Daten(un)sicherheit

Am Beispiel der Oracle Enterprise Editionen V9.x – V11gR2

Whoami

Meine Person :

- Karsten Aalderks
- Profi Einstieg mit Oracle V4/V5.1a/b/c im Jahr 1986
- Website : http://db-consult-aalderks.de/4481.html

Seit 1991 unterwegs als :

- Senior Software Architect , Database Professional

- Spezialisierungen:
  - Umsetzung von Datenschutz über Automatisierungen
  - Design/Re-Design von Datenbank Anwendungen
  - Ablaufverbesserungen über datengestützte Workflows z.B. Im Engineering f. Power Plants

Agenda

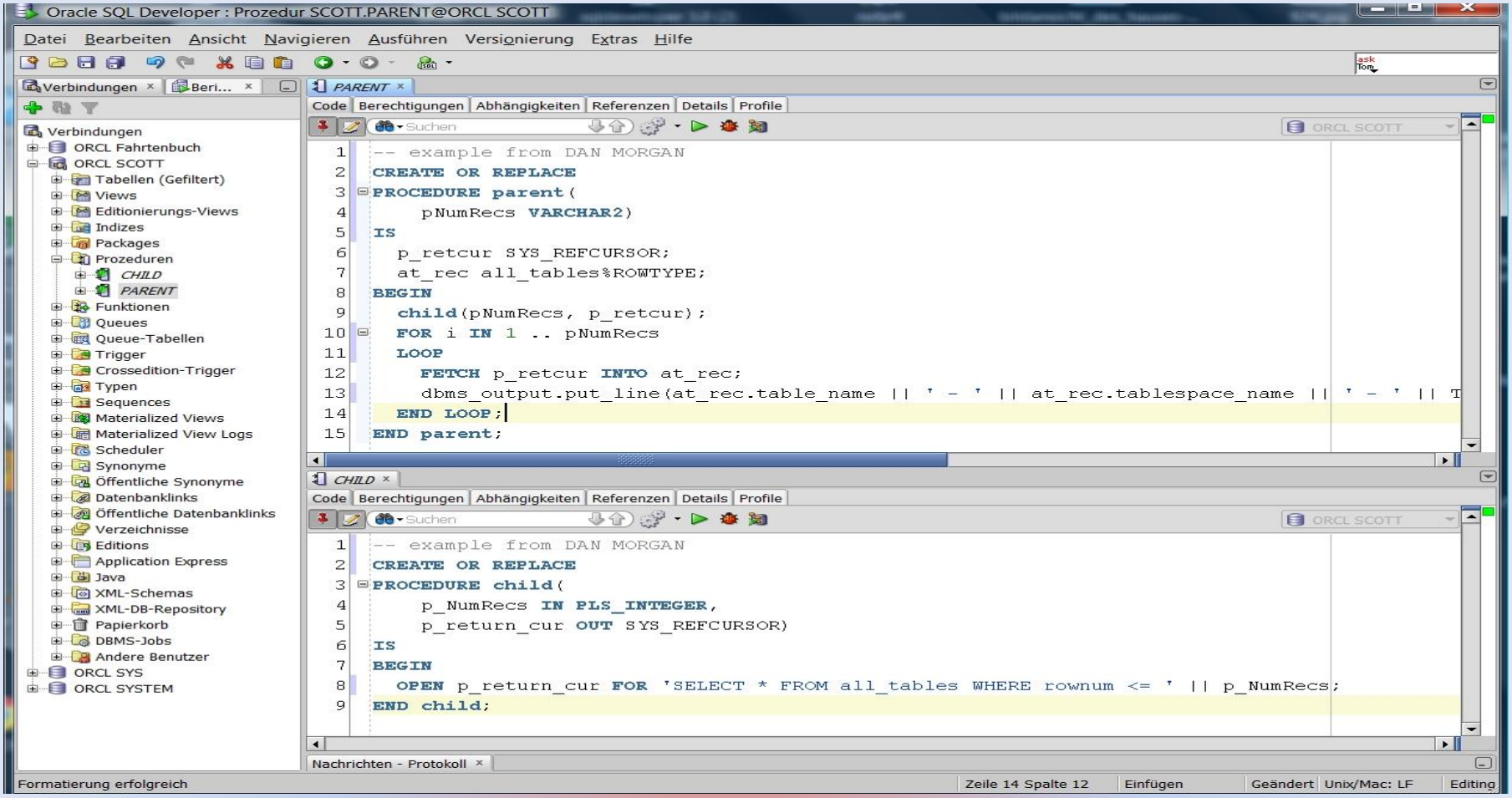
- Oracle Historie und Kernmerkmale
- Codd‘s klassische 12 Regeln
- Grundbegriffe – Oracle Jargon
- Kommunikation mit der Datenbank
- Oracle Optionen / Features
- Angriffe und Typisierungen
- Gegenmaßnahmen
- Die Hauptaufgabe beim Datenschutz
- Anregungen
- Links und Buchempfehlungen

Oracle‘s
Historie und Hauptmerkmale

- Larry Ellison und die Arbeit von Codd 1979
- Trusted Oracle – formal sicher f. CIA /NSA
- Client/Server Architecture
- SQL *Net TCP/IP – IPX/SPX - DECnet
- Transaktionssicher seit Version 4 –ACID konform
- Data Dictionary mit dyn. Perf. Views
- Stored Procedures/Packages
  - Ref Cursor Strongly /Weakly Typed
- API‘s OCI /Pro C/ Pro Fortran/…
- Lesekonsistenzmodell - Default optimistic

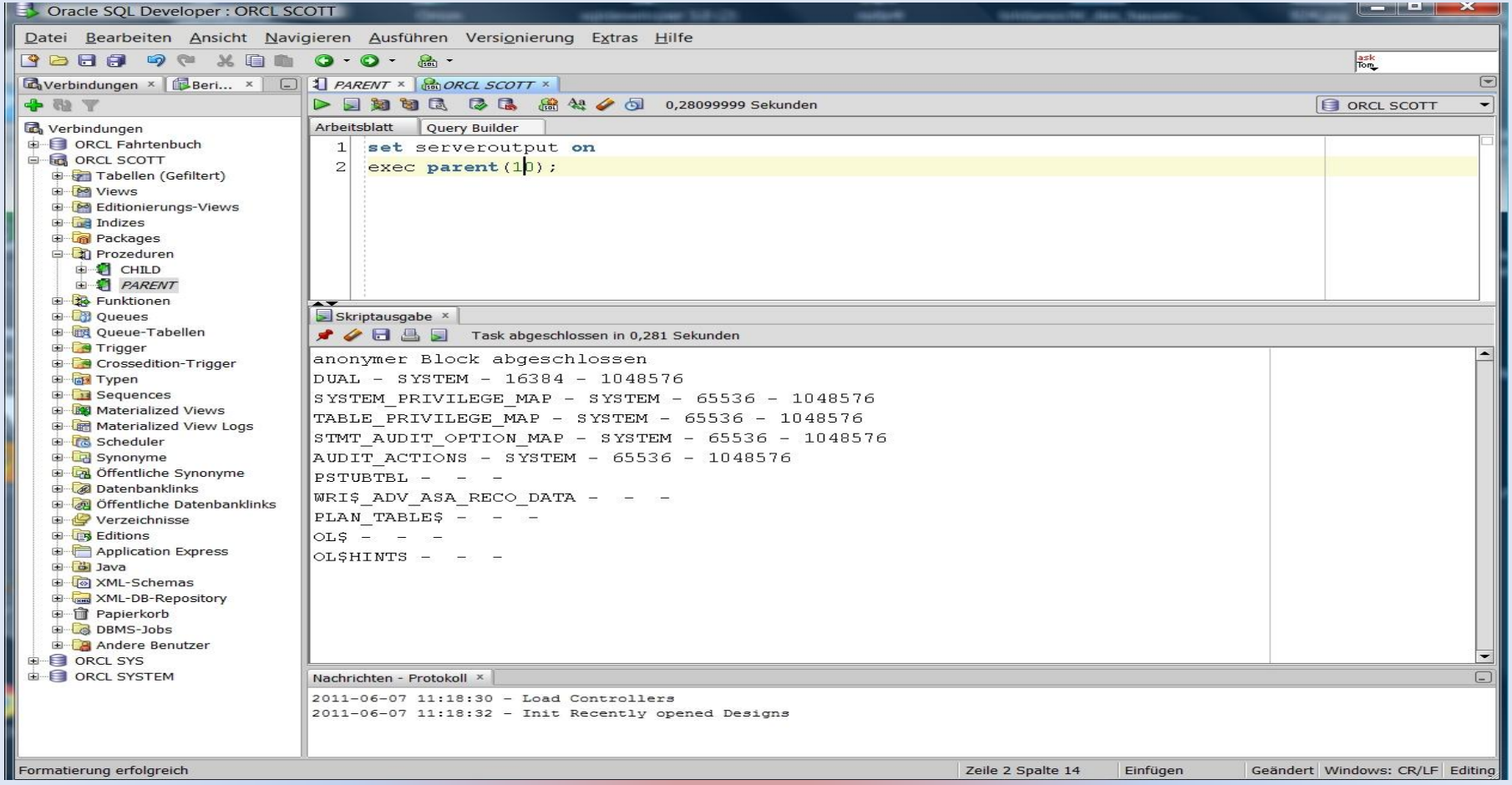
Codd‘s klassische 12 Regeln

1. **Information Rule** (Darstellung von Informationen): Alle Informationen werden durch Werte in Tabellen dargestellt.
2. **Guaranteed Access Rule** (Garantierter Zugriff): Jeder Wert muss über Tabellenname, Spaltenname und Wert des Primärschlüssels zugreifbar sein.
3. **Systematic Treatment of Null Values** (Systematische Behandlung von Null-Werten): Fehlende Werte werden unabhängig vom Datentyp mit NULL dargestellt.
4. **Dynamic On-line Catalog Based on the Relational Model** (Online-Datenkatalog): Die Datenbank und ihre Inhalte werden in einem Online-Datenkatalog (Data Dictionary) in Form von Tabellen beschrieben.
5. **Comprehensive Data Sublanguage Rule** (Metasprache): Das relationale System unterstützt Datendefinition, Datenmanipulation und Datenabfrage mit einer einheitlichen Sprache.
6. **View Updating Rule** (Aktualisierung von Sichten): Sofern theoretisch möglich, müssen Inhalte von Basistabellen auch über Sichten änderbar sein.
7. **High-level Insert, Update and Delete** (Parallele Datenmanipulation): Innerhalb einer Operation können beliebig viele Tupel bearbeitet werden.
8. **Physical Data Independence** (Physische Datenunabhängigkeit): Physische Ebene (Speicherung) und konzeptuelles Schema der Datenbank sind unabhängig voneinander.
9. **Logical Data Independence** (Logische Datenunabhängigkeit): Die Sicht einer Anwendung und das konzeptuelle Schema der Datenbank sind unabhängig voneinander.
10. **Integrity Independence** (Unabhängigkeit der Integrität): Unumgängliche Regeln (Integritätsbedingungen) müssen im Online-Datenkatalog gespeichert werden können.
11. **Distribution Independence** (Verteilungsunabhängigkeit): Logische und Physische Datenunabhängigkeit müssen ebenso für verteilte Datenbanken gelten.
12. **Nonsubversion Rule** (Kein Unterlaufen der Integrität): Integritätsbedingungen dürfen auf keinen Fall umgangen werden, auch nicht mit Hilfe von niederen Programmiersprachen.



Gefahr Weak Ref Cursor bzw. Dynamic SQL

Das fixe SQL `SELECT * FROM …` könnte mehrstufig dynamisch erzeugt werden.



Das Ergebniss – limitiert auf 10 Result Records

Mit dieser Technik lassen sich ganze Packages zur Laufzeit anlegen, weitgehend über Metadaten gesteuert.

Grundbegriffe – Abkürzungen

- Datenbankinstanz - orcl
- Userschema - scott
- Roles - connect, resource, …,dba
- Grantee - ka
- Owner - sales

- ASO - Advanced Security Option
- SAR - Secure Application Roles
- VPD - Virtual Private Database
- RLS - Row Level Security
- FGA - Fine Grained Access

Kommunikation mit der Datenbank

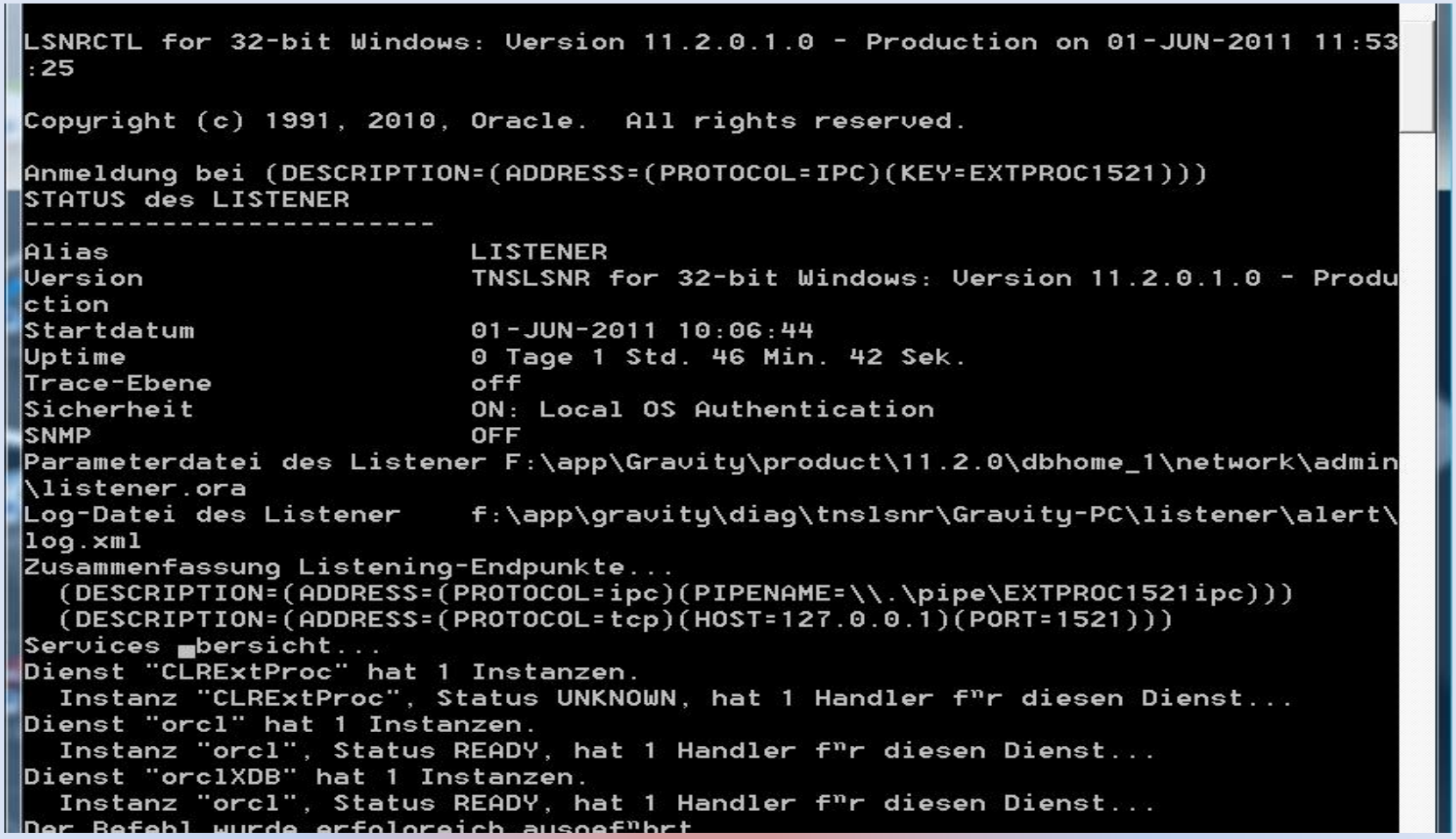
- Listener Default Port 1521

  c:\> lsnrctl status

  c:\> lsnrctl start, stop
- Dispatcher (dedicated, multithreaded MTS)
- User(Schema) / Password

  c:\> sqlplus system/manager@orcl

  c:\> exp scott/tiger@orcl



Oracle Listener

CLRExtProc ist die .Net Schnittstelle zu Oracle

orclXDB die XML Datenbank innerhalb von Oracle

Orcl ist die eigentlich Datenbankinstanz

Optionenen/Features

- Aurora Java - Single Threaded -non reentrant
- XDB XML Database
- Data Cartridge API
- APEX - Application Express
- Advanced Security Option
- Database Vault
- Audit Vault

- Nahezu jede Option und spez. Features besitzen eigene accounts/userschemata, die teils weitgehende Rechte besitzen. z.B. XDB

Angriffe

- Default User and default passwords

  c:\> sqlplus sys/change_on_install !!!

- Oracle listet mehr als 600 default accounts
- Pete Finnigan vermutet mehr als 1400 default accounts mit default password
- Viele sind mittlerweile *locked und/*oder mit abgelaufenem Password eingerichtet
- Die Rechte diverser default accounts sind mit Fleißarbeit zu überarbeiten, notfalls nur bedingt zu nutzen.

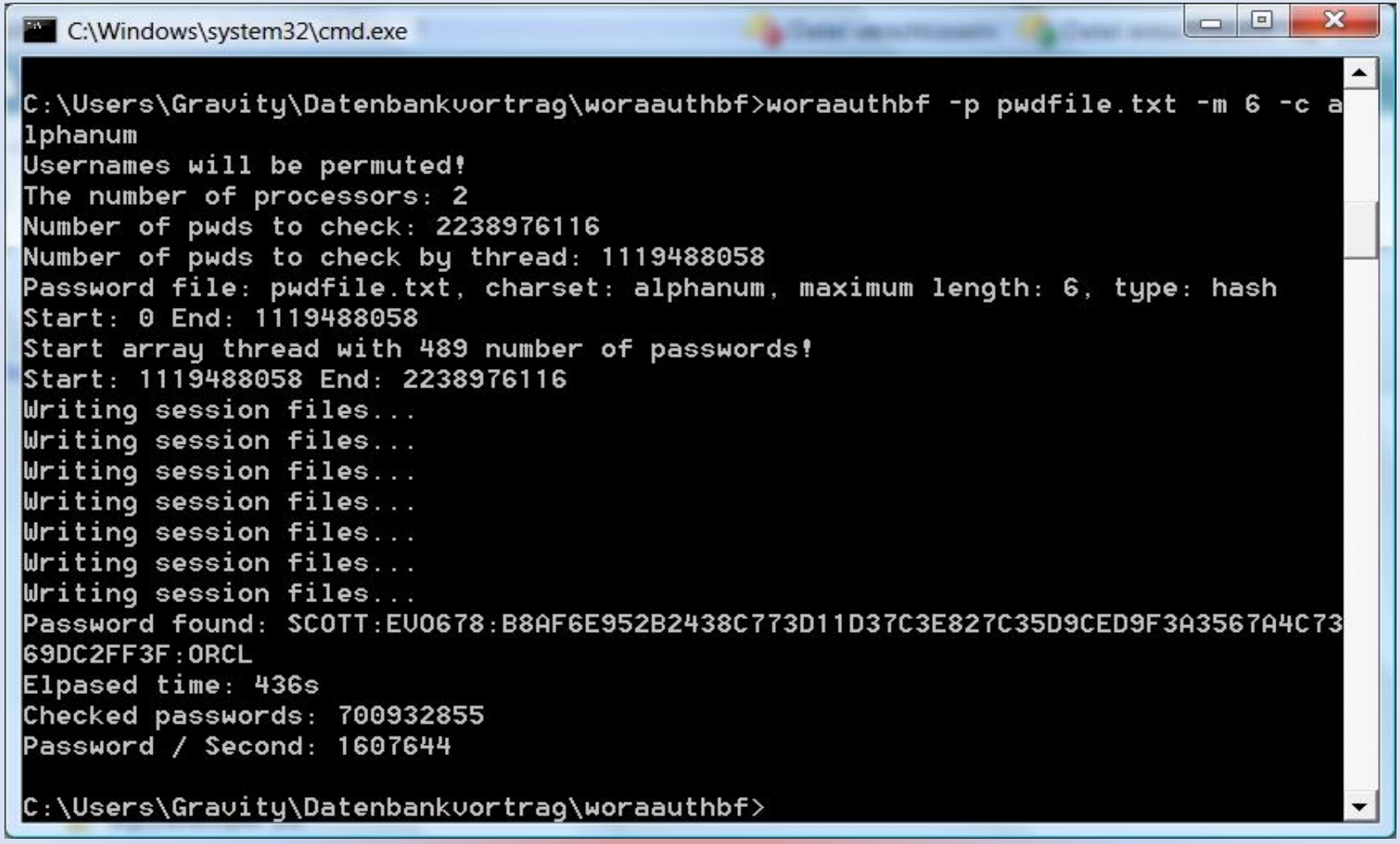
Angriffe

- Password Cracker z.B. woraauthbf
- Alphanumerisches Password bis 6 Stellen wird auf

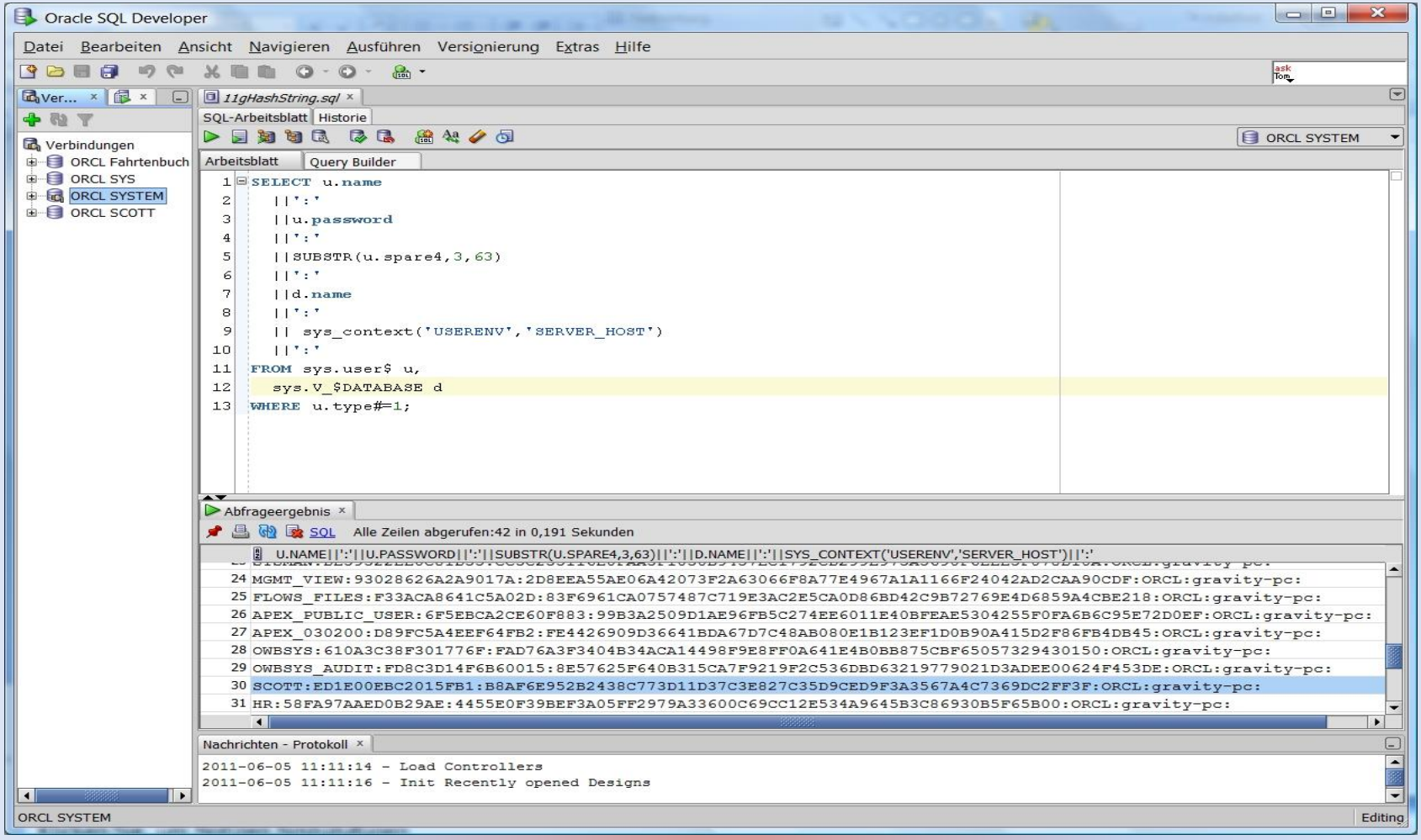
  2 Kern CPU in 30 Sekunden – 8 Minuten ermittelt.

Wie ?

- Datenbank mit gesammelten Passwörtern wird vorab abgeglichen
- Anschließend Brute Force Methode Multithreaded in C



woraauthbf on Oracle 11gR2 EE – 6 digits alphanum



11gR2 Hash Ermittlung über sys.user$

Über ´Mithören´ via z.B. DLL Injection kann der Hash ebenfals ermittelt werden.

Angriffe

Wie kommt man an den Passwort Hash bzw. das Passwort direkt ?

- Zu weitgehende Privilegien z.B. sys.user$
- Zugriff auf Password File - OS Ebene
- Zugriff auf System Tablespace z.B. über Java Stored Procedure
- Zugriff auf full Export File
- Zugriff auf Archive Logs
- …

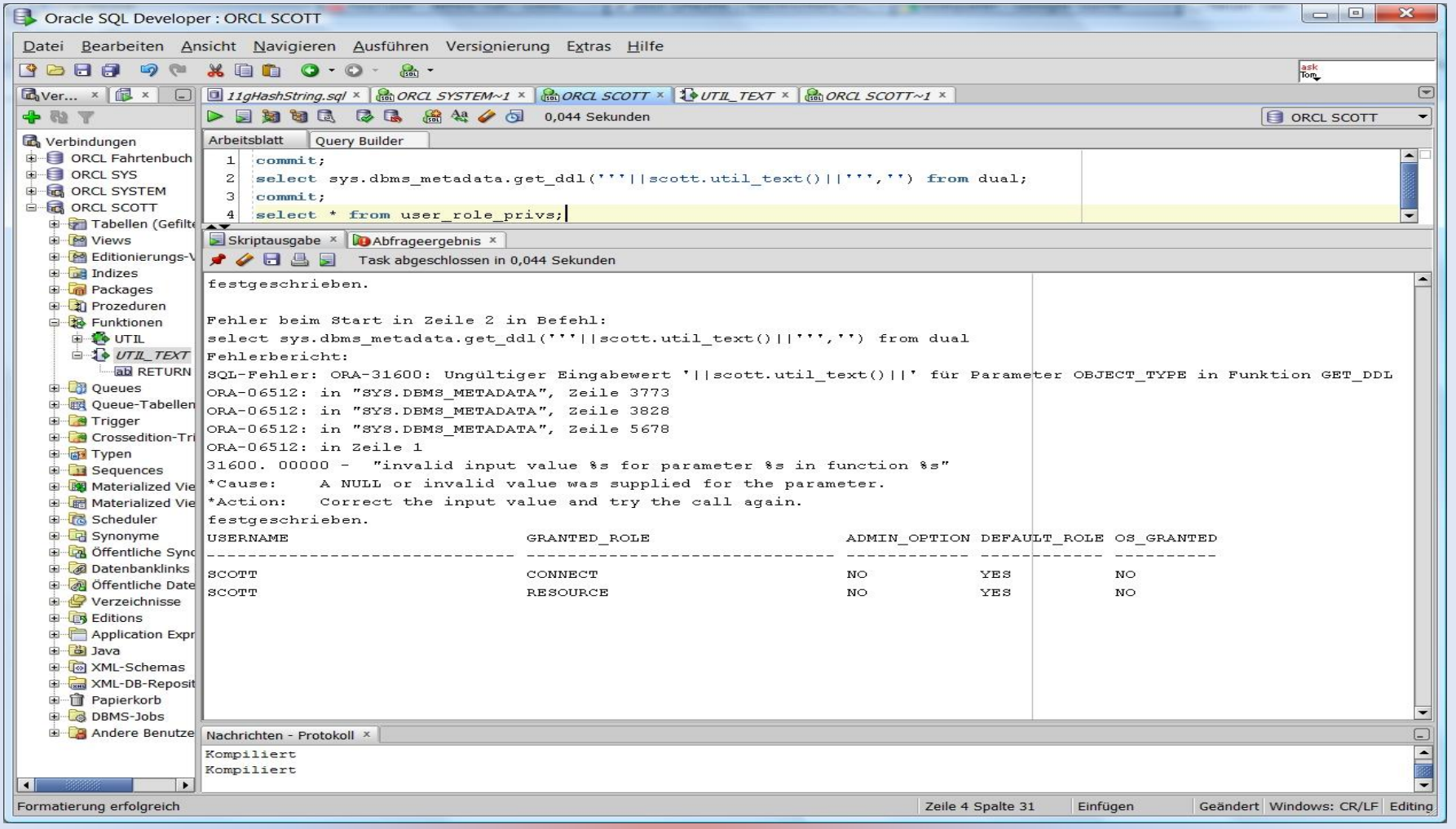
Angriffe

Was heißt zu weitgehende Privilegien ?

- Direkt auslesen aus DD - z.B. sys.user$
- Indirekt über SQL/PLSQL Injection
- Einstufig : z.B. Dynamic SQL Procedure mit Union oder Concatenate erweitern.
- Beispiel : Mit Dynamic SQL Procedure DBA Rechte erlangen und direkt auslesen

Angriffe über PL/SQL Funktionen

```
CREATE OR REPLACE
 FUNCTION scott.util_text
  RETURN VARCHAR2 authid current_user
 IS
  pragma autonomous_transaction;
 BEGIN
  EXECUTE immediate 'grant dba to scott';
  RETURN '';
 END;
```



Klappt nicht

Mit V9i wurden DBA Rechte noch erteilt !

Die Beispieldatenbank ist eine 11g R2 EE

Angriffstypisierung

- Von außen kommt die sekundäre Gefahr
  - Demonstration für alle
  - Hacker machen aufmerksam
  - Gangster klauen und verkaufen Daten
- Von innen kommt die Hauptgefahr
  - Alle Mitarbeiter sind potentielle Datendiebe
    - Unzufriedene Mitarbeiter
    - Korrupte Mitarbeiter
    - DBA‘s, Manager, Operator, einfache Anwender, …

Angriffstypisierung

- Nicht nur Datenklau ist ein Problem

  sondern nicht Geschäftsprozeß-konforme Datenmanipulation
- Hierbei ist zu unterscheiden zwischen legitimierter und nicht legitimierter Datenmanipulation
- Beispiel legitimiert: Die Jahresabschlußzahlen werden auf Anweisung eines Managers geschönt
- Beispiel nicht legitimiert : Die Verkaufszahlen eines Produkts werden von einem bezahlten Mitarbeiter der Konkurrenz je nach Marktsituation manipuliert.
- Gründe : z.B. Verhinderung bzw. Forcierung von Marketing Aktionen

Gegenmaßnahmen

- Neuste Oracle DB Version
- Wenn möglich Advanced Security Option installieren
- Aktuelle Patches bes. Security Patches
- Datensicherheitsklassifizierung aller Daten
  - Was ist wie zu schützen
- Feingranulare Datenzugriffsregelung für alle !
- Verschlüsselung
  - z.B. SSL Zugang statt klassischem Passwort
  - z.B. Datenverschlüsselung TDE
  - z.B. Backup Verschlüsselung
- Secure Application Roles !
- Prozess/Thread Kontrolle gegen Datenbankfiles und Shared Memory

Gegenmaßnahmen

- Auditing
  - Mandatory , Admin Auditing
  - Fine Grained Acces Auditing
  - Before/ After and special Select Auditing
- DML History with timestamp, user, machine, application, reasons !
- High Security Read Level

  - Datentausch auf Row Level Ebene über Algorithmus

  - erweiterbar auf Column Level

  - Hierbei wird die physikalische Rowid genutzt

Die Hauptaufgabe

- Wer hat wann das Recht auf welche Daten ?
- Ermittlung scheitert oft an dem Gesamtaufwand.
- Ein Lösungsansatz : Zentralisierte feingranulare Zugriffsregelung mit individualisierten Anwendungsrollen
  - Secure Application Roles
  - Virtual Private Database FGA/VPD/RLS
- Mit Policy Functions wird ein Select Befehl mit individueller WHERE Bedingung zur Laufzeit in Anhängigkeit des Users/Role erzeugt.

Wie setzt man die Hauptaufgabe um ?

Meine Antwort :

Virtual Private Database Generator

- On the fly or pre-generated
- Unternehmenshierarchien werden eingepflegt
- Rechteprofile erstellt
- Generierung der Policies/Functions samt Auditing

Anregungen

- Proprietäteres Trusted OS sowie Trusted Oracle
- speziell Filesysteme und Library Calls die
- Zugriff auf Shared Memory und OS Files über
  - Authentifizierte Prozesse/Threads mit z.B. dyn. Tageskennung
  - Alle anderen osuser oracle Prozesse /Threads haben keinen Zugriff
- API‘s für individuelle Authentifizierungsprozesse
- Dynamische Protokolländerungen SSL , Kerberos, Radius, …
- Dynamisch konfigurierbare High Speed Protokolle f. Cluster, Grid, Cloud, …
- Allgemein : Keine statischen Schlupflöcher zulassen

Links und Buchempfehlungen

- David Litchfield http://www.davidlitchfield.com/
- http://www.v3rity.com/
- Pete Finnigan http://www.petefinnigan.com/
- Alexander Kornbrust
- http://www.red-database-security.com/
- Lazlo Toth
- http://soonerorlater.hu/index.khtml?article_id=1
- Ron Ben Nathan - **HOWTO Secure and Audit Oracle 10g/11g**
- http://ronbennatan.sys-con.com/
- David Knox , Tyler Muth , Peter Wahl - **Applied Oracle Security**
- http://www.amazon.com/Applied-Oracle-Security-Developing-Environments/dp/0071613706

Ende

oder auch nicht …

- QA – Session direkt im Anschluß
- Optionale Demo von Lazlo Toth zur Transparent Data Encryption Aushebelung nach dem Vortrag von

  Dr. Oriana Weber