A QUICK OVERVIEW

Of the other security vulnerabilities in the top 10 list

OWASP

1. Injection (SQLi)
2. Broken authentication and session management
3. Cross Site Scripting (XSS)
4. Direct object Reference
5. Security misconfiguration
6. Sensitive data exposure
7. Missing function level access control
8. Cross site request forgery (XSRF)
9. Using components with known vulnerabilities
10. Unvalidated redirects and forwards

Security misconfiguration

In order to set up a web application, you need to configure your web server

The sad truth is that there are many ways in which this can go wrong

Security misconfiguration

Running in debug mode in production

This might hit code paths which are not tested

Have error messages which have information about implementation details

Show additional stack trace details on errors

Security misconfiguration

Having directory listing enabled on the server

Giving the user more information increases the surface area of the attack

Showing her paths which exist but she may not encounter is unnecessary exposure

Security misconfiguration

Running outdated, unpatched software

Older versions of software might have serious security vulnerabilities

These are usually well known and can be used to attack your server

Security misconfiguration

Running unnecessary services

If you don't use a service, stop or uninstall it

Running services just increase your exposure to attacks

Services expose endpoints or APIs which may be used against you

OWASP

1. Injection (SQLi)
2. Broken authentication and session management
3. cross Site Scripting (XSS)
4. Direct object Reference
5. Security misconfiguration
6. Sensitive data exposure
7. Missing function level access control
8. cross site request forgery (XSRF)
9. Using components with known vulnerabilities
10. Unvalidated redirects and forwards

Sensitive data exposure

Sensitive data should be protected at all times, when stored or when in transit

Sensitive data exposure

In transit, use HTTPS with a proper certificate

In storage, encrypt data, hash passwords, separate hash keys and encrypted data

OWASP

1. Injection (SQLi)
2. Broken authentication and session management
3. cross Site Scripting (XSS)
4. Direct object Reference
5. Security misconfiguration
6. Sensitive data exposure
7. Missing function level access control
8. cross site request forgery (XSRF)
9. Using components with known vulnerabilities
10. Unvalidated redirects and forwards

Missing function level access control

This is an authorization failure

On the server side there are usually multiple APIs (functions) for specific actions

Missing function level access control

On the server side there are usually multiple APIs (functions) for specific actions

Do not rely on client side security to control access to these functions!

Missing function level access control

On the server side there are usually multiple APIs (functions) for specific actions

Server APIs may have multiple access points (web client, mobile client)

Missing function level access control

Server APIs may have
multiple access points
(web client, mobile client)

Users may also accidentally discover functions or URLs which they may not be authorized to view e.g /admin URL on a web site

Missing function level access control

Server APIs may have multiple access points (web client, mobile client)

Users may also accidentally discover functions or URLs which they may not be authorized to view e.g /admin URL on a web site

Server side authorization of APIs is absolutely a must!

OWASP

1. Injection (SQLi)
2. Broken authentication and session management
3. cross Site Scripting (XSS)
4. Direct object Reference
5. Security misconfiguration
6. Sensitive data exposure
7. Missing function level access control
8. cross site request forgery (XSRF)
9. Using components with known vulnerabilities
10. Unvalidated redirects and forwards

Using components with known vulnerabilities

There are multiple ways in which we end up with vulnerable services

Copy-pasting 3rd party code without thorough testing

Using old plugins or installations without updating or patching them

OWASP

1. Injection (SQLi)
2. Broken authentication and session management
3. Cross Site Scripting (XSS)
4. Direct object Reference
5. Security misconfiguration
6. Sensitive data exposure
7. Missing function level access control
8. Cross site request forgery (XSRF)
9. Using components with known vulnerabilities
10. Unvalidated redirects and forwards

Unvalidated redirects and forwards

Do not allow user redirects using GET parameters which are visible in the URL

This problem becomes worse when GET parameters are un-sanitized and un-validated

Unvalidated redirects and forwards

Do not allow user redirects using GET parameters which are visible in the URL

Users can be manipulated and redirected to sites which may install malware or to any other malicious page

OWASP

1. Injection (SQLi)
2. Broken authentication and session management
3. Cross Site Scripting (XSS)
4. Direct object Reference
5. Security misconfiguration
6. Sensitive data exposure
7. Missing function level access control
8. Cross site request forgery (XSRF)
9. Using components with known vulnerabilities
10. Unvalidated redirects and forwards