Q) What is the consequence of an error made by a human in software activity?

a) Command
b) A fault
c) Documentation
d) Process

Q) What is an example of an error when performing software activity?

a) Writing incorrect code
b) User inputting wrong data
c) A designer misunderstanding a requirement.
d) System crashes

Q) What can be a result from a design fault?

a) An incorrect description in a user manual
b) A failure
c) A single error
d) Incorrect code

Q) What is a single error capable of generating?

a) Incorrect description
b) An incorrect code
c) A failure
d) Many faults

Q) When can faults in requirements documents be discovered?

a) While the system is performing as specified.
b) Before system delivery
c) During testing, or during operation and maintenance.
d) After system delivery

Q) What does a failure indicate in the context of requirements documents?

a) The requirements documents can contain faults
b) The system is operating optimally
c) The system is not performing as required
d) The system is not performing as specified

Q) What is the difference between a fault and a failure in terms of software development?

a) A fault is an inside view of the system seen by the user, whereas a failure is an outside view seen by the developer.
b) A fault is an inside view of the system seen by the developer, whereas a failure is an outside view seen by the user.
c) A fault is an outside view of the system seen by the developer, whereas a failure is an inside view seen by the user.
d) A fault is an outside view of the system seen by the user, whereas a failure is an inside view seen by the developer.

Q) What is the difference between a fault and a failure in software engineering?

a) A fault is when an error in code causes the code to fail.
b) A failure is an error in code that has not caused the code to fail.
c) Flaws are only used by security engineers.
d) A fault is an error in code that has not caused the code to fail.

Q) What is a flaw according to security engineers?

a) A fault or a failure
b) A programming mistake
c) A fault
d) A security vulnerability

Q) In which way can an attacker gain control by masquerading as the operating system?

a) Privilege escalation
b) Memory mapping
c) Code injection
d) Port scanning

Q) What happens when data overflow occurs?

a) It stays strictly within the data space.
b) It is stored on top of the operating system's data or code.
c) It spills over into an adjacent code area.
d) It is stored on top of another piece of your data.

Q) How can an attacker use an overflow attack to produce an effect?

a) Redirect execution
b) Place particular data in a predictable location
c) Put arbitrary data in the wrong place
d) Overwrite stack memory

Q) How can an attacker take control of the program in order to execute their own instructions?

a) Overwrite the program counter stored in the stack.
b) Delete the program counter stored in the stack.
c) Overwrite part of the code in high memory.
d) Replace the instructions with their own instructions.

Q) In a data driven attack, how is the program's memory best protected?

a) Overwrite the program counter and data in the stack.
b) Stay within bounds.
c) Change the data inputs to the program.
d) Reset the program counter.

Q) Which of the following should all be done by the programmer, operating system, compiler, and hardware when maintaining boundaries?

a) Confirm that array subscripts are within limits.
b) Transfer only a bounded amount of data.

c) Check lengths before writing.
d) Monitor input and accept only as many characters as can be handled.

Q) What is the process of verifying that the subject is authorized to perform an operation on an object?

a) Mediation
b) Checking procedures
c) Limiting the privileges of programs
d) Overrunning the allocated space

Q) What is an undocumented access point called?

a) A backdoor or trapdoor
b) Reference monitor
c) Unbypassable
d) Solid and complete mediation