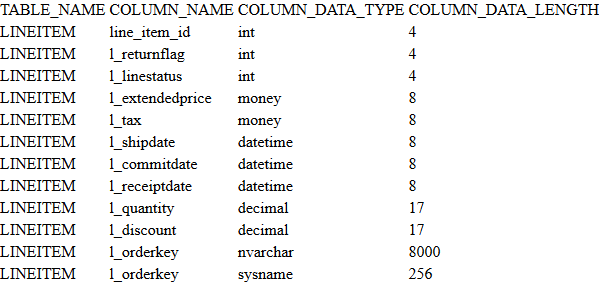
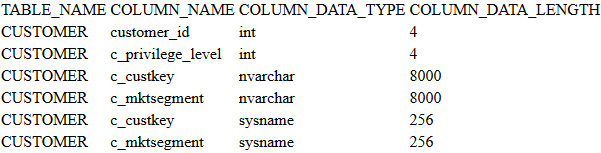
1. Table Structure Description

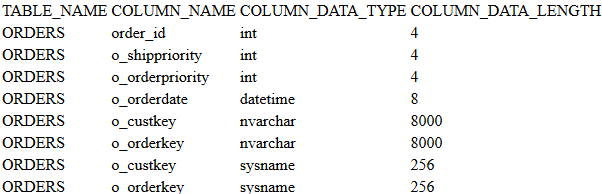
1.1 LINEITEM Table structure:



1.2 CUSTOMER Table structure:

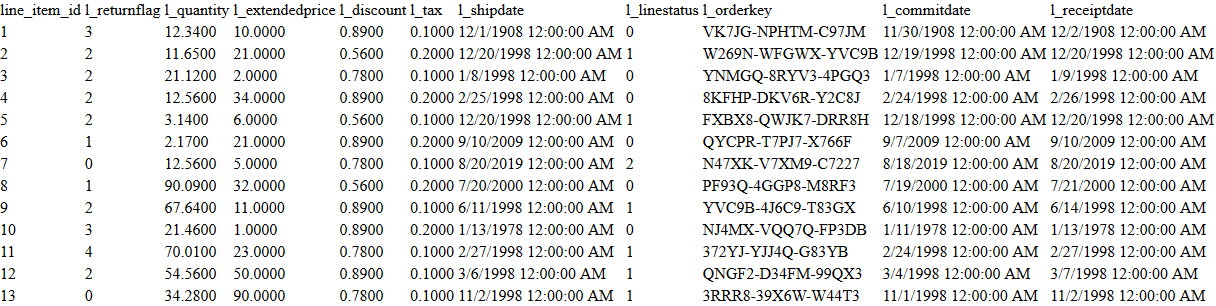


1.3 ORDERS Table structure:

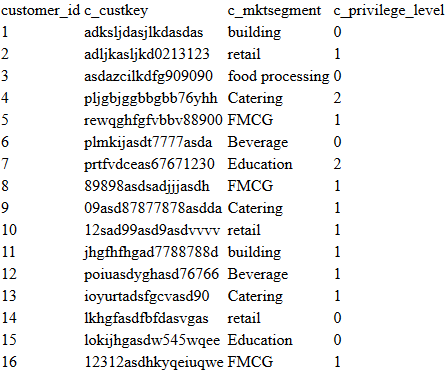


2. Table Data Description

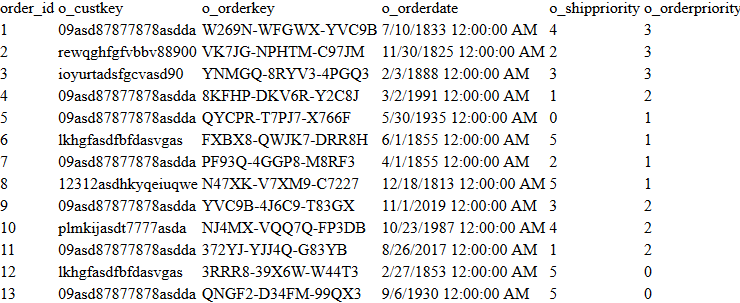
2.1 LINEITEM Data:



2.2 CUSTOMER Data:



2.3 ORDERS Data:



3. Policy Description:

Users with **level =** **2** access Table **<TEST01.SA.CUSTOMER>** with some limitations. He can only **<SELECT>** or **<UPDATE>** or **<DELETE>** records with field **c\_privilege\_level < 1** and he can not **<INSERT>** recordsintothis table**.** Others has no limitation.

Now, we have 2 users, User1 with level = 0 and User2 with level = 2. They will have different Enforcement result under the current policy.

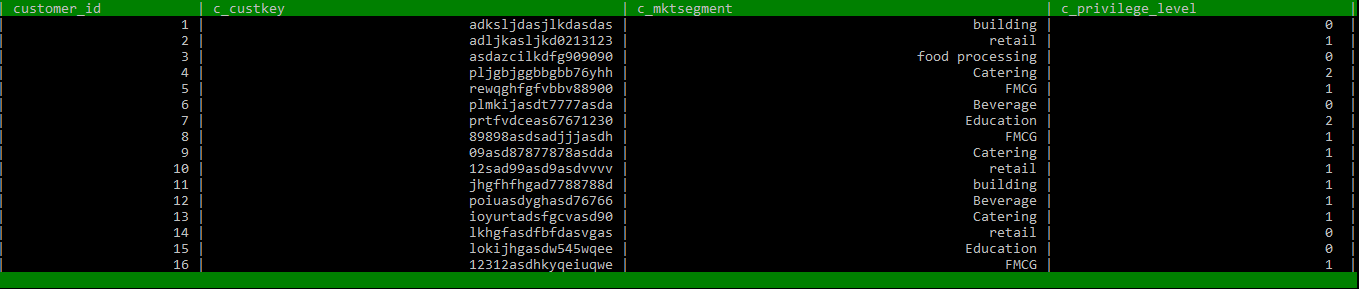
4. Enforcement Result:

4.1 Select Simple Case:

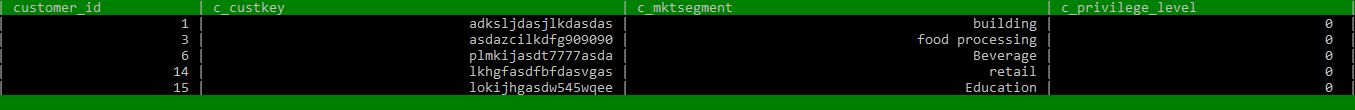
SELECT \*

FROM SA.CUSTOMER;

User1 (level = 0) has following results of enforcement:



User2 (level = 2) has following results of enforcement:



4.2 Update Simple Case:

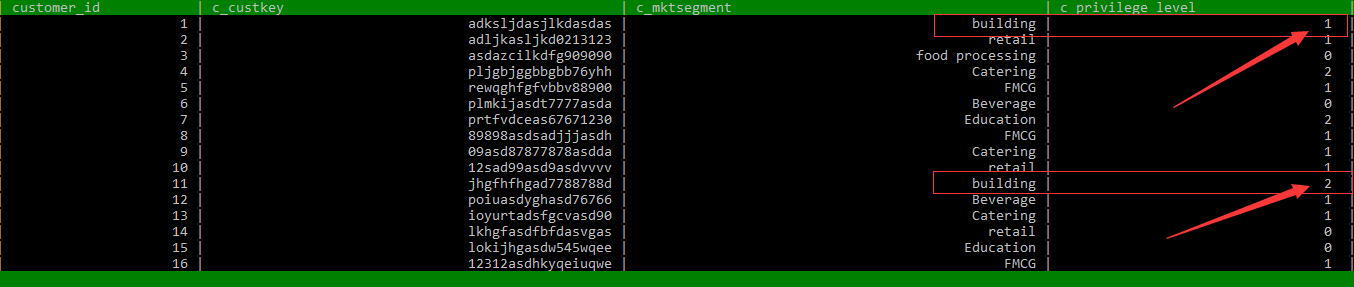
UPDATE SA.CUSTOMER

SET c\_privilege\_level = c\_privilege\_level - 1

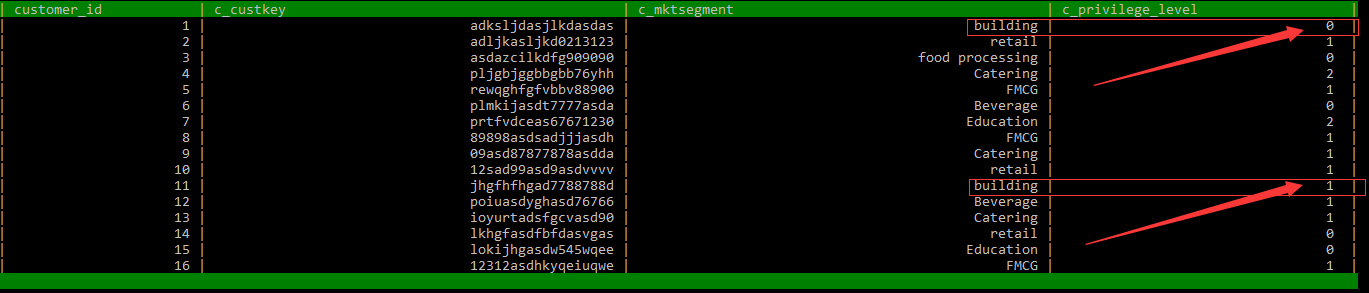
WHERE c\_mktsegment = ‘building’;

User1 (level = 0) has following results of enforcement:

Before update User1 view the table:

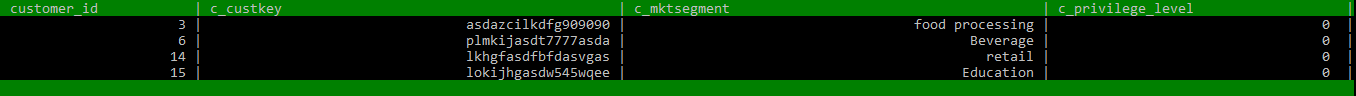


====> (2 rows affected.)

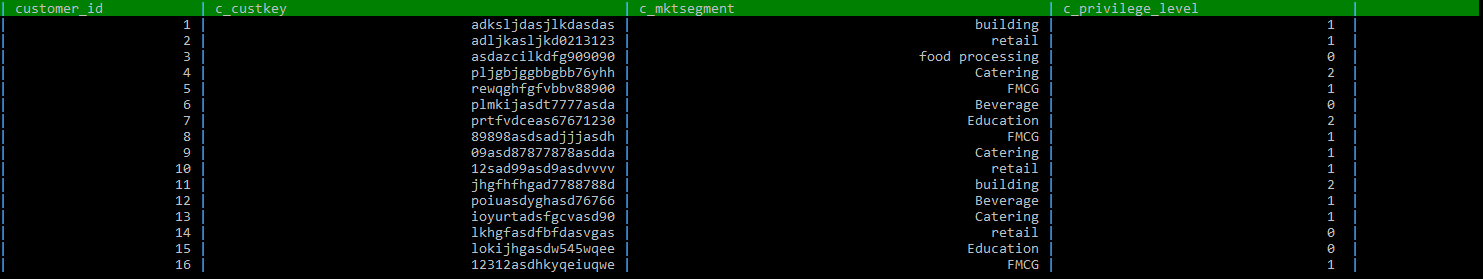


User2 (level = 2) has following results of enforcement:

Before update User2 view this table :

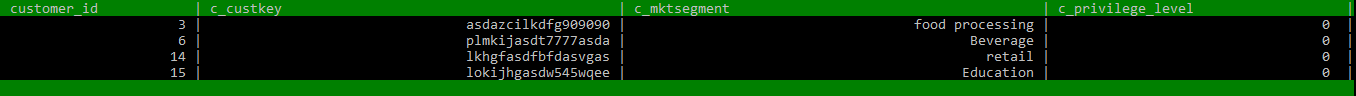


Before update User1 view this table :

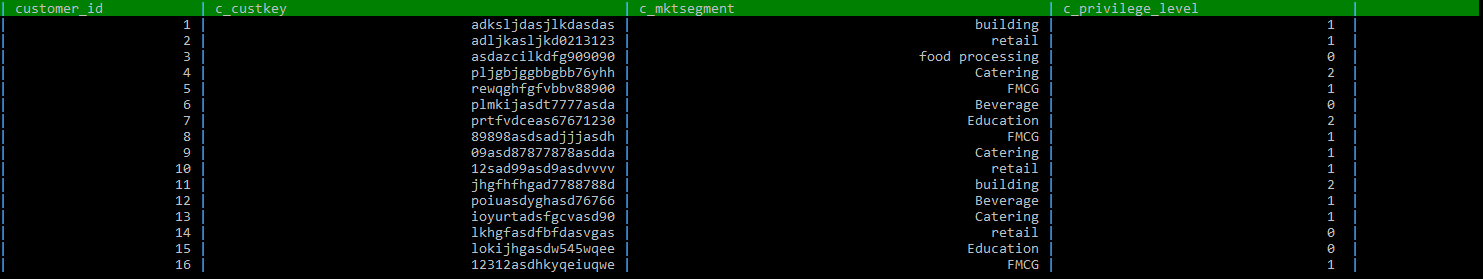


====> (nothing happened.user2’s update operation is limited)

User2 view this table:



User1 view this table:



4.3 Delete Simple Case:

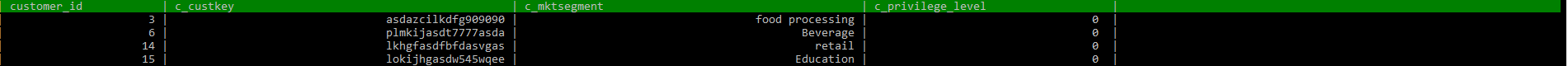
DELETE

FROM SA.CUSTOMER

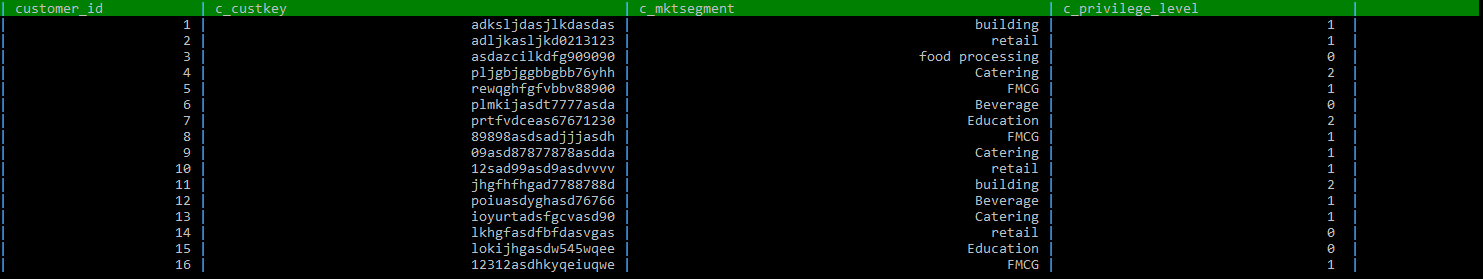
WHERE c\_mktsegment = 'FMCG';

User2 (level = 2) has following results of enforcement:

Before delete User2 view this table :

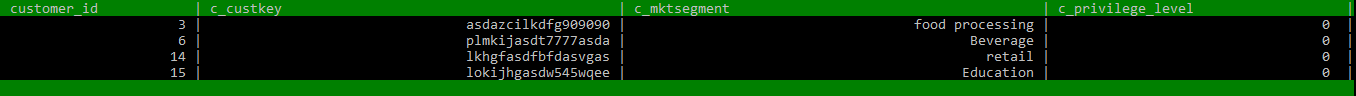


Before delete User1 view this table:

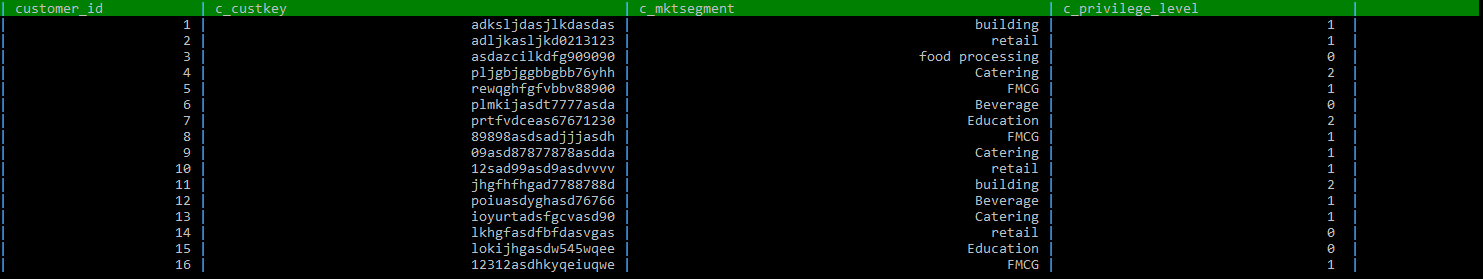


====> (nothing happened.user2’s delete operation is limited)

User2 view this table:



User1 view this table:



4.4 Insert Simple Case :

INSERT

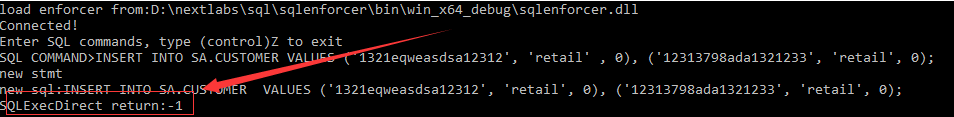
INTO SA.CUSTOMER

VALUES (‘1321eqweasdsa12312’, 'retail' , 0),

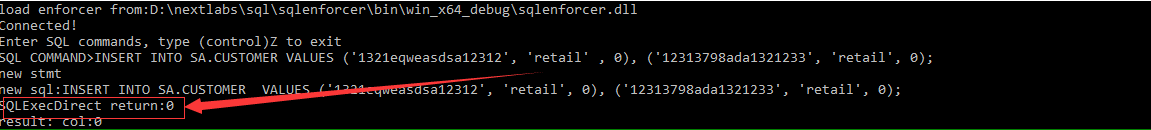
(‘12313798ada1321233’, 'retail' , 0);

User2(level = 2) has following results of enforcement:

SQLExecDirect return the SQL\_ERROR(-1), enforcer blocked user2’s insert operation



User1(level = 0) has following results of enforcement:



4.5 Select Advanced Case (Multi-Table)

SELECT l\_orderkey,

Sum(l\_extendedprice \* ( 1 - l\_discount )) AS REVENUE,

o\_orderdate,

o\_shippriority

FROM sa.customer,

sa.orders,

sa.lineitem

WHERE c\_mktsegment = 'Catering'

AND c\_custkey = o\_custkey

AND l\_orderkey = o\_orderkey

AND o\_orderdate < '1995-03-15'

AND l\_shipdate > '1995-03-15'

GROUP BY l\_orderkey,

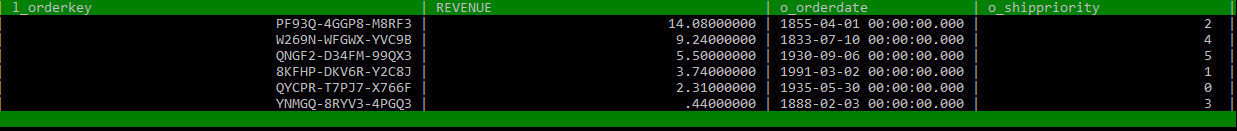
o\_orderdate,

o\_shippriority

ORDER BY revenue DESC,

o\_orderdate;

User1 (level = 0) has following results of enforcement:



User2 (level = 2) has following results of enforcement:



4.6 Advanced Select Case (Correlated-Subquery)

SELECT o\_orderpriority,

Count(\*) AS ORDER\_COUNT

FROM sa.orders

WHERE o\_orderdate >= '1983-07-01'

AND o\_orderdate < Dateadd(mm, 3, Cast('1993-07-01' AS datetime))

AND EXISTS (SELECT \*

FROM sa.customer

WHERE c\_custkey = o\_custkey)

GROUP BY o\_orderpriority

ORDER BY o\_orderpriority;

User1 (level = 0) has following results of enforcement:



User2 (level = 2) has following results of enforcement:



1. What is a <direct select statement: multiple rows> ?

Examples:

1. SELECT user\_id FROM order\_list;
2. SELECT user\_id FROM order\_list ORDER BY item\_name;
3. SELECT item\_id, user\_id, Sum(qty) FROM order\_list GROUP BY item\_id, user\_id;
4. SELECT user\_id, user\_name FROM USER UNION ALL SELECT item\_id, item\_name FROM item;

Following examples is not:

1. PREPARE test\_stmt FROM 'SELECT SQRT(POW(?,2) + POW(?,2)) AS hypotenuse';

SET @a = 3; SET @b = 4; EXECUTE stmt1 USING @a, @b;

1. SELECT \* INTO Persons\_backup FROM Persons;
2. What is a <update statement: searched> ?

Examples:

1. UPDATE item SET item\_cost = 89, item\_price = item\_cost WHERE item\_id = 100;
2. UPDATE person.address SET modifieddate = Getdate();

Following examples is not:

1. UPDATE bigorder SET UserId='123' WHERE CURRENT OF orderNum\_03\_cursor
2. What is a <delete statement: searched> ?

Examples:

1. DELETE FROM production.productcosthistory WHERE standardcost > 1000.00;
2. DELETE FROM orders;

Following examples is not:

1. DELETE FROM bigorder Where CURRENT OF orderNum\_03\_cursor
2. What is a <insert statement> ?

Examples:

1. INSERT INTO t1 DEFAULT VALUES;
2. INSERT INTO dbo.t1 (column\_2) VALUES ('Explicit value'), (‘hello world’);
3. INSERT INTO dbo.t1 (column\_2, column\_4) VALUES ('Explicit value', 'Explicit value');
4. INSERT INTO employeetitles SELECT employeekey, lastname, title FROM dbo.dimemployee WHERE enddate IS NULL;

1. The added predicate filter condition based on policys is not checked. Policy with error will generate an illegal sql expression (e.g invalid column name, illegal operator use …) .

2. Block by enforcer just have a SQL\_ERROR, error detail is not support

3. Currently, user attr info is stored in file as a xxx.ini

4. Our Enforcer solution is based on the semantic analysis of user’s script. Enforcer will do nothing if parser can not understand the script. This includes two aspects. First, some special syntax is not supported By Parser.Second, some necessary information need by enforcer does not exist in the parsing phase.