BSQL Hacker FAQ

Ferruh Mavituna (ferruh@mavituna.com)

Why there are IsNull(), NVL(), Ifnull() and similar NULL checking in attack files ?

BSQL Hacker is using a different length detection algorithm than other tools. Basically it's looking for NULL results returning from substring and similar stuff. If char identified as NULL then it believes that it found the length and return the results.

Thus it needs to check for NULL results and convert them to 0 as integer to get length detection correctly. To be honest this is an error prone way to detect length but it's generally faster and easier then doing a COUNT() or LENGTH() query.

Why is it bloody slow while exploiting Error Based stuff in Automated Attack mode?

Normally it should be so much faster than that. But Automated Attack module originally developed for Blind SQL Injection thus they wait to finish previous checks before moving on. So it's not multithreaded in enumerating it's multithreaded internally while getting the data.

Normally in error based injection you don't need multithreading internally. Error-based injection added BSQL Hacker as kind of a hack.

As a result of this combination it's slower in Error Based SQL Injections than it supposed to be. Unfortunately this is not planned to be fixed quite soon.

Why am I getting wrong results in Full Blind and Deep Blind SQL Injections?

Application database layer has limits for multiple connections to databases. While exploiting SQL Injection in **Full Blind** and **Deep Blind** every request takes couple of seconds and keeps database connection alive. When you go over parallel connection limit of database, application doesn't send your request to the database before a new slot available. This causes wrong results in time based attacks because BSQL Hacker can't figure out **true / false** responses since all of them returning longer then expected.

Thus while using time based attacks you shouldn't go over **10 threads** unless you know the application settings. Going over database pooling / parallel connection limits will result with wrong responses and this is not avoidable in any other way.

What is an Attack File?

Attack File is the name of save files which you can load to BSQL Hacker. An Attack File can be an exploit for a specific vulnerability (i.e. *PHPNuke v.66 Blind SQL Injection in Comments*) or they can be templates for a database (*i.e. MSSQL Time Based SQL Injection*).

Attack File format is XML, so you can just load it up in your favourite text editor and modify them.