

# **Архитектура ORACLE**

**Управление безопасностью**

**Лекция 9**

# Привилегии и роли

---

- ▶ GRANT / REVOKE
- ▶ Системные и объектные привилегии
- ▶ Нельзя выдавать в одном предложении



# Системные привилегии

---

- ▶ WITH ADMIN OPTION – дают право пользователю также назначать/отбирать привилегии
  - ▶ ALTER
  - ▶ ANALYZE
  - ▶ AUDIT
  - ▶ BACKUP
  - ▶ CREATE
  - ▶ DROP
  - ▶ SELECT
- ▶ ANY – для любого объекта
- ▶ ALL – для всех объектов



# Объекты грантов для системных привилегий

---

- ▶ DATABASE
- ▶ USER
- ▶ PROFILE
- ▶ TABLESPACE
- ▶ ROLE
- ▶ TABLE
- ▶ INDEX
- ▶ TRIGGER
- ▶ PROCEDURE
- ▶ SEQUENCE
- ▶ VIEW



# Объектные привилегии

---

- ▶ WITH GRANT OPTION – дают право пользователю также назначать/отбирать привилегии
  - ▶ ALTER
  - ▶ DELETE
  - ▶ EXECUTE
  - ▶ INSERT
  - ▶ UPDATE
  - ▶ SELECT
  - ▶ REFERENCES
- ▶ Снимает привилегии тот, кто их назначил



# Объекты грантов для объектных привилегий

---

- ▶ TABLE
- ▶ VIEW
- ▶ SEQUENCE
- ▶ PROCEDURE



# Аудит

---

- ▶ Определение неавторизованного доступа
- ▶ Должен быть постоянно активен для:
  - ▶ Проверки доступа пользователей (особенно неуспешные попытки и доступ в нерабочее время)
  - ▶ Изменений структуры базы данных (пользователи не могут изменять структуру БД, администратор должен вносить изменения в специально отведенное время)
  - ▶ Использования системных привилегий
- ▶ Для успешных (successful) и для неуспешных (not successful) попыток
- ▶ Необходимо включать только для необходимых объектов
- ▶ По умолчанию выключен



# Уровни аудита

---

- ▶ На уровне объектов схемы
  - ▶ На уровне SQL- выражений
  - ▶ На уровне системных привилегий
- 
- ▶ Дополнительно:
    - ▶ BY username - будут фиксироваться действия только конкретного пользователя
    - ▶ WHENEVER SUCCESSFUL / WHENEVER NOT SUCCESSFUL – фиксируются успешные или безуспешные попытки
    - ▶ BY SESSION / BY ACCESS – как фиксируются действия: одна запись на сеанс или одна на каждое действие в течение сеанса



# Детальный аудит

---

- ▶ Системные триггеры
- ▶ DML-триггеры
- ▶ Детализированный аудит (fine-grained) – пакет dbms\_fga



# Конфигурация аудита

---

- ▶ Включение аудита в параметрах: audit\_trail = db
- ▶ Проверка, что аудит включен:  

```
select name, value from v$parameter where name like 'audit%';
```
- ▶ Пока ничего не отслеживается, кроме важных действий  
(подключения привилегированных пользователей, запуск,  
останов, добавление файла данных)
- ▶ Изменения в oracle\_home/rdbms/audit по умолчанию
- ▶ Можно переопределить audit\_file\_dest в файле параметров



# Проверка отслеживаемых действий аудита

---

- ▶ Проверка, что выражения отслеживаются:

```
select * from dba_stmt_audit_opts
```

- ▶ Проверка, что привилегии отслеживаются:

```
select * from dba_priv_audit_opts;
```

- ▶ Проверка, что объекты отслеживаются:

```
select * from dba_obj_audit_opts;
```



# Формат команды включения аудита

---

- ▶ audit { statement\_option | privilege\_option } [ by user ]  
[ by {session | access} ] [whenever { successful | unsuccessful } ]
  
- ▶ Аудит могут задать пользователи с привилегией audit system:  
select \* from dba\_sys\_privs where privilege like '%audit%';
  
- ▶ Примеры:
- ▶ audit create session;
- ▶ audit create table;
- ▶ audit create index;
- ▶ audit alter trigger;
- ▶ audit drop table;
- ▶ audit execute procedure;



# Контроль журнала аудита

---

- ▶ Каждая запись аудита добавляет запись в системную таблицу aud\$ – табличное пространство system, владелец – sys
- ▶ Только sys может удалять данные из aud\$
- ▶ Роли select\_catalog\_role, delete\_catalog\_role имеют право доступа к aud\$ (добавить пользователя)
- ▶ Возможная атака – отказ в обслуживании (из-за роста размера таблицы и ограничения размера табличного пространства system)
- ▶ Тактика – регулярно копировать данные из aud\$ и усекать таблицу
- ▶ Выборки:
  - ▶ select \* from sys. aud\$;
  - ▶ select \* from dba\_audit\_trail;
  - ▶ select \* from dba\_audit\_session;



# Контроль журнала аудита

---

- ▶ Неудачные попытки входа (количество):
  - ▶ 

```
select count(*) username, terminal, to_char(timestamp,'dd/mm/yyyy')
from dba_audit_session
where returncode <>0
group by username, terminal, to_char(timestamp,'dd/mm/yyyy');
```
  - ▶ Все попытки входа (количество):
- ▶ 

```
select count(*) username, terminal, to_char(timestamp,'dd/mm/yyyy')
from dba_audit_session
group by username, terminal, to_char(timestamp,'dd/mm/yyyy');
```
- ▶ Попытки входа несуществующих пользователей:
- ▶ 

```
select username, terminal, to_char(timestamp,'dd/mm/yyyy')
from dba_audit_session
where returncode <>0 and
not exist (select 1 from dba_users
where dba_users.username = dba_audit_session.username);
```



# Контроль журнала аудита

---

- ▶ Попытки входа в нерабочее время:
  - ▶

```
select username, terminal, returncode,
       to_char(timestamp, 'DD-MON-YYYY HH24:MI:SS'),
       to_char(logoff_time, 'DD-MON-YYYY HH24:MI:SS')
  from dba_audit_session
 where timestamp between to_date('08:00:00', 'HH24:MI:SS') and
       to_date('19:00:00', 'HH24:MI:SS');
```
  - ▶ Попытки входа с разных компьютеров:
- ▶

```
select count(distinct (terminal)), username from dba_audit_session
 having count(distinct (terminal)) > 1
 group by username;
```
- ▶ Попытки входа разных пользователей с одного терминала:
- ▶

```
select count(distinct (username)), terminal from dba_audit_session
 having count(distinct (username)) > 1
 group by terminal;
```



# Контроль журнала аудита

---

- ▶ Контроль изменения объектов (работа программистов!):

```
▶ select username,
      priv_used,
      obj_name,
      to_char(timestamp, 'DD-MON-YYYY HH24:MI:SS'),
      returncode
  from dba_audit_trail
 where priv_used is not null and
       priv_used <> 'CREATE SESSION';
```

# Аудит – ИТОГИ

---

- ▶ Аудит – часть общего плана обеспечения безопасности БД
- ▶ Не предоставлять привилегий в производственной БД обычным пользователям, только в рамках согласованных ролей, обычно execute procedure
- ▶ Удалить, заблокировать, изменить пароли всех учеток по умолчанию (типичные атаки)
- ▶ Расчистить роль public – убрать большинство привилегий
- ▶ План по обслуживанию журнала аудита – перенос данных, очистка, контроль ежедневно
- ▶ Детальный аудит – fga и триггеры



# Приоритеты и роли

---

- ▶ CASCADE – каскадный отзыв
- ▶ REVOKE REFERENCES ON dept
- ▶ FROM skott
- ▶ CASCADE CONSTRAINTS;



# Вопросы?

---

