USN | | | | | | | | | | |

Continuous Internal Examination 1–September 2025

Sub:	**Computer and Network Security**							Code:	BEC714B
Date:	29/ 09 / 2025	Duration:	90 mins	Max Marks:	50	Sem:	VII	Branch:	All

Answer any 5 full questions

		Marks	CO	RBT
1	Describe the concepts of Confidentiality, Authentication, Integrity, Non-repudiation, and Availability in the context of their application to network security	10	CO1	L2
2	List and briefly define categories of passive and active security attacks.	10	CO1	L2
3	What is the difference between traffic analysis and the release of contents in network security, and how does each of these threats impact the confidentiality of communications?	10	CO1	L2
4	List and explain different types of computer viruses (e.g., boot sector infectors, macro viruses, polymorphic viruses)	10	CO1	L2
5	Define malicious logic. Explain with a suitable example from the UNIX shell script Trojan horse.	10	CO1	L1, L2
6	Differentiate between formal verification and penetration testing. Discuss their advantages and limitations with examples.	10	CO2	L2
7	Define vulnerability in computer security. Explain its significance with suitable examples.	10	CO2	L2

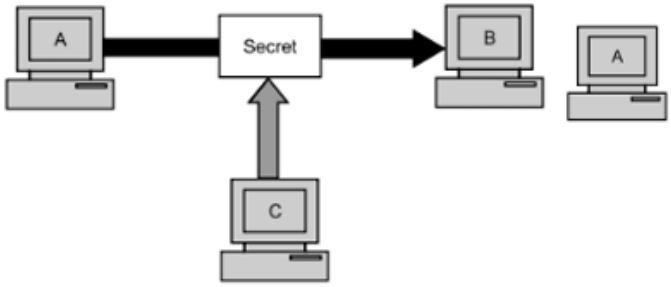
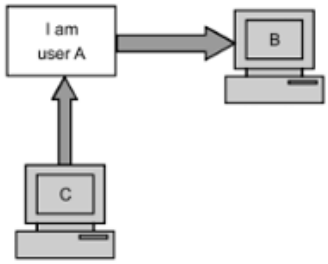
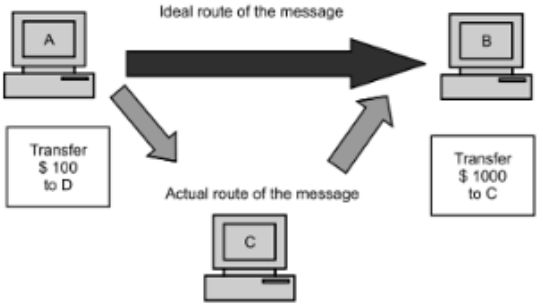
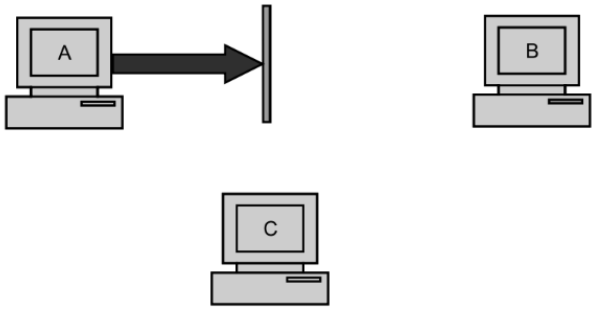
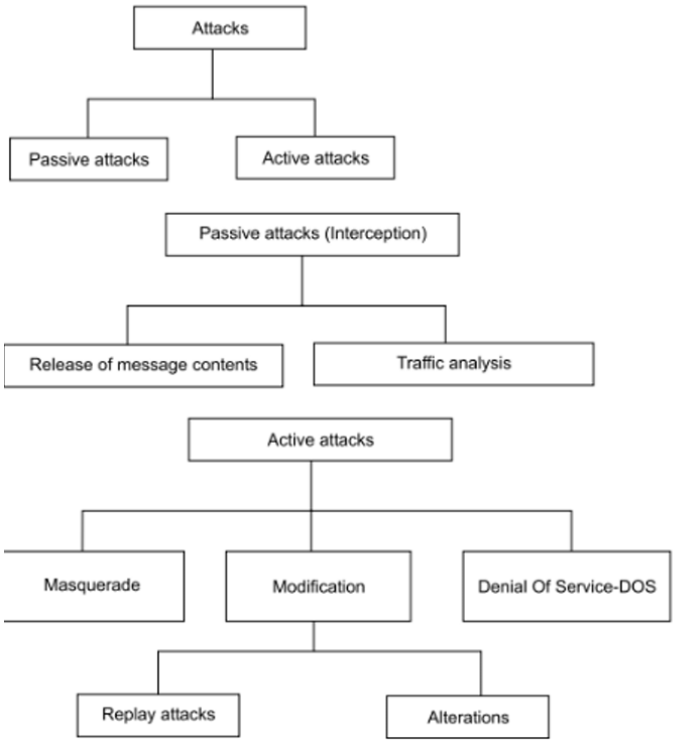
USN | | | | | | | | | | |

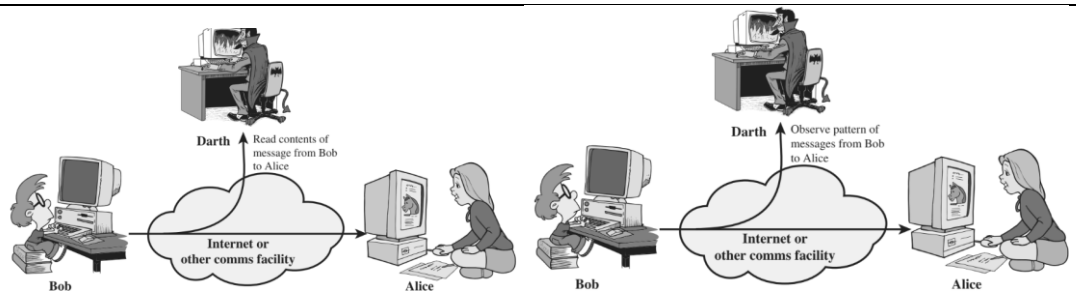
Continuous Internal Examination 1–September 2025

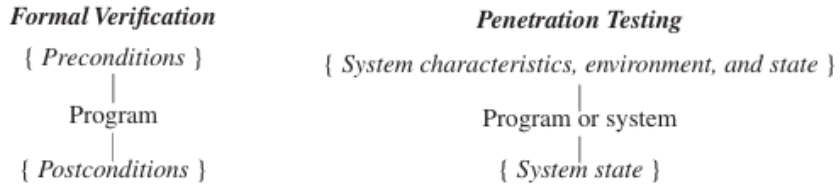
Sub:	**Computer and Network Security**							Code:	BEC714B
Date:	29/ 09 / 2025	Duration:	90 mins	Max Marks:	50	Sem:	VII	Branch:	All

Answer any 5 full questions

		Marks	CO	RBT
1	Describe the concepts of Confidentiality, Authentication, Integrity, Non-repudiation, and Availability in the context of their application to network security	10	CO1	L2
2	List and briefly define categories of passive and active security attacks.	10	CO1	L2
3	What is the difference between traffic analysis and the release of contents in network security, and how does each of these threats impact the confidentiality of communications?	10	CO1	L2
4	List and explain different types of computer viruses (e.g., boot sector infectors, macro viruses, polymorphic viruses)	10	CO1	L2
5	Define malicious logic. Explain with a suitable example from the UNIX shell script Trojan horse.	10	CO1	L1, L2
6	Differentiate between formal verification and penetration testing. Discuss their advantages and limitations with examples.	10	CO2	L2
7	Define vulnerability in computer security. Explain its significance with suitable examples.	10	CO2	L2

		Marks	CO	RBT
1	Confidentiality, Authentication, Integrity, Non-repudiation  Confidentiality  Authentication  Integrity  Non-repudiation  Attack on availability	**2 x 5**	CO1	L2
2	 Explanation passive attack Explanation Active attack	**3** **3** **4**	CO1	L2

3	 Release of contents Traffic Analysis	**5+5**	CO1	L2
4	(a) Dormant Phase Here, the virus is idle. It gets activated based on a certain action or event (e.g. the user typing a certain key or a certain date or time is reached, etc). This is an optional phase. (b) Propagation Phase In this phase, a virus copies itself, and each copy starts creating more copies of itself, thus propagating the virus. (c) Triggering Phase A dormant virus moves into this phase when the action/event for which it was waiting is initiated. Introduction to the Concepts of Security 19 (d) Execution Phase This is the actual work of the virus, which could be harmless (display some message on the screen) or destructive (delete a file on the disk). Viruses can be classified into the following categories: (a) Parasitic Virus This is the most common form of virus. Such a virus attaches itself to execut able files and keeps replicating. Whenever the infected file is executed, the virus looks for other execut able files to attach itself and spread. (b) Memory-resident Virus This type of virus first attaches itself to an area of the main memory and then infects every executable program that is executed. (c) Boot sector Virus This type of virus infects the master boot record of the disk and spreads on the disk when the operating system starts booting the computer. (d) Stealth Virus This virus has intelligence built in, which prevents anti-virus software programs from detecting it. (e) Polymorphic Virus A virus that keeps changing its signature (i.e. identity) on every execution, making it very difficult to detect. (f) Metamorphic Virus In addition to changing its signature like a polymorphic virus, this type of virus keeps rewriting itself every time, making its detection even harder.	**10**	CO1	L2
5	Malicious logic refers to set of instructions that cause site security policy to be violated. • **directory (./)**. ls $* means: Run ls on **all the arguments** that were given to the script. Trojan Horse: • Program with an *overt* purpose (known to user) and a *covert* purpose (unknown to user) Often called a Trojan • Named by Dan Edwards in Anderson Report • Example: below script is Trojan horse • Shell script on a UNIX system: — `cp /bin/sh /tmp/.xyzzy` — `chmod u+s,o+x /tmp/.xyzzy` — `rm ./ls` — `ls $*` • Place in program called “ls” and trick someone into executing it. • u+s: set user ID on execution as owner of the file, o+x:This allows all other users to execute the file. • rm ./ls : This removes (deletes) a file named ls that exists in the **current** • These set of UNIX instructions creates a copy of the UNIX shell that is setuid to the user executing this program. This program is deleted, and then the correct ls command is executed. On most systems, it is against policy to trick someone into creating a shell that is setuid to themselves. If someone is tricked into executing this script, a violation of the (implicit) security policy occurs. • Overt purpose: list files in directory • Covert purpose: create setuid shell Dan Edwards was the first to use this term. Trojan horses are often used in conjunction with other tools to attack systems Trojan horses can make copies of themselves. One of the earliest Trojan horses was a version of the game animal. When this game was played, it created an extra copy of itself. These copies spread, taking up much room. The program was modified to delete one copy of the earlier	**3** **7**	CO1	L1, L2

	version and create two copies of the modi f ied program. Because it spread even more rapidly than the earlier version, the modi f ied version of animal soon completely supplanted the earlier version. After a preset date, each copy of the later version deleted itself after it was played			
6	 **Figure 20–1 A comparison between formal verification and penetration testing. In formal verification, the "preconditions" place constraints on the state of the system when the program (or system) is run, and the "postconditions" state the effect of running the program. In penetration testing, the "preconditions" describe the state of the system in which the hypothesized security flaw can be exploited, and the "postconditions" are the result of the testing. In both verification and testing, the postconditions must conform to the security policy of the system.**	**4**	CO2	L2
	Formal Verification: • Mathematically verifying that a system satisfies certain constraints • *Preconditions* state assumptions about the system • *Postconditions* are result of applying system operations to preconditions, inputs • Required: postconditions satisfy constraints • For formal verification to prove absence, proof and preconditions must include *all* external factors o Realistically, formal verification proves absence of flaws within a particular program, design, or environment and not the absence of flaws in a computer system (think incorrect configurations, etc.) Penetration Testing: • Testing to verify that a system satisfies certain constraints • Hypothesis stating system characteristics, environment, and state relevant to vulnerability • Result is compromised system state • Apply tests to try to move system from state in hypothesis to compromised system state • Penetration testing is a *testing* technique, not a verification technique • It can prove the *presence* of vulnerabilities, but not the *absence* of vulnerabilities • Test for evaluating the strengths and effectiveness of all security controls on system • Tests system *in toto*, once it is in place o Includes procedural, operational controls as well as technological ones	**6**		
7	Define vulnerability in computer security. Explain its significance with suitable examples. *Vulnerability*, *security flaw*: failure of security policies, procedures, and controls that allow a subject to commit an action that violates the security policy • Subject is called an *attacker* • Using the failure to violate the policy is *exploiting the vulnerability* or *breaking in* The goal of vulnerability analysis is to develop methodologies that provide the following abilities. 1. The ability to specify, design, and implement a computer system without vulnerabilities. 2. The ability to analyze a computer system to detect vulnerabilities (which feeds into the Flaw Hypothesis Methodology step of penetration testing). 3. The ability to address any vulnerabilities introduced during the operation of the computer system (possibly leading to a redesign or reimplementation of the flawed components). 4. The ability to detect attempted exploitations of vulnerabilities Any one Example RISOS/NRL Taxonomy/ Asalam's Model	**3** **7**	CO2	L2