

## Cross-Site Scripting

*Haben Sie auch schon irrtümlich Post bekommen? Sei es durch die Verwechslung des Briefkastens oder weil eine Familie mit gleichem Nachnamen in der Umgebung wohnt? Wahrscheinlich schon – doch das ist nicht so schlimm, denn die Post macht den Fehler nicht systematisch und voraussehbar. Was im realen Leben zufällig passiert, kann in der virtuellen Welt provoziert werden!*

*Ein Bericht von Ivan Bütler*

Wenn Sie im Internet Ihr Online Banking nutzen, dann identifiziert Sie Ihre Bank zu Beginn Ihrer Tätigkeit durch starke Authentisierungsmaßnahmen. Nach eingehender Prüfung ihrer Identität, stellt man Ihnen fuer die Dauer der Nutzung ein Ticket aus.

Die Nutzung des Online Banking Ticket verhält sich sehr ähnlich zu einem Ticket beim Stadttheater. Sollten Sie die Vorstellung zwischendurch verlassen müssen, dann identifizieren Sie sich beim erneuten Eintreten in das Theater durch das Vorzeigen des Tickets. Das Ticket ist lediglich fuer die aktuelle Vorstellung gültig und verfällt am nächsten Tag. Auf eine vollständige Überprüfung Ihrer Identität wird verzichtet.

Fällt das Ticket während Ihrer Absenz in die Hände eines Dritten, so kann dieser problemlos in das Theater eintreten und Ihren Platz einnehmen, auch wenn sich die Sitznachbarn wundern werden.

Der Zugang zum Online Banking System ist nach erfolgreicher Identifikation (Authentisierung) ebenfalls an ein Ticket gebunden, das der Browser bei jedem Request auf den Server mitschickt. Dieses gilt es für die Dauer der Sitzung zu schützen und sicher aufzubewahren.

Stellen wir uns zwei Postkunden vor. Zum einen Herrn Meier, ein gutbürgerlicher Kunde, der die Post als Dienstleistung schätzt und nutzt. Zum anderen Herr Keller, der die Dienstleistungen der Post zu untergraben versucht und Angriffe auf die Post und Herrn Meier nicht scheut.

Nehmen wir an, dass die Post einen extrem grossen Einfluss auf Herrn Meier hat. Sagen wir, dass Herr Meier der Post „hörig“ ist. Wenn die Post Herrn Meier auffordern würde, sein Online Ticket an Herrn Keller zu senden, dann würde dies Herr Meier tun. In der realen Postwelt ist dies unwahrscheinlich, doch in der virtuellen Umgebung ist eine derartige Attacke durch Cross-Site Scripting möglich. Wir sprechen in diesem Fall von **Server-Side Cross-Site Scripting (SS-CSS)**.

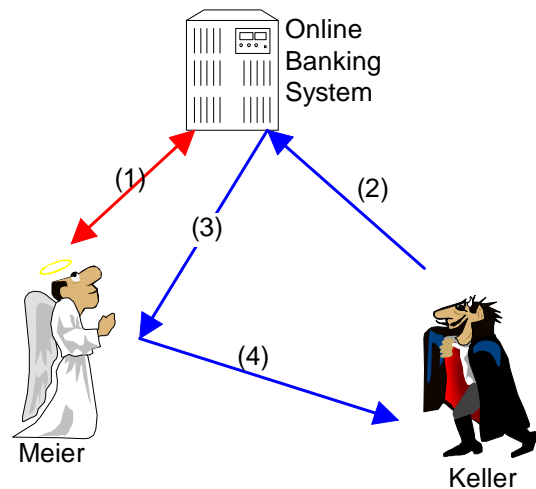


Abb. 1.0

Herr Keller nimmt in keinem Zeitpunkt direkten Kontakt mit Herrn Meier auf. Die Attacke von Herrn Keller zielt darauf ab, der Post den „Befehl“ einzuflüstern (2), dass Herr Meier das Ticket versenden soll (4) – am besten an Herrn Keller. Zuvor loggt sich Herr Meier mit (1) am System an, um das Ticket zu kriegen.

Server Side Cross-Site Scripting Sicherheitslücken sind in jüngster Zeit vermehrt im Internet bekannt geworden. Diverse Produkte sind anfällig auf solche „Einflüsterversuche“. Die Verantwortung für die Behebung dieser Schwachstelle liegt beim Betreiber der E-Business Anwendung.



Im Gegensatz zur oben beschriebenen Methode könnte man Herrn Meier dazu überreden, jegliche Post zu kopieren und selbständig an Herrn Keller auszuliefern.

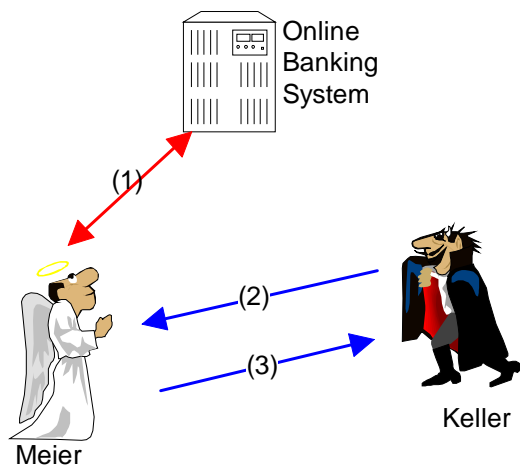


Abb. 2.0

Der Angreifer nützt dabei Sicherheitslücken der Browser Software (Internet Explorer, Netscape) aus. Die Browser senden vertrauliche Inhalte (Tickets) an Dritte (Keller) und untergraben damit die Online Bank Sicherheit. Dieses Verhalten wird als **Client-Side Cross-Site Scripting (CS-CSS)** bezeichnet.

Herr Keller muss jedoch Kontakt mit Herrn Meier aufnehmen. Die Kontaktaufnahme wird in der Realität durch anonyme HTML Mails realisiert. Oftmals sind Cross-Site Scripting Attacken auch in SPAM Mails eingewoben.

Die Verantwortung für die Behebung der Schwachstelle liegt beim Anwender der Client Software (IE, Netscape, Outlook, NotesClient).

## ActiveScripting

Cross-Site Scripting basiert auf böartigem Code in Script Sprachen, die von aktuellen Browsern und Mailclients (Internet Explorer, Netscape, Outlook) interpretiert werden können. Typischerweise sind das JavaScript, VBScript oder ganz einfach HTML Tags, welche aktive Komponenten wie ActiveX oder Applets einbinden.

Im einfachsten Fall sendet der Angreifer ein HTML Mail an sein Opfer, wobei der böartige Code im Mail enthalten ist. Eine weitere Möglichkeit besteht in der Platzierung von „böartigem Code“ in Webseiten. Dazu sucht der Angreifer nach Stellen auf einer Webseite, die es ihm ermöglicht Daten zum Server zu senden. MailForms, Chat Rooms, GuestBooks und ähnliches sind besonders gefährdet. Aber auch „normale“ E-Business Anwendungen sind nur dann sicher, wenn die Eingaben des Benutzers geprüft werden.

Eine weitere Möglichkeit des Angreifers besteht darin, dem Opfer eine E-Mail mit Link zu senden. Der Link zeigt auf eine nicht-existierende URL beim Online Banking Provider und enthält als Argumente Scripting Tags.

[http://XXX/not\\_found<SCRIPT>alert\(,TEST'\)</SCRIPT>.html](http://XXX/not_found<SCRIPT>alert(,TEST')</SCRIPT>.html)

Sobald der Benutzer diesen Link wählt, versucht der Browser die nicht existierende Seite vom Server zu laden. Entscheidend ist nun die Reaktion des Servers. Enthält die Fehlermeldung den Aufruf inklusive <SCRIPT>, dann entspricht dies im Verhältnis zum realen Post-Beispiel der Aufforderung an Herrn Meier, das Ticket an Herrn Keller zu senden. Herr Meier wird dies tun, da die Aufforderung von der offiziellen Post kam. In diesem Fall sprechen wir von Server-Side Cross-Site Scripting.

## Schutzmassnahmen

Die Anbieter von E-Business Anwendungen sollten die Dateneingaben der Benutzer sorgfältig prüfen, den aktuellen Security Patches einspielen und Cookie Settings durch Angabe „PFAD“ restriktiv einschränken.

Die Endbenutzer und Anwender von Browser und Mailclients sollten HTML Mails misstrauisch gegenüberstehen. Es ist von der gleichzeitigen Nutzung von E-Business (Online Banking) und dem Surfen auf anderen Webseiten und Mailing abzuraten. Ebenfalls sollten die Browserversionen aktuell gehalten werden, was jedoch in der Praxis nicht so einfach ist...