**DevOps Security**

**SQL Injection :** This is when untrusted data is sent as part of the query to the data base and a attacker and manipulate the database and take control of the data or server, This is usually done where the app or website takes in user input and that value is inserted straight into the query command which makes it valuable for SQL injection.

**Broken Authentication :** Authentication is a process of ensuring that whoever is accessing the account is who they say they are. Ensuring that strong usernames and passwords are used on the system so they are not easy to guess and setting up multi-factor authentication protects the system from these types of attacks. Also by limiting the number of incorrect password attempts and recording a log and alerting admin of attempted incorrect logins.

**Broken access control:**  This is when applications let different users have access to different things. For example the administrator should have certain privileges that a ordinary user wouldn’t have. One-way attackers can do this is by changing the URL in the browser by adding /admin which could then bring them to that site and have them logged in as if they were an administrator. Way to prevent this is:

Implement access control mechanisms once and re-use them throughout the application.

Model access controls should enforce record ownership, rather than accepting that the user can create, read, update, or delete any record.

Disable web server directory listing and ensure file metadata (e.g. .git) and backup files are not present within web roots.

Log access control failures, alert admins when appropriate (e.g. repeated failures).

Rate limit API and controller access to minimize the harm from automated attack tooling.

Developers and QA staff s hould include functional access control unit and integration tests.