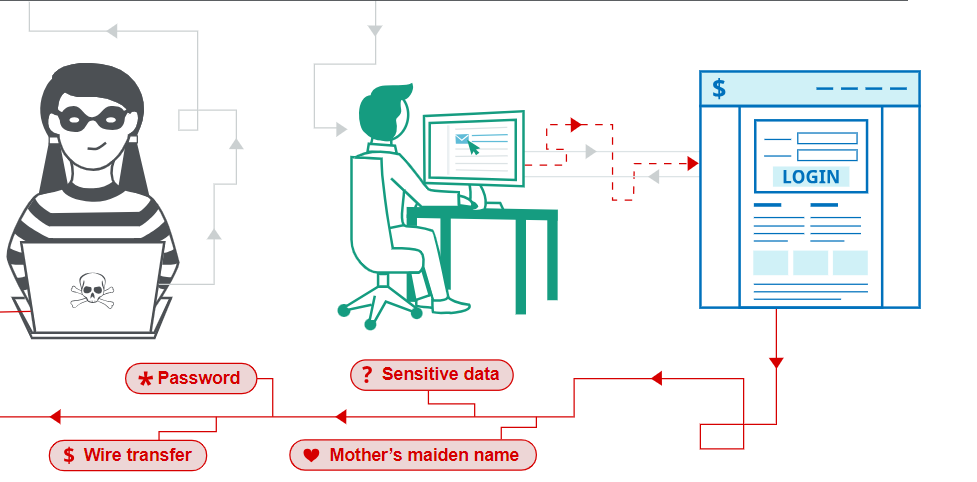
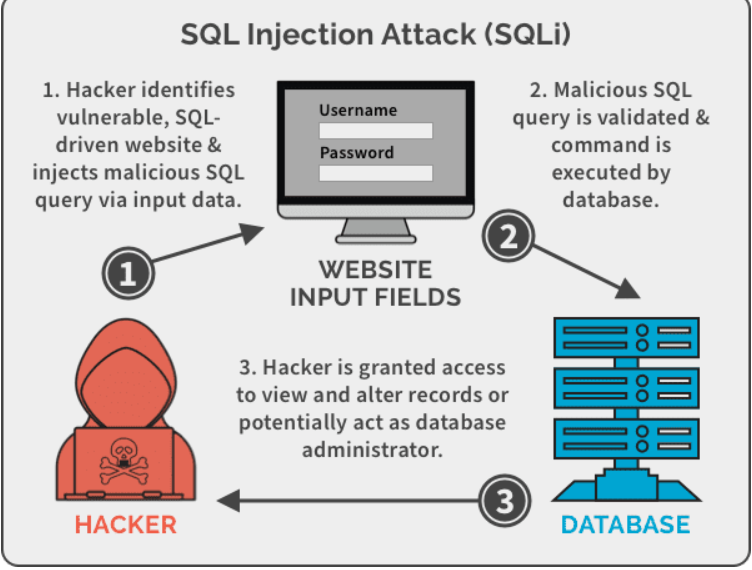
WEB APPLICATION ATTACKS

* **Cross-site scripting (XSS).** That involves an attacker uploading a piece of malicious script code onto your website that can then be used to steal data or perform other kinds of mischief. Although this strategy is relatively

unsophisticated, it remains quite common and can do significant damage.



* **SQL Injection (SQLI).**This happens when a hacker submits destructive code into an input form. If your systems fail to clean this information, it can be submitted into the database, changing, deleting, or revealing data to the attacker.
* 
* **Path traversal.**Also resulting from improper protection of data that has been inputted, these webserver attacks involve injecting patterns into the webserver hierarchy that allow bad actors to obtain user credentials,

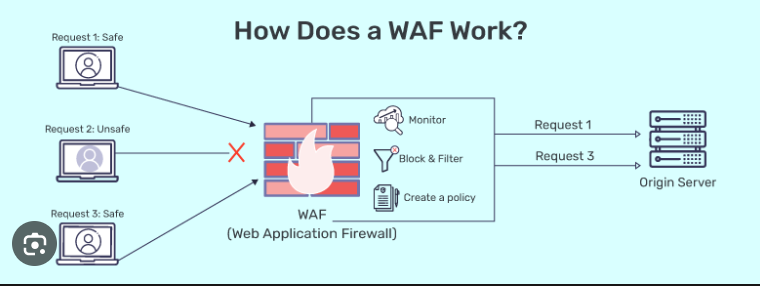
databases, configuration files, and other information stored on hard drives.



* **Local File Inclusion.** This relatively uncommon attack technique involves forcing the web application to execute a file located elsewhere on the system.
* **Distributed Denial of Service (DDoS) attacks.** Such destructive events happen when an attacker bombards the server with requests. In many cases, hackers use a network of compromised computers or bots to mount this offensive. Such actions paralyze your server and prevent legitimate visitors from gaining access to your services.
* **Automated vulnerability scanning and security testing.** These programs help you to find, analyze, and mitigate vulnerabilities, often before actual attacks occur. Investing in these preventive measures is a cost-effective way to reduce the likelihood that vulnerabilities will turn into cyber disasters.



* **Web Application Firewalls (WAFs).** These operate on the application layer and use rules and intelligence about known breach tactics to restrict access to applications. Because they can access all layers and protocols, WAFs can be highly effective gatekeepers when it comes to shielding resources from attack.



* **Secure Development Testing (SDT).** This instruction is designed for all security team members, including testers, developers, architects, and managers. It provides information about the newest attack vectors. It assists the task force in establishing a baseline and developing a practical, dynamic approach to preventing website attacks and minimizing the consequences of breaches that cannot be stopped.

