Using the IDS Auditing Subsystem

Rick Rabe

Hilton Hotels Corp.

A18

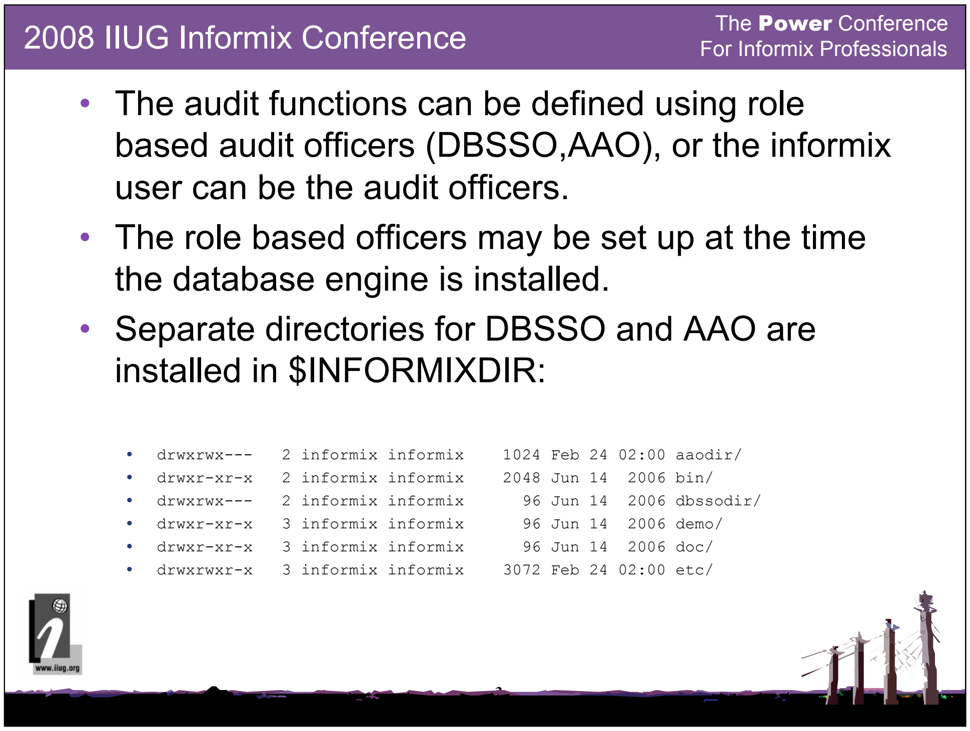
Wednesday, April 30, 2008 • 4:40 p.m. – 5:40 p.m.

2008 IIUG Informix Conference

www.iiug.org

Using the IDS Auditing Subsystem

- The audit configuration and files
- Using the auditing masks
- Strengths and weaknesses of the auditing system
- Parsing the audit logs for useful information
- Using a database for audit records



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

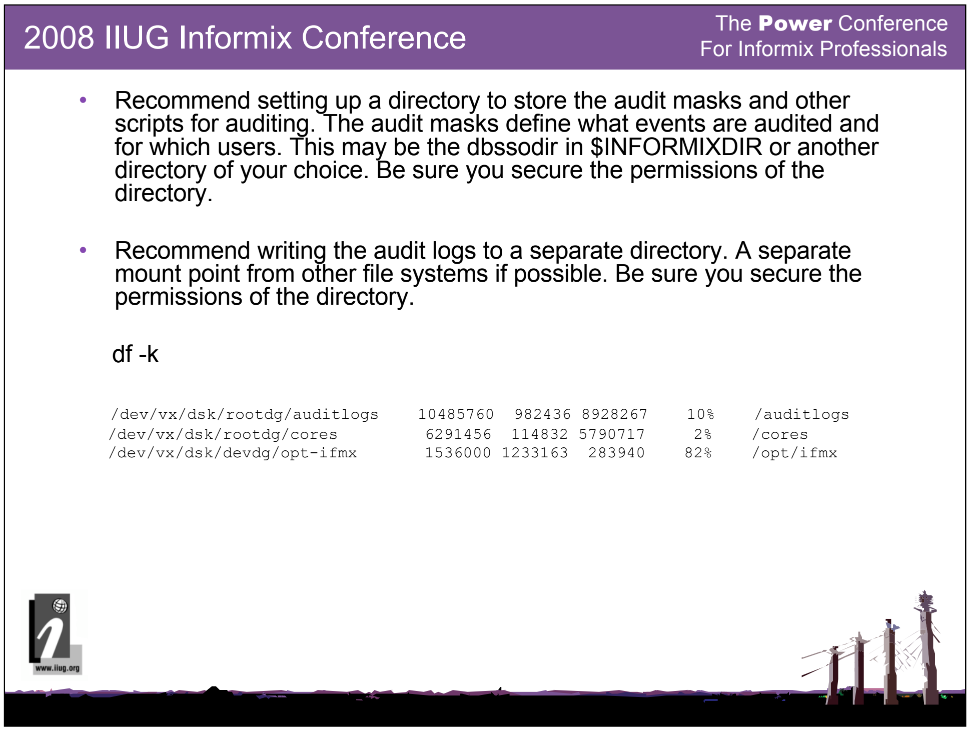
- The audit functions can be defined using role based audit officers (DBSSO,AAO), or the informix user can be the audit officers.
- The role based officers may be set up at the time the database engine is installed.
- Separate directories for DBSSO and AAO are installed in $INFORMIXDIR:

```
•  drwxrwx---   2 informix informix    1024 Feb 24 02:00 aaodir/
•  drwxr-xr-x   2 informix informix    2048 Jun 14  2006 bin/
•  drwxrwx---   2 informix informix      96 Jun 14  2006 dbssodir/
•  drwxr-xr-x   3 informix informix      96 Jun 14  2006 demo/
•  drwxr-xr-x   3 informix informix      96 Jun 14  2006 doc/
•  drwxrwxr-x   3 informix informix    3072 Feb 24 02:00 etc/
```

For simplicity, the informix user is the audit officer in this presentation.

Note the group for the 2 directories – aaodir, dbssodir - is informix since informix is the audit officer.

For more on the roles see Jonathan Leffler’s presentation M12: Auditing Informix Dynamic Server Thursday 9:20.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- Recommend setting up a directory to store the audit masks and other scripts for auditing. The audit masks define what events are audited and for which users. This may be the dbssodir in $INFORMIXDIR or another directory of your choice. Be sure you secure the permissions of the directory.

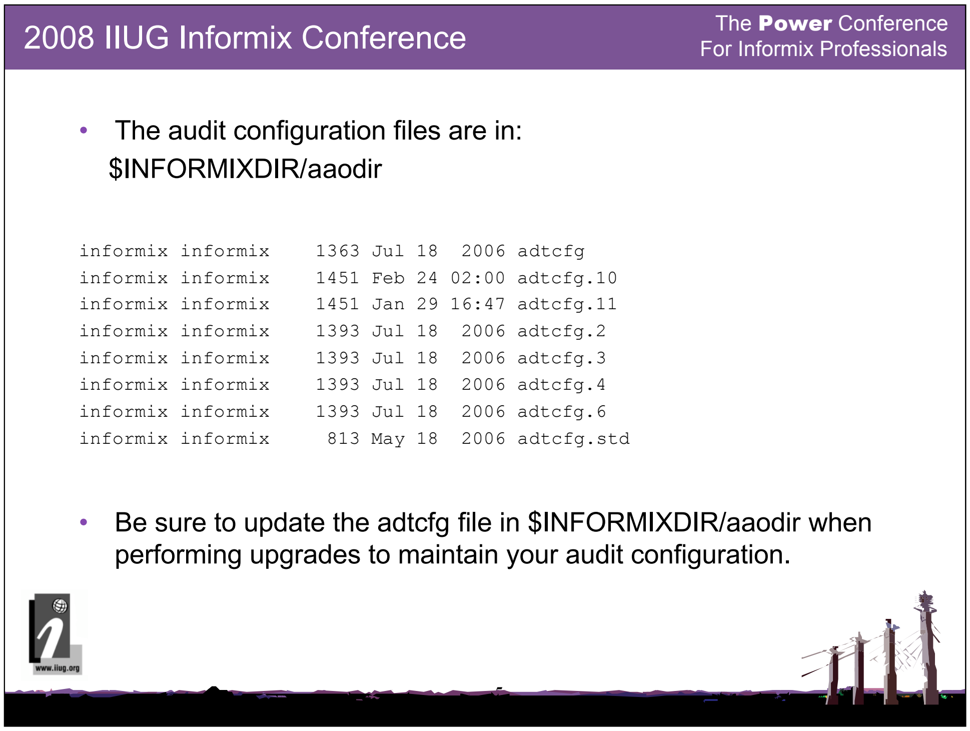
- Recommend writing the audit logs to a separate directory. A separate mount point from other file systems if possible. Be sure you secure the permissions of the directory.

df -k

```
/dev/vx/dsk/rootdg/auditlogs    10485760  982436 8928267    10%    /auditlogs
/dev/vx/dsk/rootdg/cores         6291456  114832 5790717     2%    /cores
/dev/vx/dsk/devdg/opt-ifmx       1536000 1233163  283940    82%    /opt/ifmx
```

A separate directory other than the dbsssodir for audit masks and auditing scripts makes continuity during upgrades easier.

Due to the intense write activity to the audit log, a separate mount point for the file system will reduce I/O contention.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

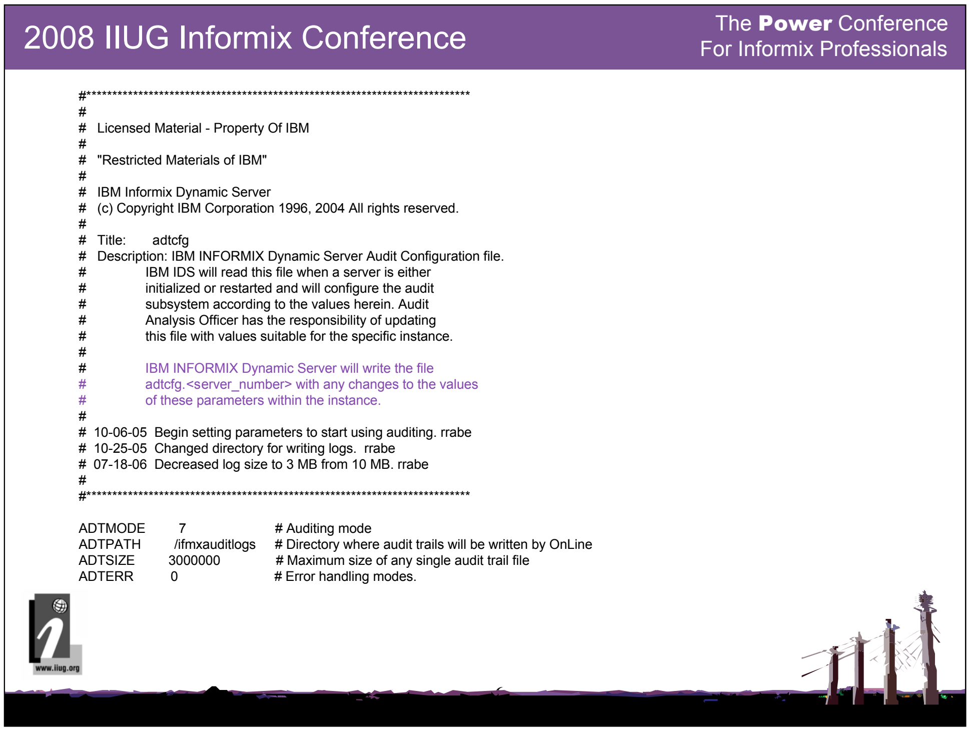
- The audit configuration files are in:
  $INFORMIXDIR/aaodir

```
informix informix    1363 Jul 18  2006 adtcfg
informix informix    1451 Feb 24 02:00 adtcfg.10
informix informix    1451 Jan 29 16:47 adtcfg.11
informix informix    1393 Jul 18  2006 adtcfg.2
informix informix    1393 Jul 18  2006 adtcfg.3
informix informix    1393 Jul 18  2006 adtcfg.4
informix informix    1393 Jul 18  2006 adtcfg.6
informix informix     813 May 18  2006 adtcfg.std
```

- Be sure to update the adtcfg file in $INFORMIXDIR/aaodir when performing upgrades to maintain your audit configuration.

As configuration changes are made using the onaudit command the engine writes the new config file as adtcfg.server_number.

At startup the engine will still use adtcfg for the audit configuration. You will need to copy adtcfg.server_number to adtcfg or manually edit the adtcfg file for the changes to be kept during the next engine restart.



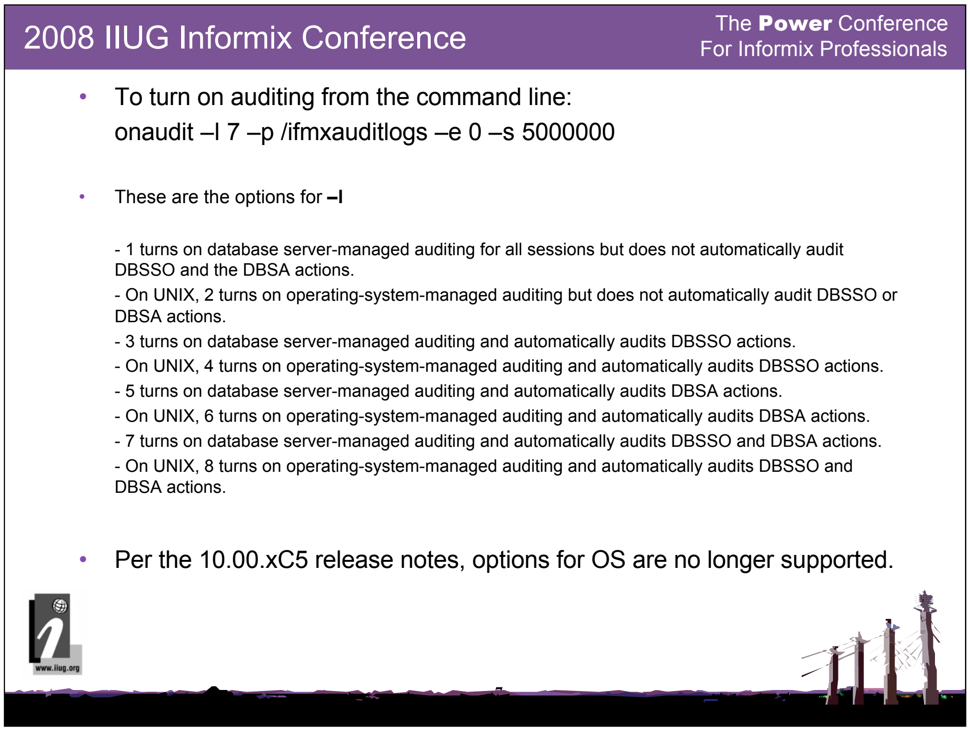
2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

```
#***************************************************************************
#
#   Licensed Material - Property Of IBM
#
#   "Restricted Materials of IBM"
#
#   IBM Informix Dynamic Server
#   (c) Copyright IBM Corporation 1996, 2004 All rights reserved.
#
#   Title:       adtcfg
#   Description: IBM INFORMIX Dynamic Server Audit Configuration file.
#               IBM IDS will read this file when a server is either
#               initialized or restarted and will configure the audit
#               subsystem according to the values herein. Audit
#               Analysis Officer has the responsibility of updating
#               this file with values suitable for the specific instance.
#
#               IBM INFORMIX Dynamic Server will write the file
#               adtcfg.<server_number> with any changes to the values
#               of these parameters within the instance.
#
#  10-06-05  Begin setting parameters to start using auditing. rrabe
#  10-25-05  Changed directory for writing logs.  rrabe
#  07-18-06  Decreased log size to 3 MB from 10 MB. rrabe
#
#***************************************************************************

ADTMODE        7                  # Auditing mode
ADTPATH        /ifmxauditlogs     # Directory where audit trails will be written by OnLine
ADTSIZE        3000000            # Maximum size of any single audit trail file
ADTERR         0                  # Error handling modes.
```

Note the comment about changes and the adtcfg.server_number. See next slide for multi-instance systems.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- To turn on auditing from the command line:

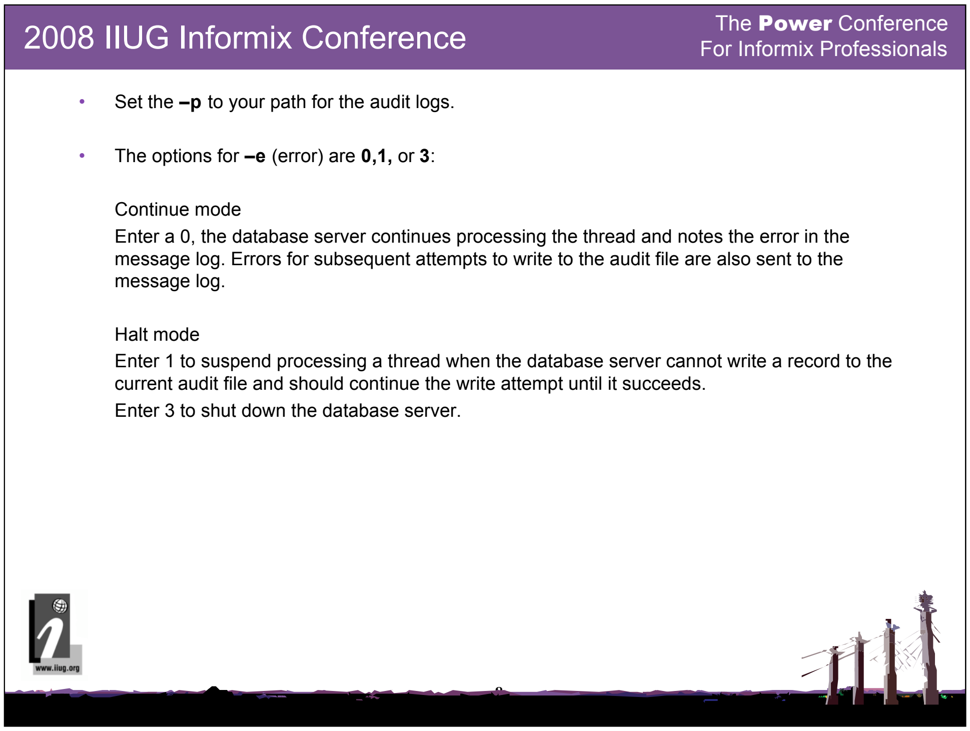
```
onaudit –l 7 –p /ifmxauditlogs –e 0 –s 5000000
```

- These are the options for **–l**

- 1 turns on database server-managed auditing for all sessions but does not automatically audit DBSSO and the DBSA actions.
- On UNIX, 2 turns on operating-system-managed auditing but does not automatically audit DBSSO or DBSA actions.
- 3 turns on database server-managed auditing and automatically audits DBSSO actions.
- On UNIX, 4 turns on operating-system-managed auditing and automatically audits DBSSO actions.
- 5 turns on database server-managed auditing and automatically audits DBSA actions.
- On UNIX, 6 turns on operating-system-managed auditing and automatically audits DBSA actions.
- 7 turns on database server-managed auditing and automatically audits DBSSO and DBSA actions.
- On UNIX, 8 turns on operating-system-managed auditing and automatically audits DBSSO and DBSA actions.

- Per the 10.00.xC5 release notes, options for OS are no longer supported.

This turns on auditing of events including for informix in our example, sets the path for the audit logs to /ifmxauditlogs, sets the auditing to continue mode, and the size of the audit logs to 5MB.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- Set the **–p** to your path for the audit logs.
- The options for **–e** (error) are **0**,**1,** or **3**:

Continue mode

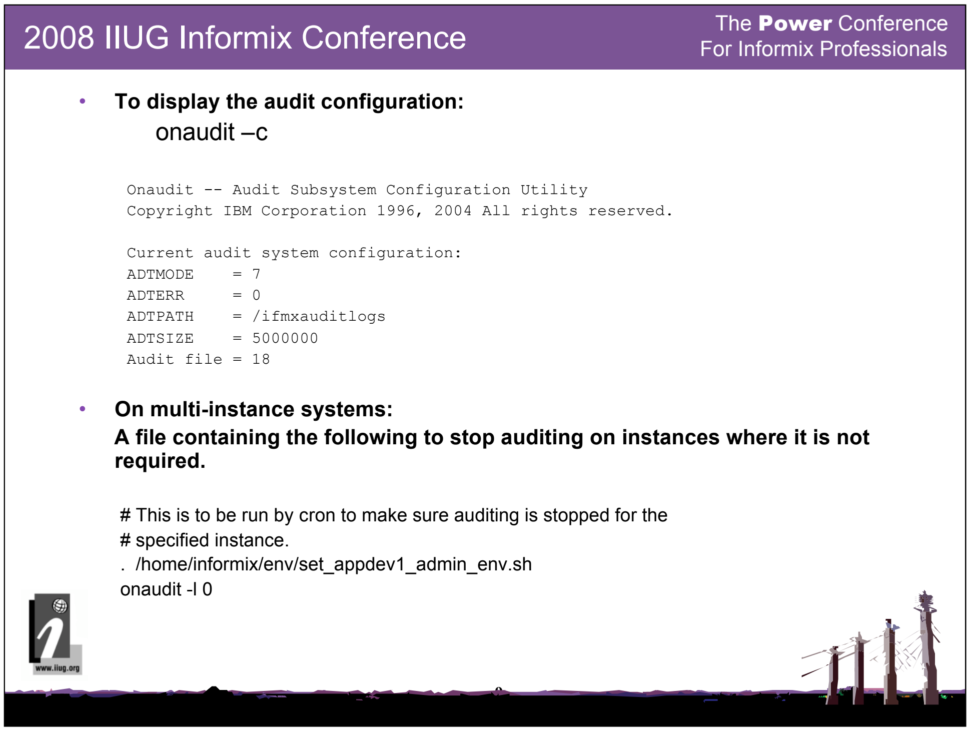
Enter a 0, the database server continues processing the thread and notes the error in the message log. Errors for subsequent attempts to write to the audit file are also sent to the message log.

Halt mode

Enter 1 to suspend processing a thread when the database server cannot write a record to the current audit file and should continue the write attempt until it succeeds.

Enter 3 to shut down the database server.

Your error setting determines whether the database and applications continue to run or not when audit logs cannot be written.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- **To display the audit configuration:**
  onaudit –c

```
Onaudit -- Audit Subsystem Configuration Utility
Copyright IBM Corporation 1996, 2004 All rights reserved.

Current audit system configuration:
ADTMODE    = 7
ADTERR     = 0
ADTPATH    = /ifmxauditlogs
ADTSIZE    = 5000000
Audit file = 18
```

- **On multi-instance systems:**
  A file containing the following to stop auditing on instances where it is not required.

```
# This is to be run by cron to make sure auditing is stopped for the
# specified instance.
.  /home/informix/env/set_appdev1_admin_env.sh
onaudit -l 0
```

The appdev1 is an instance that does not contain any sensitive information that requires auditing. The frequency of engine restarts for this instance will help determine the frequency of running the script from cron.

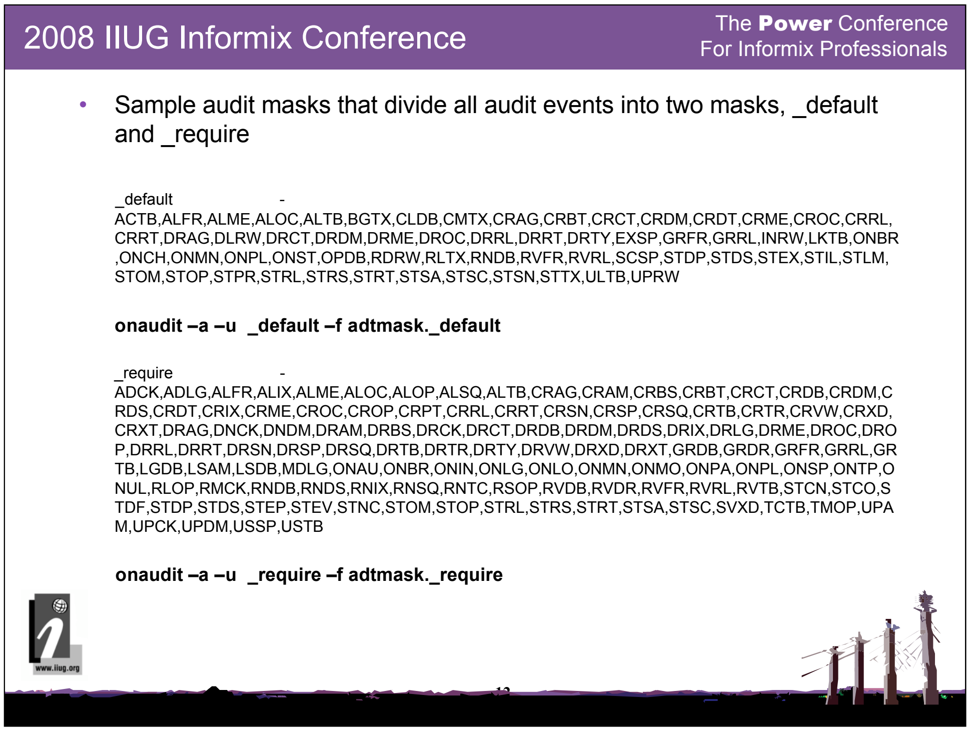
- Three types of audit masks:
  - Individual User
    - username
  - Global
    - _default
    - _require
    - _exclude
  - Template
    - _maskname

- We use _default and _require to define all audit events.
  - Default events are basic operations such as read row
  - Required events are actions such as audit mask changes

- We then use user masks to remove events from the default list.

- This allows us to not audit an application user id for default events. The application logs are used for auditing the individual user actions.

- Other user ids are audited by the _default and _require masks. Some of these users may have a mask to remove some events from their audited list. For example if a special query is to be run against very large tables you may want to remove RDRW (read row) from a specific user for the duration of the job. A user mask or template can be used to accomplish this. Use your Change Management system to track these kind of temporary audit changes.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- Sample audit masks that divide all audit events into two masks, _default and _require

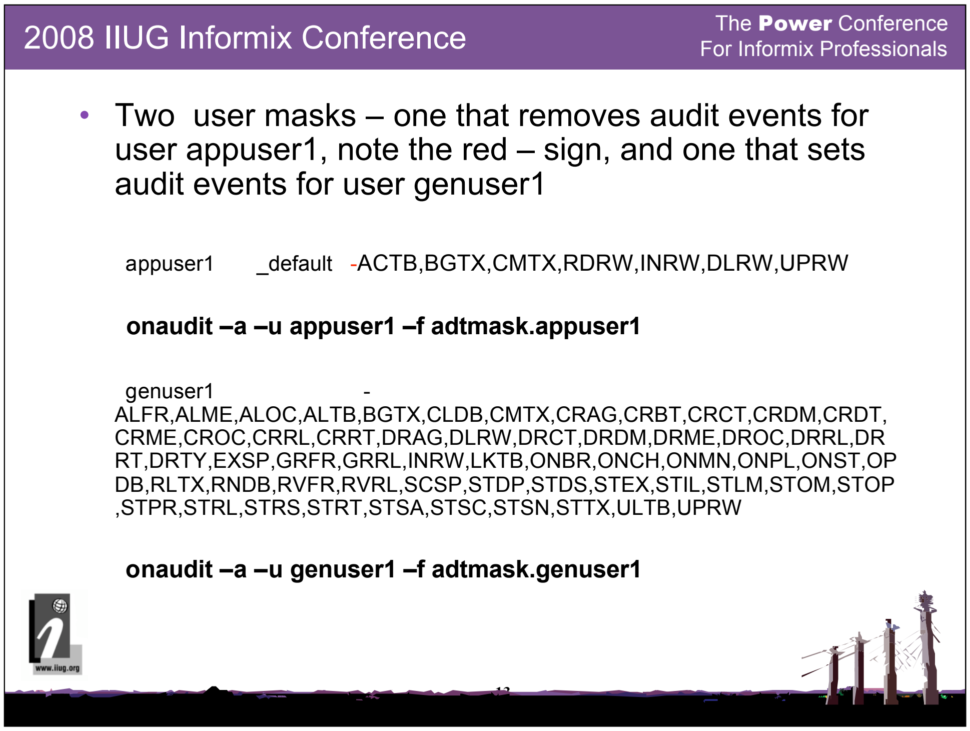
```
_default                    -
ACTB,ALFR,ALME,ALOC,ALTB,BGTX,CLDB,CMTX,CRAG,CRBT,CRCT,CRDM,CRDT,CRME,CROC,CRRL,
CRRT,DRAG,DLRW,DRCT,DRDM,DRME,DROC,DRRL,DRRT,DRTY,EXSP,GRFR,GRRL,INRW,LKTB,ONBR
,ONCH,ONMN,ONPL,ONST,OPDB,RDRW,RLTX,RNDB,RVFR,RVRL,SCSP,STDP,STDS,STEX,STIL,STLM,
STOM,STOP,STPR,STRL,STRS,STRT,STSA,STSC,STSN,STTX,ULTB,UPRW
```

onaudit –a –u _default –f adtmask._default

```
_require                    -
ADCK,ADLG,ALFR,ALIX,ALME,ALOC,ALOP,ALSQ,ALTB,CRAG,CRAM,CRBS,CRBT,CRCT,CRDB,CRDM,C
RDS,CRDT,CRIX,CRME,CROC,CROP,CRPT,CRRL,CRRT,CRSN,CRSP,CRSQ,CRTB,CRTR,CRVW,CRXD,
CRXT,DRAG,DNCK,DNDM,DRAM,DRBS,DRCK,DRCT,DRDB,DRDM,DRDS,DRIX,DRLG,DRME,DROC,DRO
P,DRRL,DRRT,DRSN,DRSP,DRSQ,DRTB,DRTR,DRTY,DRVW,DRXD,DRXT,GRDB,GRDR,GRFR,GRRL,GR
TB,LGDB,LSAM,LSDB,MDLG,ONAU,ONBR,ONIN,ONLG,ONLO,ONMN,ONMO,ONPA,ONPL,ONSP,ONTP,O
NUL,RLOP,RMCK,RNDB,RNDS,RNIX,RNSQ,RNTC,RSOP,RVDB,RVDR,RVFR,RVRL,RVTB,STCN,STCO,S
TDF,STDP,STDS,STEP,STEV,STNC,STOM,STOP,STRL,STRS,STRT,STSA,STSC,SVXD,TCTB,TMOP,UPA
M,UPCK,UPDM,USSP,USTB
```

onaudit –a –u _require –f adtmask._require

Note the actual command in the audit mask is all one line. These are displayed with line returns for better display in a slide.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- Two user masks – one that removes audit events for user appuser1, note the red – sign, and one that sets audit events for user genuser1

```
appuser1      _default   -ACTB,BGTX,CMTX,RDRW,INRW,DLRW,UPRW
```

onaudit –a –u appuser1 –f adtmask.appuser1

```
genuser1                        -
ALFR,ALME,ALOC,ALTB,BGTX,CLDB,CMTX,CRAG,CRBT,CRCT,CRDM,CRDT,
CRME,CROC,CRRL,CRRT,DRAG,DLRW,DRCT,DRDM,DRME,DROC,DRRL,DR
RT,DRTY,EXSP,GRFR,GRRL,INRW,LKTB,ONBR,ONCH,ONMN,ONPL,ONST,OP
DB,RLTX,RNDB,RVFR,RVRL,SCSP,STDP,STDS,STEX,STIL,STLM,STOM,STOP
,STPR,STRL,STRS,STRT,STSA,STSC,STSN,STTX,ULTB,UPRW
```

onaudit –a –u genuser1 –f adtmask.genuser1

Note the actual command in the audit mask is all one line. These are displayed with line returns for better display in a slide.

- For change tracking, rather than modifying an existing mask you could make a copy then edit the audit mask file. Then delete and add the user mask with the updated file.

```
onaudit –d –u genuser1
onaudit –a –u genuser1 –f newmask.genuser1
```

- Another alternative is to create user masks using a script. Create a script for each instance, for example a script called set_dev_masks.sh:

```
. /home/informix/env/set_development_admin_env.sh

onaudit -a -u davdev -r _default -e -ACTB,BGTX,CLDB,CMTX,DLRW,INRW,RDRW,UPRW
onaudit -a -u davuat -r _default -e -ACTB,BGTX,CLDB,CMTX,DLRW,INRW,RDRW,UPRW
onaudit -a -u elvis -r _default -e -ACTB,BGTX,CLDB,CMTX,DLRW,INRW,RDRW,UPRW
onaudit -a -u gpmdev -r _default -e -ACTB,BGTX,CLDB,CMTX,DLRW,INRW,RDRW,UPRW
```

- To display the audit masks use:

onaudit -o –y

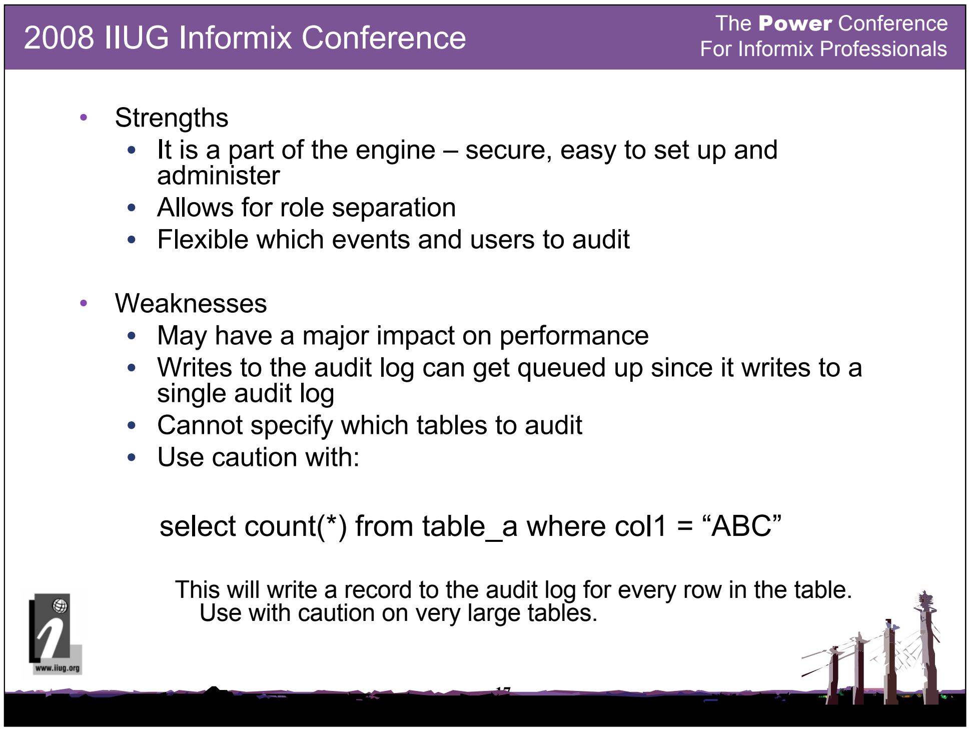
```
_default                         -
ACTB,ALTB,BGTX,CLDB,CMTX,CRIX,DLRW,EXSP,INRW,LKTB,ONCH,ONST,OPDB,RDRW,RLTX,SCSP,S
TEX,STIL,STLM,STSN,ULTB,UPRW,ALFR,STDS,STPR,STTX,STOM,STRT,STOP,GRFR,RVFR,CRRL,DRR
L,GRRL,RVRL,STDP,STRL,STSA,ONMN,RNDB,ONBR,ONPL,CRDM,DRDM,CRRT,DRRT,CRDT,CRCT,DR
CT,CRBT,DRTY,CRME,DRME,ALME,CROC,DROC,ALOC,STRS,CRAG,DRAG,STSC
_require                         -
ADCK,ADLG,ALIX,ALOP,ALTB,CRAM,CRBS,CRDB,CRDS,CRIX,CROP,CRSN,CRSP,CRTB,CRTR,CRVW,D
NCK,DNDM,DRAM,DRBS,DRCK,DRDB,DRDS,DRIX,DRLG,DROP,DRSN,DRSP,DRTB,DRTR,DRVW,GRDB,
GRTB,LGDB,LSAM,LSDB,MDLG,ONAU,ONIN,ONLG,ONLO,ONMO,ONPA,ONSP,ONTP,ONUL,RLOP,RMCK
,RNTC,RSOP,RVDB,RVTB,STCN,STDF,TMOP,UPAM,UPCK,UPDM,USSP,USTB,ALFR,STDS,STOM,STRT,
STOP,GRFR,RVFR,CRRL,DRRL,GRRL,RVRL,STDP,STRL,STSA,ONMN,RNDB,ONBR,ONPL,CRDM,DRDM,
CRRT,DRRT,CRDT,CRCT,DRCT,CRBT,DRTY,CRME,DRME,ALME,CROC,DROC,ALOC,STRS,CRAG,DRA
G,STSC,RNIX,CRSQ,RNSQ,DRSQ,ALSQ,STEV,RNDS,GRDR,RVDR,STCO,STNC,STEP,CRPT,CRXT,CRX
D,DRXT,DRXD,TCTB,SVXD
appuser1                           -
ALTB,CLDB,CRIX,EXSP,LKTB,ONCH,ONST,OPDB,RLTX,SCSP,STEX,STIL,STLM,STSN,ULTB,ALFR,STDS
,STPR,STTX,STOM,STRT,STOP,GRFR,RVFR,CRRL,DRRL,GRRL,RVRL,STDP,STRL,STSA,ONMN,RNDB,
ONBR,ONPL,CRDM,DRDM,CRRT,DRRT,CRDT,CRCT,DRCT,CRBT,DRTY,CRME,DRME,ALME,CROC,DRO
C,ALOC,STRS,CRAG,DRAG,STSC
appuat1                          -
ALTB,CLDB,CRIX,EXSP,LKTB,ONCH,ONST,OPDB,RLTX,SCSP,STEX,STIL,STLM,STSN,ULTB,ALFR,STDS
,STPR,STTX,STOM,STRT,STOP,GRFR,RVFR,CRRL,DRRL,GRRL,RVRL,STDP,STRL,STSA,ONMN,RNDB,
ONBR,ONPL,CRDM,DRDM,CRRT,DRRT,CRDT,CRCT,DRCT,CRBT,DRTY,CRME,DRME,ALME,CROC,DRO
C,ALOC,STRS,CRAG,DRAG,STSC
```

devuser1 -

ALTB,CLDB,CRIX,EXSP,LKTB,ONCH,ONST,OPDB,RLTX,SCSP,STEX,STIL,STLM,STSN,ULTB,ALFR,STDS
,STPR,STTX,STOM,STRT,STOP,GRFR,RVFR,CRRL,DRRL,GRRL,RVRL,STDP,STRL,STSA,ONMN,RNDB,
ONBR,ONPL,CRDM,DRDM,CRRT,DRRT,CRDT,CRCT,DRCT,CRBT,DRTY,CRME,DRME,ALME,CROC,DRO
C,ALOC,STRS,CRAG,DRAG,STSC

genuser1 -

ALFR,ALME,ALOC,ALTB,BGTX,CLDB,CMTX,CRAG,CRBT,CRCT,CRDM,CRDT,CRME,CROC,CRRL,CRRT,
DRAG,DLRW,DRCT,DRDM,DRME,DROC,DRRL,DRRT,DRTY,EXSP,GRFR,GRRL,INRW,LKTB,ONBR,ONC
H,ONMN,ONPL,ONST,OPDB,RLTX,RNDB,RVFR,RVRL,SCSP,STDP,STDS,STEX,STIL,STLM,STOM,STOP,
STPR,STRL,STRS,STRT,STSA,STSC,STSN,STTX,ULTB,UPRW

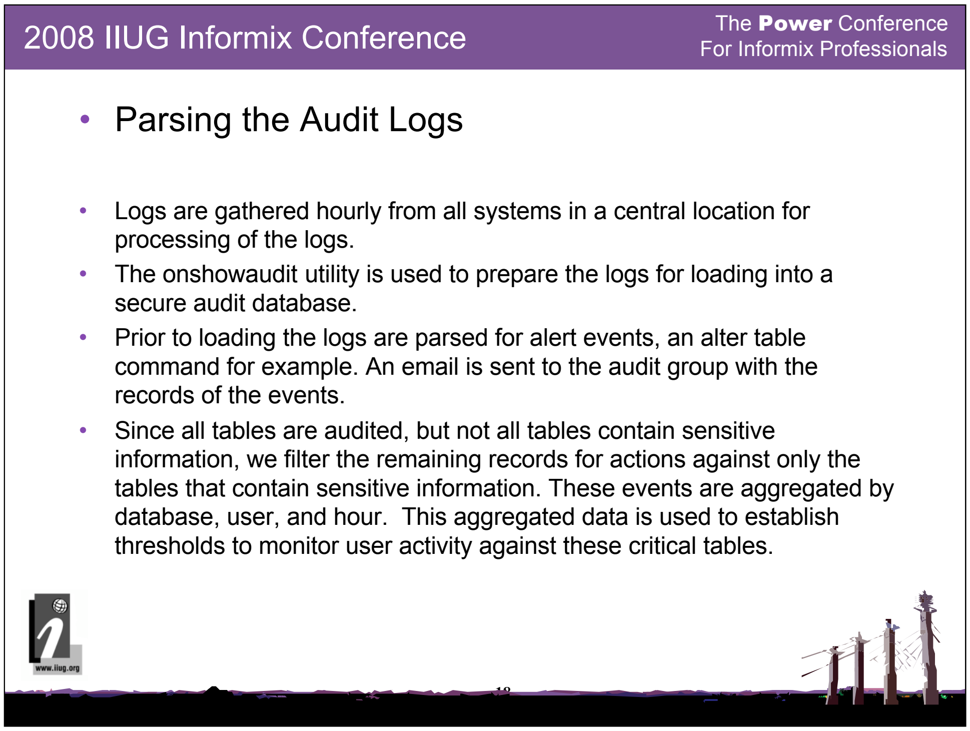


- Strengths
  - It is a part of the engine – secure, easy to set up and administer
  - Allows for role separation
  - Flexible which events and users to audit

- Weaknesses
  - May have a major impact on performance
  - Writes to the audit log can get queued up since it writes to a single audit log
  - Cannot specify which tables to audit
  - Use caution with:

```
select count(*) from table_a where col1 = “ABC”
```

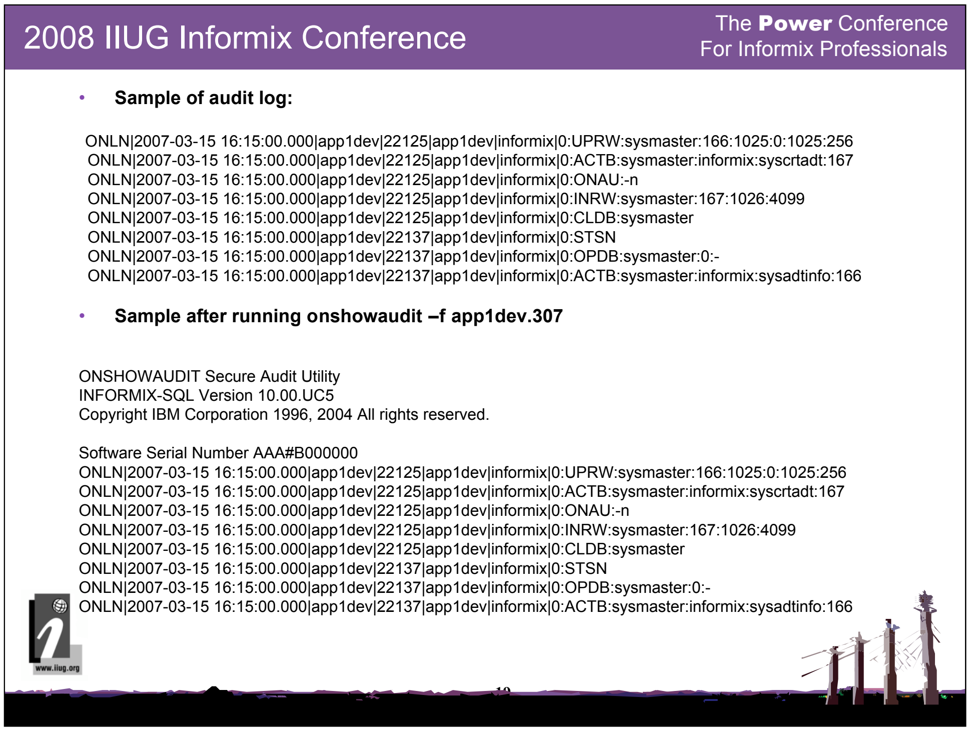
This will write a record to the audit log for every row in the table.
Use with caution on very large tables.



- Parsing the Audit Logs

- Logs are gathered hourly from all systems in a central location for processing of the logs.
- The onshowaudit utility is used to prepare the logs for loading into a secure audit database.
- Prior to loading the logs are parsed for alert events, an alter table command for example. An email is sent to the audit group with the records of the events.
- Since all tables are audited, but not all tables contain sensitive information, we filter the remaining records for actions against only the tables that contain sensitive information. These events are aggregated by database, user, and hour. This aggregated data is used to establish thresholds to monitor user activity against these critical tables.

Discuss alerts versus aggregates.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- **Sample of audit log:**

```
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:UPRW:sysmaster:166:1025:0:1025:256
 ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:ACTB:sysmaster:informix:syscrtadt:167
 ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:ONAU:-n
 ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:INRW:sysmaster:167:1026:4099
 ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:CLDB:sysmaster
 ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:STSN
 ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:OPDB:sysmaster:0:-
 ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:ACTB:sysmaster:informix:sysadtinfo:166
```

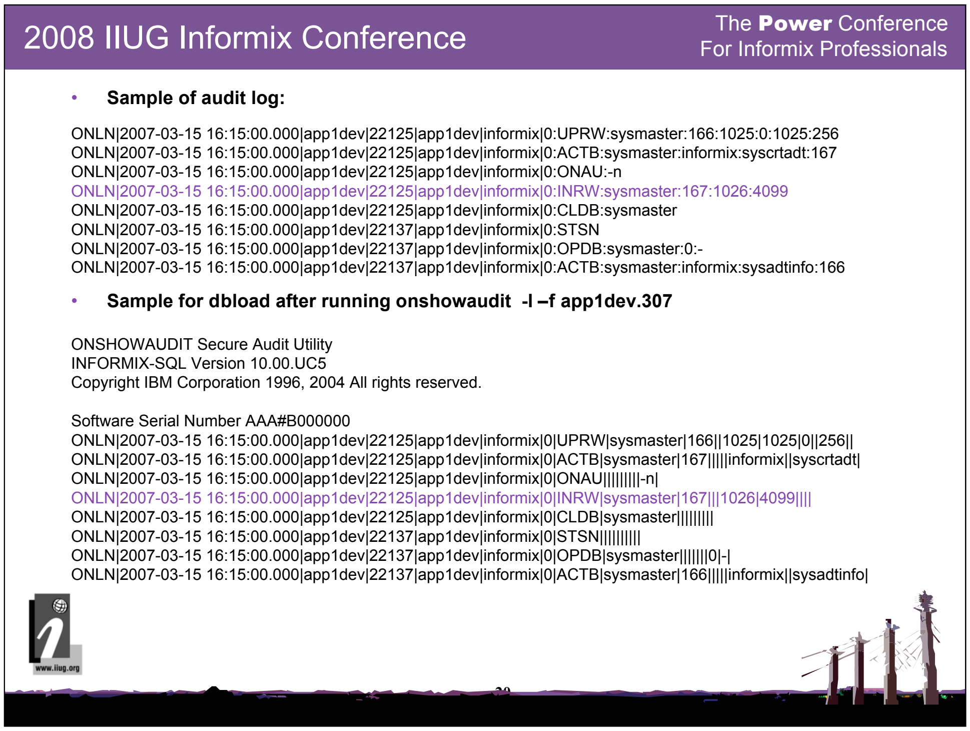
- **Sample after running onshowaudit –f app1dev.307**

```
ONSHOWAUDIT Secure Audit Utility
INFORMIX-SQL Version 10.00.UC5
Copyright IBM Corporation 1996, 2004 All rights reserved.

Software Serial Number AAA#B000000
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:UPRW:sysmaster:166:1025:0:1025:256
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:ACTB:sysmaster:informix:syscrtadt:167
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:ONAU:-n
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:INRW:sysmaster:167:1026:4099
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:CLDB:sysmaster
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:STSN
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:OPDB:sysmaster:0:-
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:ACTB:sysmaster:informix:sysadtinfo:166
```

We run the entire audit log through onshowaudit and the parse the output to do the aggregations.

Note the header lines to remove before running load or dbload.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- **Sample of audit log:**

```
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:UPRW:sysmaster:166:1025:0:1025:256
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:ACTB:sysmaster:informix:syscrtadt:167
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:ONAU:-n
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:INRW:sysmaster:167:1026:4099
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0:CLDB:sysmaster
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:STSN
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:OPDB:sysmaster:0:-
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0:ACTB:sysmaster:informix:sysadtinfo:166
```

- **Sample for dbload after running onshowaudit -l –f app1dev.307**

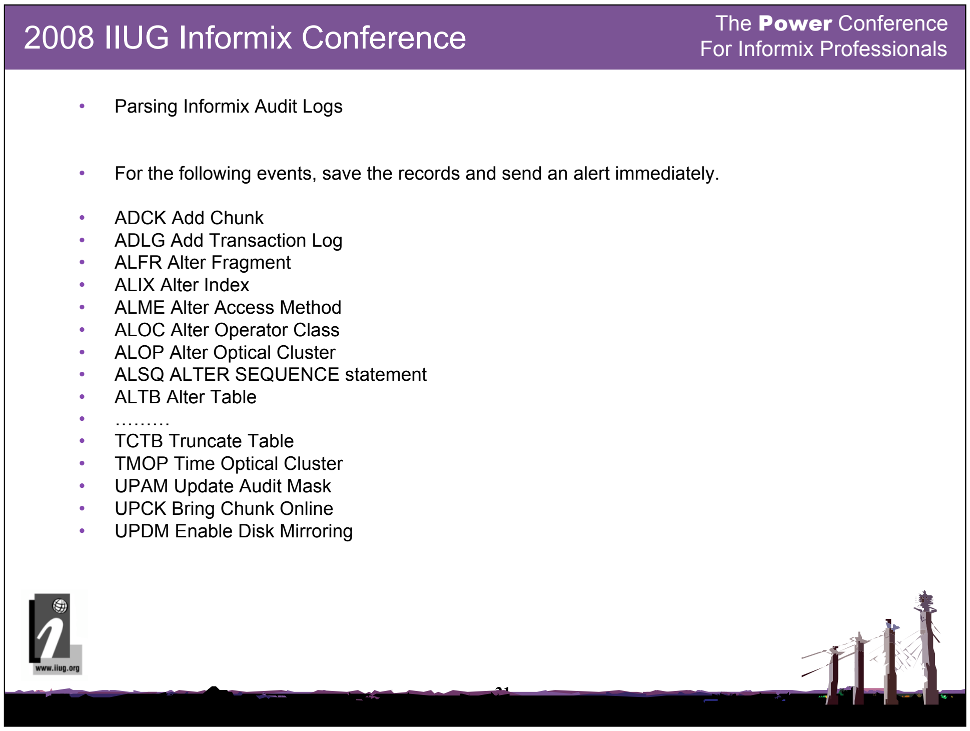
```
ONSHOWAUDIT Secure Audit Utility
INFORMIX-SQL Version 10.00.UC5
Copyright IBM Corporation 1996, 2004 All rights reserved.

Software Serial Number AAA#B000000
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0|UPRW|sysmaster|166||1025|1025|0||256||
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0|ACTB|sysmaster|167|||||informix||syscrtadt|
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0|ONAU|||||||||-n|
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0|INRW|sysmaster|167|||1026|4099||||
ONLN|2007-03-15 16:15:00.000|app1dev|22125|app1dev|informix|0|CLDB|sysmaster|||||||||
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0|STSN||||||||||
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0|OPDB|sysmaster|||||||0|-|
ONLN|2007-03-15 16:15:00.000|app1dev|22137|app1dev|informix|0|ACTB|sysmaster|166|||||informix||sysadtinfo|
```

www.iiug.org

Note the headers to be removed before loading.

Note the onshowaudit utility knows how to place the pipe delimiters for loading into a database. Cannot tell just looking at the colon-separated fields.



2008 IIUG Informix Conference

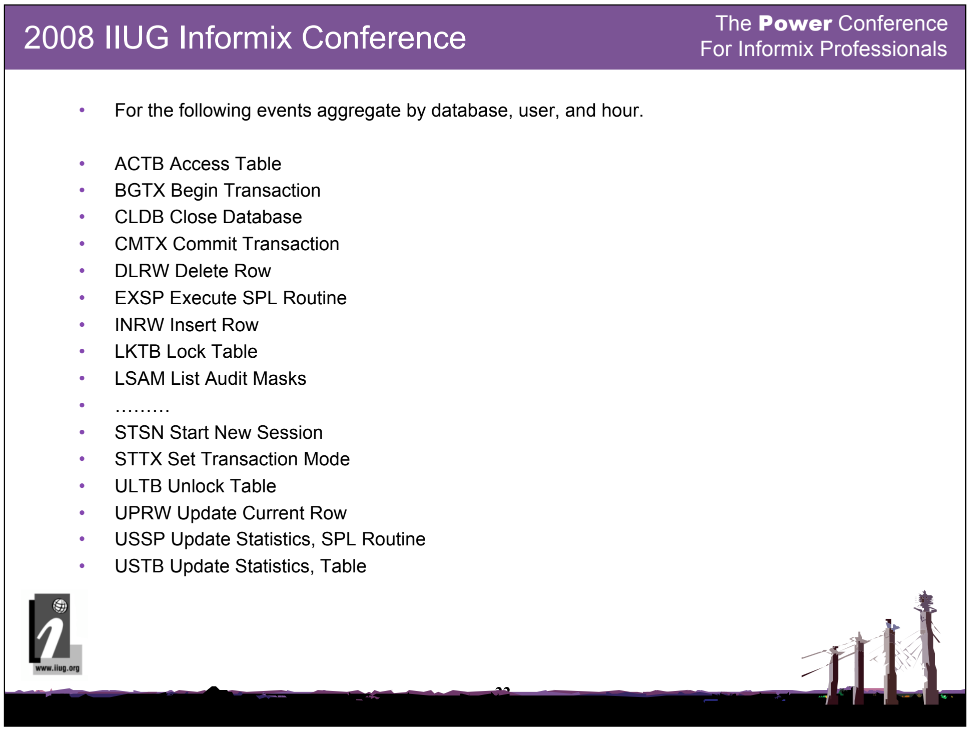
The **Power** Conference
For Informix Professionals

- Parsing Informix Audit Logs

- For the following events, save the records and send an alert immediately.

- ADCK Add Chunk
- ADLG Add Transaction Log
- ALFR Alter Fragment
- ALIX Alter Index
- ALME Alter Access Method
- ALOC Alter Operator Class
- ALOP Alter Optical Cluster
- ALSQ ALTER SEQUENCE statement
- ALTB Alter Table
- ………
- TCTB Truncate Table
- TMOP Time Optical Cluster
- UPAM Update Audit Mask
- UPCK Bring Chunk Online
- UPDM Enable Disk Mirroring

These are the events we send alerts for as soon as the log is parsed.



- For the following events aggregate by database, user, and hour.

- ACTB Access Table
- BGTX Begin Transaction
- CLDB Close Database
- CMTX Commit Transaction
- DLRW Delete Row
- EXSP Execute SPL Routine
- INRW Insert Row
- LKTB Lock Table
- LSAM List Audit Masks
- ………
- STSN Start New Session
- STTX Set Transaction Mode
- ULTB Unlock Table
- UPRW Update Current Row
- USSP Update Statistics, SPL Routine
- USTB Update Statistics, Table

These are the events we aggregate. After a period of time we determine thresholds for normal activity.

- Sample of tables that events are aggregated for:

  Tables to be audited and records retained by Informix auditing. 01/25/06
  cci_data
  client
  gnr_cci
  gst_credit
  hgnr
  msg_cci
  rjct_cci

- **Sample of an email alert for an alter table event on a production system:**

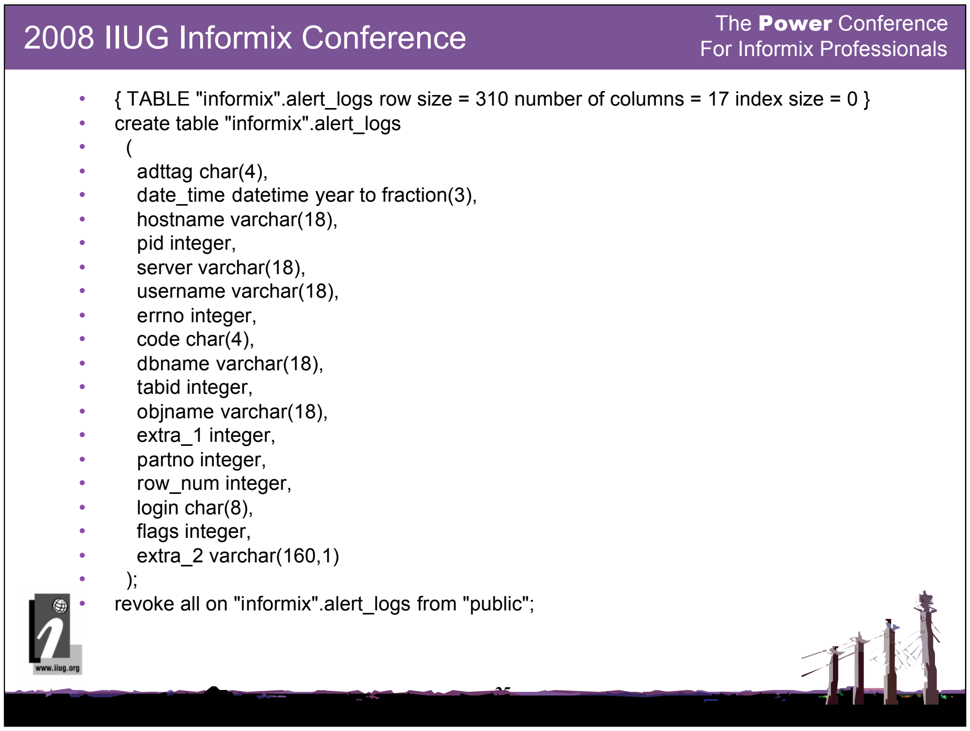
  Audited statements detected during last parsing operation.

```
ONLN|2007-02-07 14:14:00.000|app3host|6963|app3prod|informix|0|ALTB|app3db|1128||1133|||||40894489|
```

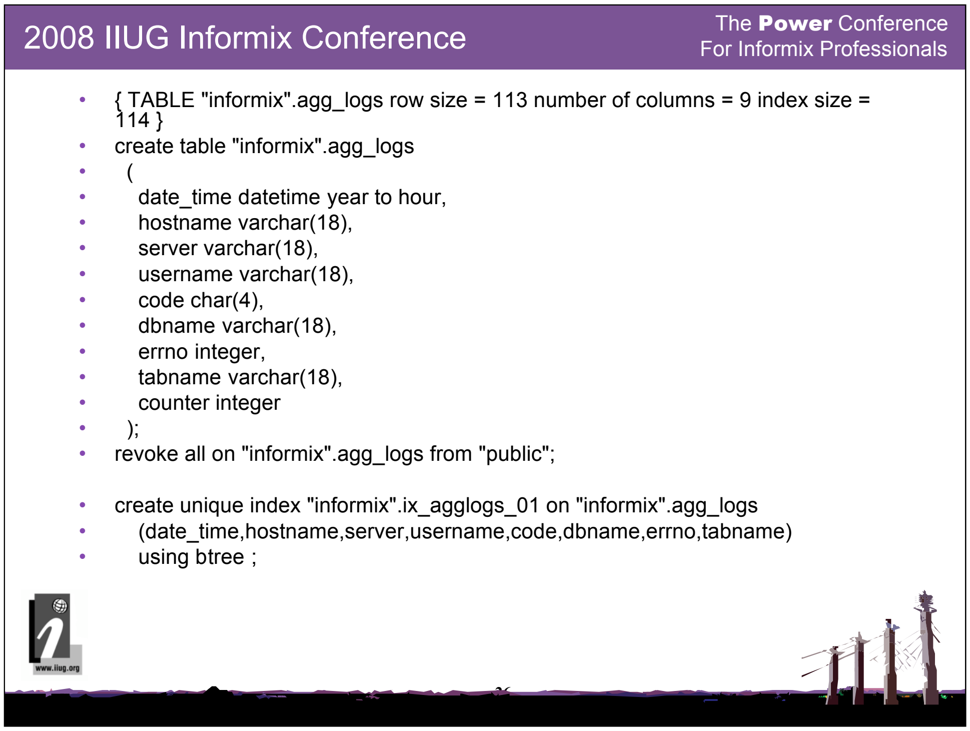
- **Since it was done by user informix and a planned change, this message can essentially be ignored. The records for these events are stored in the audit database in case they are needed for research later.**

Using a database for audit records

- The database contains two tables. One for the alert events and one for the aggregated events.
- Alert events are loaded after the alert email has been sent.
- Aggregated records are loaded after the aggregation has been done. Table numbers in the audit records are changed to table names using the tabnums for the appropriate systems.
- The schema for the two tables follows.



```
{ TABLE "informix".alert_logs row size = 310 number of columns = 17 index size = 0 }
create table "informix".alert_logs
  (
    adttag char(4),
    date_time datetime year to fraction(3),
    hostname varchar(18),
    pid integer,
    server varchar(18),
    username varchar(18),
    errno integer,
    code char(4),
    dbname varchar(18),
    tabid integer,
    objname varchar(18),
    extra_1 integer,
    partno integer,
    row_num integer,
    login char(8),
    flags integer,
    extra_2 varchar(160,1)
  );
revoke all on "informix".alert_logs from "public";
```



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

```
{ TABLE "informix".agg_logs row size = 113 number of columns = 9 index size =
114 }
create table "informix".agg_logs
  (
    date_time datetime year to hour,
    hostname varchar(18),
    server varchar(18),
    username varchar(18),
    code char(4),
    dbname varchar(18),
    errno integer,
    tabname varchar(18),
    counter integer
  );
revoke all on "informix".agg_logs from "public";

create unique index "informix".ix_agglogs_01 on "informix".agg_logs
    (date_time,hostname,server,username,code,dbname,errno,tabname)
    using btree ;
```

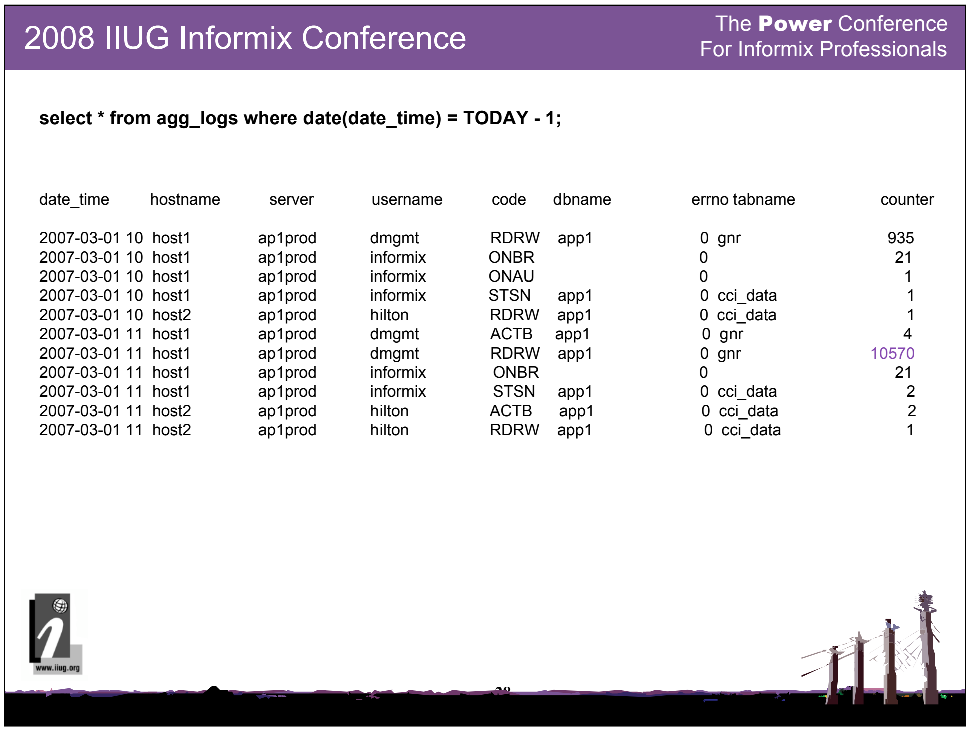
Note datetime is year to hour since we aggregate on an hourly basis.

The errno is kept to determine success or failure of the action.

The counter is the aggregate count for the event.

select * from alert_logs where date(date_time) = TODAY - 1;

```
adttag       ONLN
date_time    2007-02-28 15:23:35.000
hostname     app1dev
pid          13275
server       comuat
username     informix
errno        0
code         RVTB
dbname       app1
tabid        337
objname
extra_1      2
partno
row_num
login        informix
flags        0
extra_2      public
```



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

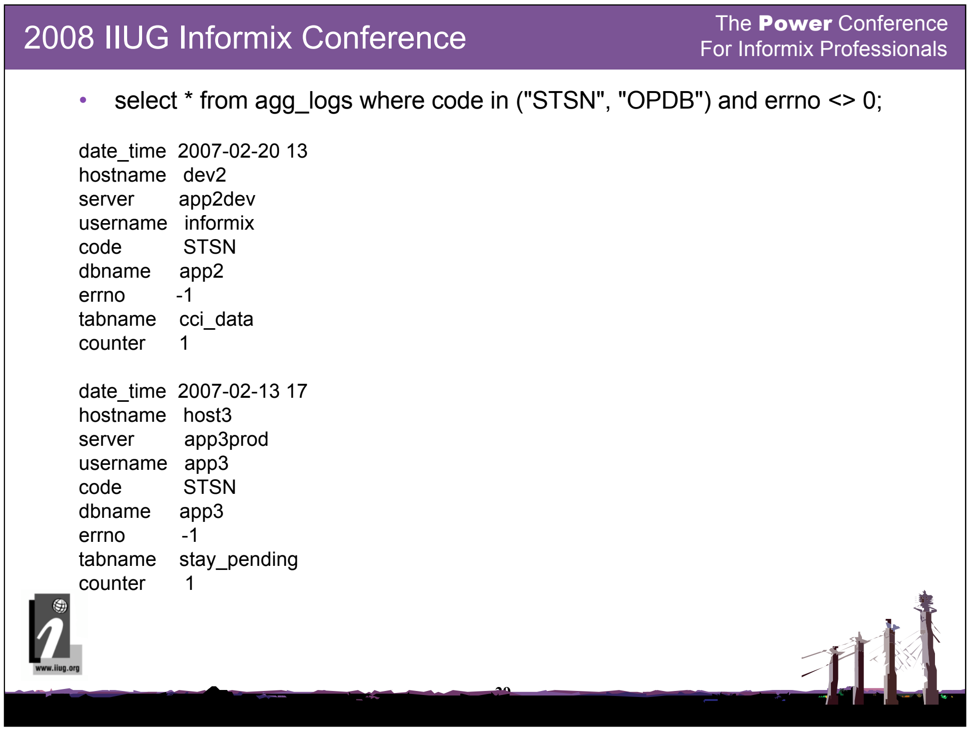
select * from agg_logs where date(date_time) = TODAY - 1;

```
date_time      hostname     server     username    code  dbname        errno tabname       counter

2007-03-01 10  host1        ap1prod    dmgmt       RDRW   app1           0  gnr               935
2007-03-01 10  host1        ap1prod    informix    ONBR                  0                     21
2007-03-01 10  host1        ap1prod    informix    ONAU                  0                      1
2007-03-01 10  host1        ap1prod    informix    STSN   app1           0  cci_data            1
2007-03-01 10  host2        ap1prod    hilton      RDRW   app1           0  cci_data            1
2007-03-01 11  host1        ap1prod    dmgmt       ACTB   app1           0  gnr                 4
2007-03-01 11  host1        ap1prod    dmgmt       RDRW   app1           0  gnr             10570
2007-03-01 11  host1        ap1prod    informix    ONBR                  0                     21
2007-03-01 11  host1        ap1prod    informix    STSN   app1           0  cci_data            2
2007-03-01 11  host2        ap1prod    hilton      ACTB   app1           0  cci_data            2
2007-03-01 11  host2        ap1prod    hilton      RDRW   app1           0  cci_data            1
```

Note the datetime column has the date and hour aggregations.

Note the high count for the dmgmt user.



2008 IIUG Informix Conference

The **Power** Conference
For Informix Professionals

- select * from agg_logs where code in ("STSN", "OPDB") and errno <> 0;

```
date_time  2007-02-20 13
hostname   dev2
server     app2dev
username   informix
code       STSN
dbname     app2
errno      -1
tabname    cci_data
counter    1

date_time  2007-02-13 17
hostname   host3
server     app3prod
username   app3
code       STSN
dbname     app3
errno      -1
tabname    stay_pending
counter    1
```

This is a query for failed session attempts or connects to the database. A failed STSN may also show up in the informix log as user not trusted or with an invalid password. A failed OPDB (user without connect privilege) will not show up in the informix log.

- For documentation see the IBM website:

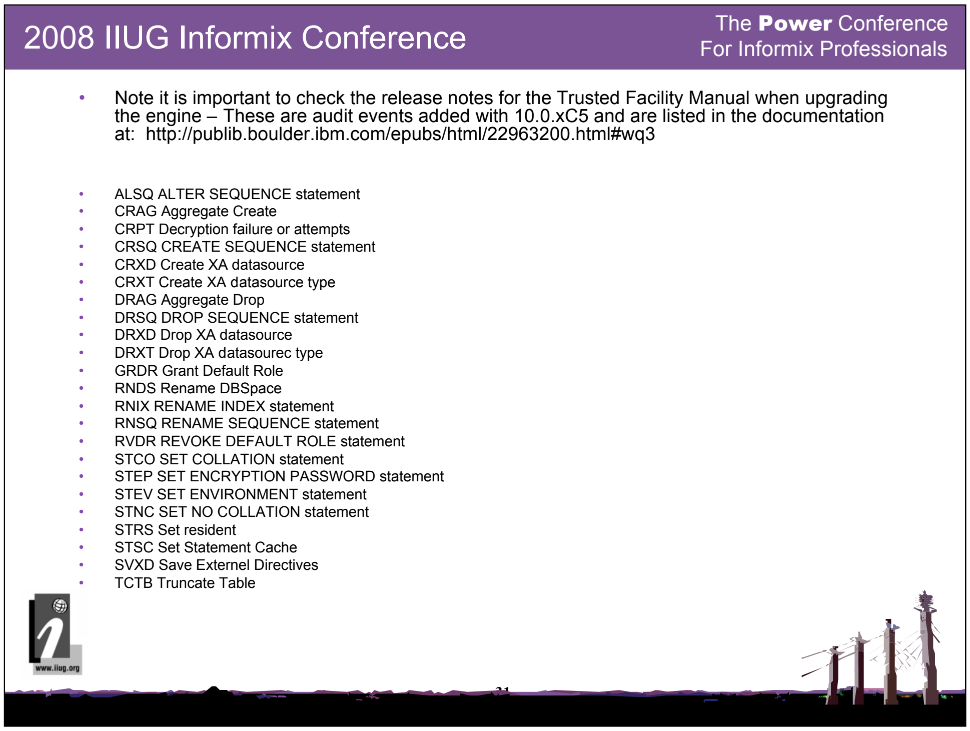
http://www-306.ibm.com/software/data/informix/pubs/library/ids_100.html

- For the Trusted Facility Manual (10.00.xC5) release notes:

http://publib.boulder.ibm.com/epubs/html/22963200.html

- For the Trusted Facility Manual (10.00.xC6) release notes:

http://publib.boulder.ibm.com/epubs/html/c2362420.html

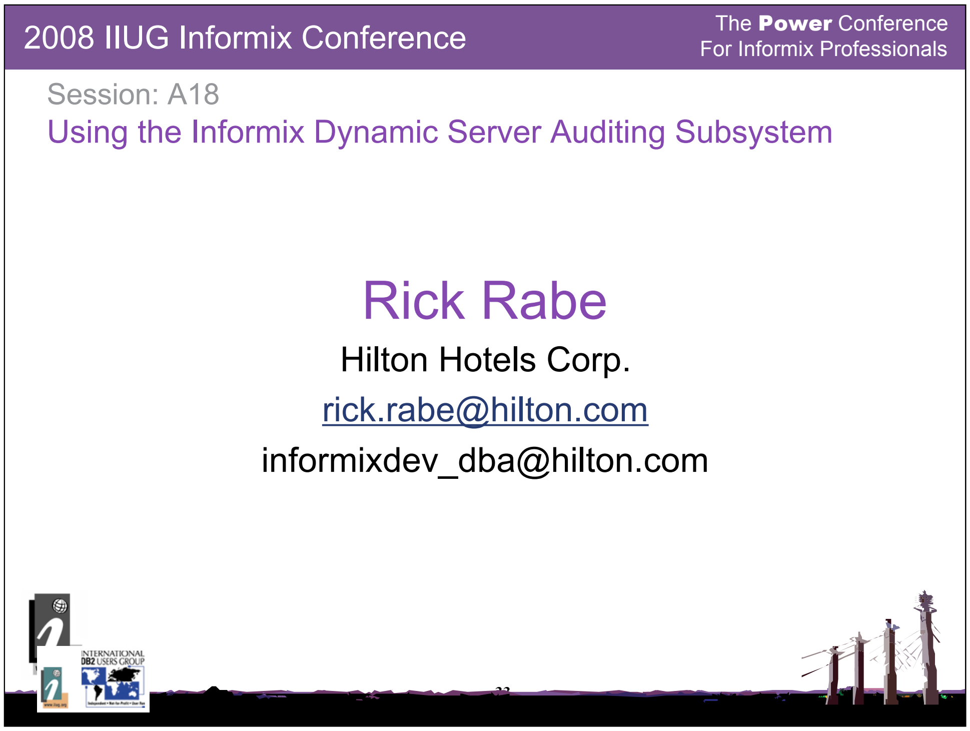


- Note it is important to check the release notes for the Trusted Facility Manual when upgrading the engine – These are audit events added with 10.0.xC5 and are listed in the documentation at: http://publib.boulder.ibm.com/epubs/html/22963200.html#wq3

- ALSQ ALTER SEQUENCE statement
- CRAG Aggregate Create
- CRPT Decryption failure or attempts
- CRSQ CREATE SEQUENCE statement
- CRXD Create XA datasource
- CRXT Create XA datasource type
- DRAG Aggregate Drop
- DRSQ DROP SEQUENCE statement
- DRXD Drop XA datasource
- DRXT Drop XA datasourec type
- GRDR Grant Default Role
- RNDS Rename DBSpace
- RNIX RENAME INDEX statement
- RNSQ RENAME SEQUENCE statement
- RVDR REVOKE DEFAULT ROLE statement
- STCO SET COLLATION statement
- STEP SET ENCRYPTION PASSWORD statement
- STEV SET ENVIRONMENT statement
- STNC SET NO COLLATION statement
- STRS Set resident
- STSC Set Statement Cache
- SVXD Save Externel Directives
- TCTB Truncate Table

There were 23 audit events added with 10.00.xC5.

Questions?



Session: A18

Using the Informix Dynamic Server Auditing Subsystem

Rick Rabe

Hilton Hotels Corp.

rick.rabe@hilton.com

informixdev_dba@hilton.com