

Michael Hnat

Schutz vor Website-Hacking



Alle Patches installiert, unnötige Ports gesperrt: Der Webserver ist sicher. Aber ist auch die Website sicher? Manche Fehler in der Programmierung können fatale Folgen haben und liefern im Ergebnis manipulierbare Datenbanken oder Webseiten. MX Magazin zeigt Ihnen, wo die Fallen lauern und wie Sie sich schützen.

Sicherheitsproblematiken sind schwierig zu beschreiben: Auf der einen Seite müssen Entwickler wissen, welche Probleme es gibt, um sichere Websites zu programmieren. Andererseits können Entwickler die Problematik erst begreifen, wenn Sie die Folgen falscher Programmierung deutlich vor Augen haben – und hier öffnet sich möglichen Angreifern Tor und Tür. Die im Folgenden aufgezeigten Probleme sollen Entwicklern helfen: Trotzdem ist es nicht zu verhindern, dass feindlich gesinnte Angreifer die hier besprochenen Hinweise für ihre widerrechtlichen (!) Angriffe nutzen. Um keine Website zu kompromittieren, haben wir in diesem Artikel auf Beispiel-Screenshots verzichtet.

Man kann sehr schnell selbst Opfer feindlicher Angriffe werden, und die-

se können ernsthafte Konsequenzen haben. Gegen die hier besprochenen Angriffe kann man sich leicht wehren: Dafür ist jedoch die Einsicht nötig, dass man Sicherheit nur über eine fundierte Kenntnis der Lücken erlangen kann. Dieser Artikel enthält viele Konjunktive, viel „Wenn“ und „Aber“. Die gezeigten Lücken müssen nicht unbedingt auftreten. Viele Programmiersprachen lösen einige Probleme automatisch – darauf sollte man sich aber nicht verlassen. Es ist wichtig, ein Bewusstsein für solche Probleme zu entwickeln und seine Anwendungen im Vorfeld ausgiebig zu testen.

SQL-Injection. Unter SQL-Injection versteht man die Kompromittierung von SQL-Anweisungen. Eine SQL-Abfrage lässt sich mit einfachen Mitteln so manipulieren, dass zum Beispiel die Änderung von Datenbankeinträgen oder sogar das Löschen von Datenban-

ken möglich sind. Alles, was ein der Website feindlich gesinnter Angreifer dazu benötigt, sind ein Formular oder ein URL sowie ein Entwickler, der manipulierbaren Code programmiert hat.

SQL-Injection beschränkt sich nicht auf eine Plattform. Es handelt sich dabei um ein grundlegendes Problem, dass bei nahezu allen Programmiersprachen und allen Datenbanken in irgendeiner Art und Weise funktioniert. Die Beispiele sind zwar alle für CFMX programmiert, lassen sich jedoch ebenfalls bei den Serversprachen ASP, PHP etc. mit verschiedenen Datenbanken teilweise nachvollziehen.

Problematik. Das Problem der SQL-Injection liegt nicht in irgendwelchen Sicherheitslücken des Webserver, sondern in Fehlern in der Programmierung. Grundlage ist eine häufig vorkommende SQL-Abfrage der nachfolgenden Art:

```
WebCode GR0438
```