



Deynao

Oracle supervision and diagnostics

Deynao — The supervision account

A dedicated, read-only Oracle account — created by the script we provide, verifiable in one minute

Audience — DBA · System administrator · IT department

Level — Beginner — guided step by step (running the script requires SYSDBA access)

Guide version — 1.0

Deynao — The supervision account

To read the state of your databases, Deynao needs **an Oracle account**. Deynao's firm recommendation — and the industry **best practice** — is to use a **dedicated, read-only account**, rather than `SYS` or an application account.

You do not have to write it yourself: Deynao **provides the script** that creates the account and grants it exactly what it needs. This guide takes you from the **script** to the **account connected in Deynao**, then shows you how to **check for yourself** what that account can — and cannot — do.

 **This guide is the prerequisite for the others.** The account described here is the one you will enter in the connection form (**Getting started** guide) and the one used by **autonomous supervision** 24/7 (**Autonomous supervision** guide). Create it once, use it throughout.

Step 1 — Why a dedicated account

Three reasons, in this order.

Reason	What it gives you
Least privilege	The account receives <code>CREATE SESSION</code> and <code>SELECT</code> on system views, nothing else . No write privilege, no DBA role, no access to your business data.
Read-only	Monitoring must never be able to modify the database it observes. Here it is structural : the right to write was simply never granted.
Traceable	Deynao's connections are identifiable in <code>V\$SESSION</code> : they carry the dedicated account's name, and the <code>MODULE</code> column shows the <code>Deynao-Supervision-RO</code> identifier. You know who does what.

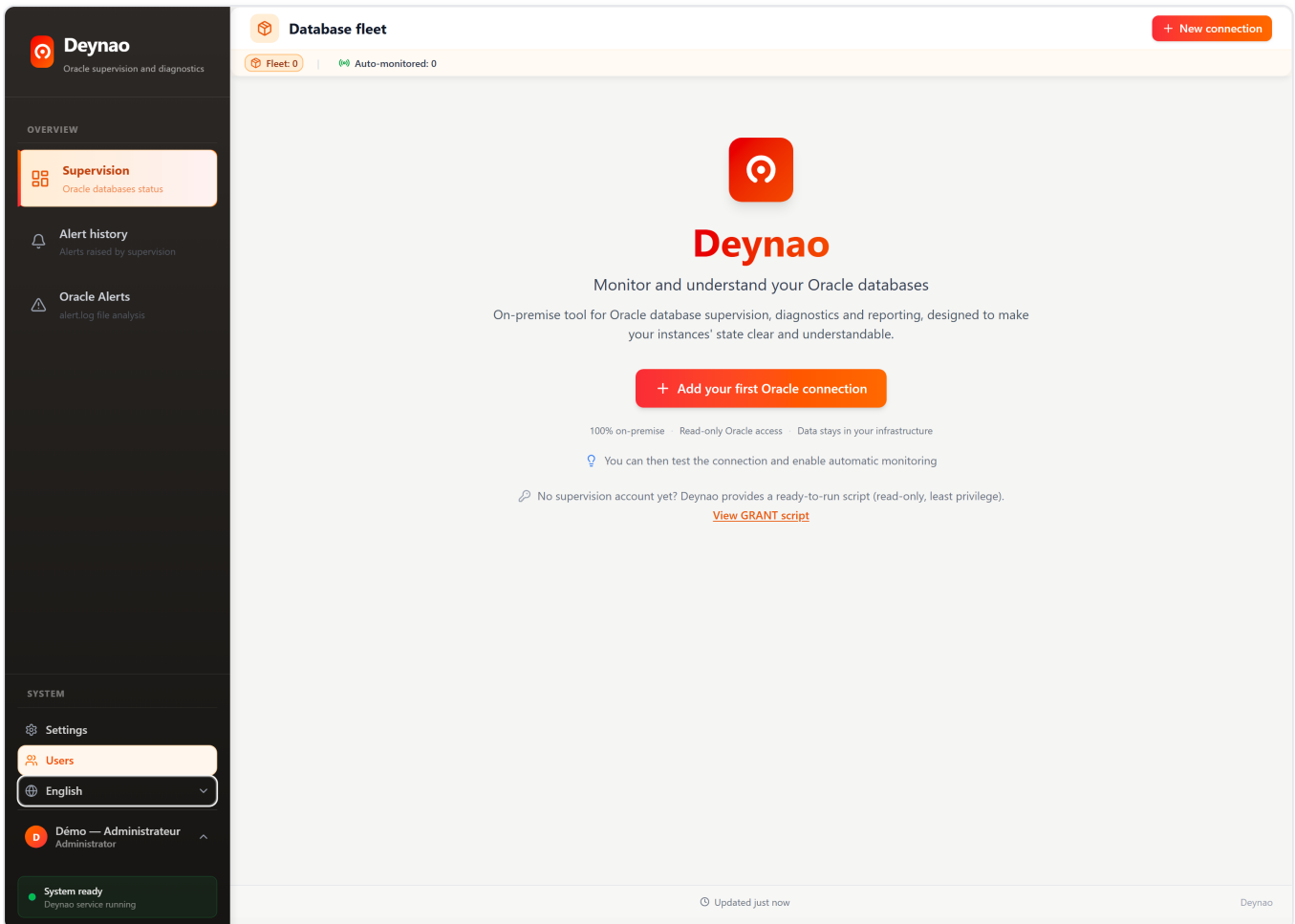
 **Why not `SYS` or the application account?** Monitoring with `SYS` means handing a read-only tool an account that can do **everything** — a needless risk. Monitoring with an application account mixes two purposes and **blurs traceability**. A dedicated account is **safer, clearer, and sufficient** for all monitoring. An incident on this account **never** exposes your business data.

One account per database. The account is **local to each database**: it must be created on **every** database you want to monitor. Fleet tip: keep **the same name everywhere** (the script suggests `DEYNAO_MON` as an example) — you will recognise the supervision account at a glance, on any database.

Step 2 — Get the script

The script is called `deynao_grant_privileges_template.sql`. It is **generic**: no account name, no server, no IP address hardcoded. Three ways to obtain it, whichever suits you.

a) From the home screen, as long as no database is registered yet: *"No supervision account yet? Deynao provides a ready-to-run script (read-only, least privilege)."* followed by the **"View GRANT script"** link.



The whole script is displayed, with two buttons: **Copy** and **Download**.

```
deynao_grant_privileges_template.sql Copy Download

-- =====
-- Deynao - Dedicated supervision account (creation + read-only privileges)
-- =====
-- This script CREATES the account (if missing) then grants it CREATE SESSION
-- + read-only SELECT on system views. No write privilege, no DBA role, no access
-- to application data. Generic (no account/server/IP hardcoded).
-- Compatible with Oracle 11g / 12c / 19c / 21c.
--
-- HOW TO RUN: connect as SYS, then run this file. The script prompts for the
-- supervision account name + a password (masked), creates the account if needed
-- and grants the privileges. Safe to re-run.
-- - Windows : Command Prompt on the Oracle server, then
```

b) From the connection form (New connection button), at any time: the "Dedicated supervision account (recommended)" section expands onto the same script.



New Connection

Connect an Oracle database to monitor

Connection Name

Production Oracle

Server *

srv-oracle-prod or 192.168.1

Port *

1521

Service *

ORCL

Username *

system



Connect as SYSDBA

Check only if this account has SYSDBA/SYSOPER privileges



Dedicated supervision account (recommended)



Best practice: supervise with a dedicated READ-ONLY account, rather than SYS or an application account.

Ready-to-use script to create this account (run as SYS):

deynao_grant_privileges_template.sql



Copy



Download

```
-- =====
-- Deynao - Dedicated supervision account (creation + read-only privil
-- =====
-- This script CREATES the account (if missing) then grants it CREATE
-- + read-only SELECT on system views. No write privilege, no DBA role
-- to application data. Generic (no account/server/IP hardcoded).
-- Compatible with Oracle 11g / 12c / 19c / 21c.
```

```
--  
-- HOW TO RUN: connect as SYS, then run this file. The script prompts  
-- supervision account name + a password (masked), creates the account  
-- and grants the privileges. Safe to re-run.  
-- - Windows : Command Prompt on the Oracle server, then
```

Environment *

Production

 Test connection

Cancel

Connect

c) **From the kit** delivered by Deynao, where the file is provided as-is.

 **It is exactly the same file.** Whether you copy it from the home screen, from the form or from the kit, Deynao serves **the same script**. The **Download** button writes it **without a BOM**: one invisible character at the start of the file would make every statement fail (`ORA-00911`). If you use **Copy**, paste the content into a **plain text** editor (Notepad, `vi`) — never into a word processor.

Step 3 — Run the script

3.1 The prerequisite: connect as `SYS AS SYSDBA`

This is the only prerequisite, and there is **no way around it**:

 **Why `SYS` and not `SYSTEM` ?** The `v$...` views are owned by `SYS` . A `SYSTEM` account, even with the DBA role, **cannot** grant privileges on them: the attempt fails with `ORA-01031` (insufficient privileges). That is an Oracle rule, not a Deynao choice. `SYS` is involved **only here**, just long enough to create the account — monitoring itself runs **solely** with the dedicated account.

3.2 The command

Place the file on the Oracle server, then, depending on your case:

Your situation	The command
On the Oracle server, Windows	Open the Command Prompt , then <code>sqlplus / as sysdba</code> <code>@C:\path\deynao_grant_privileges_template.sql</code>
On the Oracle server, Linux	In a shell, as the <code>oracle</code> user: <code>sqlplus / as sysdba</code> <code>@/path/deynao_grant_privileges_template.sql</code>
Remotely (Oracle client installed)	<code>sqlplus sys/<password>@<host>:<port>/<service> as sysdba @<file_path></code>
GUI tool (SQL Developer, Toad...)	Open the file and run it as a script (F5) — not statement by statement.

 **Multitenant database (12c and later).** For a **PDB**, connect **to the target PDB**: the supervision account is a **local** account there. On the server, after `sqlplus / as sysdba`, the command `ALTER SESSION SET CONTAINER=<pdb>;` places the session in the target PDB — that is what the screenshot below shows. If you run the script at the `CDB$ROOT` level, Oracle requires the `C##` prefix on the account name (otherwise **ORA-65096**). On a non-multitenant database (11g, or 12c+ without PDBs), there is nothing special to do.

3.3 The two questions asked

The script asks you **two things**, and nothing else:

Question	What you enter
<i>Deynao supervision account name</i> (e.g. <code>DEYNAO_MON</code>)	The account name. Keep the same one on all your databases .
<i>Password for this account (masked)</i>	A password of your choosing. Input is masked : it is neither displayed nor written to a spool file. Avoid double quotes .

The script then does its work and ends with a **summary**:

```
Command Prompt

C:\tmp\deynao>sqlplus / as sysdba

SQL*Plus: Release 21.0.0.0.0 - Production on Thu Aug 27 00:06:04 2026
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle. All rights reserved.

Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL> ALTER SESSION SET CONTAINER=XEPDB1;

Session altered.

SQL> @deynao_grant_privileges_template.sql
=====
Deynao - Supervision account setup (read-only)
=====

Deynao supervision account name (e.g. DEYNAO_MON): DEYNAO_MON
Password for this account (masked):
Account DEYNAO_MON already exists: creation skipped.
Granting read privileges (any errors will be shown below)...

=====
Summary
=====

CREATE SESSION : OK

Read views granted : 109

Done - account DEYNAO_MON ready for Deynao supervision.
An ORA-xxxxx error above = view absent on this Oracle version (no impact
on the other privileges).
=====
Disconnected from Oracle Database 21c Express Edition Release 21.0.0.0.0 - Pro
duction
Version 21.3.0.0.0

C:\tmp\deynao>
```

3.4 What the script does, exactly

- It **creates the account if it does not exist** (message *"Account ... created."*). If it already exists, creation is **skipped without error** (*"Account ... already exists: creation skipped."*) and the password you typed has **no effect**: the script **never** resets an existing password.
- It grants **CREATE SESSION** — without it, the account could not connect at all.
- It grants **SELECT** on the system views Deynao needs, grouped by purpose: **critical** (instance state, performance, capacity, sessions, locks, objects, Oracle alerts, security), **important** (logging and archivelog, files, UNDO, parameters), then **optional** (RMAN backups, scheduled jobs, advanced auditing, licence indicators) and the **detailed** dashboard views.
- It **wipes the password** from the SQL*Plus session memory as soon as creation is done.
- It prints a **verifiable summary** (see Step 4).

💡 **A missing view blocks nothing.** The script is **compatible with Oracle 11g, 12c, 19c and 21c**, and is written to **carry on** when an error occurs. A view that does not exist on **your** version (for example `unified_audit_trail`, introduced in 12c) produces an **ORA-...** error — typically `ORA-00942`, *"table or view does not exist"* — which is **displayed then ignored**, with no effect on the other privileges. The script says so itself at the end of its run. **It is safe to re-run**, as many times as needed.

⚠️ **The shortcut at the end of the script: not for production.** At the very bottom of the file you will see two **commented-out** lines (`SELECT_CATALOG_ROLE` and `SELECT ANY DICTIONARY`). They cover the views in one go and can help out in **pre-production or a lab**. In **production**, leave them commented: they grant access to the **whole** data dictionary, far beyond what Deynao asks for — precisely the opposite of the least privilege sought here.

Step 4 — Verify

4.1 The script's summary

At the end of its run, the script prints two lines that **answer the only question that matters**:

```
CREATE SESSION : OK
```

```
Read views granted : 109
```

- **CREATE SESSION : OK** — the account can connect. If you read **MISSING**, the account was not created (re-read the messages above: the profile's password policy, or the `C##` prefix required in `CDB$ROOT`).

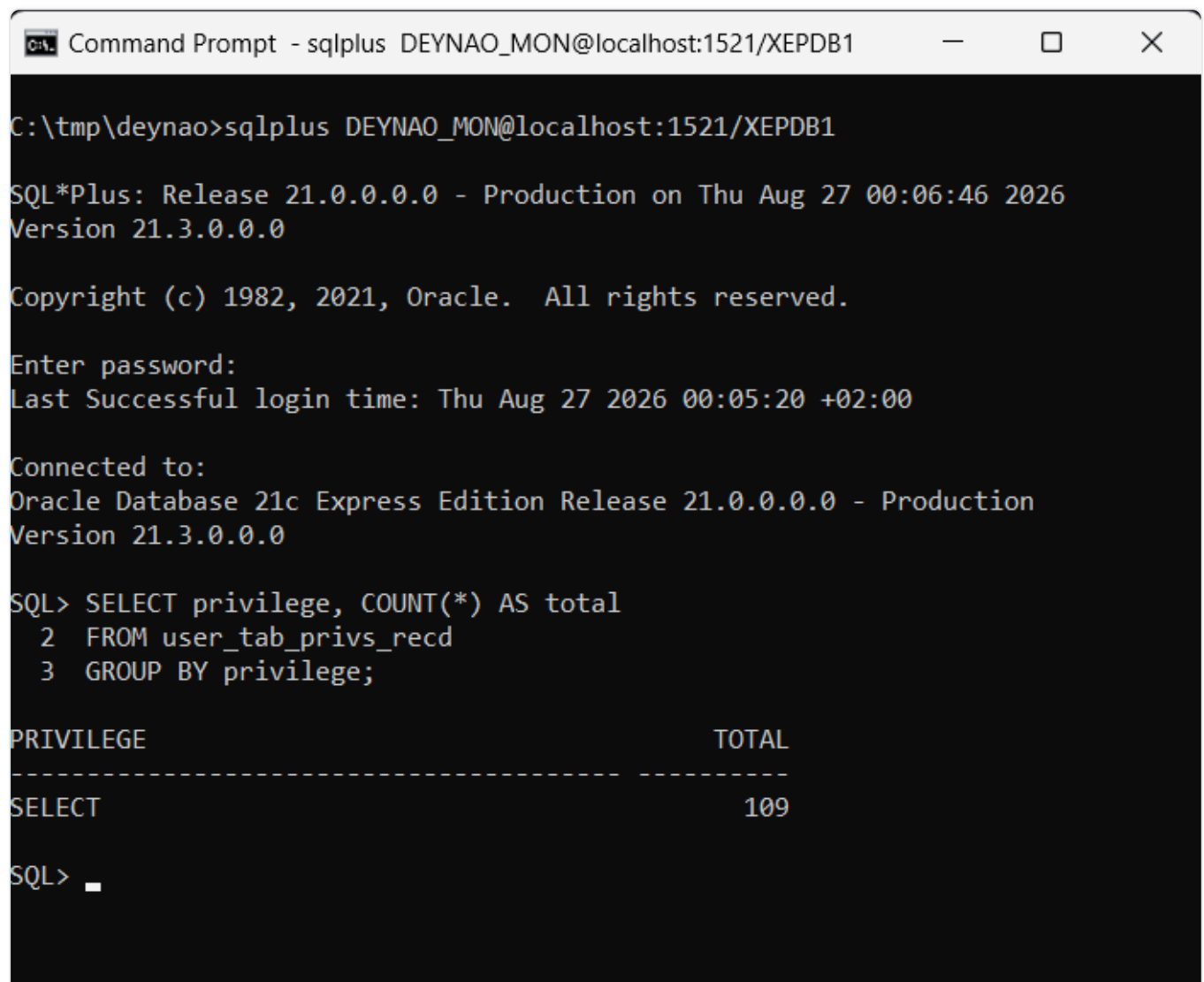
- **Read views granted** — the number of views the account can read. **This number depends on your Oracle version:** on the demonstration database (Oracle 21c), **all 109** views of the script exist and are granted; on an older database a few views are missing and the total is lower — **that is normal**.

4.2 The independent check (recommended)

The best check requires **no special privilege**: connect **with the supervision account itself** and ask it for its own list of privileges.

```
sqlplus <supervision_account>@<host>:<port>/<service>
```

```
SELECT privilege, COUNT(*) AS total
FROM   user_tab_privs_recd
GROUP  BY privilege;
```



```
C:\tmp\deynao>sqlplus DEYNAO_MON@localhost:1521/XEPDB1

SQL*Plus: Release 21.0.0.0.0 - Production on Thu Aug 27 00:06:46 2026
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle. All rights reserved.

Enter password:
Last Successful login time: Thu Aug 27 2026 00:05:20 +02:00

Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL> SELECT privilege, COUNT(*) AS total
      2  FROM user_tab_privs_recd
      3  GROUP BY privilege;

PRIVILEGE                                TOTAL
-----
SELECT                                109

SQL> _
```

✅ **What this check proves.** A **single** row comes back, and it reads **SELECT** . No **INSERT** , no **UPDATE** , no **DELETE** , no **EXECUTE** : the account **cannot write**. You do not have to take our word for it — the proof is produced by **your** database, in one command, at any time.



From the DBA side. If you prefer to audit from an administrator account, the equivalent query is `SELECT table_name, privilege FROM dba_tab_privs WHERE grantee = UPPER('<supervision_account>') ORDER BY table_name;` . It lists **every** granted view. The **Oracle Transparency** document provided by Deynao details, view by view, **why** each one is requested.

Step 5 — Connect the account in Deynao

In Deynao, open **New connection** and fill in the database coordinates and the account you have just created:

Field	Value
Server · Port · Service	The coordinates of the database to monitor.
Username	The supervision account (not <code>SYS</code>).
Connect as SYSDBA	Leave it unchecked. The dedicated account does not have — and does not need — those privileges.
Password	The one you chose in Step 3.
Environment	Development, Test or Production (it tunes how sensitive alerting is).

Click **Test connection before** saving: Deynao checks **the database** and **the credentials**, and answers in plain language.



New Connection

Connect an Oracle database to monitor

Connection Name

Production Oracle

Server *

<host>

Port *

<port>

Service *

<service>

Username *

<supervision_account>



Connect as SYSDBA

Check only if this account has SYSDBA/SYSOPER privileges



Dedicated supervision account (recommended)



Best practice: supervise with a dedicated READ-ONLY account, rather than SYS or an application account.

Environment *

Production



Password *

.....





Keep this database in my fleet

Connection and (encrypted) credentials saved — supervise this database anytime.



Database reachable, credentials valid



Test connection

Cancel

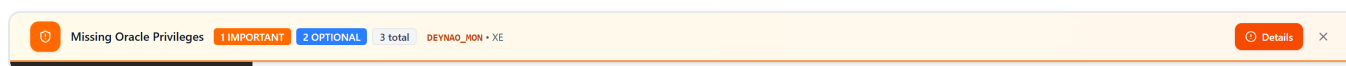
Connect



Test first, save second. The test keeps nothing. Once you get the green message, **Connect** saves the connection (credentials **encrypted**) and the database joins your fleet. What comes next — dashboards, domains, recommendations — is covered by the **Getting started** guide; enabling 24/7 monitoring, by **Autonomous supervision**.

Step 6 — If the "Missing Oracle Privileges" banner appears one day

One day, perhaps, an orange banner will appear at the top of a dashboard:



This is not a failure. It is Deynao telling you, honestly, that it is missing a **read** privilege needed to display part of what it can do. The usual causes are mundane:

- a **Deynao update** added a view to its list, and the script has not been re-run;
- the database was **restored** or **recreated**, and the privileges did not follow;
- the connection uses an **application account** instead of a dedicated supervision account.

The **Details** button opens the "**Oracle Privileges Analysis**" window, which tells you everything:

Oracle Privileges Analysis

DEYNAO_MON • XE

Incomplete Oracle Supervision

This account is missing some privileges for complete Oracle supervision. **3 missing privileges in total** (1 important, 2 optional). A few advanced features are not available.

MISSING PRIVILEGES

3

v\$rman_backup_job_details

OPTIONAL

RMAN backup job details unavailable

dba_scheduler_job_run_details

OPTIONAL

Jobs execution history unavailable

v\$diag_alert_ext

IMPORTANT

Real-time alert.log monitoring unavailable

FIX PRIVILEGES

Recommended — dedicated supervision account
Rather than patching this application account, create a dedicated read-only account (least-privilege) — the recommended path for clean, safe supervision.
[View the dedicated account script](#)

Or quickly unblock this account:

SQL command examples to adapt — Replace values according to your environment before execution.

✓ Online

SYSDBA

< 1 min

Oracle 11.2+

Copy

Download

```
-- =====
-- Script: Deynao supervision privileges fix
-- Target account: DEYNAO_MON
-- Prerequisites: connect as SYS AS SYSDBA. V$/DBA_* views are owned
-- by SYS; SYSTEM CANNOT grant privileges on them (ORA-01031).
-- Usage: sqlplus / as sysdba @grant_privileges_deynao_mon.sql
-- Remote: sqlplus sys/<password>@<host>:<port>/<service> as sysdba @<file>
-- =====
SET ECHO ON
```

Close

What the window shows	What you do with it
The level of impact (<i>Incomplete · Limited · Impossible Oracle Supervision</i>)	You know straight away whether it is a detail or a blocker.
Each missing view , with its severity (CRITICAL · IMPORTANT · OPTIONAL) and its plain-language impact	You know what you lose, screen by screen.
"Recommended — dedicated supervision account"	The right path: create (or reconnect) a dedicated account, with this guide's script.
"Or quickly unblock this account"	A targeted fix script, asking only for the privileges actually missing.

Deynao — The supervision account

Page 14 / 17


Oracle Privileges Analysis
DEYNAO_MON • XE
✕

```

-- A view missing on a given Oracle version must not stop the rest of the script:
WHenever SQLERROR Continue

PROMPT =====
PROMPT Granting Deynao supervision privileges to account DEYNAO_MON
PROMPT (any ORA-xxxxx errors are shown below, without blocking)
PROMPT =====

-- [IMPORTANT] Strongly recommended privileges
GRANT SELECT ON sys.v_$diag_alert_ext TO DEYNAO_MON;

-- [OPTIONAL] Advanced features
GRANT SELECT ON sys.v_$rman_backup_job_details TO DEYNAO_MON;
GRANT SELECT ON sys.dba_scheduler_job_run_details TO DEYNAO_MON;

-- =====
-- VERIFICATION: count the supervision views from this script actually granted
-- (expected: 3; a lower number = a view missing on this version or a failed GRANT)
-- =====
SET HEADING OFF
SELECT 'Supervision views granted to DEYNAO_MON : ' || COUNT(*) || ' / 3'
FROM dba_tab_privs
WHERE grantee = UPPER('DEYNAO_MON')
  AND owner = 'SYS'
  AND privilege = 'SELECT'
  AND table_name IN ('V_$DIAG_ALERT_EXT', 'V_$RMAN_BACKUP_JOB_DETAILS', 'DBA_SCHEDULER_JOB_RUN_DETAILS');
SET HEADING ON

-- =====
-- ROLLBACK (revoke these grants if needed – uncomment the desired lines):
-- =====
-- REVOKE SELECT ON sys.v_$diag_alert_ext FROM DEYNAO_MON;
-- REVOKE SELECT ON sys.v_$rman_backup_job_details FROM DEYNAO_MON;
-- REVOKE SELECT ON sys.dba_scheduler_job_run_details FROM DEYNAO_MON;

-- =====
-- ALTERNATIVE (pre-prod / lab): instead of the individual grants above,
-- SELECT_CATALOG_ROLE + SELECT ANY DICTIONARY cover all views.
-- Not recommended in PRODUCTION (grants access to the whole data dictionary).
-- GRANT SELECT_CATALOG_ROLE TO DEYNAO_MON;
-- GRANT SELECT ANY DICTIONARY TO DEYNAO_MON;
-- =====

PROMPT Done. Check the number of granted views above.

EXIT;

```

 Test these commands in a development or test environment before executing them in production. Adapt values to your infrastructure.

☐ Read only — Deynao never modifies your database.

Close

This fix script is **built for your case**: it contains only the missing `GRANT` statements, carries the same prerequisites (**SYSDBA**, Oracle 11.2+, under a minute), embeds **its own verification query** and even the **rollback** lines (commented out). **Copy** and **Download** buttons, just like the main script.

 **The simplest reflex.** When in doubt, **re-run the full script** from Step 3: it is safe to re-run, it skips creating the existing account, and it catches up **all** the privileges at once. Then, as the window says, **reconnect** with the account to get full monitoring back.

Step 7 — Troubleshooting

Symptom	Likely cause	Fix
ORA-01031 when the GRANT statements run	You are not connected as SYS AS SYSDBA	The v\$... views are owned by SYS : go back to Step 3.1 .
ORA-65096 when the account is created	You are at the CDB\$ROOT level of a multitenant database	Connect to the target PDB , or prefix the account name with C## .
The script reports a creation failure	The Oracle profile's password policy rejects the password	Choose a password that complies with your profile, then re-run the script.
ORA-00911 : invalid character , on every line	The file was saved with a BOM or came from a word processor	Download the script again with the Download button, or save it as plain text .
A few ORA-00942 during the run	Those views do not exist on your Oracle version	No action needed : this is expected, the remaining privileges are granted.
CREATE SESSION : MISSING in the summary	The account was not created	Re-read the error messages just above the summary (password, or C##).
The connection test fails in Deynao	Coordinates, password, or unreachable database	Check <host>:<port>/<service> ; test with sqlplus first (Step 4.2).
The "Missing Oracle Privileges" banner appears	A read privilege is missing	Step 6 — normal behaviour, fix script provided.

 **Oracle 11g databases.** Deynao needs **Oracle Instant Client** (recommended: version 19c) to connect to them. The supervision account itself is created **in exactly the same way**.

What this account cannot do

This may be the most important part of this guide — the part to show your **security team**.

This account cannot...	Why
Write anything to your databases	No <code>INSERT</code> , <code>UPDATE</code> , <code>DELETE</code> or <code>EXECUTE</code> is granted to it (Step 4.2 proves it in one command).
Read your business data	The privileges cover system views (<code>V\$...</code> , <code>DBA_...</code>) — never your application tables.
Administer the database	No DBA role , no <code>ANY</code> privilege: it can neither create, modify nor drop an object.
Expose you to an Oracle bill	No view subject to a paid option (neither Diagnostics Pack <code>DBA_HIST_*</code> /ASH, nor Tuning Pack). Compatible with Standard Edition .
Be used elsewhere	It is local to one database and only makes sense for monitoring.

 **A structural guarantee, not a promise.** Deynao cannot read a view it was not granted: the query would fail. The list of privileges requested is **public** (the script you have just run), the list of privileges **actually held** is **verifiable by you** (Step 4.2), and the **Oracle Transparency** document explains the **why** behind each family of views. Nothing is hidden, and you stay in control.

You are ready

Your databases have a **dedicated supervision account, read-only**, created by a **generic** and **re-runnable** script, verifiable in one command, and connected in Deynao. This is the foundation for everything else.

What next? Explore the dashboards and the 14 domains with the **Getting started** guide, then enable **automatic monitoring** and **e-mail alerting** with the **Autonomous supervision** guide.

Deynao — Oracle supervision and diagnostics, 100% on-premise. Your data stays with you.