HTML image tags

- Images are static content with no authority
- Any problems with images?

Social engineering: add logos to fool a user
- Impersonate site
- Impersonate credentials

Web security policy goals

- Safe to visit an evil web site
- Safe to visit two pages at one time
  - Address bar distinguishes them
- Allow safe delegation
  - Frame inside a frame
  - Each frame = origin of the content within it
  - Enforce same-origin policy

Background: Frames and iFrames

- Browser window may contain frames from different sources
  - Frame = rigid division as part of frameset
  - iFrame = floating inline frame
- Why use them?
  - Delegate screen area to content from another source
  - Browser provides isolation based on frames
  - Parent can continue to function even if frame is broken
Web application security model: **same-origin policy**

A browser permits scripts in one page to access data in a second page only if both pages have the same origin:

\[
\text{origin} = \{ \text{URI scheme}, \text{hostname}, \text{port number} \}
\]

- **Same origin**
- **Different origin**
  - https://www.poopybrain.com/index.html
    - different URI scheme (https)
    - different port
    - different host

**Ideas behind the same-origin policy**

- Each origin has client-side resources
  - Cookies: simple way to implement state
    - Browser sends cookies associated with the origin
  - DOM storage: key-value storage per origin
    - JavaScript namespace: functions & variables
  - DOM tree: JavaScript version of the HTML structure
- Each frame is assigned the origin of its URL
- JavaScript code executes with the authority of its frame’s origin
  - If cnn.com loads JavaScript from jQuery.com, the script runs with the authority of cnn.com
- Passive content (CSS files, images) has no authority
  - It doesn’t (and shouldn’t) contain executable code

**Mixed content: http & https**

- HTTPS page may contain HTTP content:
  `<script src="http://www.mysite.com/script.js"> </script>`
    - Active network attacker may now hijack the session
    - Content over the network is plain text
- Safer approach
  `<script src="//www.mysite.com/script.js"> </script>`
    - Served over the same protocol as the embedding page (frame)
- Some browsers warn you of mixed content
  - Some warning may be unclear to the user

**Extended Validation Certificates**

For SSL/TLS authentication to be meaningful, the server’s X.509 certificate must belong to the party the user believes it belongs to:

- **Domain validated certificates**
  - Only require proof of domain control
  - Do not prove that a legal entity has a relationship with the domain
- **Extended validation (EV) certificates**
  - Belong to the legal entity controlling the domain (or software)
  - Certificate Authority must validate the entity’s identity
  - More stringent validation: check company incorporation, domain registration, position of applicant, etc.

**Extended Validation Certificates**

EV certificate will contain
- Government-registered serial number
- Physical address
- ... 

**Extended Validation Certificates**

- Browsers would show a lock icon for any SSL/TLS connection
  - [www.cs.rutgers.edu](http://www.cs.rutgers.edu)
- This led to a false sense of security
  - Fraud sites would use TLS to let users think they are legitimate
- Modern browsers
  - Identify & validate EV certificates
  - Present a security indicator that identifies the certificate owner
  - [JPMorgan Chase and Co. www.chase.com](http://www.chase.com)
Browser Status Bar

Mouseover shows link target

https://www.paypal.com/signin/

Trivial to spoof with JavaScript

<a href="http://www.paypal.com/signin"
   onclick="this.href = 'http://www.evil.com/';">
   PayPal
</a>

The end