Chapter 9
Web and Browser Security

9.1 Web review: domains, URLs, HTML, HTTP, scripts ......................... 246
9.2 TLS and HTTPS (HTTP over TLS) ............................................ 252
9.3 DOM objects and HTTP cookies ............................................. 255
9.4 Same-origin policy (DOM SOP) ............................................... 257
9.5 Authentication cookies, malicious scripts and CSRF ...................... 260
9.6 More malicious scripts: cross-site scripting (XSS) ....................... 262
9.7 SQL injection ........................................................................... 266
9.8 Usable security and the web ..................................................... 269
9.9 End notes and further reading ................................................ 274
References ................................................................................ 276
Chapter 9

Web and Browser Security

We now aim to develop an awareness of what can go wrong on the web, through browser-server interactions as web resources are transferred and displayed to users. When a browser visits a web site, the browser is sent a page (HTML document). The browser renders the document by first assembling the specified pieces and executing embedded executable content (if any), perhaps being redirected to other sites. Much of this occurs without user involvement or understanding. Documents may recursively pull in content from multiple sites (e.g., in support of the Internet’s underlying advertising model), including scripts (active content). Two basic security foundations discussed here are the same-origin policy (SOP), and how HTTP traffic is sent over TLS (i.e., HTTPS). HTTP proxies and HTTP cookies also play important roles. As representative classes of attacks, we discuss cross-site request forgery, cross-site scripting and SQL injection. Many aspects of security from other chapters tie in to web security.

As we shall see, security requirements related to browsers are broad and complex. On the client side, one major issue is isolation: Do browsers ensure separation, for content from unrelated tasks on different sites? Do browsers protect the user’s local device, filesystem and networking resources from malicious web content? The answers depend on design choices made in browser architectures. Other issues are confidentiality and integrity protection of received and transmitted data, and data origin authentication, for assurance of sources. Protecting user resources also requires addressing server-side vulnerabilities. Beyond these are usable security requirements: browser interfaces, web site content and choices presented to users must be intuitive and simple, allowing users to form a mental model consistent with avoiding dangerous errors. Providing meaningful security indicators to users is among the most challenging problems.

9.1 Web review: domains, URLs, HTML, HTTP, scripts

We first briefly review some essential web concepts. The Domain Name System (DNS) defines a scheme of hierarchical domain names, supported by an operational infrastructure. Relying on this, Uniform Resource Locators (URLs), such as those commonly displayed