Blinde SQL - Injektion - Gefahren und Maßnahmen

== Beschreibung ==
Die Lücken in den Codierungen von SQL sind schon relativ bekannt. Weniger bekannt hingegen, ist eine Technik, namens „blind exploitation“ zu deutsch „blindes Ausnutzen“. Ich möchte mich hierbei nicht auf die gesamte Bandbreite der Informationen über dieses Thema stürzen, sondern auf die Entdeckungstechniken und Wiederherstellung von Daten in einer uns nicht bekannten Umgebung.(hört sich irgendwie konfus an oder? ;). Da stellt sich die Frage, wie bestimme ich jetzt, ob meine durchgeführte Abfrage das Ergebnis aufweist, welches ich erwartet habe. Aus dieser Fragestellung, resultiert das „blind progression “, blindes Fortschreiten.

== „blind progression “ blindes Fortschreiten ==
Wir haben herausgefunden, dass unsere Seite lückenhaft, bezogen auf SQL ist. Die Argumente dieser Seite werden über URL übertragen (method="get"). Das Feld, welches eine Lücke aufweist, ist ein digitales und dessen magic quotes sind aktiv. Bei dem Versuch etwas einzuschleusen, bekamen wir eine Error – Message, durch dessen Aussage, können wir nun die Lücke identifizieren.Man kann z.B. den Wert OR 1=1 angeben, doch dann ist es unmöglich festzustellen, ob das Ergebnis positiv oder negativ ist. Es scheint, als könne man die gefundene Lücke im System nicht nutzen, zu mindestens wenn man den timing – Angriff umleitet.Das mit dem timing – Angriff ist eh schon so eine eigenartige Sache, dieser beschreibt die Zeit, die das von Server ausgeführte Skript gebraucht hat, um irgend eine Seite aufzubauen. Wenn dieses Skript nun eine Anfrage an die SQL stellt, die sehr Zeitaufwendig ist, wird die Ladezeit der Seite beeinträchtigt. Man kann also eine Abfrage einschleusen und diese durch ein bestimmtes Kriterium testen. Je nach dem, welches Ergebnis wir erhalten, rufen wir einen Vorgang auf, der viel Zeit in Anspruch nimmt. Jetzt vergleicht man die Ladezeit einer normalen Seite, mit unserer künstlich injizierten Abfrage. Wenn die Zeiten gleich bleiben, dann wurden auch die Bedingungen geprüft. In einem Skript bedeutet es „injection for F()“. Um diese Tests durchzuführen, brauchen wir eine empfindliche Seite, welche diese Art simuliert.

Hier geht es weiter -> [wiki]Empfindliches PHP - Skript[/wiki]