MDN Web security docs outline

Background

At the Secure the Web Forward workshop in 2023, writers from Open Web Docs (OWD) led a discussion about the state of web security docs on MDN. One outcome from this was that OWD would draft an outline for some improved security documentation for MDN. This document presents that outline.

It's mostly taken from a couple of sources:

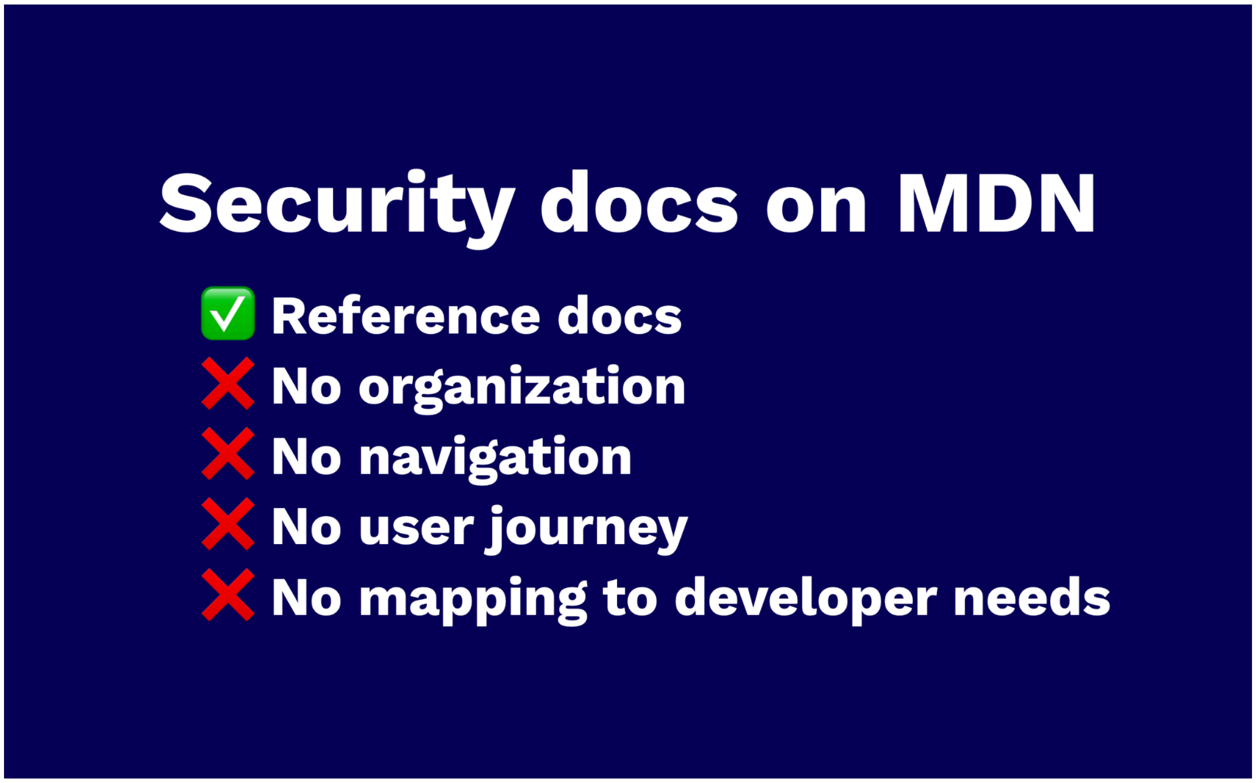
- https://cheatsheetseries.owasp.org/
- https://www.cyber.gc.ca/en/guidance/security-considerations-your-website-itsm60005
- (to do: also refer to https://github.com/OWASP/ASVS/tree/v4.0.3)

It really consists of two pieces:

- A high-level organization for web security docs on MDN.
- A set of individual docs to live within that organization.

Organization

In the workshop OWD observed that the existing docs on MDN don't have a clear organization:



Security docs on MDN

- ✅ Reference docs
- ❌ No organization
- ❌ No navigation
- ❌ No user journey
- ❌ No mapping to developer needs

In this outline we've created four categories of security docs:

- **Theory**: background/conceptual explanations, non-actionable.
- **Practices**: practices a web developer should follow.
- **Attacks**: specific types of attack.
- **Tools**: web security features that are intended to protect against types of attack.

Probably each item in each category is a page.

I don't know if we will necessarily want to document all of these things: or at least, we will prioritize them. I think "Practices" is the most important category because it connects with their needs, and that's probably the category we will concentrate on.

Theory

Same-origin policy

Secure product design

- least privilege, defense in depth, zero trust, ACL/RBAC models
- don't trust the client (e.g. don't rely on client-side input validation)

Practices

Web security 101

The idea for this came from the workshop, and it is that there are some fundamental things that you can do, that have a big impact on the security of the site. This includes:

- Use HTTPS for everything including third party resources
- Redirect HTTP to HTTPS
- Have a CSP
- Set good headers for cookies
- Don't set CORS headers without a good reason
- Validate all form data both client side and server side.
- Escape text and code
- Use a proven framework and follow its security directives
- Don't use GET for operations that change a database
- …?

It's quite important that this page is not exhaustive and is not too complicated: the idea is that these are relatively low-effort/high reward things. It's like: even if you don't do anything else, do this.

Authentication

- passwords: strength, storage, transmission, forgotten password flow
- supporting password managers
- MFA
- Taking care around auth responses/error messages
- login throttling
- CAPTCHA
- security questions
- passwordless protocols: OAuth, OpenID, …
- HTTP authentication
- Web Authentication API

Authorization/access control

- Checking REST calls, API keys
- Separation of users and admins
- Least privilege and default deny.

Session management

- Use session management toolkits
- Use random session identifiers
- Session expiry
- Cookies: secure cookie handling
- CSRF protection
- reauthenticate for high risk operations

Configuration

- turn off directory browsing
- remove unnecessary files (e.g. git files)
- disable unnecessary plugins and services
- Secrets management (don’t keep secrets in the repo, what to do if you did)
- Keep dev and live databases separate, turn off debug screens in live DB (see also)

Handling third-party software

- third-party JS (HTTPS, CSPs, SRI)
- npm (managing dependencies)
- Patching

User input validation

- Validation, sanitization, input hints
- Buffer overflows
- File upload

Operations

- Automate deployment
- Monitor
- Patching strategy

Attacks

We already have pages for many of these, so perhaps this is mostly a question of making sure they are up to date and in a consistent location.

Clickjacking

CSRF

Denial of service

Phishing/social engineering

SSRF

XSS

Man in the middle

Tools

CSP

CORS

HSTS

HTTPS

iframe sandboxing

Logging security events

SRI

TLS

Permissions policy