ifmacedo Update SQLinjection History

1 contributor

39 lines (32 sloc) | 2.06 KB

```
Presentation:
Security vulnerability: SQL Injection.
Vulnerability Type: Injections.
Affected Component: Affected function on database access code.
Software: mConnect MV.
Versions: 02.001.00.00, 2013.1.6.8 (discontinued).
Bussiness area: Health, Medicine.

Describe the bug/issue:
SQL injection in Logon Page of MV's mConnect application, versions v02.001.00 and 2013.1.6.8, allows an attacker to use a non existing user with a generic password to connect to th
and get access to unauthorized information.

Have you searched the internet or Github for an answer?
Yes.

To Reproduce:
1. The application used to demonstrate the security flaw by SQL Injection is mConnect version 02.001.00. This version was discontinued by the developer and is no longer
offered to the customers, newer and free-flaw versions have been available for update and use.
2. The database used by the application is an Oracle. The application does not handle the 'user' field, returning an Oracle error when an unexpected character is inserted,
as a single quotes.
3. A basic test of an SQL Injection: We insert a 'single quotation mark' in one of the fields and analyze the return of the application. The application returns an Oracle
error (ORA-01756), meaning that it is vulnerable t0 SQL Injection.
4. Now injecting the payload 'OR '1'='1'-- and no password, the application returns "Invalid Password", meaning that it accepted this payload as a valid user. But, if
we send a invalid user, it returns "Invalid User".
5. SQLMap could be used to automate injection, pointing to User field. All database's information will be disclosed.

Expected behavior:
Parameterized SQL queries.

Bug Fix:
No bug fix. Discontinued software.

Additional context:
Using a blacklist filter is not recommended in this scenario, because SQL/MySQL queries can be written in varied and unexpected ways, see e.g.:
http://cwe.mitre.org/data/definitions/89.html
https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html
which recommends parametrization and white lists, among other solutions.

CVE ID: CVE-2020-23282
```