Oracle 23c DB Nest in ~~*practical*~~ use

Challenge of using a long-awaited feature

October 2023

Stefan Oehrli

Stefan Oehrli – Data Platforms

Oracle ACE Pro

stefan.oehrli@accenture.com

Terraform ASSOCIATE – HashiCorp CERTIFIED

Tech Architecture Manager

- Since 1997 active in various IT areas
- More than 25 years of experience in Oracle databases
- Focus: Protecting data and operating databases securely
  - Security assessments and reviews
  - Database security concepts and their implementation
  - Oracle Backup & Recovery concepts and troubleshooting
  - Oracle Enterprise User and Advanced Security, DB Vault, ...
  - Oracle Directory Services
- Co-author of the book The Oracle DBA (Hanser, 2016/07)

oradba.ch

@stefanoehrli

>

DATA PLATFORMS

WHY? We are the game changer for our client's data platform projects

HOW? Maximum automation, maximum efficiency, maximum quality!

WHAT? We build innovative data platforms based on our blueprints, assets and tools.

3 key benefits

1 Architecture expertise from hands-on projects

2 Delivery of tailor-made data platforms

3 Integrated Teams - Like a Rowing team, perfect alignment and interaction.

Tools and Blueprints

Key enabler for the implementation of modern data platforms at a high speed and quality.

Continuous Optimization

Tools and Blueprints are continuously optimized to the customer and project’s needs.

Expertise

Expert group for modern data platforms from technical implementation to project management and organization

Oracle DBNest

A different approach to protect your DB, PDB and Environment

1 Introduction

2 Oracle DB Nest Architecture

3 Look behind the Scenes

4 Oracle DB Nest Configuration

5 A few Attempts

6 Challanges

7 Alternative Security Measures

8 Conclusion

1

Introduction

Why do you need a DBNest at all?

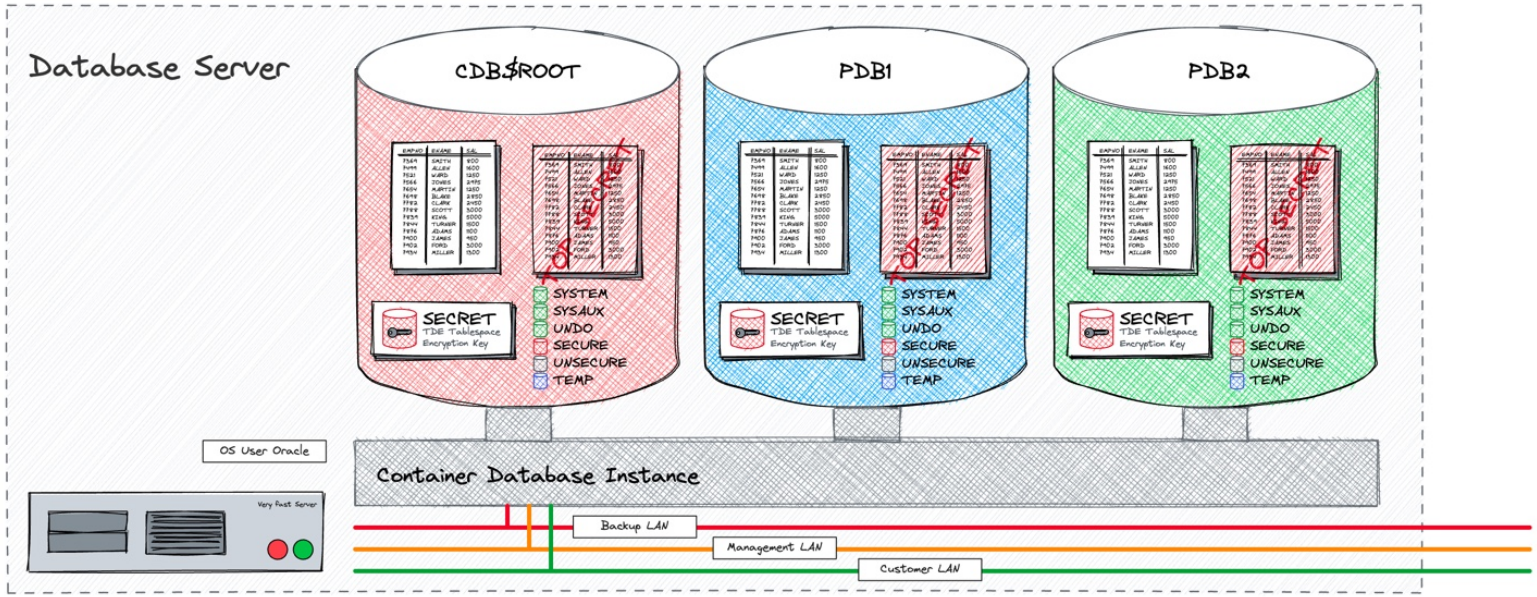
Container Database Usage

How is the Container Environment used?

- Simple replacement of the classic database (none-CDB) architecture with just one PDB
- Consolidation using multiple PDBs
- Private DBAAS
- Public DBAAS

Corporate structure e.g., all one legal entity or subsidiaries?

- Infrastructure architecture
- Cloud, Hybrid or on-premises?
- Dedicated hardware
- Virtual environments
- Engineered System

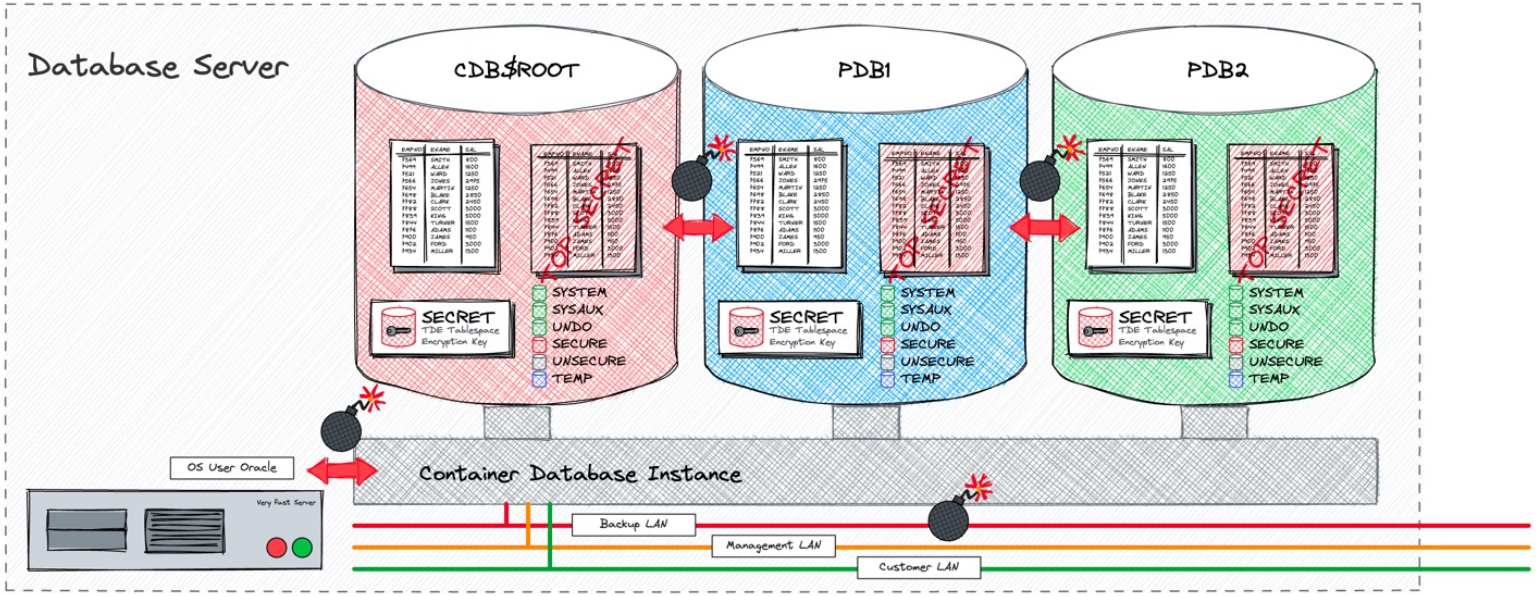


Insulation Requirements Vary

Risks in a DBaaS environment

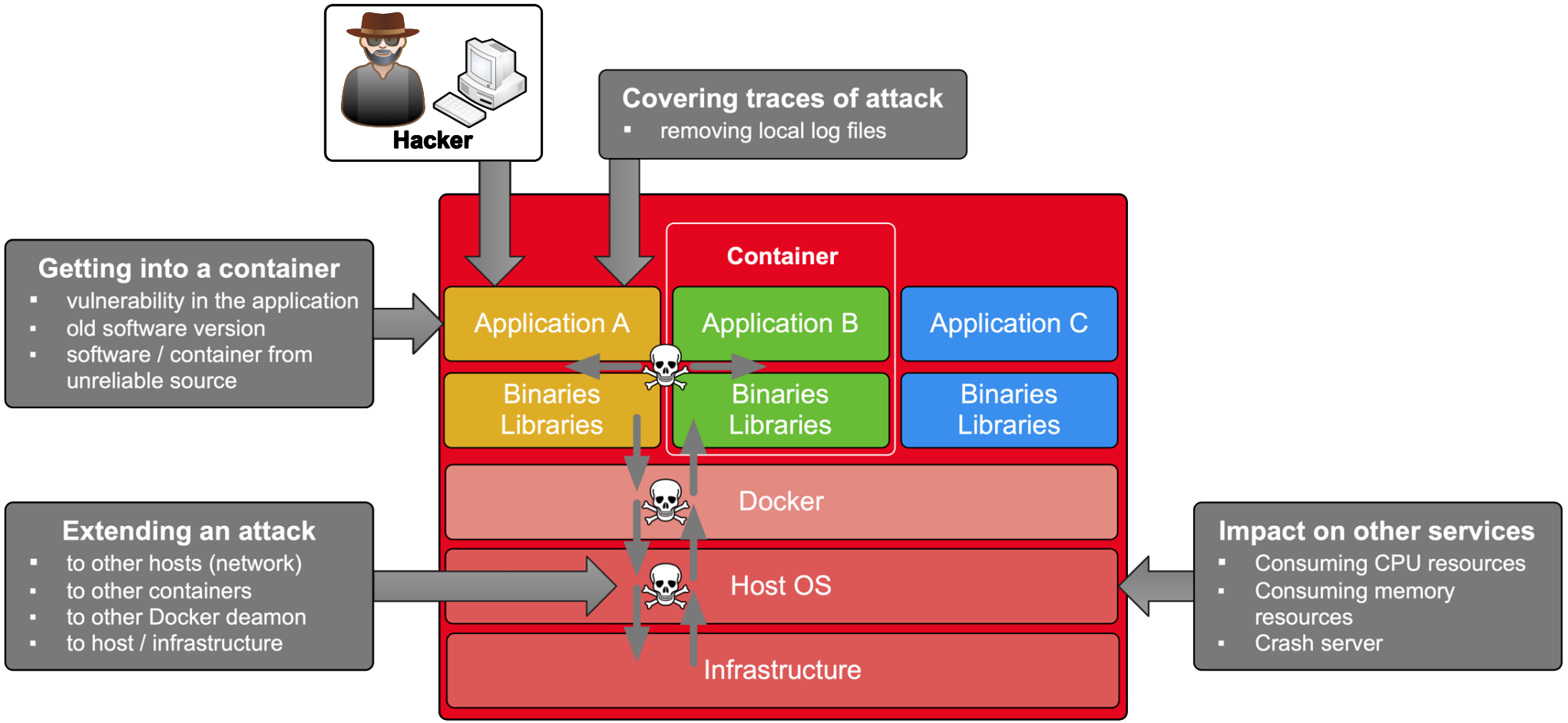
What can happen?

- PDB admin use **privilege escalation**
- Access sensitive data via shared resources e.g., backup or management LAN.
- Break out of PDB and get OS access as **oracle**.
- Gain access to the **root container** (cdb$root)
- Gain access to other PDBs.
- Gain access to the **network**.
- **Excessive** use of shared resources
- Use of **critical features** like
  - Oracle JVM
  - DBMS_SCHEDULER
  - External table pre-processor



Are we alone with this topic?

What about other container technologies?



Honestly, this is only for large deployments, right?

Sure, if the containers look like this

Honestly, this is only for large deployments, right?

… but what about this container…

Honestly, this is only for large deployments, right?

… or even worse?

But why isolation at all?

Protect your resources and environment

Not only for cloud deployments…

- … database consolidation on-premises
- … new default architecture as of 21c
- … mixed classification of data
- … mixed availability requirements

Noisy neighbors can only disturb in best case …

- … or sink the ship in worst case

A few reasons to consider

- Vulnerabilities and bugs
- Security / compliance requirements
- Shared resources (OS, Memory, CPU,…)

Still not Convinced?

Some features can do more than you think...

- Only basic database security is not enough...
- PDB admin and user do have comprehensive privileges (DBAAS).
  - Full DBA role
  - ALTER SYSTEM, ALTER SESSION,...
  - PL/SQL packages and procedures
- Oracle bugs and feature allow to escape the boundaries of a PDB.
  - Scheduler jobs including OS calls
  - External table pre-processor scripts
  - PL/SQL Library or Java OS calls
- Resource management beyond the scope of PDBs
- Compromised Oracle processes

2

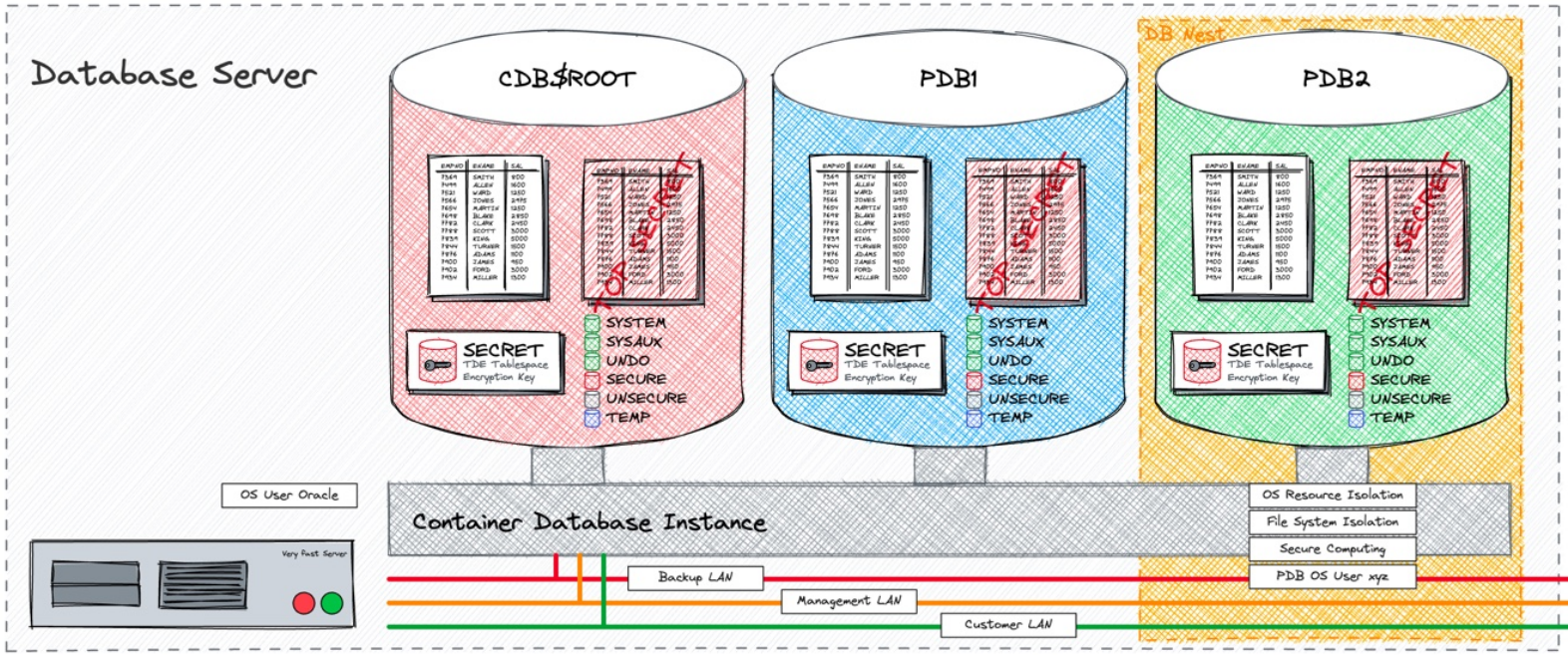
Oracle DB Nest Architecture

What does the whole thing look like?

Oracle DB Nest

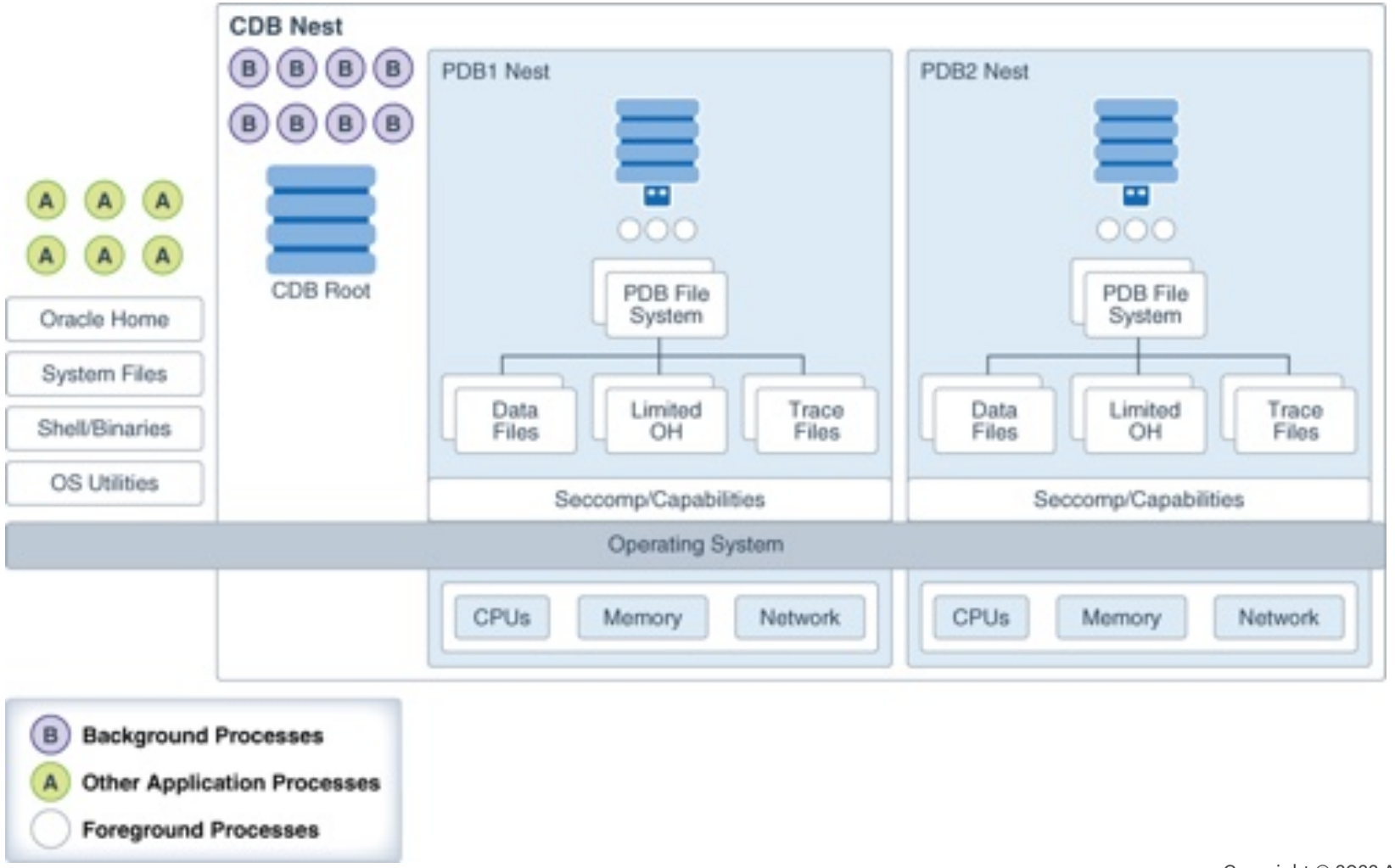
Basic Idea behind the DBNest

- Available in Oracle 21c
- Control and isolation of…
  - … OS resources used by a PDB
  - … File system isolation per PDB
  - … Secure computing
- Concept analogue to container technologies like Docker
  - Use of Linux Namespaces
  - Use of CGROUPS



Architecture of a CDB Nest

Oracle® Database Security Guide 23c - Securing and Isolating Resources Using DbNest



Goal of DB Nest

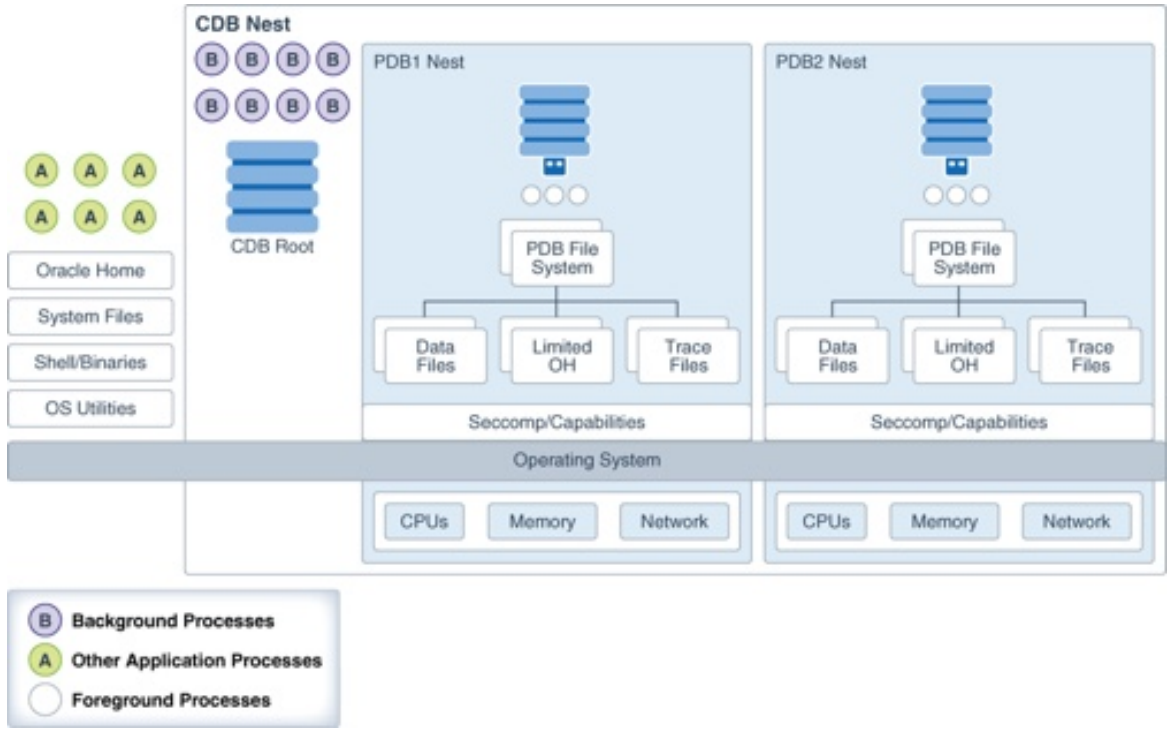
What is to be Achieved?

- DB Nest is the Oracle solution for database instance and PDB protection
- Enables a database instance to run in a protected, virtualized environment.
- DB Nest isolate database instance from…
  - … another database instance
  - … other applications
  - … as well as PDBs from each other and from the CDB

DB Nest Properties

What can be configured

- Operating system isolation
  - OS resources like process ID, user, and mount
- File system isolation
  - Visibility for file system entities
  - A **pivot root** in Linux namespaces is equivalent to chroot
  - A **bind mount** enables the contents of one directory to be accessible in a different directory
- Resource management
  - Control and monitor the resources of a nest
- Secure computing mode (seccomp)
  - seccomp to filter out system calls



3

Look behind the Scenes

What is behind Oracle DBNest?

Linux Kernel Namespaces

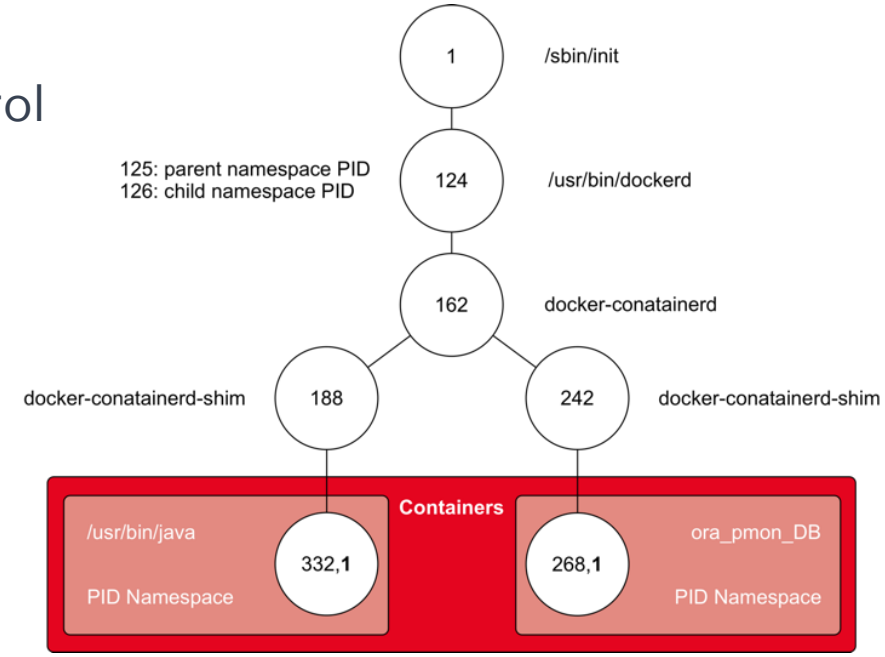
Kernel namespaces are a feature in modern Linux operating systems

Namespaces are the **foundation** of **containerization** technologies like **Docker** and **Kubernetes**

- **Isolate Resources** Namespaces enable resource isolation, ensuring that different instances of resources are isolated from one another
- **Segregate Processes** Kernel namespaces allow processes to run independently in isolated environments
- **Resource Control** Kernel namespaces provide fine-grained control over resources

When a DB Nest is launched, Oracle creates a set of namespaces

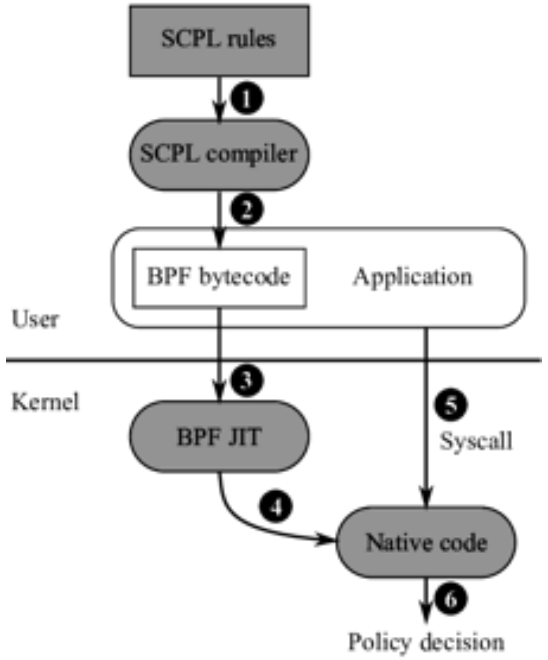
- Process namespace
- User ID namespace
- Mount namespace



Secure Computing Mode (sccomp)

Feature to enhances security in operating systems

- **seccomp** is a security feature that provides isolation and protection
- prevents untrusted code or potentially malicious applications from tampering with system resources
- Filter out system calls that are…
  - … unnecessary
  - … malicious
- Restrict the actions available within the container
- **seccomp** uses Berkeley Packet Filters (BPF)
- Well known / used in Container environments e.g., Docker, Kubernets



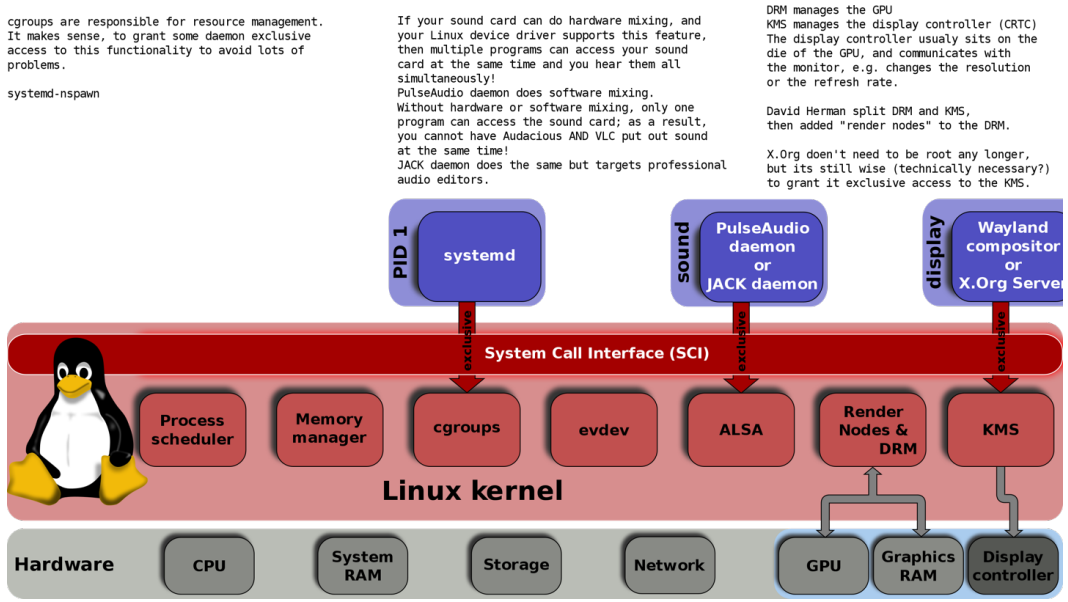
Control Groups (cgroups)

Linux feature for resource management and process control

- **cgroups** is a Linux kernel feature
  - mainlined into the Linux kernel since 2007

- **Enables** controlling and limiting **resource usage** (CPU, memory, etc.) of processes or groups of processes.
- Essential for **resource allocation** in containers, virtualization, and managing system performance.
- Ensures **fair resource distribution**, prevents resource contention, and enhances system stability.

cgroups are a fundamental tool for efficient resource management, critical in modern computing environments.



Wikipedia https://en.wikipedia.org/wiki/Cgroups

4

Oracle DB Nest Configuration

Let's try a simple configuration

Oracle DB Nest Configuration

New configuration parameters

- Introduction of new *init.ora* parameter
  - **DBNEST_ENABLE** Enables or disables DB Nest can have value **NONE** or **CDB_RESOURCE_PDB_ALL**
  - **DBNEST_PDB_FS_CONF** Specifies the location of an optional file system configuration file. Set this parameter in the CDB root.
- Use of a dedicated broker configured in *listener.ora* by **DEDICATED_THROUGH_BROKER_LISTENER**
- Introduction of new command line tools *dbnest* and *dbnestinit*
  - Allows to create, initialize and test DB Nests
- Requires additional OS package (don't think this is documented either)
  - **nscd** A Name Service Caching Daemon (nscd)
  - **sssd** System Security Services Daemon

Basic DB Nest Configuration

Simple configuration

- Configure a static listener entry for you database
- Configure a dedicated broker in listener.ora

```
DEDICATED_THROUGH_BROKER_LISTENER=ON
```

- Enable the broker

```
ALTER SYSTEM SET use_dedicated_broker=TRUE;
```

- Enable DB Nest and restart the database, em connect via listener…

```
ALTER SYSTEM SET dbnest_enable=cdb_resource_pdb_all SCOPE=SPFILE;
```

- Restart the database and check the *alert.log* for DB Nest

```
PDB1A(4):Creating (PDB1A) Nest for PDB(4)
PDB1A(4):DB Nest (PDB00004, 3250948838) open successful
```

The DB Nest

Check what is displayed by dbnest

```
oracle@db23:~/ [CDB23A] dbnest list
------------------------------------------------------------------------------
 Id : Nest              : Parent           : : Tag              :    State
------------------------------------------------------------------------------
  1 : ORA_CDB23A_CDB23A_b9b08f44 :                 :  ORA_CDB23A_CDB23A_b9b08f44 :     OPEN
      Net State         :
      Namespace State   : (pid=0,cnid=4026531836,pnid=4026531836,no namespace,type=0x0)
      Resources         : (cpu=0)
      Property enabled : resources
      Seccomp status    : (level=none)
      FS Isolation      : (disabled)
------------------------------------------------------------------------------
...
------------------------------------------------------------------------------
  4 : PDB00005          : ORA_CDB23A_CDB23A_b9b08f44 :  PDB2A (uid=3573573158) :     OPEN
      Net State         :
      Namespace State   : (pid=312021,cnid=4026532368,pnid=4026531836,type=0x7)
      Resources         : (cpu=0)
      Property enabled : namespaces,resources
      Seccomp status    : (level=strict1)
      FS Isolation      : (default-config)
------------------------------------------------------------------------------
Number of active nest namespaces = 4
------------------------------------------------------------------------------
```

Entering DB Nests

What can we do with the next?

- Use dbnest to enter the namespace of a nest e.g. opening a shell in this namespace

```
dbnest enter ORA_CDB23A_CDB23A_b9b08f44
Entering nest namespace : ORA_CDB23A_CDB23A_b9b08f44
...
```

- Try the PDB nest

```
oracle@db23:~/ [CDB23A] dbnest enter PDB00005
Entering nest namespace : PDB00005
shell not found : errno = 2
Exiting nest namespace : PDB00005
```

New tools for DB Nest

DB Nest command line utility

- New commandline tools to configure / administer Oracle DB Nests
- Currently not yet documented
- Highly try and error to use it
- Documentation seems to have been forgotten. Once again

```
oracle@db23:~/ [CDB23A] dbnest -h
 Usage : dbnest <command> [options]

 List of options and commands.

 init [options]                          Initialize nest environment
   --stage <staging area path>           Nest staging area path, Used for
                                         storing nest conf files, skeleton
                                         directories for nests etc.
   --cgroup <base cgroups>               Valid base cgroup path if
                                         required to override the default
                                         path available on the system.

 destroy [options]                       Destroy nest environment
```

5

A few Attempts

What about the useful / helpful features

Ok how does this look now?

Let's walk through the configuration

- What does the database look like?
- What can you do?
- What is the *dbnest* command?

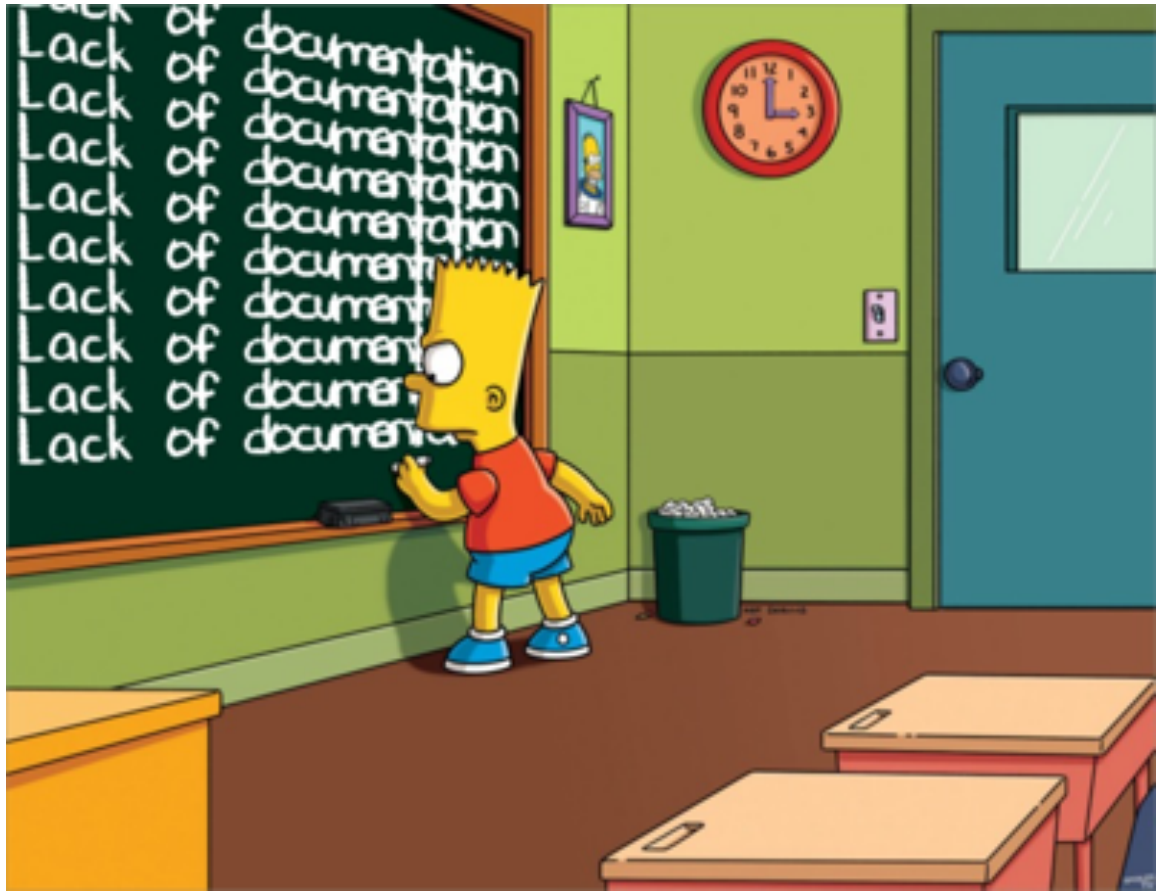
6

Challanges

What are the challenges?

The biggest Challenge

Where to start?



- Documentation for the different tools
- Missing resource management for **noisy neighbours**
- Bunch of hidden parameter
  - Needed, helpful?
- Ideas for concept and implementation
- Example use cases
- What happens to other OS? Like Windows?

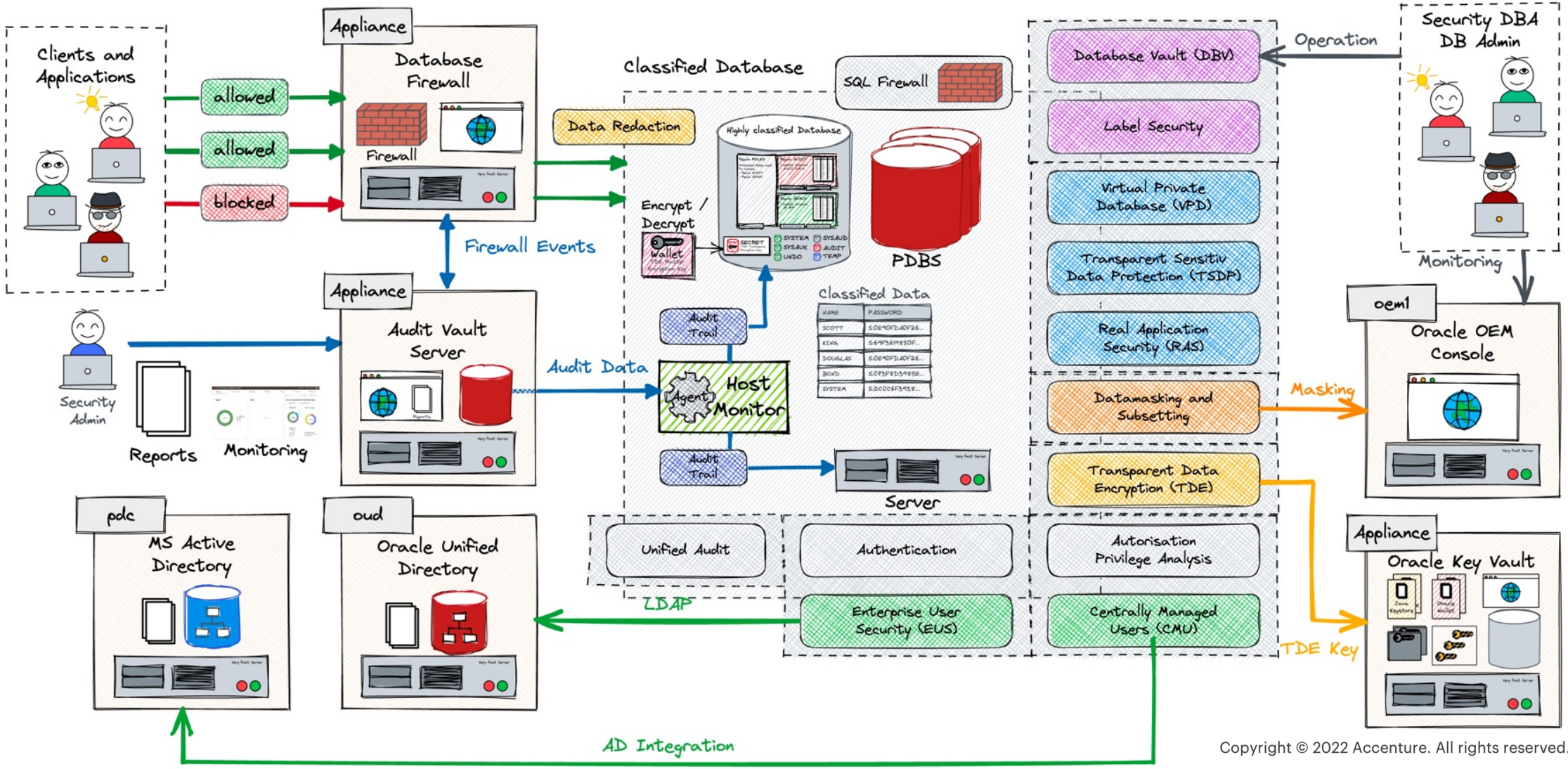
7

Alternative Security Measures

What alternative Measures Do I Have?

Maximum Data Security Architecture

Which Feature could Help?

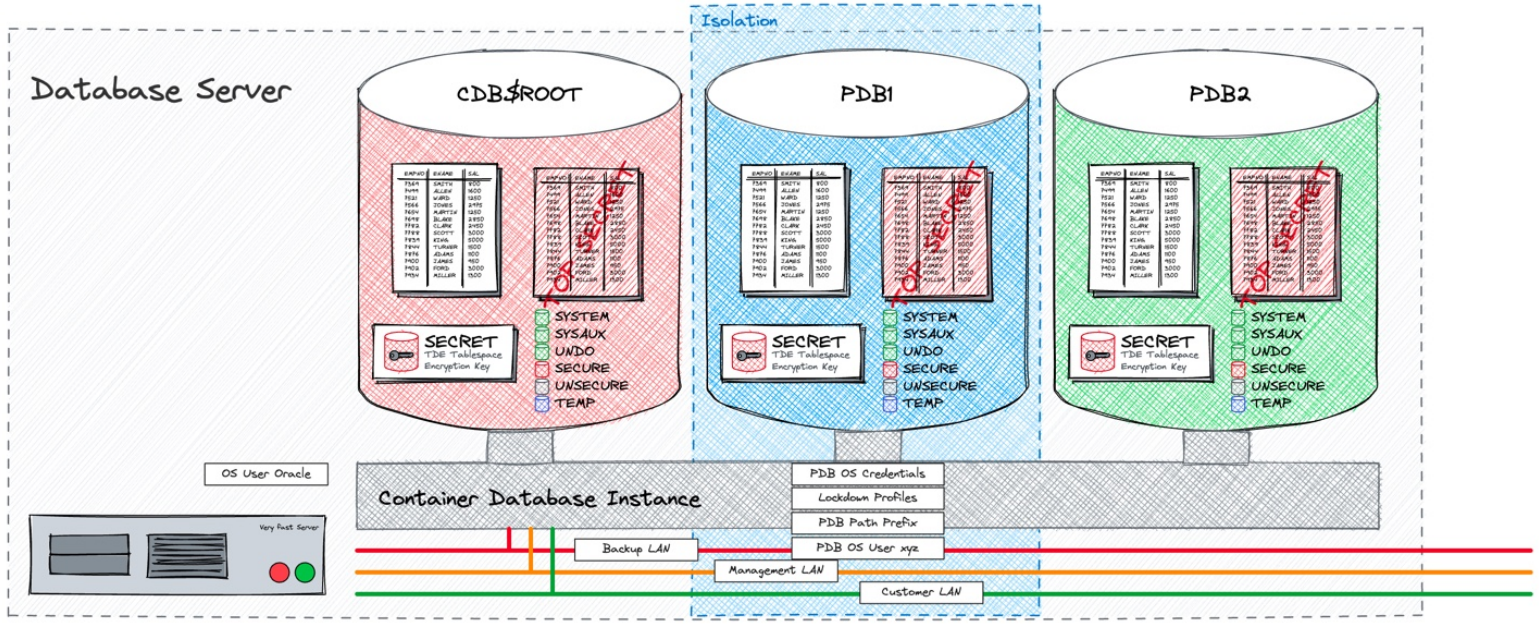


Possibilities for risk mitigation

What can we do with default features…

A multitenant container database provides the following features beyond regular security measures:

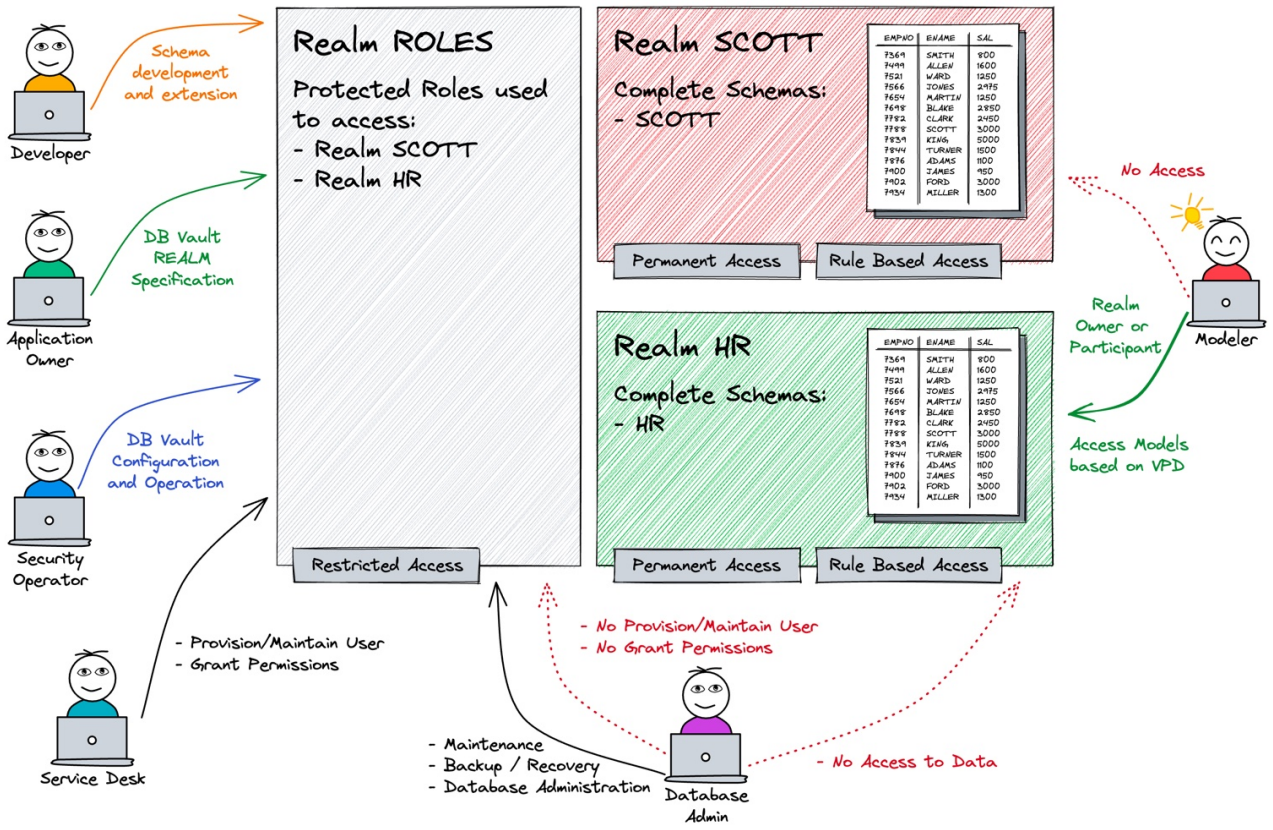
- **PATH_PREFIX** and **CREATE_FILE_DEST** clause to limit data files and directory objects to certain paths.
- **PDB_OS_CREDENTIAL** parameter assigning a dedicated user account for OS interactions
- **Lockdown profiles** to restrict certain operations or functionalities in a PDBs



Oracle Database Vault?

Oracle Database Vault…

- …provides advanced controls for sensitive data
  - Basic security concept is still necessary respectively even mandatory
- … integrated with existing security measures and features
- … implements a few basic security measures by just switching it on.
  - Update existing database roles
  - Modify some commands by adding command rules
    - Change some *init.ora* parameter
- Requires additional License
- There is always somebody who still can use resources



8

Conclusion

Ready for Production?

Conclusion

Ready for Production?

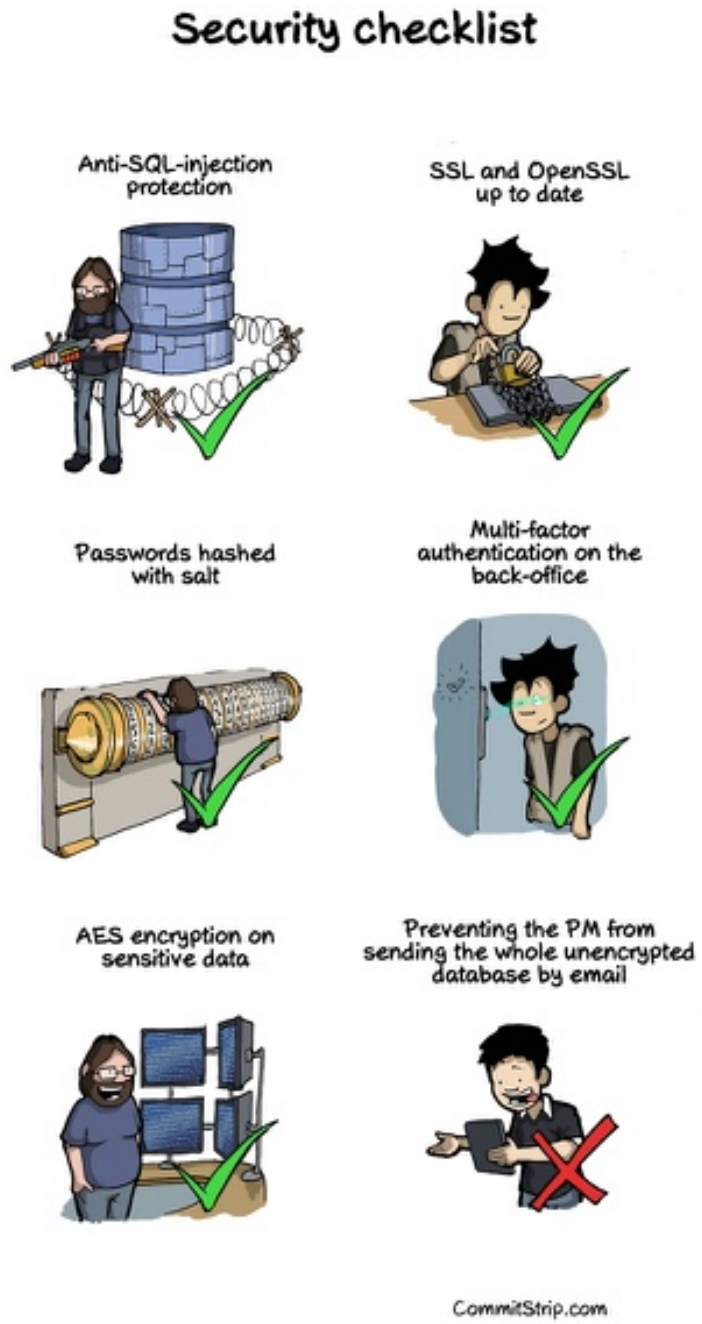
Basic configuration does work but…

- … long way to go to have a smooth and easy deployment
- … a few stuff does change
- … **lack** of documentation and examples

A few stuff needs clarification

- What does the **DB Nest roadmap** look like?
- Basic **concept** ideas?
- Differentiation from other tools and integration with features
- What about other **OS**?

The concept of Linux namespaces, cgroups, sccomp is proven



The practical use of DB Nest will last for a while.

Continue to rely on alternative measures in your security concept.

Thank You

Oracle Database Nest

Documentation, White Papers, Support Notes and other Links

- Oracle® Database Security Guide 21c - Securing and Isolating Resources Using DbNest
- OraDBA Blog PDB Isolation and Security

>