Enhance the Informix Auditing C code

Description

IBM has shared the source for Event-driven fine-grained auditing with Informix.
https://www.ibm.com/developerworks/data/library/techarticle/dm-0410roy/index.html

The informix has been configured with auditing tool as per the instruction, it is able to generate the xml audit output. The output xml is truncating to 30 charachter even a column value has more than 30 character.

Goal of the challenges are
1) Figuring out way to fix the code which accomodate the full column value and the code should consider the column data type for string/blob/byte. Verify the column value by generating the output.
2) Optionaly convert the output to json format

Prerequisites

None.

Development was done using Ubuntu 18.04 LTS, however setup/configuration and testing steps should work on different distribution / OS as well

Production Build and Installation

Please follow provided steps in order to set-up and run the testing environment

S1. Install missing packages on informix container

S1.1 run a container for informix as root

```
sudo docker run -it -u root appiriodevops/informix:6f3884d bash
```

S1.2 within container install make and gcc

```
yum install make
```

```
yum install gcc
```

S2 Change user to informix

```
su informix
```

S3 cd trunk

S4 Install tocoder db

S4.1 oninit -vy

S4.2 ant reinstall_db (if fails run again)

S4.3 onmode -ky

S5 Change informix database configuration

S5.1 Create sbspace for blob data type

S5.1.1 Configure mirroring

edit (with your preferred editor) /opt/IBM/informix/etc/onconfig.informixoltp_tcp

vi /opt/IBM/informix/etc/onconfig.informixoltp_tcp

set up mirror:

MIRROR 1

set up MIRRORPATH to empty

MIRRORPATH (eg. delete default value)

S5.1.2 Startup informix

oninit -vy

S5.1.3 Create files where blob will be stored

touch /tmp/rawdev1

touch /tmp/rawdev2

chmod 660 /tmp/rawdev1

chmod 660 /tmp/rawdev2

S5.1.4 Create dbspace

```
onspaces -c -S sbsp4 -p /tmp/rawdev1 -o 500 -s 20480 -m
/tmp/rawdev2 500 -Ms 150 -Mo 200 -Df "AVG_LO_SIZE=32"
```

S5.2 Add sbspace to informix configuration

S5.2.1 Stop informix

onmode -ky

S5.2.2 edit (with your preferred editor) /opt/IBM/informix/etc/onconfig.informixoltp_tcp

vi /opt/IBM/informix/etc/onconfig.informixoltp_tcp

change SBSPACENAME configuration

SBSPACENAME sbsp4

S5.2.3 Start informix

oninit -vy

S5.3 Copy modified code into docker container

S5.3.1 Create a dev folder where code will be compiled

mkdir /home/informix/trunk/dev

S.5.3.2

From a separate terminal (not docker) run

sudo docker cp auditing_modified.zipmkdir /opt/IBM/informix/extend/auditing <<containerid>>:/home/informix/trunk/dev/auditing_modified.zip

S5.3.3 Within docker dev folder unzip the code

unzip auditing_modified.zip

S5.4 Compile code

make -f UNIX.mak

S5.5 Create required folders within informix

mkdir /opt/IBM/informix/extend/auditing

S5.6 Install compiled code

make -f UNIX.mak INSTALL

S5.7 Prepare database for testing (a testing database was created for testing instead of using an existing table from schema. Testing database contains only one table which makes it easier to test it)

S5.7.1 Create database (supporting sql files are provided in zip archive)

dbaccess - createdatabase.sql

5.7.2 create table

dbaccess -e testdb createtable.sql

Running Tests

S1. Create a file named test.log in /tmp – this will be used for blob database

vi /tmp/test.log (you can write a short text inside)

S2. From trunk/dev run following commands

dbaccess -e testdb auditing1

dbaccess -e testdb auditing2

dbaccess -e testdb saveAuditOnTable.sql

dbaccess -e testdb saveAuditOnFile.sql

S3. In /tmp folder you should see several json files (eg. audit23_0.json, audit23_1.json, audit23_2.json) – check they contain the json logs – also you can validate the json using https://jsonlint.com/

S4 Using an sqlclient (can be dbaccess) check there is an audittable in testdb database – audit table will contain the json audit strings

S5 Remove triggers

dbaccess -e testdb auditing1_d

dbaccess -e testdb auditing2_d

Troubleshooting

If at any point running dbaccess commands informix is complaining about memory – restart informix server

onmode -ky

onmode -vy

Notes

Code changes

Following changes were made to the code

1. UNIX.mak - change make file to compile on CentOS

2. audit_util.h

function do_cast was renamed to do_castl and a new param mi_integer was added. The new parameter specify field length and is used when performing casting

2. audit_util.c

a. function do_cast was renamed to do_castl and a new param mi_integer was added. The new parameter specify field length and is used when performing casting

new code was added as well to read precision instead of hardcodding it

b. In functions doInsertCN, doSelectCN, doDeleteCN, doUpdateCN

all sprintf methods were changed to write json tags instead of xml

all do_cast calls changed to do_castl

4. auditing2.c

all xml references and names changed to json