Web Application

- Same Origin Policy
  - Only accept requests from the same origin domain.
- CORS
  - Cross-Origin Resource Sharing. Can specify allowed origins in HTTP headers. Sends a preflight request with options set asking if the server approves, and if the server approves, then the actual request is sent (eg. should client send auth cookies).
- HSTS
  - Policies, eg what websites use HTTPS.
- Cert Transparency
  - Can verify certificates against public logs
- HTTP Public Key Pinning
  - Deprecated by Google Chrome
- Cookies
  - httponly - cannot be accessed by javascript.
- CSRF
  - Cross-Site Request Forgery.
  - Cookies.
- XSS
  - Reflected XSS.
  - Persistent XSS.
  - DOM based /client-side XSS.
  - `<img scr=””>` will often load content from other websites, making a cross-origin HTTP request.
- SQLi
  - Person-in-the-browser (flash / java applets) (malware).
  - Validation / sanitisation of webforms.
- POST
  - Form data.
- GET
  - Queries.
  - Visible from URL.
- Directory Traversal
  - Find directories on the server you’re not meant to be able to see.
  - There are tools that do this.
- API Security
  - Think about what information they return.
  - And what can be sent.
- BeEF Hook
  - Get info about Chrome extensions.
- User Agents
  - Is this a legitimate browser? Or a botnet?
- Browser Extension Takeovers
  - Miners, cred stealers, adware.
- Local File Inclusion

- Remote File Inclusion
  - Not as common these days
- SSRF
  - Server Side Request Forgery.
- Web Vuln Scanners
- SQLmap
- Malicious Redirects