**11.5 State the similarities and differences between command injection and SQL injection attacks.**

Both SQL injection and command injection attacks are caused by improper handling of input to the program, and both can be prevented by sanitizing input. Command injection occurs in the shell, whereas SQL injection is SQL specific. Therefore, the methods for sanitizing inputs are slightly different.

**11.9 Define input fuzzing. State where this technique should be used.**

Input fuzzing is a method for testing programs for bugs or security flaws by generating a large amount of random inputs and feeding those to the program. Fuzzing should be used to find basic flaws in the way that the program handles input.

**11.12 Identify several concerns associated with the use of environment variables by shell scripts.**

Malicious environment variables such as PATH or LD\_LIBRARY\_PATH can lead to a program to load or run a program that was not intended. Changing the IFS variable can cause shell scripts to be interpreted differently than was intended. Malicious environment variables that are really long can also cause buffer overflows when a program is executed.

**22.4 What is DKIM?**

DKIM (DomainKeys Identified Mail) is a method for verifying that an email came from who it says its from by using public-private key cryptography at the domain level to verify the validity of the sender's domain.

**22.6 What is the difference between an SSL connection and an SSL session?**

An SSL connection is a temporary communication channel between two systems. An SSL session can encompass multiple SSL connections, and contains a set of security parameters such as keys that are negotiated at the start of each session.

**23.1 What are the principal elements of a Kerberos system?**

A Kerberos system involve clients, servers, and a specialized authentication server to mediate authentication between the clients and the servers.

**23.4 What is X.509?**

X.509 is a standard format for distributing certificates for public keys.

**23.11 What are some key problems with current public key infrastructure implementations?**

One problem is user ignorance of security issues, and users making poor decisions once a certificate has been rejected (such as proceeding into unsafe waters anyhow). Another issue is that the certificate authorities are implicitly trusted and believed to be perfectly secure. A third issue is that different implementation of the public key schemas can have different trust stores.