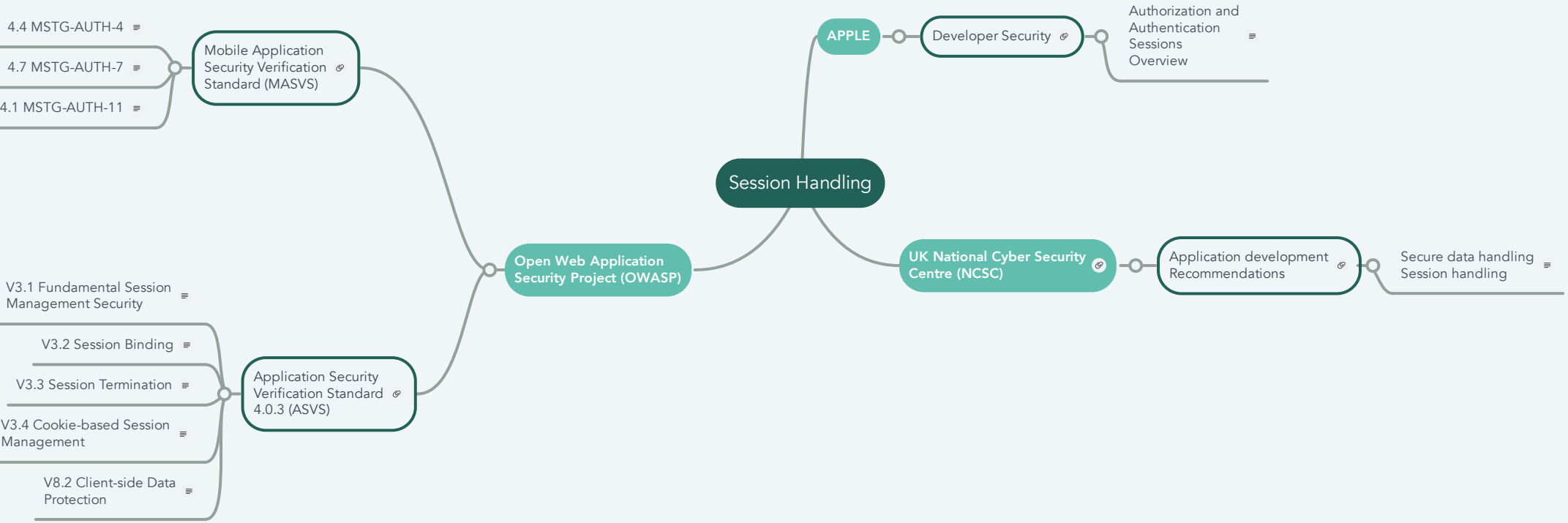
Session Handling
APPLE
Developer Security
Authorization and Authentication Sessions Overview
UK National Cyber Security Centre (NCSC)
Application development Recommendations
Secure data handling
Session handling
Open Web Application Security Project (OWASP)
Mobile Application Security Verification Standard (MASVS)
4.4 MSTG-AUTH-4
4.7 MSTG-AUTH-7
4.1 MSTG-AUTH-11
Application Security Verification Standard 4.0.3 (ASVS)
V3.1 Fundamental Session Management Security
V3.2 Session Binding
V3.3 Session Termination
V3.4 Cookie-based Session Management
V8.2 Client-side Data Protection

Session Handling

1. APPLE

1.1. Developer Security

Link: https://developer.apple.com/documentation/security

1.1.1. Authorization and Authentication Sessions Overview

> Use the Security.AuthSession API to work with session management and inquiry functions.

2. UK National Cyber Security Centre (NCSC)

Link: https://www.ncsc.gov.uk/collection/application-development/generic-application-development

2.1. Application development Recommendations

Link: https://www.ncsc.gov.uk/collection/application-development/generic-application-development

2.1.1. Secure data handling Session handling

> Session handling requires appropriate controls to be placed on the backend server to which the application connects.
> Ensure the backend server treats the application (and its user) as untrusted, until they can provide appropriate authentication.
> Ensure that sessions timeout periodically and require the user or application to repeat the authentication process.

3. Open Web Application Security Project (OWASP)

3.1. Application Security Verification Standard 4.0.3 (ASVS)

Link: https://raw.githubusercontent.com/OWASP/ASVS/v4.0.3/4.0/OWASP%20Application%20Security%20Verification%20Standard%204.0.3-en.pdf

3.1.1. V3.1 Fundamental Session Management Security

> 3.1.1 Verify the application never reveals session tokens in URL parameters.

3.1.2. V3.2 Session Binding

3.2.1 Verify the application generates a new session token on user authentication.

3.2.2 Verify that session tokens possess at least 64 bits of entropy.

3.2.3 Verify the application only stores session tokens in the browser using secure methods such as appropriately secured cookies (see section 3.4) or HTML 5 session storage.

3.1.3. V3.3 Session Termination

3.3.1 Verify that logout and expiration invalidate the session token, such that the back button or a downstream relying party does not resume an authenticated session, including across relying parties.

3.3.2 If authenticators permit users to remain logged in, verify that re-authentication occurs periodically both when actively used or after an idle period.

3.3.3 Verify that the application gives the option to terminate all other active sessions after a successful password change (including change via password reset/recovery), and that this is effective across the application, federated login (if present), and any relying parties.

3.3.4 Verify that users are able to view and (having re-entered login credentials) log out of any or all currently active sessions and devices.

3.1.4. V3.4 Cookie-based Session Management

3.4.1 Verify that cookie-based session tokens have the 'Secure' attribute set.

3.4.2 Verify that cookie-based session tokens have the 'HttpOnly' attribute set.

3.4.3 Verify that cookie-based session tokens utilize the 'SameSite' attribute to limit exposure to cross-site request forgery attacks.

3.4.4 Verify that cookie-based session tokens use the "__Host-" prefix so cookies are only sent to the host that initially set the cookie.

3.4.5 Verify that if the application is published under a domain name with other applications that set or use session cookies that might disclose the session cookies, set the path attribute in cookie-based session tokens using the most precise path possible.

3.1.5. V8.2 Client-side Data Protection

> 8.2.3 Verify that authenticated data is cleared from client storage, such as the browser DOM, after the client or session is terminated.

3.2. Mobile Application Security Verification Standard (MASVS)

Link: https://github.com/OWASP/owasp-masvs/releases/tag/v1.4.2

3.2.1. 4.4 MSTG-AUTH-4

> The remote endpoint terminates the existing session when the user logs out.

3.2.2. 4.7 MSTG-AUTH-7

> Sessions are invalidated at the remote endpoint after a predefined period of inactivity and access tokens expire

3.2.3. 4.1 MSTG-AUTH-11

> The app informs the user of all sensitive activities with their account. Users are able to view a list of devices, view contextual information (IP address, location, etc.), and to block specific devices.