


<https://swayam.gov.in>

[https://swayam.gov.in/nc\\_details/NPTEL](https://swayam.gov.in/nc_details/NPTEL)

harshith.savanur01@gmail.com ▾

NPTEL (<https://swayam.gov.in/explorer?ncCode=NPTEL>) » Information Security - 5 - Secure Systems Engineering (course)



Click to register  
for Certification  
exam

([https://examform.nptel.ac.in/2025\\_01/exam\\_form/dashboard](https://examform.nptel.ac.in/2025_01/exam_form/dashboard))

If already  
registered, click  
to check your  
payment status

## Course outline

About NPTEL  
( )

How does an  
NPTEL  
online  
course  
work? ( )

Week 1 ( )

Week 2 ( )

Week 3 ( )

# Week 3 : Assignment 3

The due date for submitting this assignment has passed.

Due on 2025-02-12, 23:59 IST.

Assignment submitted on 2025-02-11, 09:30 IST

1) Which of the following statements about **ASLR** is **false**?

1 point

- ☐ ASLR randomizes the base addresses of memory regions such as the stack, heap, and shared libraries at runtime
- ☐ ASLR can be bypassed if an attacker is able to leak memory addresses during program execution
- ☒ ASLR guarantees complete protection against buffer overflow and ROP attacks
- ☐ ASLR is more effective when combined with other security mechanisms like DEP and stack canaries

Yes, the answer is correct.

Score: 1

Accepted Answers:

*ASLR guarantees complete protection against buffer overflow and ROP attacks*

2) What is a potential consequence of a **buffer overread** vulnerability?

1 point

- ☐ Arbitrary code execution by overwriting the return address on the stack
- ☒ Exposure of sensitive data, such as passwords or cryptographic keys, from adjacent memory regions
- ☐ Writing malicious data into protected memory regions, causing privilege escalation.
- ☐ Preventing the execution of code due to stack-based protections like ASLR

ASLR (part 1)  
(unit?  
unit=35&less  
n=36)

ASLR (part 2)  
(unit?  
unit=35&less  
n=37)

Buffer  
overreads  
(unit?  
unit=35&less  
n=38)

Demonstration  
of Load Time  
Relocation  
(unit?  
unit=35&less  
n=39)

Demonstration  
of Position  
Independent  
Code (unit?  
unit=35&less  
n=40)

PLT  
Demonstration  
(unit?  
unit=35&less  
n=41)

Week 3  
Feedback  
Form :  
Information  
Security - 5 -  
Secure  
Systems  
Engineering!!  
(unit?  
unit=35&less  
n=42)

Quiz: Week 3 :  
Assignment 3  
(assessment?  
name=147)

Week 4 ()

Yes, the answer is correct.

Score: 1

Accepted Answers:

*Exposure of sensitive data, such as passwords or cryptographic keys, from adjacent memory regions*

3) What was the primary cause of the **Heartbleed vulnerability** in OpenSSL? **1 point**

- ☐ A buffer overflow that allowed attackers to execute arbitrary code on the server
- ☒ An improperly validated length field in a heartbeat request, allowing attackers to read more memory than intended
- ☐ A misconfigured SSL certificate that allowed attackers to impersonate the server
- ☐ A race condition in the SSL handshake process, leading to memory corruption

Yes, the answer is correct.

Score: 1

Accepted Answers:

*An improperly validated length field in a heartbeat request, allowing attackers to read more memory than intended*

4) In a dynamically linked executable, the **Procedure Linkage Table (PLT)** is primarily responsible for resolving addresses of statically linked functions at runtime. (True or False) **1 point**

- ☐ True
- ☒ False

Yes, the answer is correct.

Score: 1

Accepted Answers:

*False*

5) Which of the following statements about the **Global Offset Table (GOT)** is true? **1 point**

- ☐ The GOT is used to store the actual implementation of dynamically linked functions
- ☐ The GOT holds the addresses of global variables in statically linked executables
- ☒ The GOT is used to store addresses of dynamically linked functions and is updated during runtime by the dynamic linker
- ☐ The GOT is responsible for resolving function addresses during compile time

Yes, the answer is correct.

Score: 1

Accepted Answers:

*The GOT is used to store addresses of dynamically linked functions and is updated during runtime by the dynamic linker*

6) Which of the following statements best describes the relationship between **Position-Independent Code (PIC)** and **Address Space Layout Randomization (ASLR)**? **1 point**

- ☒ PIC is required for ASLR to work because it allows code to execute correctly regardless of its memory location

**Week 5 ()****Week 6 ()****Week 7 ()****Week 8 ()****Download  
Videos ()****Text  
Transcripts ()****Books ()****Lecture  
Material ()**

- ☐ ASLR disables the use of PIC because it randomizes memory addresses at runtime
- ☐ PIC ensures that the stack is non-executable, which is a requirement for ASLR
- ☐ ASLR and PIC are unrelated, as ASLR only affects the heap and stack memory, not the code segment

Yes, the answer is correct.

Score: 1

Accepted Answers:

*PIC is required for ASLR to work because it allows code to execute correctly regardless of its memory location*

7) Which of these techniques is essential for achieving PIC?

**1 point**

- ☐ Using absolute addressing for all memory references
- ☒ Employing relative addressing and indirect jumps
- ☐ Avoiding any data access within the code
- ☐ Storing all instructions in a separate memory segment

Yes, the answer is correct.

Score: 1

Accepted Answers:

*Employing relative addressing and indirect jumps*

8) Match the following

**1 point**

- |             |                               |
|-------------|-------------------------------|
| 1. W^X      | A) Buffer overflow protection |
| 2. Canaries | B) Memory access control      |
| 3. PLT      | C) Dynamic linking            |
| 4. GOT      | D) Address randomization      |
| 5. ASLR     | E) Function resolution        |

- ☒ 1:B 2:A 3:E 4:C 5:D
- ☐ 1:D 2:C 3:B 4:A 5:E
- ☐ 1:C 2:B 3:E 4:D 5:A
- ☐ 1:C 2:B 3:D 4:E 5:A

Yes, the answer is correct.

Score: 1

Accepted Answers:

*1:B 2:A 3:E 4:C 5:D*

9) In a program using dynamic linking, the **PLT** has 4 entries for 4 different functions. Each entry in the **PLT** consists of 4 bytes for the jump instruction and 4 bytes for the address of the function in the **GOT** (Global Offset Table).

How many bytes of memory are required for the **PLT** table to store the entries for these 4 functions?

Yes, the answer is correct.

Score: 1

Accepted Answers:

(Type: Numeric) 32

**1 point**

10) [True or False] **Position-Independent Code (PIC)** can only be used in statically linked executables, as it requires fixed memory addresses to function correctly.

**1 point**

☐ True

☒ False

Yes, the answer is correct.

Score: 1

Accepted Answers:

*False*