Chapter 20
Web Security and Single Sign-On Protocols

Abstract *Single Sign-On* protocols (SSO) are used to grant a web user authenticated access to many web applications after a single manual login. On the client side, SSO protocols like OAuth, OpenID Connect, or SAML rely on standard web browser features. The blueprint for modern SSO protocols is the *Kerberos protocol* [56] described in chapter 14.

An existing login to Google can be used to log in to another web application with a single mouse click on the corresponding "Sign in with Google" button (Figure 20.1). These widely used SSO protocols are based on standards such as OAuth, OpenID Connect, or SAML.

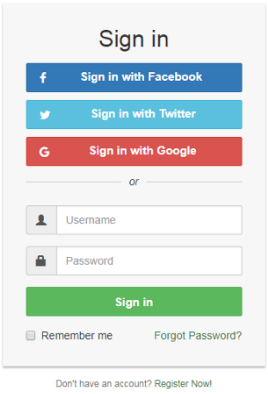

Fig. 20.1 Single-Sign-On dialogue for a web application.

SSO protocols differ from the cryptographic protocols presented so far: they do not have their own client-side implementation but have to rely on features provided by modern web browsers. Typically, cryptographically secured messages, serialized JSON or XML files, are transmitted as HTML forms, URLs, or HTTP cookies. We will, therefore, briefly introduce these browser features. Since the security of SSO

J. Schwenk, *Guide to Internet Cryptography*, Information Security and Cryptography,
https://doi.org/10.1007/978-3-031-19439-9_20

is directly linked to the security of web applications, we will also describe the most critical attacks on such applications: Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), and SQL Injection (SQLi). A presentation of the most important SSO protocols in use today concludes the chapter.

20.1 Web Applications

A web application is a distributed client-server application whose application logic is split between the web browser as the client and a web server. Communication is mainly done via HTTP. Smartphone apps can also be implemented as web applications – in this case, the GUI is realized with HTML5, and the system browser of Android or iOS is used for the app's communication with a server.

The basic building blocks of web applications are HTTP (section 9.2) and HTML [16]. In addition to the actual HTML markup, a web page contains active script code (JavaScript) and precise layout instructions (Cascading Style Sheets). Both operate on an abstract object model of the web page, the *Document Object Model*. As a central security mechanism, the *Same Origin Policy* is intended to prevent foreign scripts from reading or modifying sensitive data of a web page (e.g., HTTP session cookies, passwords).

20.1.1 Architecture of Web Applications

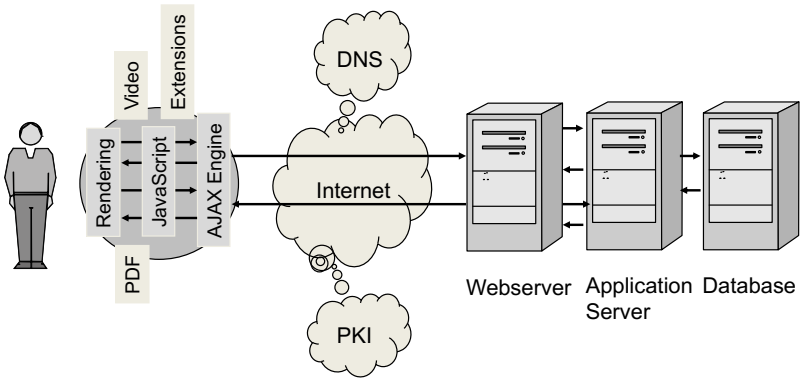

Fig. 20.2 Components of a Web Application.

A modern web application consists of many components. A typical scenario is given in Figure 20.2. On the client side, the browser implements various parsers