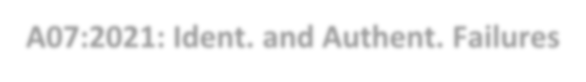
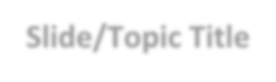
Identification and authentication failures can occur when functions related to a user's identity, authentication, or session management are not implemented correctly or not adequately protected by an application. Attackers may be able to exploit identification and authentication failures by compromising passwords, keys, session tokens, or exploit other implementation flaws to assume other users' identities, either temporarily or permanently.

**Slide/Topic Title**



* Brute force / Credential staffing
* Session Hijacking / Session guessing
* User Enumeration

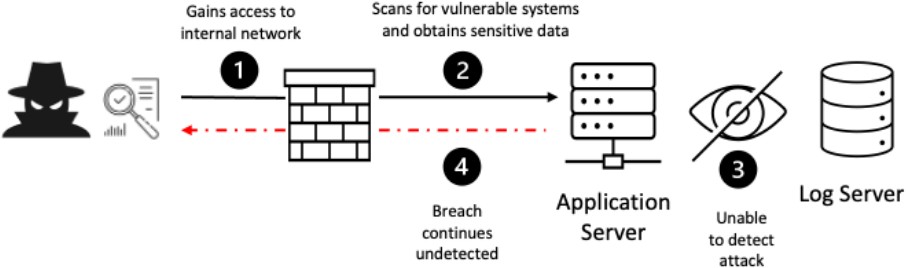
Most web applications that use components such as libraries, frameworks or plugins always execute then with full privileges and flaws in any component can result in serious impact.

If those components contains bugs/vulnerability then the application may compromise without any vulnerability in application.

As example – wordpress plugins are vulnerable.

Software and data integrity failures relate to code and infrastructure that does not protect against integrity violations. This can occur when you use software from untrusted sources and repositories or even software that has been tampered with at the source, in transit, or even the endpoint cache (**supply chain attack**). Attackers can exploit this to potentially introduce unauthorized access, malicious code, or system compromise as part of the following attacks:

* Cache Poisoning
* Code injection
* Command execution
* Denial of Service

Failure to sufficiently log, monitor, or report security events, such as login attempts, makes suspicious behavior difficult to detect and significantly raises the likelihood that an attacker can successfully exploit your application. For example, an attacker may probe your application or software components for known vulnerabilities over a period. Allowing such probes to continue undetected increases the likelihood that the attacker ultimately finds a vulnerability and successfully exploits the flaw.

Server-side request forgery (SSRF) flaws occur whenever a web application is fetching a remote resource without validating the user-supplied URL. The vulnerable web application will often have privileges to read, write, or import data using a URL. To execute an SSRF attack, the attacker abuses the functionality on the server to read or update internal resources. The attacker can then force the application to send requests to access unintended resources, often bypassing security controls