Securing Authentication and Authorization

1. Intro

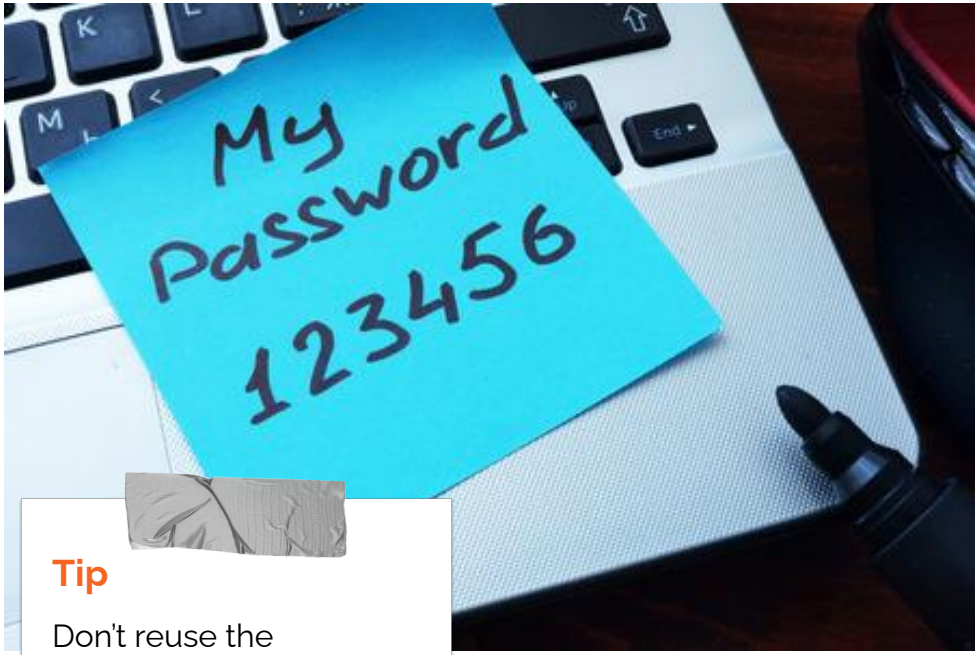
We will be covering these topics in this presentations

- ➔ What is Authentication.
- ➔ Weaknesses
- ➔ Securing Authentication
- ➔ What is Authorization
- ➔ Weaknesses
- ➔ Securing Authorization

What is Authentication

Authentication is the process of verifying that an individual, entity or website is whom it claims to be. In the context of web application it is being carried out by submitting the username and the password.

Weaknesses in the Authentication

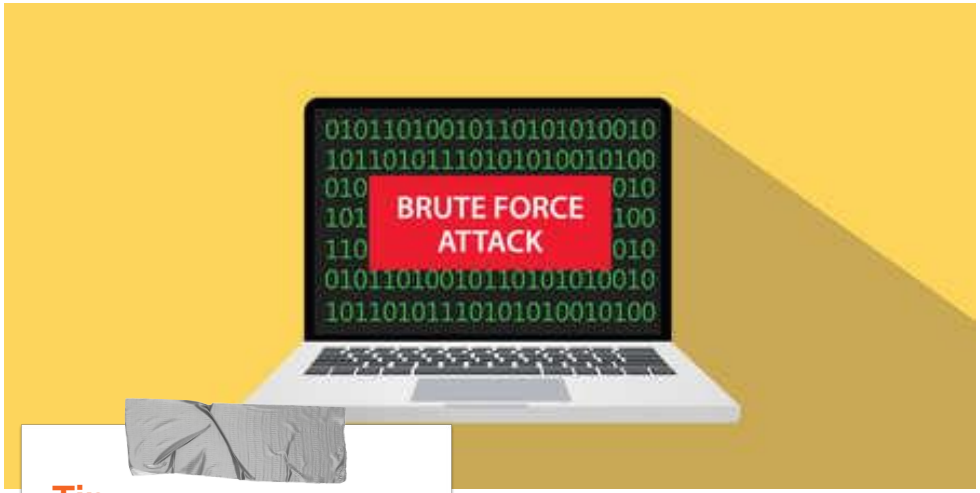


Tip

Don't reuse the password on different website and chose a secure password which contains alphanumeric characters with special one's

Weak and Reused Passwords

The password which is most used in the World is 123456. Users were able to use these type of password because there were no password policies implement. Even though many users reuse their password of different website, so If one website is breached their account on other may also be compromised



Tip

Implement a brute force protection on the website.

Password Brute Force

If the application is not using the captcha or any protection against the brute force then a malicious attacker can easily crack the password by using a list of known words.

Securing Authentication

How to Secure Authentication

Authentication can always be secured by using the layers or authentication which is usually called the multi factor authentication or MFA.

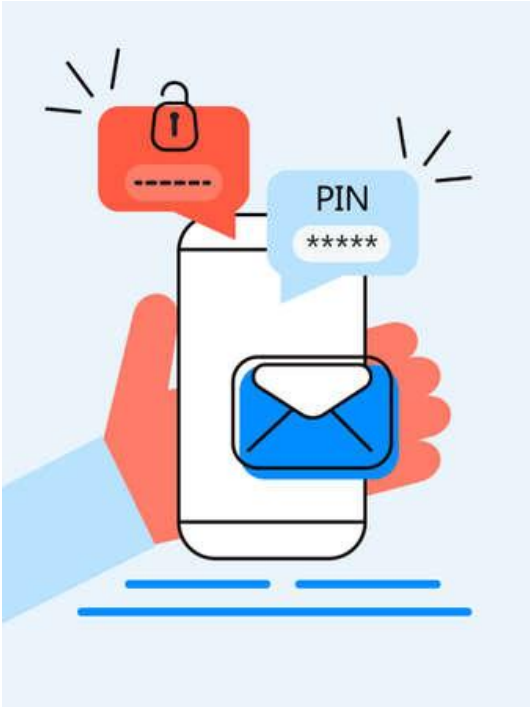
Most of the Web Applications uses 2FA to authenticate the users which provides an extra layer of authentication and secure the users from weak or password reuse attacks.

What is 2FA

Users can enable this on their requirements. This uses couple of different methods

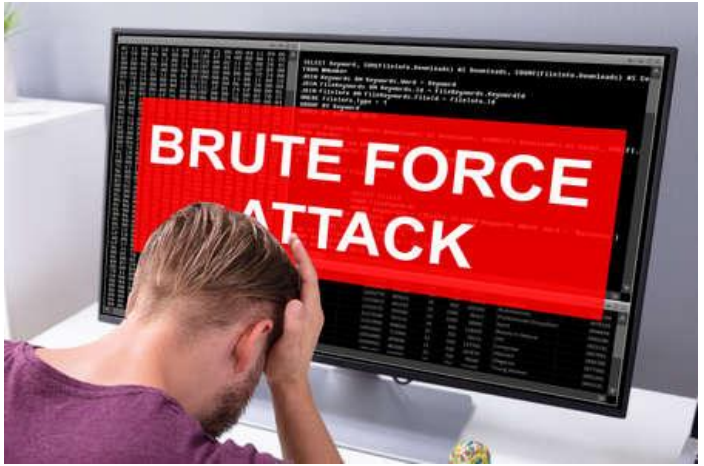
1. OTP on email or phone
2. Google Authentication or Authy apps
3. Secret Questions

So after submitting a Valid username and password user need to use any of the above method to prove their identity.



Example

Attackers have access to hundreds of millions of valid username and password combinations for credential stuffing, default administrative account lists, automated brute force, and dictionary attack tools. Session management attacks are well understood, particularly in relation to unexpired session tokens.



What is Authorization ?

What is Authorization

Authorization is the process by which we verify if an individual entity have access to certain resources or not.

It is different from authentication as it only verifies the whether user have permission to access some of the resources or not.

Weaknesses in Authorization

Weakness with Authorization

There are a couple of vulnerabilities that occur due to the authorization issues.

These are

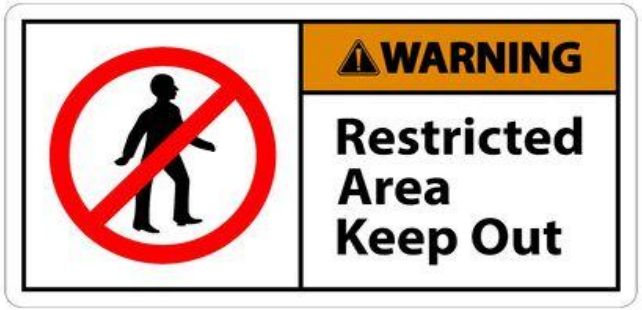
1. Directory Traversal
2. IDOR
3. Privilege Escalation

Why it occurs

When Users and their Inputs are not properly validated they lead to the authorization issues .

Due to which issues like accessing the sensitive files, elevating to the admin privileges may occurs.

A normal user can elevate this privileges to perform the actions that are only intended to perform by the admin.

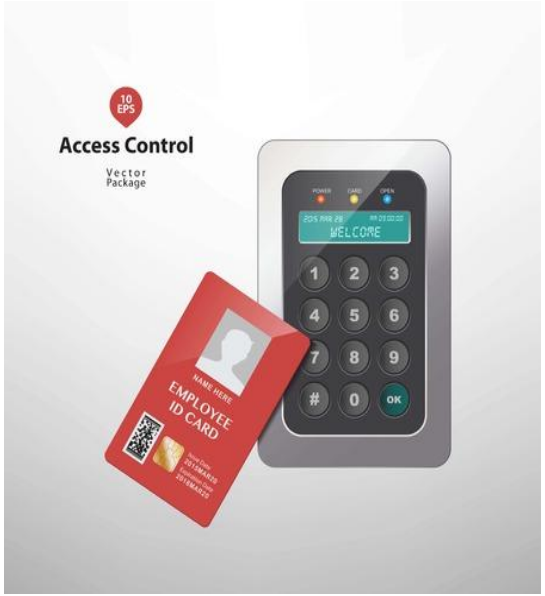


Example

Consider a website “somesite.com”, that has a feature that allows users to check their account balance only after post-authentication. The details fetched with account balance also includes the account holder's name, email address, and account number. To retrieve the personal details website submits a request with an account number to the endpoint listed below.

https://somesite.com/customer/balance.aspx?acc_number=33761

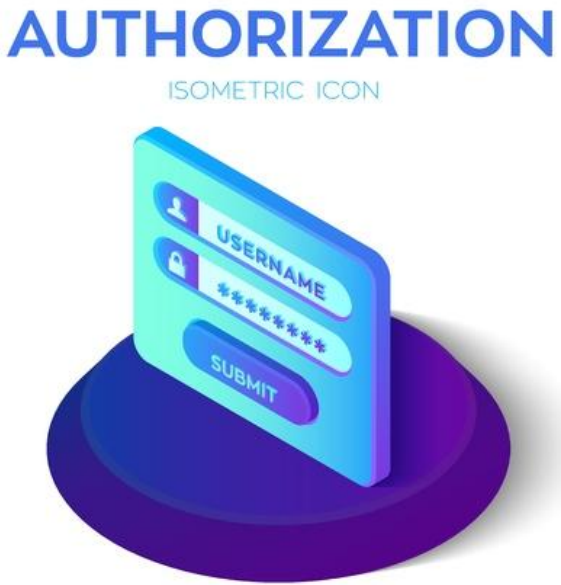
The account number is explicitly used to perform queries on the database. As there is no backend authorization check on this endpoint, the attacker can simply modify the value of account number to another user’s account and can retrieve the details of other users.

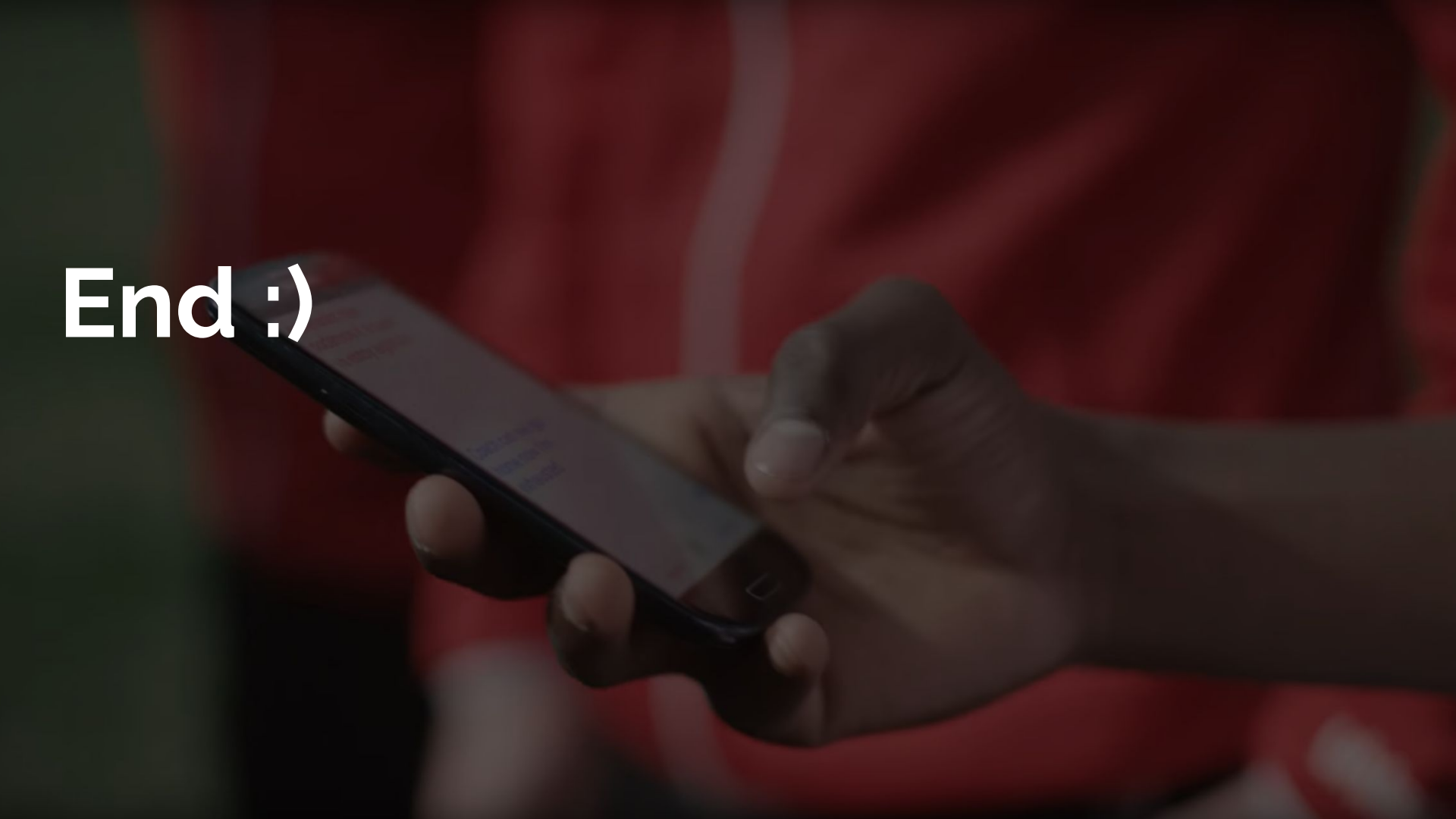


Securing Authorization

Securing Authorization

1. Never rely on obfuscation alone for access control.
2. Unless a resource is intended to be publicly accessible, deny access by default.
3. Wherever possible, use a single application-wide mechanism for enforcing access controls.
4. Thoroughly audit and test access controls to ensure they are working as designed.




End :)