Oracle

Primavera
Analytics Security Guide for On-Premises

Version 20
December 2020

Contents

Primavera Analytics Security Overview

During the installation and configuration process for Primavera Analytics and Primavera Data Warehouse, several options are available that impact security. Depending on your organization's needs, you might need to create a highly secure environment for Primavera Analytics and Primavera Data Warehouse. Use the following guidelines to plan your security strategy for Primavera Analytics and Primavera Data Warehouse:

- Review all security documentation for applications and hardware components that interact or integrate with Primavera Analytics and Primavera Data Warehouse. Oracle recommends you harden your environment (where required).
- Read through the summary of considerations for Primavera Analytics and Primavera Data Warehouse included in this document. Areas covered include: safe deployment, authentication options, authorization, confidentiality, sensitive data, and reliability.

Tips

As with any software product, be aware that security changes made for third-party applications might affect Primavera Analytics and Primavera Data Warehouse applications.

Safe Deployment of Primavera Analytics and Primavera Data Warehouse

To ensure overall safe deployment, you should carefully plan security for all components, such as database servers and client computers that are required for and interact with Primavera Analytics and Primavera Data Warehouse. In addition to the documentation included with other applications and hardware components, follow the Primavera Analytics and Primavera Data Warehouse-specific guidance below.

Administrative Privileges Installing and Operating Primavera Analytics and Data Warehouse

As the administrator, you should determine the minimum administrative privileges or permissions needed to install, configure, and operate Primavera Analytics and Primavera Data Warehouse.

Physical Security Requirements for Primavera Analytics and Primavera Data Warehouse

You should physically secure all hardware hosting Primavera Analytics and Primavera Data Warehouse to maintain a safe implementation environment. See the *Primavera Analytics Planning and Sizing Guide.*

Files to Back Up after Installing Primavera Analytics and Primavera Data Warehouse

Once Primavera Analytics and Primavera Data Warehouse installation and configuration is compete, you should back up the files that are not needed for daily operations. Use your discretion to determine the complete list of files to be backed up, but Oracle recommends that you include the following:

- **dwstaretl.properties**, which contains DB user names and encrypted passwords, along with applications database host URLs. The user who installs, configures, and runs the ETL process must have read/write privileges to this file. The default location is the **<star_home>\star\config\res** folder.
- **etl_calculations.tcsv**, which contains metadata calculations. The default location is the **<star_home>\star\etl_homes\staretl\res\metadata** folder.
- Any custom scripts that you have created. The default location for custom scripts is the **<star_home>\star\etl_homes\staretl\scripts\user_scripts** folder.

Authentication Options

When you set up Primavera Analytics, you can choose one of the following authentication modes:

- **Native** is the default mode for the application. In Native mode, the database acts as the authority and the application handles the authentication of the user who is logging into that application.
- **Single Sign-On (SSO)** controls access to Web applications. In SSO mode, the applications are protected resources. When a user tries to login to one, a Web agent intercepts the login and prompts the user for login credentials. The Web agent passes the user's credentials to a policy server, which authenticates them against a user data store. With SSO, once the users login, they are logged into all Web applications during their browser session (as long as all Web applications authenticate against the same policy server).
- **Lightweight Directory Access Protocol (LDAP)** authenticates users through a directory and is available for all applications. The application supports LDAP referrals with Oracle Internet Directory and Microsoft Windows Active Directory. LDAP referrals allow authentication to extend to another domain. You can also configure multiple LDAP servers, which supports failover and enables you to search for users in multiple LDAP stores. In LDAP mode, an LDAP directory server database confirms the user's identity when they attempt to log in to an application.

Authorization for P6 EPPM, Primavera Unifier, and Primavera Data Warehouse

Grant appropriate authorization to all P6 EPPM, Primavera Unifier, and Primavera Data Warehouse users. See the following documents for details on the most secure application security options:

- *Primavera Data Warehouse Installation and Configuration Guide*
- *Primavera Analytics Installation and Configuration Guide*
- *P6 EPPM Security Guide*
- *Primavera Unifier Administration Guide*

Authentication for Primavera Analytics depends on your authorization method for Oracle Business Intelligence (OBI); however, all of the user names must match exactly in the following products:

- P6 EPPM
- Primavera Unifier
- Primavera Data Warehouse
- OBI WebLogic server

Maintaining Confidentiality for Primavera Analytics and Primavera Data Warehouse

Confidentiality ensures only authorized users see stored and transmitted information. In addition to the documentation included with other applications and hardware components, follow the guidance below.

- For data in transit, use Secure Socket Layer (SSL)/Transport Layer Security (TLS) to protect network connections among modules. If you use LDAP or SSO authentication, ensure you use LDAP over SSL to connect to the directory server.
- For data at rest, refer to the documentation included with the database server for instructions on securing the database.

Sensitive Data in Primavera Analytics and Primavera Data Warehouse

Protect sensitive data in Primavera Analytics and Primavera Data Warehouse, such as user names, passwords, and email addresses. Use the process below to help during your security planning:

- Determine which products and interacting applications display or transmit data that your organization considers sensitive. For example, costs and secure codes.
- Implement security measures in Primavera Analytics and Primavera Data Warehouse to carefully grant users access to sensitive data. For example, in P6 EPPM, use a combination of Global Profiles, Project Profiles, and OBS access to limit access to data. In Primavera Unifier, use Company level permissions to grant access to companywide data for all projects and Project level permissions to grant access to a project-specific data.
- Implement security measures for applications that interact with Primavera Analytics and Primavera Data Warehouse, as described in the documentation included with those applications.

- Implement consent notices in the source applications configured with Primavera Analytics and Primavera Data Warehouse to gather the consent of users to store, use, process, and transmit personal information (PI) and to alert users when there is a risk of PI being exposed.
- Implement a password policy by default. Password policies must be applied for:
  - Application users

    For source applications, it is their responsibility.
  - Database-level password policy or Weblogic user password policies.

    The DBA is responsible for their specific policies
  - Oracle recommends users have a secure password policy for Oracle databases and for WebLogic.

Transparent Data Encryption (TDE)

Transparent Data Encryption (TDE) is an Oracle Advanced Security feature that is used for Oracle Database encryption. TDE provides strong protection from malicious access to database files by encrypting data before it is written to storage, decrypting data when being read from storage, and offering built-in key management.

For TDE implementation instructions, refer to the **readme.txt** file in the P6 EPPM physical media or download at database\scripts\common\tde.

For more information about TDE, refer to the ***Oracle Advanced Security Guide.*** ***https://docs.oracle.com/en/database/oracle/oracle-database/18/asoag/introduction-to-oracle-advanced-security.html#GUID-81BF34FA-6044-47D4-BF31-12DEF178BA6B***

To start using TDE, you must create a wallet and set a master key. The wallet can be the default database wallet shared with other Oracle Database components or a separate wallet specifically used by TDE. In an effort to exercise these security practices, Oracle strongly recommends using a separate wallet to store the master encryption key.

Specifying a Wallet Location

If you choose to use a wallet specifically for TDE, specify a wallet location in the sqlnet.ora file by using the ENCRYPTION_WALLET_LOCATION parameter. The wallet location specified by this file and parameter is used to create the master encryption key. If the ENCRYPTION_WALLET_LOCATION parameter is not present in the sqlnet.ora file, then the WALLET_LOCATION value is used. A new wallet is created if one does not exist already.

If no wallet location is specified in the sqlnet.ora file, then the default database wallet location, ORACLE_BASE/admin/DB_UNIQUE_NAME/wallet or ORACLE_HOME/admin/DB_UNIQUE_NAME/wallet, is used. Here, DB_UNIQUE_NAME is the unique name of the database specified in the initialization parameter file.

If an existing auto-login wallet is present at the expected wallet location, then a new wallet is not created.

Setting the Master Encryption Key

Before you can encrypt or decrypt database columns or tablespaces, you must generate a master encryption key.

Opening the Encrypted Wallet

The database must load the master encryption key into memory before it can encrypt or decrypt columns/tablespaces. Opening the wallet allows the database to access the master encryption key. Once the wallet has been opened, it remains open until you shut down the database instance or close it explicitly.

Encrypting Tablespaces

TDE tablespace encryption encrypts and decrypts data during read/write operations.

To create an encrypted tablespace, run the following:

```
CREATE
   [ BIGFILE | SMALLFILE ]
   { permanent_tablespace_clause
   | temporary_tablespace_clause
   | undo_tablespace_clause
   } ;
```

where

```
permanent_tablespace_clause=
TABLESPACE tablespace
.........
ENCRYPTION [USING algorithm]
.........
storage_clause
.........
```

where

```
storage_clause=
.........
[ENCRYPT]
.........
```

For example:

```
CREATE TABLESPACE securespace
DATAFILE '/home/user/oradata/secure01.dbf'
SIZE 150M
ENCRYPTION USING 'AES128'
DEFAULT STORAGE(ENCRYPT);
```

> **Note**: An existing tablespace cannot be encrypted; however, you can import data into an encrypted tablespace using the Oracle Data Pump utility.

The keystore is container-level and keys can be separate for pdbs. To configuring a keystore, complete the following steps:

1) Set the location of the wallet in sqlnet.ora.
2) After logging into the database with SYSDBA or at least SYSKM role, create a password protected wallet:

```
SQL> administer key management create keystore '/PATH/TO/ORACLE/WALLET/DIR'
identified by tdecdb;
```

3) Open the keystore.

```
SQL> administer key management set keystore open identified by tdecdb
container=all;
```

4) With the wallet open, a TDE key can be created. For multitenant environments, a TDE key can be used by all PDBs or each PDB can have a dedicated TDE key.

```
SQL> administer key management set key using tag 'cdb_shared' identified by tdecdb
with backup using '/tmp/wallet.bak' container=all;
```

5) Create encrypted tablespaces:

```
CREATE SMALLFILE TABLESPACE STAR_DAT1 DATAFILE 'star_dat1.dbf' SIZE 100M
AUTOEXTEND ON NEXT 10M MAXSIZE UNLIMITED LOGGING EXTENT MANAGEMENT LOCAL UNIFORM
SIZE 1M SEGMENT SPACE MANAGEMENT AUTO
       ENCRYPTION USING 'AES128'
       DEFAULT STORAGE(ENCRYPT);
CREATE SMALLFILE TABLESPACE STAR_HST1 DATAFILE
       'star_hst1.dbf' SIZE 10M AUTOEXTEND ON NEXT 100M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M SEGMENT SPACE MANAGEMENT AUTO
       ENCRYPTION USING 'AES128'
       DEFAULT STORAGE(ENCRYPT);
```

Applying TDE to an Existing Star Schema

An existing tablespace cannot be encrypted. To apply TDE to an existing Star Schema:

1) Create tablespaces that are encrypted. For more information, see ***Encrypting Tablespaces*** (on page 9).
2) Move data from existing tablespaces to the new ones using table redefinition. The files that are used are preferences.txt, grants_before_tde.sql, migrate.sql, and grants_after_tde.sql.
   a. Edit the preferences.txt file so that it matches your environment and security requirements:

```
alg varchar2(7 char) NOT NULL := 'AES128';
owner varchar2(30 char) NOT NULL := 'STARUSER';
source_dir varchar2(128 char) NOT NULL := '/u01/opt/oradata/orcl/';
target_dir varchar2(128 char) NOT NULL := '/u02/opt/oradata/orcl/';
```

3) Replace 'STARUSER' with the user name applicable to your environment.

Note:

- `source_dir` is the folder that contains the current clear text application tablespaces.
- `Target_dir` can point to the same or another folder or partition which allows you to relocate the encrypted tablespaces.
- `grants_before_tde.sql` grant the required privileges to STARUSER (change STARUSER to your schema user in case it is different). Run the script as sys user.
- `migrate.sql` migrates all objects in the given clear text tablespace into its encrypted counterpart. Run this script from command prompt for both Star_Dat1 and Star_Hst1 tablespaces as staruser.
- `migrate_tables_part.sql` migrates the remaining tables that have row level security enabled. Run this script as staruser and post migrate.sql finishes.
- `grants_after_tde.sql` will revoke all the privileges that were provided to staruser for creating and migrating data to encrypted tablespaces. Run the script as sys user.

Post-migration steps

Once the migration of all tablespaces is complete, manually complete the following steps:

1) Rename all tablespaces and take the clear text tablespaces offline (as 'STARUSER'):

```
alter tablespace STAR_DAT1 rename to STAR_DAT1_backup;
alter tablespace STAR_DAT1_ENC rename to STAR_DAT1;
alter tablespace STAR_DAT1_backup offline normal;
alter tablespace STAR_HST1 rename to STAR_HST1_backup;
alter tablespace STAR_HST1_ENC rename to STAR_HST1;
alter tablespace STAR_HST1_backup offline normal;
```

2) After confirming that the ETL process and OBIEE reports runs seamlessly off encrypted tablespaces, the original clear-text tablespaces can now be deleted:

```
drop tablespace STAR_DAT1_backup including contents and datafiles;
drop tablespace STAR_HST1_backup including contents and datafiles;
```

3) Log in as `sysdba` and run `grants_after_tde.sql`.

Troubleshooting Common Errors

Primavera Data Warehouse writes detailed process information to the log files. The Logs contain information about the installation, as well as about each run of the STARETL process. If an error occurs, the log files may include diagnostic information. Analyzing these files can help lead you to the resolution or to the file or process that caused the error.

Cannot auto-create wallet (ORA-28368)

If you encounter the above error while running the following line:

```
ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY "admin";
```

- Make sure the folder where you want your encryption keys to be placed exists.
- If it is not already created, create one and re-run the above line.

Encryption wallet, auto login wallet, or HSM is already open (ORA-28354)

If you encounter the above error, run:

```
alter system set  encryption wallet close identified by "<password>";
```

then

```
alter system set encryption wallet open identified by "<password>";
```

TNS: wallet open failed (ORA-12578)

- Check that sqlnet.ora file does not contain any extra spaces before the WALLET_LOCATION line.
- Make sure you have not missed any parenthesis.
- If the above 2 do not work, edit your sqlnet.ora file and modify SQLNET.WALLET_OVERRIDE parameter value from TRUE to FALSE.

Auto login wallet not open but encryption wallet may be open (ORA-28390)

If you encounter the above error while trying to close the wallet with `alter system set encryption wallet close`, correct the statement to:

```
alter system set  encryption wallet close identified by "<password>";
```

AES Customer-Supplied Encryption

Advanced Encryption Standard (AES) is an encryption tool used to encrypt and decrypt text using an AES encryption algorithm. With this customer-supplied encryption, passwords for all schema required to move data from Primavera databases into the Primavera Data Warehouse will be encrypted.

Configuring AES and Primavera Data Warehouse

Before running configStar, complete the following steps to create the keystores and configure them for use with Primavera Data Warehouse:

> **Note**: If there is no need of keystores, ignore this procedure and the passwords will be encrypted as AE04.

1) Open a command prompt/terminal window and navigate to **<PDW Install Folder>\star\config folder**.
2) Run the following command:

```
java -classpath lib/prm-common.jar com.primavera.common.KeyStoreInstaller
-createnew.
```

3) When prompted, enter a password for the keystore.
4) Open a command prompt/terminal window and navigate to **<PDW Install Folder>\star\config folder**.
5) Run the configStar.cmd (with Windows) or configStar.sh (with UNIX or Linux) to create a new staretl source.
6) After configuring the new staretl source, perform the following steps to access the keystore:
   a. Copy the created keystore files (p6keystore.jks and p6kspass.pwf) located in **/star/config folder to etl_homes/staretlX/** where x is the number of the ETL source.
   b. Change the keystore:
      1. Open **p6kspass.pwd** and delete its content.
      2. Enter the password that has been used to generate the keystore in this file in clear text.

      > **Note**: Be careful not to introduce new spaces or lines.

      3. Save and exit.
      4. After the first ETL run, the process will encrypt the password.
   c. Regenerate the password
      1. Go to the **staretl** folder.
      2. Regenerate the password (use the password as used to create the keystore)

```
java -classpath lib/prm-common.jar com.primavera.common.KeyStoreInstaller
-genpassfile
```

7) In the WebApp section:
   a. Change bin/startWeblogic.sh to specify the location of primavera.bootstrap.home (line 99)

```
JAVA_OPTIONS="${SAVE_JAVA_OPTIONS}
-Dprimavera.bootstrap.home=/home/g/staretl-cloud/config"
```

Reliability for Primavera Analytics and Primavera Data Warehouse

Take the following steps to protect against attacks that could cause a denial of service:

- Install the latest security patches on all Primavera Analytics and Primavera Data Warehouse servers.
- (P6 EPPM only) Replace the default Admin Superuser (admin) immediately after a manual database installation or an upgrade from P6 7.0 and earlier.
- Ensure log settings meet the operational needs of the server environment. Do not use "Debug" log level in production environments.
- Document the configuration settings used for servers and create a process for changing them.
- Protect access to configuration files with physical and file system security.

Cookies Usage in Primavera Analytics and Primavera Data Warehouse

When using Primavera Analytics and Primavera Data Warehouse, the server may generate cookies and send them to the user’s browser. The user’s machine stores the cookies, either temporarily by the browser, or permanently until they expire or are removed manually.

Oracle might use cookies for authentication, session management, remembering application behavior preferences and performance characteristics, and to provide documentation support.

Also, Oracle might use cookies to remember your login details, collect statistics to optimize site functionality, and deliver marketing based on your interests.

Primavera Data Warehouse Security

Primavera Data Warehouse maintains security similarly to P6 EPPM and Primavera Unifier. In P6 EPPM, the security being maintained consists of Project/Cost security, Resource security, and OBS security. In Primavera Unifier, the security being maintained consists of Company level permissions and Project level permissions, which both work in tandem with the permissions set in Primavera Analytics for a user.

Primavera Data Warehouse has row-level security that is built into the Oracle Enterprise Edition database.

Primavera Analytics Security

Primavera Data Warehouse row-level security is enforced when queries are processed from the OBI server. To apply the proper security and ensure users have access to their data, confirm that the following user names match:

- P6 EPPM
- Primavera Unifier
- Primavera Data Warehouse
- OBI WebLogic server

WebLogic Embedded Security

Primavera Analytics leverages the existing WebLogic embedded security model. This means that Primavera Analytics supports all the various security implementations that a traditional WebLogic server supports. This section lists are some common security methods.

WebLogic Authenticator Provider

To allow users to access Primavera Data Warehouse, create a group in the provider called **p6rdbusers** and assign each administrator that will run the Primavera Analytics web application to this group.

For details on configuring the WebLogic Authenticator Provider, see http://docs.oracle.com/cd/E17904_01/web.1111/e13707/atn.htm#i1206556.

LDAP Providers

WebLogic supports a variety of LDAP providers. Refer to the documentation from your LDAP provider for details on adding users and groups to the store. The only requirement for Primavera Analytics is that you create a group in the LDAP store called **p6rdbusers** and assign each administrator that will run the Primavera Analytics web application to this group. Primavera Analytics has been certified in the Red Stack using Oracle Internet Directory (OID) Authentication provider.

For details on WebLogic LDAP providers and how to configure the server, see http://docs.oracle.com/cd/E17904_01/web.1111/e13707/atn.htm#i1216261.

Identity Assertion Providers (Single Sign-on):

WebLogic supports configuration of a SSO provider. Refer to your SSO documentation for information on configuration and administration. Primavera Analytics has been certified using Oracle Internet Directory (OID) in coordination with Oracle Access Manager (OAM).

For details on configuring OAM with WebLogic, see http://docs.oracle.com/cd/E15523_01/core.1111/e10043/osso.htm#CHDGCACF.

Copyright

Oracle Primavera Analytics Security Guide for On-Premises

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information on content, products and services from third-parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.