# Auxiliary Inject Functions in Oracle’s eBusiness Suite 11.5

**David Litchfield**

**2nd June 2016**

Auxiliary inject functions are functions that execute an arbitrary SQL statement and therefore allow the execute of any SQL including DML and DDL even from a SELECT statement.(This is achieved by specifying the [illegible] pragma in the declaration block)

## APPS.ASG_CUSTOM_PVT.EXEC_CMD

This function returns a VARCHAR.

Example:

[illegible]

[illegible]

## APPS.WIP_MASS_LOAD_UTILITIES.DYNAMIC_SQL

This function returns a NUMBER.

Example:

[illegible]

[illegible]

[illegible]

## APPS.MSC_GET_NAME.EXECUTE_SQL_GETCOUNT

This function returns a NUMBER.

Example:

[illegible]

[illegible]

## APPS.BSC_UPDATE_UTIL.EXECUTE_IMMEDIATE

This function returns a NUMBER.

Example:

[illegible]

[illegible]

Note: does not exist in EBS R12

## APPS.PSB_WS_ACCT1.DSQL

This function returns a NUMBER.

Example:

```
[illegible]
[illegible]
```

Note: does not exist in EBS R12

Note: okc_wf.exec_wf_plsql and apps.okc_p_util.execute_sql can be used in DML inject points (their code issues a savepoint)

**Executing SQL as SYS via APPS in eBusiness Suite 11.5**

By default APPS can execute DBMS_SYS_SQL. This allows APPS to run SQL as SYS.

```
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
```

**Executing SQL as SYS via APPS (via SYSTEM) in eBusiness Suite 12.2**

In R12.2, APPS no longer has the privileges to directly execute DBMS_SYS_SQL. However, SYSTEM does. Additionally, AD_APPS_PRIVATE owned by SYSTEM is vulnerable to SQL injection and it can be executed by the APPS user. By chaining these flaws together we can still execute SQL as SYS from APPS.

```
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
```

Appendix A - SQL to look for “EXEC” functions:

```
[illegible]
[illegible]
```