Средства контроля за доступом в Informix Dynamic Server

В процессе эксплуатации ИС (информационной системы) частенько возникает необходимость отслеживать доступ к информации, то есть следить за действиями определенных пользователей, следить за определенным типом действий и т.д. Возможное решение этой проблемы - триггеры. Между тем, триггеры являются частью схемы базы данных, и не всегда удобно менять схему ради контроля за действиями пользователей. Помимо этого, не существует триггеров на оператор выборки (SELECT).

Informix Dynamic Server предоставляет специальный механизм контроля за действиями пользователей - механизм аудитинга. С помощью этого механизма можно в полном объеме отследить кто, когда и что делал с конкретными данными. Интересно, что наличие самой возможности полного аудитинга, невидимого для пользователя, может предостеречь пользователя от попыток несанкционированных действий.)

1. Как работает аудитинг?

Встроенный механизм аудитинга позволяет фиксировать в специальных файлах (журналах) любые действия любых пользователей.

Для обеспечения работы механизма аудитинга в СУБД Informix дополнительно вводятся два новых типа пользователей - “заведующий безопасностью базы данных” (DBSSO - Database System Security Officer) и “аналитик по вопросам безопасности” (AAO - Audit Analysis Officer). Первый конфигурирует механизм аудитинга, при необходимости включает или выключает его и т.д. Второй имеет право просматривать журнал действий пользователя.

Конфигурирование механизма аудитинга подразумевает:

- указание действий, которые в обязательном порядке будут фиксироваться в журналах для любого пользователя;
- указание пользователей, для которых надо (или, наоборот, не надо) фиксировать те или иные действия;
- указание действий, которые надо протоколировать для того или иного

пользователя.

В качестве фиксируемых действий могут выступать практически любые действия по доступу и модификации данных в БД, а также и административные действия.

Необходимо отметить, что никто, кроме заведующих безопасностью базы данных и системных администраторов не может знать, включен в данный момент времени аудитинг или нет. А когда аудитинг включен, то нельзя узнать, протоколируется ли то или иное действие для того или иного пользователя, или нет.

Для того, чтобы можно было использовать аудитинг, необходимо выполнить следующие действичя:

1. Определить перечень пользователей, имеющих право управлять аудитингом и просматривать журнал аудитинга;
2. Определить, что будет фиксировать в журнале механизм аудита (определить протоколируемые действия пользователей);
3. Сконфигурировать механизм аудита
4. Запустить механизм аудитинга.

Рассмотрим эти действия подробнее.

Определения специальных пользователей

Для работы с механизмом аудитинга в СУБД Informix дополнительно вводятся два новых типа пользователей - “заведующий безопасностью базы данных” (DBSSO) и “аналитик по вопросам безопасности” (AAO). Наличие или отсутствие таких специальных типов пользователей определяется в процессе инсталляции сервера Informix. По умолчанию (когда не предпринимать никаких специальных действий) пользователь informix может и должен выполнять роль и заведующего безопасностью, и аналитика.

Пользователь может выполнять роль DBSSO, когда он принадлежит группе пользователей (имеется в виду группа пользователей с точки зрения ОС), которая владеет директорией $INFORMIXDIR/dbssodir) . Пользователь может выполнять роль

AAO, когда он принадлежит группе пользователей (имеется в виду группа пользователей с точки зрения ОС), которая владеет директорией $INFORMIXDIR/aaodir. Например, в следующем примере видно, что пользователи из группы inf_aoo являются аналитиками, а пользователи из группы inf_dbsso могут выполнять роль DBSSO:

```
  # ls -al
total 390
drwxr-xr-x 16 informix informix 512 Nov 30 05:17 .
drwxrwxr-x 11 informix informix 512 Jan 30 12:12 ..
drwxrwx--- 2 a_aao ix_aao 512 Jan 6 14:11 aaodir
drwxrwxr-x 2 informix informix 1536 Oct 6 19:44 bin
drwxrwx--- 2 a_dbsso ix_dbsso 512 Sep 25 13:19 dbssodir
drwxrwxr-x 2 informix informix 1536 Jan 30 11:56 etc
. . . . . .
```

Для того, чтобы во время установки сконфигурировать сервер базы данных так, что бы для вопросов безопасности были выделены пользователи, отличные от пользователя informix (то есть вопросы безопасности были отделены от вопросов администрирования базы данных) необходимо установить переменную окружения INF_ROLE_SEP в какое-либо значение, например “1”:

```
  INF_ROLE_SEP=1
export INF_ROLE_SEP
```

и только после этого запускать утилиту инсталляции сервера.

Добавление новых пользователей, которые смогут администрировать процесс аудитинга или анализировать его результаты, сводится к созданию нового пользователя - члена соответствующей группы или, когда речь идет об уже существующем пользователе, приписывания его к соответсвующей группе.

2. Конфигурирование списков протоколируемых событий

Что такое “маска”

Маска определяет перечень действий, информация о которых будет фиксироваться. Существуют глобальные маски, применимые ко всем пользователям, и индивидуальные маски, применимые к конкретному пользователю. Каждая маска имеет имя. Имеются следующие три глобальные маски:

- _default - маска по умолчанию, которая применяется ко всем пользователям, у которых нет индивидуальной маски.

- _require - маска, в которой указаны действия, которые будут протоколироваться для всех пользователей.
- _exclude - маска, которая содержит перечень всех действий, которые не будут протоколироваться для любого пользователя.

Для каждого конкретного пользователя может быть создана индивидуальная маска. В этом случае, имя индивидуальной маски для пользователя совпадает с системным именем пользователя. Например, для пользователя andy индивидуальная маска тоже будет называться “andy”. Естественно, названия индивидуальных масок не должны совпадать с именами глобальных масок (_default, _require, _exclude).

Алгоритм применения масок следующий. Как только некоторый пользователь выполняет какое-либо действие, встроенный в сервер баз данных механизм аудитинга проверяет надо ли протоколировать это действие. когда у пользователя есть индивидуальная маска, то она просматривается. когда индивидуальной маски нет, то просматривается маска _default. Затем просматривается глобальная обязательная маска _require. когда ни одна из этих масок не содержит описание выполняемого действия, то данное действие не протоколируется. когда же выполняемое действие в одной из этих масок указано, то оно будет запротоколировано, но только когда данное действие не указано в маске _exclude.

Другими словами, для пользователя никогда не будут протоколироваться действия, указанные в глобальной маске _exclude. когда же действие не указано в маске _exclude, но указано в ндивидуальной маске (или в маске _default, когда индивидуальной маски нет) или в глобальной маске _require, то оно будет запротоколировано.

Итак, маска - это набор действий. Следовательно, нужно уметь различать, идентифицировать возможные действия. Каждое событие, каждое действие в системе аудитинга имеет свой 4-х символьный код. Полный перечень кодов приведен в системной документации. Здесь, в качестве примера, дадим перечень некоторых действий, наиболее интересных с точки зрение отслеживания действий пользователей по доступу к информации:

- ACTB - доступ к таблице;
- LSDB - получение списка баз данных;
- OPDB - открытие базы данных;
- RDRW - чтение одной записи.

Просмотр, создание, модификация и удаление масок

Только пользователи из группы DBSSO могут просматривать, создавать, удалять и модифицировать маски. Для выполнения данных действий служит утилита onaudit. Для того, чтобы вывести на экран информацию о всех существующих масках, необходимо выполнить команду

```
onaudit -o
```

когда выполнить эту команду сразу после инсталляции сервера Informix Dynamic Server, то вы не увидите ни одной существующей маски (в том числе и глобальные маски). Далее, требуемые маски надо создать. Предположим, мы желаем для всех пользователей в обязательном порядке протоколировать открытие базы данных. Для этого надо в глобальную маску _require добавить событие OPDB (“открытие базы данных”)

```
onaudit -a -u _require -e OPDB
```

Утилита onaudit, пример использования которой приведен выше, имеет следующий ситаксис для создания маски с некоторым набором событий:

```
onaudit -a -u <имя маски> -e <код события>, <код события>, ……
```

С помощью утилиты onaudit можно проводить и модификацию существующей маски - добавлять или удалять события маски. Для этого используется опция “-m”. Добавляемые в маску события надо указать после опции “-e”, а удаляемые из маски события указываются после опции “-e” и знака минус:

```
onaudit -m -u <имя маски> -e - <код удаляемого события>, ……  onaudit -m -u <имя маски> -e <код добавляемого события>, ……
```

Опция “-d” утилиты onaudit задает команду удаления маски:

```
onaudit -d -u <имя маски>
```

Например, следующий набор команд создает индивидуальную маску для пользователя micky, содержащую фиксацию всех операций чтения данных, из глобальной маски _default удаляет фиксацию события RDRW (чтение какого-либо ряда из таблицы), и целиком удаляет индивидуальную маску для пользователя andy:

```
onaudit -a -u micky -e OPDB, ACTB, LSDB, RDRW
onaudit -m -u _default -e - RDRW
onaudit -d -u andy
```

Задание дополнительных условий на событие

Система аудитинга, реализованная в Informix Dynamic Server, позволяет дополнительно специфицировать протоколируемые действия. А именно, можно указать, что надо не всегда протоколировать некоторое действие, а только в том случае, когда данное действие было успешно выполнено. Или наоборот, мы можем задать протоколирование только неудачной попытки выполнить некоторое действие. Для этого в командах создание (onaudit -a) или модификации (onaudit -m) списка событий для маски перед кодом действия надо указать символ “F” (для протоколирования неуспешных попыток) или символ “S” (для протоколирования успешных событий). Например:

```
onaudit -m -u micky -e FOPDB, SACTB
```

3. Задание файлов, запуск и остановка механизма аудитинга

Задание файлов для записи протокола

Для протоколирования действий пользователей используются файлы. Одним из обязательных шагов по запуску системы аудитинга должно быть задание расположение файлов протокола. Файлы протокола - это обычные файлы ОС, куда включенный механизм аудитинга будет помещать записи о действиях пользователя. Аналитик по безопасности должен специфицировать директорию, где эти файлы будут располагаться.

Изначально файлы протокола располагаются в директории, на которую указывает параметр ADTPATH файла $INFORMIXDIR/aaodir/adtcnf. когда сразу после инсталляции сервера посмотреть на данный параметр, то он будет установлен в /tmp. Временная директория /tmp не самое удачное место для хранения файлов протокола. Можно поменять значение данного параметра, но это должно быть сделано до инициализации экземпляра сервера (то есть до выполнения команды oninit -i). когда не менять значение данного параметра, то указываемая им директория и будет первой директорией для файлов протокола. Поменять эту директорию можно после запуска механизма аудитинга командой

```
onaudit -p <директория>
```

например

```
onaudit -p /usr/audit
```

когда менанизм аудитинга не включен или на указываемую директорию аналитик по безопасности не имеет прав записи и чтения, то будет выдано сообщение об ошибке.

В директории, указанной для хранения протоколов будут автоматически создаваться

файлы с названиями вида <имя сервера БД>.<порядковый номер>. В качестве имени сервера будет выступать значение параметра INFORMIXSERVER, а порядковый номер - просто номер файла протокола. Дело в том, что система аудитинга имеет ограничение по размеру файлов протокола (параметр ADTSIZE файла $INFORMIXDIR/aaodir/adtcnf) и по достижению очередного файла протокола произойдет автоматический переход на новый файл. Можно и принудительно перейти к следующему файлу протокола, выполнив команду

```
onaudit -n
```

Данная команда может быть выполнена только при включенном механизме аудитинга. Пример содержимого директории с протоколами для сервера по имени shm_ncr1:

```
# echo $INFORMIXSERVER
shm_ncr1

# ls -l /usr/audit
total 3
-rw-rw---- 1 informix ix_aao 234 Mar 5 12:41 shm_ncr1.0
-rw-rw---- 1 informix ix_aao 78 Mar 5 13:18 shm_ncr1.1

# onaudit -n

Onaudit -- Audit Subsystem Configuration Utility
Copyright (C) Informix Software, Inc., 1996


# ls -l
total 5
-rw-rw---- 1 informix ix_aao 234 Mar 5 12:41 shm_ncr1.0
-rw-rw---- 1 informix ix_aao 78 Mar 5 13:18 shm_ncr1.1
-rw-rw---- 1 informix ix_aao 0 Mar 5 13:24 shm_ncr1.2
```

Включение и выключение механизма аудитинга.

Для того, чтобы узнать включен ли или выключен механизм аудитинга в данный момент времени, управляющий безопасностью может выполнить команду

```
onaudit -c
```

Данная команда, среди нескольких параметров, выдает и параметр ADTMODE. Этот параметр указывает на текущее состояние системы аудитинга. когда он равен 0, то механизм аудитинга выключен, никакого протоколирования не ведется. когда он находится в диапазоне от 1 до 8, то это означает, что механизм аудитинга включен:

```
 # onaudit -c
Onaudit -- Audit Subsystem Configuration Utility
Copyright (C) Informix Software, Inc., 1996


Current audit system configuration:
ADTMODE = 3
ADTERR = 0
ADTPATH = /usr/audit
ADTSIZE = 50000
Audit file = 3
```

Разные значения параметра ADTMODE соответствуют разным уровням и способам протоколирования. Не вдаваясь в подробности, которые всегда можно найти в документации, будем считать что нам для нормального протоколирования нужен уровень “3”. Соответственно, перевод механизма аудитинга на данный уровень с другого уровня, или просто включение механизма аудитинга происходводится командой:

```
onaudit -l 3
```

4. Анализ протокола

Анализом протоколов занимается аналитик системы безопасности (ААО). При запущенном механизме аудитинга в файлы с протоколами пишется информация о действиях пользователей. Эти файлы имеют обычный текстовый формат и могут быть просмотрены стандартными средствами ОС. Например:

```
 # cat /usr/audit/shm_ncr1.3
ONLN|1998-03-05 14:47:09.000|ncr1|15646|shm_ncr1|roma|0:OPDB:sysmaster:0:-
ONLN|1998-03-05 15:00:27.000|ncr1|15681|shm_ncr1|roma|0:OPDB:sysmaster:0:-
ONLN|1998-03-05 15:00:27.000|ncr1|15681|shm_ncr1|roma|0:UPRW:sysmaster:149:10
ONLN|1998-03-05 15:00:27.000|ncr1|15681|shm_ncr1|roma|0:ACTB:sysmaster:infor
ONLN|1998-03-05 15:00:27.000|ncr1|15681|shm_ncr1|roma|0:ONAU:-l 3
ONLN|1998-03-05 15:00:27.000|ncr1|15681|shm_ncr1|roma|0:INRW:sysmaster:150:18
ONLN|1998-03-05 15:00:27.000|ncr1|15681|shm_ncr1|roma|0:CLDB:sysmaster
```

По данной информации можно увидеть, что 5-го Марта, пользователь roma два раза (в 14:47 и в 15:00) обращался к базе данных с действиями, подлежащими протоколированию. Оба раза он находился на компьютере ncr1. Первый раз он открыл базу данных sysmaster. Второй раз он не только открыл базу данных, но и обновил запись (событие с кодом UPRW), прочитал таблицу (ACTB), выполнил некоторые действия по аудитингу (ONAU), вставил новую запись (INRW) и закрыл базу данных (CLDB). Полностью способ представления записей в файле протокола приведен в документации (Informix Dynamic Server/Trusted Facility Manual).

Для более направленного анализа файлов протокола все же лучше воспользоваться утилитой onshowaudit. Только аналитик (AAO) может запускать данную утилиту. Данная утилита позволяет проводить целенаправленный поиск по пользователям или серверам баз данных. Данная утилита также может осуществлять загрузку информации из файла протокола в базу данных.

Для того, чтобы выяснить, какие действия выполнял некоторый пользователь, используется следующий вариант утилиты onshowaudit:

```
onshowaudit -l -u <имя пользователя>
```

Для того, чтобы выяснить, какие действия выполнялись с некоторым сервером базы данных Informix, используется следующий вариант утилиты onshowaudit:

```
onshowaudit -l -s <имя сервера БД>
```

Утилиту onshowaudit можно использовать и совместно со стандартными утилитами операционной системы UNIX, такими, как grep, awk и т.д. Например, для того, чтобы выяснить, кто получал списки доступных баз данных на сервере shm_ncr1, надо выполнить сдледующую Unix-команду:

```
onshowaudit -l -s shm_ncr1 | grep LSDB
```

5. Практические советы или Что делать, когда вы хотите…

Выяснить, кто выполняет определенные действия

Предположим, нам надо выяснить, кто меняет структуру таблиц в базе данных base1 на сервере online1.

1. Проверьте, что аудитинг включен (onaudit -c). когда аудитинг выключен, включите его (onaudit -l 3).
2. Уберите из маски _exclude событие ALTB (onaudit -m -u _exclude -e - ALTB).
3. Введите в маску _require событие ALTB (onaudit -m -u _require -e ALTB).
4. Через некоторое время выполните команду onshowaudit и проанализируйте ее результаты (onshowaudit -l -s online1 | grep ALTB | grep base1)

Выяснить, когда кто-то приступает к работе с ИС и когда завершает

Предположим, нам надо выяснить, когда пользователь kate начинает работу с сервером базы данных online1, а когда завершает. Другими словами, когда она стартует свое приложение, а когда выходит из него.

1. Проверьте, что аудитинг включен (когда нет, включите его).
2. Уберите из маски _exclude события OPDB и CLDB (onaudit -m -u _exclude -e - OPDB, CLDB).
3. Введите в индивидуальную маску пользователя kate события OPDB и CLDB (onaudit -m -u kate -e OPDB, CLDB).
4. Через некоторое время выполните команду onshowaudit и проанализируйте ее результаты (onshowaudit -l -u kate)