---	---	--	---	---	---

RISK RATING GUIDE

<b>CRITICAL</b>	Immediate threat to business operations or data.
<b>HIGH</b>	High risk with significant impact.
<b>MEDIUM</b>	Moderate risk; should be addressed.
<b>LOW</b>	Low risk; minor impact.

A GOOD REPORT IS...

- ✓ Clear and easy to understand
- ✓ Accurate and evidence-based
- ✓ Actionable and prioritized
- ✓ Aligned with business goals
- ✓ Delivered on time

BEST PRACTICES

Obtain written authorization before testing.	Clearly define scope and objectives.	Maintain continuous communication with stakeholders.	Document all activities and findings.	Protect confidential information at all times.	Provide actionable recommendations and guidance.	Conduct retesting after remediation to validate fixes.
--	--------------------------------------	--	---------------------------------------	--	--	--

KEY TAKEAWAY

Effective management of a penetration test ensures a controlled, transparent, and value-driven assessment. From careful planning to clear reporting, every step helps organizations understand their risks and build stronger defenses.

REMEMBER:

Penetration testing is not about finding all the vulnerabilities, it's about improving security and reducing risk.



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

## Basic Linux Exploits

### 1. Stack Operations

What is a Stack?

A **stack** is a special memory area used by a program to store temporary information while it runs.

Think of it like a **stack of plates**:

📦 Put a plate on top → **PUSH**

📦 Remove the top plate → **POP**

The last item placed on the stack is the first one removed.

Why is it Important?

The stack stores:

- Function information
- Temporary data
- Return locations for programs

If the stack is corrupted, a program may crash or behave unexpectedly.

---

### 2. Buffer Overflows

What is a Buffer?

A **buffer** is a small memory space used to store data temporarily.

Example:

A box can hold **10 balls**.

📦 Capacity = 10 balls



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

What is a Buffer Overflow?

If you try to put **20 balls** into the same box:

 →  Overflow

The extra balls spill outside the box.

Similarly, when a program receives more data than its buffer can hold, it may overwrite nearby memory.

### Effects

- ✗ Program crashes
- ✗ Data corruption
- ✗ Security vulnerabilities

---

## 3. Local Buffer Overflow Exploits

What Does "Local" Mean?

The person must already have access to the computer.

A local vulnerability exists inside a program running on that system.

### Example

Suppose a login application accepts only 20 characters.

If it does not properly check input length, very long input may cause:

- Program failure
- Memory corruption
- Unexpected behavior

Why Is It Dangerous?



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

An attacker with local access may try to misuse such weaknesses to gain higher privileges or disrupt the system.

---

### 4. Exploit Development Process

Security researchers study vulnerabilities to understand and fix them.

#### Step 1: Find a Vulnerability

- 🔍 Discover a weakness in a program.

#### Step 2: Analyze the Problem

- 🧠 Understand why the vulnerability exists.

#### Step 3: Create a Test

- 🔧 Safely demonstrate the issue in a controlled environment.

#### Step 4: Assess the Risk

- 📊 Determine how serious the vulnerability is.

#### Step 5: Develop a Fix

- 🔑 Correct the coding error.

#### Step 6: Test the Fix

- ✓ Ensure the vulnerability is removed.

#### Step 7: Release the Patch

- 📦 Update the software to protect users.
- 

## Modern Linux Protections



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

Linux includes built-in security features:

### ASLR

Randomly changes memory locations to make attacks harder.

### Stack Canaries

Detects unexpected changes in stack memory.

### NX (No Execute)

Prevents programs from running code in unsafe memory areas.

### SELinux/AppArmor

Adds extra security controls to protect the system.

## Windows Exploits

### What Are Windows Exploits?

A **Windows exploit** is a method used to take advantage of a weakness (vulnerability) in a Windows program or operating system. Security researchers study these weaknesses to help developers fix them and improve security.

## 1. Compiling and Debugging Windows Programs

### Compiling

Compiling means converting source code into a Windows executable file (.exe) that can run on a computer.

### Debugging

Debugging means finding and fixing errors in a program. Security researchers use debuggers to understand how a program behaves and why it crashes.

---



# NARSIMHA REDDY ENGINEERING COLLEGE

An Autonomous Institution | Affiliated to JNTUH | Approved by AICTE  
Accredited by NBA & NAAC with 'A' Grade

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

### 2. Writing Windows Exploits

Security researchers analyze vulnerable programs to understand:

- What causes the vulnerability
- How serious it is
- How it can be fixed

The purpose is to improve security and develop patches.

---

### 3. Understanding Structured Exception Handling (SEH)

**SEH (Structured Exception Handling)** is Windows' error-handling system.

When a program encounters a problem, such as:

- Invalid memory access
- Missing files
- Divide-by-zero errors

SEH helps the program manage the error instead of crashing immediately.

---

### 4. Understanding Windows Memory Protections

Modern Windows versions include built-in security features.

**DEP (Data Execution Prevention)**

Prevents code from running in memory areas meant only for data.

**ASLR (Address Space Layout Randomization)**

Randomly changes memory locations, making attacks more difficult.

**SafeSEH**

## VULNERABILITY ASSESSMENT AND PENETRATION TESTING (23CY701)

Protects the Windows exception-handling process.

SEHOP

Provides additional protection for exception handling.

Stack Cookies

Detects memory corruption and helps stop attacks before they succeed.

## 5. Bypassing Windows Memory Protections

Security experts study how attackers might attempt to overcome memory security protections. This helps software developers:

- Discover weaknesses
- Improve defenses
- Release security updates
- Better protect users

**WINDOWS EXPLOITS SIMPLE EXPLANATION**

Windows exploits take advantage of weaknesses in software or the operating system. Security researchers study these weaknesses to help fix them and make systems more secure.

### 1 COMPILING & DEBUGGING WINDOWS PROGRAMS

**COMPILING**  
Compiling converts source code into an executable file (.exe) that Windows can run.

**DEBUGGING**  
Debugging helps find and fix errors (bugs) in a program before it is released.

Tools help researchers:

- Examine program behavior
- Find crashes and bugs
- Inspect memory
- Understand weaknesses

### 2 WRITING WINDOWS EXPLOITS

An exploit is a method used to demonstrate that a vulnerability exists in software.

Researchers use exploits in a controlled environment to:

- Understand the weakness
- Measure the risk
- Help developers fix the problem
- Improve overall security

**IMPORTANT**  
The goal is to improve security, not to harm systems. Responsible research leads to stronger software and safer systems.

### 3 UNDERSTANDING STRUCTURED EXCEPTION HANDLING (SEH)

SEH (Structured Exception Handling) is Windows' built-in error-handling system.

When a program encounters a problem, such as:

- Invalid memory access
- Missing files
- Divide-by-zero errors
- Other unexpected issues

SEH helps the program manage the error instead of crashing immediately.

SEH improves software stability and reliability by handling errors in a controlled way.

### 4 UNDERSTANDING WINDOWS MEMORY PROTECTIONS

Modern Windows versions include several built-in security features that protect against memory-related attacks.

<b>DEP (Data Execution Prevention)</b>	Prevents code from running in memory areas that are meant only for data.
<b>ASLR (Address Space Layout Randomization)</b>	Randomly changes memory locations each time a program runs, making attacks harder.
<b>SafeSEH (Safe Structured Exception Handling)</b>	Ensures that only valid exception handlers can be used. Helps prevent SEH exploitation.
<b>SEHOP (SEH Overwrite Protection)</b>	Adds an extra layer of protection for SEH to stop manipulation of exception chains.
<b>Stack Cookies (Canaries)</b>	Places a random value (cookie) on the stack to detect memory corruption and stop attacks.

### 5 BYPASSING WINDOWS MEMORY PROTECTIONS

Attackers may try to bypass security protections to exploit vulnerabilities. Security researchers study these techniques so that developers and Microsoft can:

- Discover weaknesses
- Improve defenses
- Release security updates
- Better protect users

Understanding bypass methods helps build stronger protections and more secure Windows systems.

### WINDOWS SECURITY EVOLUTION

WINDOWS XP SP3	WINDOWS VISTA	WINDOWS 7	WINDOWS SERVER 2008
• DEP • SafeSEH • Basic protections	• DEP • ASLR • SafeSEH • Improved security	• Improved ASLR • SEHOP • Better kernel protections	• Stronger memory protections • Enhanced access controls • Better monitoring

**KEY TAKEAWAY – IN SIMPLE WORDS**

- COMPILING**: Turns code into a program.
- DEBUGGING**: Finds and fixes errors.
- SEH**: Windows system that handles errors.
- DEP**: Stops code from running in data memory.
- ASLR**: Randomizes memory locations.
- SAFESEH & SEHOP**: Protect Windows error handling.
- STACK COOKIES**: Detects memory corruption.

Remember: Ethical hacking and security research help make the digital world safer for everyone. Secure Today, Safer Tomorrow