Sicherheitsrisiken mit aktiven Webseiten

Mit dem World Wide Web, das 1989 im CERN entworfen wurde, entstand ein Netz miteinander verbundener Hypertext-Dokumente. Der Benutzer kann sich über die Verweise in den Dokumenten im Hypertext-Raum bewegen. Der Nachteil dieser Arbeitsweise bestand darin, daß der Benutzer wenig Möglichkeiten der Interaktion hatte. Deshalb waren spätere Erweiterungen darauf gerichtet, dieses Medium interaktiv benutzbar zu machen. Dazu ist es notwendig, daß der Benutzer Eingaben in das System vornehmen kann, die mit Hilfe von Programmen verarbeitet werden und entsprechende Reaktionen und Antworten erzeugen. Die erste Möglichkeit wurde mit der Einführung der Formularbefehle und des „Common Gateway Interface" (CGI) geschaffen. Hierbei werden die Daten, die der Benutzer in das Formular einträgt, über das Netz zum Server transportiert. Auf diesem wird dann ein Programm gestartet, das die Verarbeitung übernimmt. Es soll hier nicht weiter auf dieses Prinzip eingegangen werden, da die Sicherheitsprobleme vor allem im Server auftreten und vom Administrator gelöst werden müssen. Der Benutzer solcher Formulare sollte nur bedenken, daß die eingegebenen Daten im allgemeinen ungeschützt über das Netz transportiert werden. Es besteht dadurch die Möglichkeit, daß sie während des Transportes gelesen und auch geändert werden können.

Ein weiterer Schritt in der Entwicklung des WWW bestand darin, daß die Möglichkeit geschaffen wurde, Programme in die Dokumente einzubinden. Sie werden beim Aufruf der entsprechenden Seiten vom WWW-Server zum lokalen Rechner transportiert und häufig, ohne den Nutzer davon zu informieren, dort gestartet. Dies birgt natürlich die Gefahr, daß Programme geladen werden, die wissentlich oder unwissentlich Funktionen enthalten, die den lokalen Rechner schädigen oder vertrauliche Informationen dieses Rechners weitergeben. Deshalb müssen Sicherheitsmechanismen eingeführt werden, die dies verhindern. Für die Erstellung der Programme werden die Sprachen Java und JavaScript eingesetzt. Nachfolgend sollen die Besonderheiten dieser Sprachen hinsichtlich der Sicherheit gezeigt werden.

Java

Java wurde als allgemeine, objektorientierte Programmiersprache von der Firma Sun Microsystems entwickelt. Ein großer Vorteil der Sprache besteht darin, daß die Programme, die mit Hilfe dieser Sprache geschrieben wurden, auf (fast) allen Rechnertypen direkt abgearbeitet werden können. Das wird dadurch erreicht, daß der Quellcode in einen rechnerunabhängigen Code (Bytecode) übersetzt wird. Dieser wird dann von einem virtuellen Rechner (Java Virtual Machine) interpretiert und dadurch abgearbeitet. Bei der Definition der Programmiersprache wurde darauf geachtet, daß sie möglichst wenig Elemente enthält, die zu fehlerhaften Programmen führen können. Andererseits soll es aber möglich sein, Java-Programme zu erstellen, die beliebige Funktionen im Computer ausführen können. Dazu ist es notwendig, daß auch auf die lokalen Ressourcen (lokale Festplatte, Netzwerk u. a.) zugegriffen werden kann. Es wäre aber ein sehr hohes Risiko, solche Programme aus dem anonymen Internet zu laden. Deshalb wurden den Programmen, die in den WWW-Dokumenten eingebettet sind (sog. Applets), bisher folgende Beschränkungen auferlegt:

- Applets dürfen keine Dateien auf dem lokalen Rechner lesen oder schreiben.
- Applets können keine Netzwerkverbindungen zu anderen Rechnern als dem, von dem sie geladen wurden, aufbauen.
- Applets können keine Programme starten.
- Applets können keine Programmbibliotheken laden.
- Applets haben nur begrenzten Zugriff zu den Systeminformationen des lokalen Rechners.

Diese auch als „Sandbox" bezeichnete Methode soll dazu führen, daß die von einem unbekannten Server des Internets geladenen Programme keinen Schaden auf dem lokalen Rechner anrichten können. Die Überwachung der oben genannten Einschränkungen übernimmt der Browser, in dem ein Java-Interpreter (Bytekode-Interpreter) integriert ist. Damit wird auch deutlich, daß die Sicherheit bei dieser Methode davon abhängt, wie sorgfältig diese Überwachungsfunktionen im Browser implementiert wurden. Die Vergangenheit hat gezeigt, daß immer wieder Sicherheitslücken in einzelnen Browserversionen aufgetreten sind, die dazu führen konnten, daß spezielle Applets Zugriff auf die lokale Festplatte erhielten oder andere unerwünschte Funktionen ausführten [1] [2].

Die Sandbox-Methode hat aber auch den Nachteil, daß eventuell nützliche Funktionen mit Applets nicht realisiert werden können, wenn dazu Zugriffe auf die lokalen Ressourcen des Rechners benötigt werden. Deshalb hat die Firma Netscape für ihren Browser der Version 4 (Communicator 4.0x) die Sicherheitsstrategie geändert. Die Applets können eine digitale Unterschrift vom Hersteller erhalten. Diese Unterschrift wird aus einer Art Prüfsumme vom Programm und einem persönlichen Schlüssel des Unterzeichners gebildet. Dadurch kann ein unterschriebenes Programm nachträglich nicht unbemerkt geändert werden. So unterschriebene Applets können dann einen erweiterten Zugriff auf lokale Ressourcen anfordern. Der Browser erkennt bei der Sicherheitsprüfung diese Funktionen und meldet sie dem Benutzer, wobei gleichzeitig eine

Information zur Unterschrift und zum Unterzeichner angezeigt wird (siehe Bild). Der Benutzer kann dann entsprechend seinem Vertrauen zum Programmierer entscheiden, ob diese Funktion ausgeführt werden soll. Der Vorteil dieser Methode besteht darin, daß es unwichtig ist, auf welchem Server sich das Objekt befindet und auf welchem (unsicheren) Wege es zum Benutzer gelangt, da es durch diese Unterschrift vor Veränderung geschützt ist. Die Schwierigkeit für den Leser besteht darin einzuschätzen, welchem Programmierer von Applets er vertrauen kann.

JavaScript

JavaScript ist eine objektbasierte Sprache, die zur Einbindung von Programmen in HTML-Dokumenten dient. Sie wurde von der Firma Netscape zunächst unter dem Namen Live-Script entwickelt. Ziel war es, eine Programmiersprache zu schaffen, die es auch nichtprofessionellen Programmierern erlaubt, schnell einfache Programme zu schreiben. Um eine gewisse Ähnlichkeit mit der Sprache Java zu suggerieren, wurde sie später in JavaScript umbenannt. Trotz mancher vergleichbarer Eigenschaften gibt es auch entscheidende Unterschiede. Der wohl auffälligste besteht darin, daß JavaScript als Quelltext (Script) in die HTML-Dokumente eingebunden wird. Diese Scripte werden nach dem Einlesen des Dokuments vom Browser interpretiert und abgearbeitet. Die Sicherheitsstrategie besteht darin, daß die Sprache keine Elemente enthält, die einen Zugriff auf die lokale Festplatte ermöglichen. Trotzdem sind auch hier Sicherheitslücken aufgetreten. Diese sind aber mehr dem Bereich der Verletzung der Privatsphäre zuzuordnen, d. h. es können Informationen über den lokalen Rechner oder über den Nutzer unbemerkt abgerufen werden. So haben Mitarbeiter der Bell Labs im Juli diesen Jahres einen Fehler in den Browsern von Microsoft wie auch Netscape entdeckt, der dazu führt, daß von einem Dokument ein JavaScript-Programm gestartet werden kann, das, selbst nachdem diese Seite verlassen wurde, die Internet-Aktivitäten des Benutzers sammelt und versendet [3]. Dabei kann protokolliert werden, welche Seiten (URL) später besucht wurden. Weitaus gefährlicher ist aber, daß auch das Ausfüllen von HTML-Formularen überwacht werden kann. Da dies direkt auf dem lokalen Rechner des Nutzers geschieht, kann dieser Eingriff auch nicht durch einen Firewall-Rechner oder durch Verschlüsselung der Formulardaten während der Datenübertragung verhindert werden. In den neuesten Versionen der Browser wurde dieser Fehler beseitigt (Keine Lösung gibt es für Netscape Navigator Version 2.0x.).

Zusammenfassend kann festgestellt werden, daß trotz aller Begrenzungen für die Internet-Programme zur Zeit nicht mit Sicherheit verhindert werden kann, daß Java- oder JavaScript-Programme, die gemeinsam mit den Dokumenten geladen und automatisch gestartet werden, dem Nutzer schaden können. Deshalb sollten folgende Hinweise beachtet werden:

- Bei Computern, die sicherheitsrelevante Daten enthalten, sollten die Funktionen Java und JavaScript im Browser ausgeschaltet werden.
- Wer Java und JavaScript nutzen will, sollte immer die neueste Version eines Browser installieren, da dann zumindest die bekannten Sicherheitslücken beseitigt sind.
- Jede Warnung oder Meldung des Browsers sollte aufmerksam gelesen werden, bevor sie akzeptiert wird.

Die Informationen zur Sicherheit beim Browsen in HTML-Dokumenten gelten auch für das Lesen von Mails mit Hilfe des Netscape Navigators. Es können vollständige HTML-Dokumente inklusive entsprechender Java- oder JavaScript-Programme in eine Mail integriert werden. Wenn sie vom Mail-Programm des Browsers geöffnet werden, werden diese Programme gestartet.

Cookies

Eine weitere Funktion im WWW, die häufig Anlaß für Diskussionen zur Sicherheit bietet, sind die sog. „Cookies“. Das sind kleine Informationseinheiten, die vom WWW-Server an den Browser gesendet werden. Darin sind enthalten der Domain-Name des Servers, der Pfad auf dem Server, ein Verfallsdatum, der Name des Eintrags und ein variabler Teil. Diese Information wird zunächst im Browser gespeichert. Erst wenn der Browser geschlossen wird und das Verfallsdatum noch nicht erreicht wurde, wird ein Cookie auf die Festplatte des lokalen Rechners gespeichert (Datei cookies.txt unter MS Windows oder cookies unter UNIX). Bei jedem Verbindungsaufbau mit einem Server, dessen Cookies gespeichert sind, werden diese zum Server übertragen. Damit soll ein Nachteil des Übertragungsprotokolls (HTTP) überwunden werden, der darin besteht, daß es keine permanente Verbindung zwischen WWW-Server und lokalem Rechner gibt. Jeder neue

Aufruf einer Seite oder das Absenden eines ausgefüllten Formulars ist eine eigenständige Übertragung, wobei keinerlei Informationen zu vorherigen Verbindungen mitgeliefert werden. Hierdurch ist es schwierig, komplexe, mehrstufige Formulare aufzubauen. Der Einsatz von Cookies ermöglicht es, den Zustand einer Verbindung zu speichern und bei der nächsten Datenübertragung an den Server zu senden, so daß seine Reaktion sowohl von der aktuellen als auch von vorherigen Anfragen abhängen kann.

Die Gefahren durch Cookies bestehen darin, daß auf dem Server eine Statistik über den Besuch der einzelnen Seiten und somit über die Vorlieben des Benutzers geführt werden kann. Weiterhin werden dadurch Übertragungs- und Speicherkapazität gebunden. Durch die Definition einer maximalen Anzahl und Größe von Cookies wird dieses Problem aber begrenzt [4]. Abgesehen von der oben erwähnten Statistik stellen Cookies kein Sicherheitsrisiko dar. Trotzdem bieten die modernen Browser von Microsoft und Netscape die Möglichkeit, das Schreiben von Cookies zu verbieten.

Literatur:

[1] ZDNet News: Vorsicht: Platten-Crash beim Explorer.
http://www.pcpro.de/news/artikel/1997/09/09002-wf.htm

[2] Sun Microsystems, Inc.: Chronology of security-related bugs.
http://www.javasoft.com/sfaq/chronology.html

[3] Vinod Anupam: JavaScript Related Browser Vulnerability.
http://www-db.research.bell-labs.com/user/anupam/vulnerability/

[4] Netscape Communications Corporation: Persistent Client State HTTP Cookies.
http://home.netscape.com/newsref/std/cookie_spec.html

Lothar Wendroth