Auditing The Oracle Database

PFCATK – A Toolkit to Help

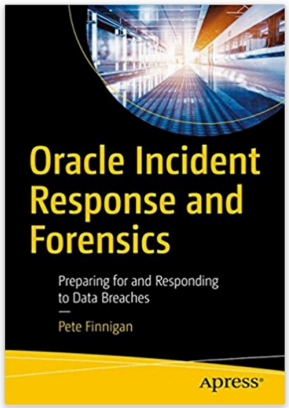
Legal Notice

Auditing The Oracle Database

Published by
PeteFinnigan.com Limited
9 Beech Grove
Acomb
York
England, YO26 5LD

PFCL
PETEFINNIGAN.COM LIMITED

Sorry I cannot be there in person but we are working on something later in the year



Pete Finnigan – Background, Who Am I?

- Oracle Security specialist and researcher
- CEO and founder of PeteFinnigan.com Limited in February 2003
- Writer of the longest running Oracle security blog
- Author of the Oracle Security step-by-step guide and “Oracle Expert Practices”, “Oracle Incident Response and Forensics” books
- **Oracle ACE for security**
- **Member of the OakTable**
- Speaker at various conferences
  - UKOUG, PSOUG, BlackHat, more..
- Published many times, see
  - http://www.petefinnigan.com for links
- Influenced industry standards
  - And governments

PFCL
PETEFINNIGAN.COM LIMITED

Agenda

- Do people use Oracle audit trails?
- A bit of history
- The Focus - **The PFCLATK toolkit**
- An overview
- Deployment
- Hacking
- Audit results

PFCL
PETEFINNIGAN.COM LIMITED

State of the (Audit Trail) Nation

- Extensive experience visiting customer sites
  - Performing security audits
  - Reacting to incidents, breaches or attacks
- I see one common theme
  - No audit trails **OR**
  - Very limited audit trails **OR**
  - Of those that do have audit trails very few use them interactively
- Sometimes people collect audit because they have to (**regulations**)
- I have even seen some sites collect audit and delete it (**regulations**)

PFCL
PETEFINNIGAN.COM LIMITED

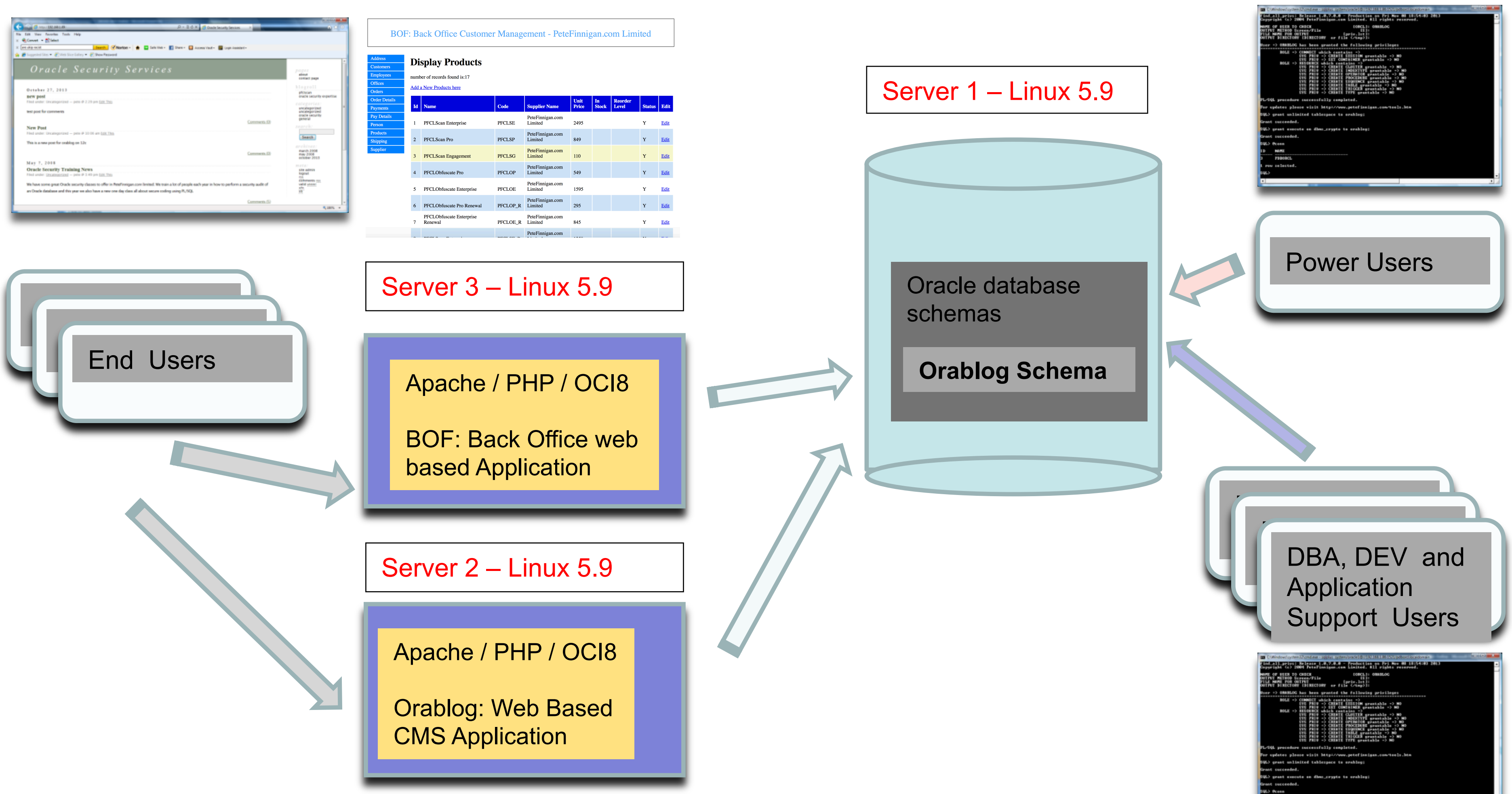
History of the Toolkit and Talks

- In 2009 piece of work to help design audit trails
  - Site had limited staff, little time to design, deploy, maintain any audit trails
  - I came up with some simple ideas, proof of concepts – to package up audit trails for them; inc policy based audit, IPS and simple firewall
  - They spent limited time to deploy a useful audit trail
- Similar piece of work in 2011 where limited team needed to deploy audit
- 2012 to 2015 extended the toolkit
- I wrote a presentation back in 2012 and presented it just once at a SIG on practical audit trails where I mentioned this toolkit for the first time
- This then became the basis of a one day class on the same subject
- Reworked that presentation in UKOUG 2015 conference
- Customer in 2016 needed an audit trail to deploy quickly
- Deployed now to customers in UK, Ireland and Germany

PFCL
PETEFINNIGAN.COM LIMITED

- Oracle Linux
- Oracle SE1 Database
- Applications (Front Facing Website, back office customer processing)

My Sample Application Architecture



Demo Hacking

Demo:

- Enable secconf.sql to get standard audit
- Run audit.sql to see audit configuration
- Test some SQL Injection as an attacker
- SQL Injection attack as unauthenticated web user
- SQL Injection attack as database user with just CREATE SESSION
- Access data as a DBA with %ANY% rights
- View audit trail generated

The Goal of the Toolkit

- As simple as `SQL> @atk` and a sophisticated audit trail is up and running
- Making it simple for organisations to deploy audit trails simply, with no resources
- No design, implement, test etc as we have done it for you already
- Used in ATC mode – space also is managed in each target database audited
- Simple to configure or not configured at all
- A complete solution to know what is happening at the database engine level for sites with limited resources

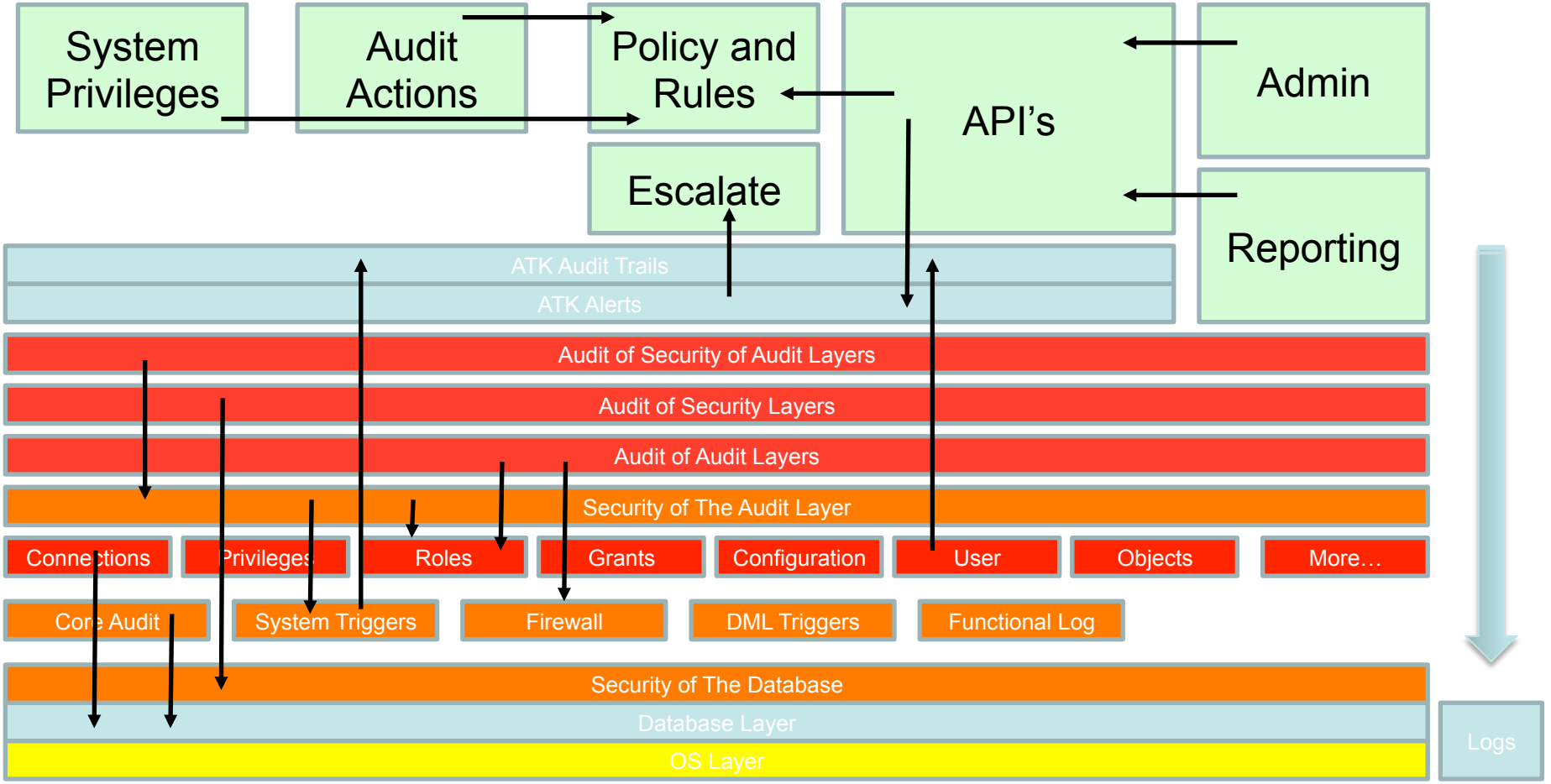
PFCLATK – “A”udit “T”rail tool”K”it

- Toolkit to aid audit trail deployment easily
- Simple pre-configure
- Policy based
- Alert based
- Multiple audit trails sources
- Add in factors (input hints)
- Separated schema design
- Manual 25 pages currently
- Version 1.7.2.0 currently
- Layered audit

PFCL
PETEFINNIGAN.COM LIMITED

- Free PL/SQL and SQL based toolkit – 14k lines of code.
- Audit the database engine itself

PFCLATK Block Overview

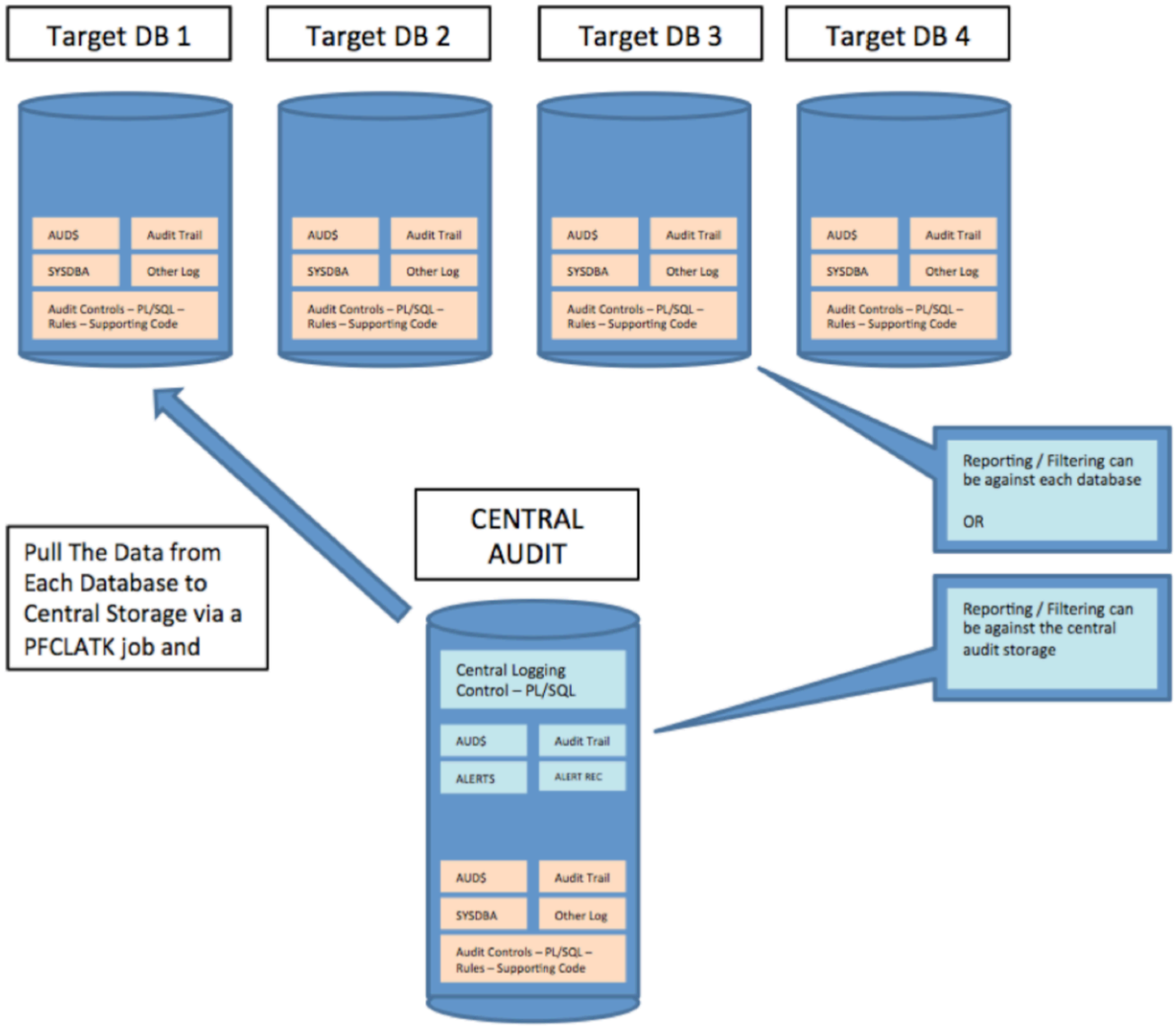


PFCLATC – "A"udit "T"oolkit "C"entralised

- PFCLATK can be deployed to each "target Oracle Database"
- PFCLATC is an additional layer to add centralisation of those audits automatically
  - Simple configuration to link each target with the central storage
  - Uses links and a PUL mechanism
  - Audit trails are check summed
  - Audit trails are PUL'd
  - Audit trails purged from the target
- Manage target space needed for audit trails (limited by PUL)
- The toolkit also audits the PFCLATC target (if required)
- Central reporting possible over many databases

PFCLATK Architecture

- The PFCLATK toolkit is designed to be deployed to a target or central database
- When enabled simply adding target link details to the ATC database starts the PUL process automatically



PFCL
PETEFINNIGAN.COM LIMITED

Database Engine Audit

- Sites often have application level audit trails
  - In the application layer itself
  - Sometime also in the database (RLA in Oracle E-Business Suite for instance)
- Sites sometimes have audit enabled at the operating system level
- Auditing of the database is often
  - Application related
  - Regulation related
- **Audit is needed at the database engine layer to capture abuse against the database itself**

Alerts

- React in real time to attacks
  - SQL Injection
  - Privilege abuse
  - Error conditions
- React in
  - Real time where possible
  - Semi-real time if not possible
- Other reactions can be slower or not at all
- Alerts are configured in policies along with raw audit collected – post filtering is more powerful than unified audit pre-filtering because we can filter across more domains

PFCL
PETEFINNIGAN.COM LIMITED

Separated Schemas and Roles

- Schemas
  - ATKD – The owner of tables, views, sequences
  - ATK – The owner of the main API. Also runs jobs that payloads and filters based on
- Roles
  - ATK_ADMIN - Any user granted this role can set up PFCLATK rules, policies, jobs, filters, credentials and factors
  - ATK_REPORT – Any user granted this role can view the alerts and alert details and audit trail details

PFCL
PETEFINNIGAN.COM LIMITED

Configuration

```
169
170 -- ------------------------------------------------------------------------
171 -- There variables can control the operation of this install script
172 -- ------------------------------------------------------------------------
173
174 define DEBUG = "OFF"          -- turn debug ON or OFF, results in the O/P file
175 define TBLSPC = "USERS"       -- define the tablespace for ATK, must be created first
176 define ATC = "ON"             -- Turn on or off to install the ATC objects and code
177 define DROPATK = "OFF"        -- if ON call drop anyway, if OFF test if ATK is installed
178                               -- before dropping ATK
179
180 -- ------------------------------------------------------------------------
181 -- End of customer changeable values
182 -- ------------------------------------------------------------------------
183
```

The user configurable settings are simple and at the top of atk.sql

Factors

Some factors are re-defined, some should be edited and more can be added easily

Factors allow the toolkit to be customised for a specific site

```
100   -- ----------------------------------------------------------------
101   atk.pfclatk.addfactor('SUPPORT-IP','192.168.56.1');
102   -- ----------------------------------------------------------------
103   -- LOW-FAILED: Define the number of failed logins per minute above
104   --             which an alert should be raised. This should be specified
105   --             per 30 minutes. So a number of 3 per minute would be set
106   --             to 90 for the 30 minute period.
107   -- ----------------------------------------------------------------
108   atk.pfclatk.addfactor('LOW-FAILED','60');
109   -- ----------------------------------------------------------------
110   -- HIGH-FAILED: Define the number of failed logins above which an alert
111   --             would be raised. This should be specified per 30 minutes
112   --             So for a number of 50 per minute that would be 30*50 =
      --             1500. This could indicate a scripted attack.
      -- ----------------------------------------------------------------
      atk.pfclatk.addfactor('HIGH-FAILED','1500');
      -- ----------------------------------------------------------------
      -- DEV-IP: Define the IP adress of all of your developers terminals
      --         here.
      --
      -- NOTE: For multiple developers add multiple entries here.
      -- ----------------------------------------------------------------
      atk.pfclatk.addfactor('DEV-IP','192.168.56.1');
      -- ----------------------------------------------------------------
      -- DBA-USER: Define the DBA user accounts allowed to be used as DBA.
      --           This could be SYS and SYSTEM but should really be seperate
      --           user accounts for each DBA
      --
      -- NOTE: For multiple DBA add multiple entries here.
      -- ----------------------------------------------------------------
      atk.pfclatk.addfactor('DBA-USER','SYS');
      atk.pfclatk.addfactor('DBA-USER','SYSTEM');
132   -- ----------------------------------------------------------------
133   -- ERROR-RATE: This is the trigger rate for the number of errors that
134   --             can occur for a single user/IP that could indicate an
135   --             attack.
136   -- ----------------------------------------------------------------
137   atk.pfclatk.addfactor('ERROR-LIMIT','8');
138   --
139 end;
140 /
```

PFCL
PETEFINNIGAN.COM LIMITED

Audit Policies

```
270  begin
271    -- create the policy
272    atk.pfclatk.createpolicy('PROFILEPRIVILEGE');
273
274    -- audit profiles
275    atk.pfclatk.createandaddrule('PROFILEPRIVILEGE','Create Profile','CORE-S','create profile');
276    atk.pfclatk.createandaddrule('PROFILEPRIVILEGE','Drop Profile','CORE-S','drop profile');
277    atk.pfclatk.createandaddrule('PROFILEPRIVILEGE','Alter Profile','CORE-S','alter profile');
278    atk.pfclatk.createandaddrule('PROFILEPRIVILEGE','Profile','CORE-S','profile');
279
280    -- ----------------------------------------------------------------
281    -- Add filter job to detect non-ligitimate profile changes - i.e. use
282    -- of PROFILE system privileges; not use of statement PROFILE and not use
283    -- of a DBA IP address and not use of a DBA account; so if a DBA
284    -- uses a non DBA account from his own IP it should be detected
285    -- ----------------------------------------------------------------
286
287    atk.pfclatk.addfilter('NON-AUTH-PROFILE-CHANGE','PROFILEPRIVILEGE','HALFHOUR',
288        '[Alert] Non-legitimate profile privilege change',
289        'select ''A non-authorised user change {''||a.action_name||''} on {''||a.obj_name||''} by
290
291    -- enable the policy
292    atk.pfclatk.enablepolicy('PROFILEPRIVILEGE');
293  end;
294  /
```

- Policies declare collection of raw data and also events
- PFCLATK policies are different to Unified audit – we filter on collected data after storage to look for abuse; Unified audit filters before storage
- Core audit, DML, System triggers

- Alerts are jobs that run to parse the collected audit trails
- Each policy can include raw collect and also alert jobs (filters)

Alert Jobs

```
74  -- ------------------------------------------------------------------
75  -- In this setup we create a number of database jobs that can be used
76  -- to attach payloads to. Each time a job runs it queries the payloads
77  -- table and runs all payloads that are attached to the respective job
78  -- frequency that is running.
79  --
80  -- We will create seven jobs (BUT more can be created as necessary)
81  --
82  -- MINUTES  : 2 minutes
83  -- HALFHOUR : 30 minutes
84  -- ONEHOUR  : 1 Hour (used for PUL)
85  -- TWOHOUR  : 2 Hours (Used for PUL)
86  -- HALFDAY  : 12 Hours
87  -- WEEK     : 1 Week
88  -- MONTH    : 1 Month
89  -- YEAR     : 1 year
90  --
91  -- IMMEDIATE: immediate
92  --
93  -- There is also a job type of IMMEDIATE but this is not run as a job
94  -- in the database as a trigger or code elsewhere can execute these
95  -- job types and their payloads.
96  -- ------------------------------------------------------------------
97
98  -- create the IMMEDIATE job. This is slightly different as it does
99  -- not create a DBMS_JOB job.
100
101 @@check.sql "IMMEDIATE: Create the immediate job"
102
103 declare
104   lv_job varchar2(100);
105 begin
106   -- create the job
107   lv_job:=atk.pfclatk.createjob('IMMEDIATE',NULL,100001);
108   -- start the job
109   atk.pfclatk.enablejob(lv_job);
110   --
111 end;
112 /
```

Audit of Audit

- A multi-layer approach is needed
  - Audit of core trail tables such as AUD$
  - Audit of core audit settings such as AUDIT$
  - Audit of triggers (Event, DDL and DML)
  - Audit of custom logs
  - Audit of audit functionality, packages and other objects
  - All can be set up as policies in PFCLATK

Configure and Deploy

- Edit atk.sql
  - Edit required settings needed for the toolkit
- Edit conf.sql
  - Add connection details
- Demo deployment
  - Run atk.sql

Demo Hacking

Demo:

- All ATK policies are enabled
- Test some SQL Injection as an attacker
- SQL Injection attack as unauthenticated web user
- SQL Injection attack as database user with just CREATE SESSION
- Access data as a DBA with %ANY% rights

Demo: look at the audit trails

Reports

- A few sample reports exist that highlight issues
  - Audit_report.sql
  - Car.sql
- Alerts are viewed via the ATKD.PFCLATK_ALERTS table
- Alert details in ATKD.PFCLATK_ALERT_ROWS
  - tr.sql shows high level summary of alerts

Live Training in Ljubljana, Slovenia

- I am sorry that I could not be there in person today!
- If you would like to learn much more in details about audit trails in Oracle then I offer a 1 day class that we are planning to hold in November with Palsit
- The class details are here - http://www.petefinnigan.com/training/Practical_Audit_Trail_Class_Flyer.pdf
- If you are interested please speak to Palsit or myself

PFCL
PETEFINNIGAN.COM LIMITED

Conclusions

- Start to audit the database engine
- Understand what people are doing at the database engine level
- Take advantage of a simple to use idea to enable policies and factors
- Deploy with a simple command
- Close the gap between OS and application audit
- GDPR is coming and you need to detect attacks (successful or not)

Auditing The Oracle Database

PFCATK – A Toolkit to Help