Task 5

Server-Side Template Injection (SSTI):
SSTI attacks exploit vulnerabilities in template engines used by web applications. Attackers can inject malicious code into templates, potentially leading to remote code execution and data leaks.

HTTP Response Splitting Attack:
HTTP response splitting attacks involve manipulating the server's response to inject malicious headers or content. This can lead to cache poisoning, session hijacking, or cross-site scripting vulnerabilities.

HTTP Parameter Pollution (HPP):
HPP attacks exploit inconsistencies in how web applications handle multiple instances of the same parameter. Attackers manipulate parameter values to potentially modify application behavior or gain unauthorized access.

Session Fixation Attack:
In a session fixation attack, an attacker sets a victim's session identifier, often through links or cookies. This allows the attacker to hijack the victim's session after they log in. **Clickjacking Attack:**
Clickjacking involves tricking users into clicking on hidden or transparent elements on a web page. Attackers overlay the actual content with malicious content, potentially leading to actions the user didn't intend.

Server-Side Request Forgery (SSRF): SSRF attacks manipulate a web application into making requests to internal or external servers. Attackers can use SSRF to scan internal networks, access restricted resources, or carry out other malicious actions.

DOM-Based Attacks:
DOM-based attacks manipulate the Document Object Model (DOM) of a web page to execute malicious scripts. These attacks can lead to data theft, session hijacking, or other client-side exploits.

Session Prediction/Session Replay Attack:
Attackers predict or capture valid session tokens to gain unauthorized access. This can involve replaying intercepted session data or predicting session IDs to impersonate users. **Web Cache Poisoning**:
Cache poisoning attacks manipulate the cache of a web server or proxy to serve malicious content to users. This can lead to users being served compromised content or performing unintended actions.

LDAP Injection:
LDAP injection attacks target applications that use Lightweight Directory Access Protocol (LDAP) to authenticate users. Attackers manipulate input to inject malicious LDAP queries, potentially leading to unauthorized access.