Symantec Data Loss Prevention Oracle 12c Standard Edition 2 Installation and Upgrade Guide

Version 15.7

Last updated: April 6, 2021

Table of Contents

About this guide

About updates to the Symantec Data Loss Prevention Oracle 12c Standard Edition 2 Installation and Upgrade Guide

This guide is occasionally updated as new information becomes available.

The following table provides the history of updates to this version of the *Symantec Data Loss Prevention Oracle 12c Standard Edition 2 Release 2 Installation and Upgrade Guide*:

Table 1: History of updates to the *Symantec Data Loss Prevention Oracle 12c Standard Edition 2 Release 2 Installation and Upgrade Guide*

Date	Description
6 April 2021	Corrected the sequence for configuring the database connection (on both Windows and Linux platforms).
13 February 2020	Added the following updates to procedures applicable to Windows and Linux systems: • Updated steps for converting LOB tables from BasicFiles LOB storage to SecureFiles LOB storage to apply to previous Symantec Data Loss Prevention versions. • Clarified the files you must download when using Oracle 12.1.0.2 and 12.2.0.1. • Corrected the steps for installing the Oracle database software to account for single-tenant and multitenant response file locations. • Corrected the procedure for creating the Symantec Data Loss Prevention database. • Updated the sequence for post-database install verification.

Preparing to install or upgrade to Oracle 12c Standard Edition 2

About installing or upgrading to Oracle 12c Standard Edition 2

You can use the following Oracle 12c Standard edition 2 versions with Symantec Data Loss Prevention 14.6 through 15.7 for new installations and upgrades:

- Oracle 12c Standard Edition 2 Release 1 (12.1.0.2 [12c SE2])
- Oracle 12c Standard Edition 2 Release 2 (12.2.0.1 [12c SE2 R2])

You can download the Symantec-licensed version of Oracle 12c SE2 R2 and use this guide to install or upgrade based on your Symantec Data Loss Prevention implementation.

If you implement a three-tier installation, you must install the Oracle Client (SQL*Plus and Database Utilities) on the Enforce Server. Installation of the Oracle Client enables database communications between the Oracle database server and the Enforce Server. The Symantec Data Loss Prevention installer needs SQL*Plus to create tables and views on the Enforce Server. For this reason, the Windows or Linux user account that is used to install Symantec Data Loss Prevention needs access to SQL*Plus. For full details on how to install the Oracle 12c Database Client software, see the platform-specific documentation from Oracle Corporation, available from the Oracle Documentation Library at https://docs.oracle.com/database/122/nav/install-and-upgrade.htm.

NOTE

After you create the Symantec Data Loss Prevention database and complete the Symantec Data Loss Prevention installation, you can change the database password. To change the database password, you use the Symantec Data Loss Prevention DBPasswordChanger utility. For more information about the Symantec Data Loss Prevention DBPasswordChanger utility, see the Symantec Data Loss Prevention Help Center.

About the Oracle multitenant environment

Symantec Data Loss Prevention supports the Oracle multitenant Containerized Database (CDB)/Pluggable Database (PDB). Symantec Data Loss Prevention version 15.1 is the first version that supports CDB/PDB.

NOTE

The steps in this guidecontent assume you are using a CDB that contains a single PDB.

About deploying Oracle to Amazon Web Services (AWS)

You can deploy the Oracle database server or Oracle RDS on Amazon Web Services (AWS). You do not have to modify the servers or perform any special configurations to deploy the Oracle database Server on AWS. For deploying Oracle RDS on AWS, you must configure TLS as described in About securing communications between the Enforce Server and Amazon RDS for Oracle.

See the *Symantec Data Loss Prevention Deployment Guide for Amazon Web Services* available at Related Documents on the Tech Docs Portal.

Before upgrading to Oracle 12c Standard Edition 2

Prepare for the Oracle upgrade by completing the following

1. Back up the Oracle database.

   If the upgrade fails you can use a backup to restore your system. See the *Symantec Symantec Data Loss Prevention System Maintenance Guide* at Related Documents for instructions on backing up your database, available here:

2. Convert your Large Object (LOB) tables from BasicFiles LOB storage to SecureFiles LOB storage:

   Converting your LOB tables from BasicFiles LOB storage to SecureFiles LOB storage

3. Use the Update Readiness Tool to verify that your database is ready to upgrade. See the Symantec Data Loss Prevention Help Center.

4. Upgrade to Symantec Data Loss Prevention version 15.0, 15.1, 15.5, or 15.7.

   See the *Symantec Data Loss Prevention Upgrade Guide* available at Related Documents.

5. Download Oracle 12.2.0.1 or 12.1.0.2 database files.

   Downloading the Oracle 12c SE2 Release 2 software

Converting your LOB tables from BasicFiles LOB storage to SecureFiles LOB storage

This solution applies to all supported databases and requires that you shut down the system during the conversion process.

Unlike BasicFiles LOB storage, SecureFiles LOB storage tracks deleted LOBs and makes that space available after the retention period expires. After converting to SecureFiles LOB storage, you do not need to run a script to reclaim LOB space in your database. Space reclamation is handled automatically.

If you are using an Oracle 12c Standard database that still includes BasicFiles LOB storage tables, you should convert them as soon as possible to take advantage of the improved functionality of the SecureFiles LOB storage format. You must convert your tables to SecureFiles format before running the Upgrade Readiness Tool when upgrading to the next release of Symantec Data Loss Prevention.

You can manually convert your Oracle 12c LOB tables from BasicFiles to SecureFiles using the following procedure:

1. Back up the Oracle database before making any changes.

2. Shut down all DLP services on your Enforce Server.

   Refer to the topics "Starting and stopping services on Linux" and "About starting and stopping services on Windows" in the *Symantec Data Loss Prevention Administration Guide*.

3. On the Oracle server, stop the Oracle Listener service. This will prevent external connections to the database that may interfere with the export/import process. The remaining steps will need to be executed on the Oracle server directly.

4. Estimate that there is enough space on the database hard drive for the SecureFiles export by running the following queries:

   ```
   expdp protect/<protect password> NOLOGFILE=YES ESTIMATE_ONLY=YES TABLES='MESSAGELOB'

   expdp protect/<protect password> NOLOGFILE=YES ESTIMATE_ONLY=YES TABLES='MESSAGECOMPONENTLOB'

   expdp protect/<protect password> NOLOGFILE=YES ESTIMATE_ONLY=YES TABLES='CONDITIONVIOLATIONLOB'
   ```

   Use the estimates that the queries provide to confirm whether there is sufficient space on the database hard drive. If there is enough space, proceed to step 5.

5. Export the MESSAGELOB, MESSAGECOMPONENTLOB, and CONDITIONVIOLATIONLOB database tables to the data pump directory by running the following queries:

   ```
   expdp protect/<protect password> dumpfile=protect_messagelob.dmp logfile=protect_messagelob.log
   directory=DATA_PUMP_DIR tables='MESSAGELOB'
   ```

```
expdp protect/<protect password> dumpfile=protect_messagecom.dmp logfile=protect_messagecom.log
directory=DATA_PUMP_DIR tables='MESSAGECOMPONENTLOB'

expdp protect/<protect password> dumpfile=protect_cvlob.dmp logfile=protect_cvlob.log
directory=DATA_PUMP_DIR tables='CONDITIONVIOLATIONLOB'
```

6. Verify that the tables appear in the data pump directory by runnig the following command:

```
select DIRECTORY_NAME, DIRECTORY_PATH from dba_directories where DIRECTORY_NAME =
'DATA_PUMP_DIR';
```

7. Import the tables from the data pump directory by running the following commands:

```
impdp protect/<protect password> dumpfile=protect_messagelob.dmp
logfile=protect_import_message.log directory=DATA_PUMP_DIR table_exists_action=REPLACE
transform=LOB_STORAGE:SECUREFILE

impdp protect/<protect password> dumpfile=protect_messagecom.dmp
logfile=protect_import_messagecom.log directory=DATA_PUMP_DIR table_exists_action=REPLACE
transform=LOB_STORAGE:SECUREFILE

impdp protect/<protect password> dumpfile=protect_cvlob.dmp logfile=protect_import_cv.log
directory=DATA_PUMP_DIR table_exists_action=REPLACE transform=LOB_STORAGE:SECUREFILE
```

8. Run the following query to verify that the tables are in SecureFiles LOB storage format:

```
select table_name, securefile from user_lobs where table_name like '%LOB%';
```

   The query returns **yes** in the **securefile column** to indicate that the tables are in SecureFiles LOB storage format.

9. Restart the Oracle Listener service on the Oracle server.
10. Restart all DLP services on your Enforce Server.

Downloading the Oracle 12c SE2 Release 2 software

You should have received a Symantec Serial Number Certificate with your order that lists a serial number for each of your products. If you did not receive the certificate, contact Symantec Support as described at https://support.symantec.com/en_US/contact-support.html.

Go to http://fileconnect.symantec.com/ and enter the serial number. Proceed to the list of available downloads and download and extract the appropriate files. Oracle 12c SE2 files to download lists the files you download.

NOTE

Because you must license the Oracle 12.1.0.2 product files, you must download them from the Oracle website. However, you must download the installation tools file from MySymantec.

Table 2: Oracle 12c SE2 files to download

File name	Description
12.1.0.2 • For Windows, `12.1.0.2_64_bit_Installation_Tools.zip` • For Linux, `12.1.0.2_64_bit_Installation_Tools.tar.gz` **12.2.0.1** • For Windows, `12.2.0.1_64_bit_Installation_Tools.zip` • For Linux, `12.2.0.1_64_bit_Installation_Tools.tar.gz`	These ZIP files contain the Oracle database template, the database user SQL script, and the response (`.rsp`) files, which you use during the installation and configuration of Oracle 12.1.0.2 or 12.2.0.1. If you are using Symantec Data Loss Prevention 14.6 or 15.0, you can download these ZIP files at MySymantec directly. If you are using Symantec Data Loss Prevention 15.1, 15.5, or 15.7, these files are available in the Platform ZIP files: • Windows: `Symantec_DLP_15.7_Platform_Win-IN.zip` • Linux: `Symantec_DLP_15.7_Platform_Lin-IN.zip`
12.2.0.1 • For Windows, `Oracle_12.2.0.1.0_Server_Win64_1of2.zip` and `Oracle_12.2.0.1.0_Server_Win64_2of2.zip` • For Linux, `Oracle_12.2.0.1.0_Server_Lin64_1of2.zip` and `Oracle_12.2.0.1.0_Server_Lin64_2of2.zip`	This ZIP file contains the Oracle server.
For Windows, `Oracle_12.2.0.1.0_Client_Win64.zip` For Linux, `Oracle_12.2.0.1.0_Client_Lin64.zip`	This ZIP file contains the Oracle Client (SQL*Plus and Database Utilities). If you implement a three-tier installation, you must install the Oracle Client on the Enforce Server. About installing or upgrading to Oracle 12c Standard Edition 2

Setting privileges for the Oracle user

You must set privileges for the Oracle user if you are currently running Symantec Data Loss Prevention 15.1 through 15.7. Set privileges before you upgrade Symantec Data Loss Prevention.

1. Stop all Symantec Data Loss Prevention services.
2. Grant select on v_$version to protect; (or to your schema user) as sysdba user by running the following command:

```
sqlplus sys/<password> as sysdba

Grant select on v_$version to protect;
```

Preparing the upgrade software

If you are updating to 15.1 or later, install the new version where the existing version is running. See the *Symantec Data Loss Prevention Upgrade Guide* available at Related Documents.

Follow the upgrade path for your hardware profile

Upgrade paths are unique depending on your database server hardware. Oracle version and hardware profile describes scenarios you may be using and provides links where you can go for steps to upgrade to Oracle Database 12c Standard Edition 2.

Table 3: Oracle version and hardware profile

Oracle version	Hardware profile	More information
Oracle 11g SE1 or Oracle 11g SE	Servers with two (or fewer) CPU sockets	Upgrading from Oracle 11g SE1 or Oracle 11g SE on servers with two (or fewer) CPU sockets
Oracle 11g SE	Servers with more than two CPU sockets on a two-tier installation	Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a two-tier installation
	Servers with more than two CPU sockets on a single-tier installation	Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation

Installing Oracle 12c Standard Edition 2 on Windows

Steps to install Oracle 12c Release 1 Standard Edition 2 on Windows

Preparing to install Oracle 12c Release 1 Standard Edition 2 on Windows provides the Oracle 12c installation process. You can find additional detail for each step of the process as indicated in the table.

Table 4: Steps to install Oracle 12c Release 1 Standard Edition 2 on Windows

Step	Action	More information
1	Install Oracle 12c Release 1 Standard Edition 2.	Preparing to install Oracle 12c Release 1 Standard Edition 2 on Windows
2	Create the Symantec Data Loss Prevention database.	Creating the Symantec Data Loss Prevention database on Windows
3	Complete additional steps to verify the database if you are installing the database to a multitenant environment.	Verifying the CDB and PDB database (on Windows)
4	Create the database listener.	Creating the TNS Listener on Windows
5	Configure the local net service name.	Configuring the local net service name on Windows
6	Complete the following steps if you are installing the database to a multitenant environment: • Verify that the PDB listener is created and registered. • Set the protect PDB to autostart. • Add tablespaces to the PDB database.	Verifying that the PDB listener is created and registered (on Windows) Setting the protect PDB to autostart (on Windows) Adding required tablespaces to the PDB database (on Windows)
7	Create the Symantec Data Loss Prevention database user.	Creating the Oracle user account for Symantec Data Loss Prevention (Windows)
8	Verify the Oracle database.	Verifying the Symantec Data Loss Prevention database

Preparing to install Oracle 12c Release 1 Standard Edition 2 on Windows

The Enforce Server uses the Oracle thin driver and the Oracle Instant Client (for three-tier deployments). Symantec Data Loss Prevention packages the JAR files for the Oracle thin driver with the Symantec Data Loss Prevention software.

You must install the Oracle Instant Client using the `Admin` option if you implement a three-tier system. The Symantec Data Loss Prevention installer needs SQL*Plus to create tables and views on the Enforce Server. Therefore, the Windows user account that is used to install Symantec Data Loss Prevention must be able to access SQL*Plus.

About installing or upgrading to Oracle 12c Standard Edition 2

NOTE

Before starting the installation process, confirm that the Windows host name does not contain invalid characters (for example, underscores [_].) Using invalid characters causes the Oracle installation to fail. If the Windows host name contains invalid characters, go to **Control Panel** > **System** and change the host name. Restart the computer for the new host name to take effect.

Installing Oracle 12c SE2 on Windows

To install Oracle 12c SE2 on Windows

1. Shut down the following services if they are running in Windows Services:
   - All Oracle services: OracleService<ServiceID>, Oracle<HOME_NAME>TNSListener
   - Distributed Transaction Coordinator service

   To view the services go to **Start > Control Panel > Administrative Tools > Computer Management**, and then expand **Services and Applications** and click **Services**.
2. Extract the two ZIP files containing your Oracle 12c SE2 software into a single temporary directory: `C:\temp\Oracle`. The contents of both extracted `database` directories should be in the temporary directory `C:\temp\Oracle\database`.
3. Extract the ZIP file for your Oracle 12c SE2 version into the temporary directory `C:\temp\Oracle\tools`.
   - `12.1.0.2_64_bit_Installation_Tools.zip` for Oracle 12c SE2 Release 1
   - `12.2.0.1_64_bit_Installation_Tools.zip` for Oracle 12c SE2 Release 2
4. Use the command prompt to navigate to the temporary directory where you extracted the Oracle 12c files.
5. Run one of following commands based on your Oracle version and tenant type.

   The command includes the paths to the temporary directories where you extracted the zip files in steps Extract the two ZIP files containing your Oracle 12c SE2 software into a single temporary directory: C:\temp\Oracle. The contents of both extracted database directories should be in the temporary directory C:\temp\Oracle\database. and Extract the ZIP file for your Oracle 12c SE2 version into the temporary directory C:\temp\Oracle\tools. 12.1.0.2_64_bit_Installation_Tools.zip for Oracle 12c SE2 Release 112.2.0.1_64_bit_Installation_Tools.zip for Oracle 12c SE2 Release 2:

   NOTE

   Line breaks added for legibility.

   - Oracle 12c SE2 Release 1
     - Run the following command for a single-tenant database installation:

       ```
       C:\temp\Oracle\database\setup.exe -noconfig -responsefile
       C:\temp\Oracle\tools\responsefiles\singleinstance\
       \Oracle_12.1.0.2_Standard_Edition_Installation_WIN.rsp
       ```

     - Run the following command for a multitenant database installation:

       ```
       C:\temp\Oracle\database\setup.exe -noconfig -responsefile
       C:\temp\Oracle\tools\responsefiles\mutitenant
       \Oracle_12.1.0.2_Standard_Edition_Installation_CDB_WIN.rsp
       ```

   - Oracle 12c SE2 Release 2
     - Run the following command for a single-tenant database installation:

       ```
       C:\temp\Oracle\database\setup.exe -noconfig -responsefile
       C:\temp\Oracle\tools\responsefiles
       \Oracle_12.2.0.1_Standard_Edition_Installation_WIN.rsp
       ```

     - Run the following command for a multitenant database installation:

       ```
       C:\temp\Oracle\database\setup.exe -noconfig -responsefile
       C:\temp\Oracle\tools\responsefiles\mutitenant
       \Oracle_12.2.0.1_Standard_Edition_Installation_CDB_WIN.rsp
       ```

   The installation wizard appears with pre-selected values drawn from the installation response file. You can confirm these values and click through the panels without needing to enter information where noted.

6. On the **Configure Security Updates** panel, **I wish to receive security updates via My Oracle Support** is selected. Click **Next**.
7. On the **Select Installation Options** panel, **Install database software only** is selected. Click **Next**.
8. On the **Grid Installation Options** panel, **Single instance database installation** is selected. Click **Next**.
9. On the **Select Database Edition** panel, **Standard Edition** is selected. Click **Next**.
10. On the **Oracle Home User** panel, enter a user name and password for the Oracle Home User. The default name for the Oracle Home User is Oracle.

    NOTE

    The Oracle Home User is the Windows user account that runs Windows services for `%ORACLE_HOME`. It is not the Symantec Data Loss Prevention Oracle user account.

    Confirm the password, then click **Next**.
11. On the **Specify Installation Location** panel, the **Oracle Base** and **Software Location** paths fields are populated. Click **Next**.

    Oracle Base: `c:\oracle`

    Software Location: `c:\oracle\product\[version]\db_1`

    Replace [version] with the Oracle 12c version you are running (either 12.1.0.2 or 12.2.0.1).
12. On the **Prerequisite Check** panel, click **Next** to begin the prerequisite check process.
13. On the **Summary** panel, click **Install** to begin the installation.

    The installer application installs the Oracle 12c software to your computer. This process may take several minutes to complete.
14. On the **Finish** panel, click **Close** to exit the installer application. You can safely ignore the configuration note that displays on this panel.

For Symantec Data Loss Prevention installation on Linux systems, follow this procedure to install Oracle 12c SE2 Release 1 or SE2 Release 2.

Creating the Symantec Data Loss Prevention database on Windows

Follow this procedure to create the Symantec Data Loss Prevention database on Windows systems.

To create the Symantec Data Loss Prevention database on Windows

1. Set the ORACLE_HOME environment variable for your new installation. Open a command prompt, and enter:

   ```
   set ORACLE_HOME=c:\oracle\product\[version]\db_1
   ```

   Replace [version] with the Oracle 12c version you are running (either 12.1.0.2 or 12.2.0.1). If you installed Oracle 12c to a different location, substitute the correct directory in this command.
2. Navigate to the `C:\temp\Oracle\tools` folder where you extracted the `12.2.0.1_64_bit_Installation_Tools.zip` file.
3. Copy one of the following database files based on your database environment from the `C:\temp\Oracle\tools` folder to the `c:\oracle\product\12.2.0.1\db_1\assistants\dbca\templates` folder:
   - Single tenant database: `Oracle_12.2.0.1_Template_for_64_bit_WIN.dbt`
   - Multi tenant database: `Oracle_12.2.0.1_Template_for_64_bit_PDB_WIN.dbt`

4. (Optional) Rename the `OraDb12c_home1` section of the **Windows Start** menu item to the Oracle 12c version you are running.
5. Open a command prompt, and execute one of the following commands based on your Oracle version and tenant type.

   NOTE

   Line breaks added for legibility.

   - Oracle 12.1.0.2:
     - Run the following command for a single tenant environment:
       ```
       %ORACLE_HOME%\bin\dbca
       -ProgressOnly
       -responseFile C:\temp\Oracle\tools\responsefiles\Oracle_12.2.0.1_DBCA_WIN.rsp
       ```
     - Run the following command for a multitenant environment:
       ```
       %ORACLE_HOME%\bin\dbca
       -ProgressOnly
       -responseFile C:\temp\Oracle\tools\responsefiles\multitenant
       \Oracle_12.2.0.1_DBCA_PDB_WIN.rsp
       ```
   - For Oracle 12.2.0.1:
     - Run the following command for a single tenant environment:
       ```
       %ORACLE_HOME%\bin\dbca
       -createDatabase
       -ProgressOnly
       -responseFile C:\temp\Oracle\tools\responsefiles\Oracle_12.2.0.1_DBCA_WIN.rsp
       ```
     - Run the following command for a multitenant environment:
       ```
       %ORACLE_HOME%\bin\dbca
       -createDatabase
       -ProgressOnly
       -responseFile C:\temp\Oracle\tools\responsefiles\multitenant
       \Oracle_12.2.0.1_DBCA_PDB_WIN.rsp
       ```
6. Enter the SYS password when you are prompted.
7. Create the SYSTEM password when you are prompted.

   Follow these guidelines to create acceptable passwords:

   - Passwords cannot contain more than 30 characters.
   - Passwords cannot contain double quotation marks, commas, or backslashes.
   - Avoid using the `&` character.
   - Passwords are case-sensitive by default. You can change the case sensitivity through an Oracle configuration setting.
   - If your password uses special characters other than `_`, `#`, or `$`, or if your password begins with a number, you must enclose the password in double quotes when you configure it.

   The progress of the Symantec Data Loss Prevention database creation displays on the terminal window.
8. Enter the Oracle Home User password when you are prompted.

   The password you enter here is the same password you created in step On the Oracle Home User panel, enter a user name and password for the Oracle Home User. The default name for the Oracle Home User is Oracle. The Oracle Home User is the Windows user account that runs Windows services for %ORACLE_HOME. It is not the Symantec Data Loss Prevention Oracle user account. Confirm the password, then click Next. of Installing Oracle 12c SE2 on Windows.

9. If the database services OracleServicePROTECT and Distributed Transaction Coordinator are down, start them using Windows Services: **Start > Control Panel > Administrative Tools > Computer Management > Services and Applications > Services**.

Verifying and PDB database for RAC on Windows

After you complete the CDB and PDB database installation on RAC, you verify components of the installation. Specifically, you confirm that the CDB name is dlpcdb and that the PDB name is protect.

You complete these steps only if you are setting up a PDB environment or a PDB running in a RAC environment.

1. Confirm the CON_NAME by running the following command.

```
sqlplus sys/<password> as sysdba
show con_name
```

   The command output should display a message similar to the following message:

```
CON_NAME
------------------------------
CDB$ROOT
```

2. Confirm the PDBS name by running the following command:

```
show pdbs
```

   The command output should display a message similar to the following message:

```
CON_ID CON_NAME   OPEN MODE RESTRICTED
---------- ------------------------------ ---------- ----------
2 PDB$SEED                READ ONLY NO
3 PROTECT                  READ WRITE NO
```

Creating the TNS Listener on Windows

Perform the following procedure to create a TNS listener for the Symantec Data Loss Prevention database.

Before you create the TNS listener, confirm that the local host name can be resolved using the DNS server name or a hosts file. If no DNS server resolution exists, the Net Configuration Assistant (NETCA) does not start. If you use a host file (at `\windows\system32\drivers\etc\hosts`), it must contain IP-address-to-host-name mappings that point to the DNS server name. Add two entries to the `/hosts` file, one that resolves the static IP and one that resolves the local host IP. For example, use the following:

```
[IP address or DNS] myhost.mydomain.com myhost
127.0.0.1 myhost.mydomain.com myhost
```

Replace myhost with the actual host name.

To create the TNS Listener

1. (Optional) If you logged on as a domain user, you must set the `sqlnet.ora` file `SQLNET.AUTHENTICATION_SERVICES=()` value to `none`. Otherwise, proceed to step 2.

   To set the `sqlnet.ora` file `SQLNET.AUTHENTICATION_SERVICES=()` value, perform the following steps in this order:

   - Open `sqlnet.ora`, located in the `%Oracle_Home%\network\admin` folder (for example, `c:\oracle\product\12.2.0.1\db_1\NETWORK\ADMIN`), using a text editor.
   - Change the `SQLNET.AUTHENTICATION_SERVICES=(NTS)` value to `none`:
     `SQLNET.AUTHENTICATION_SERVICES=(none)`
   - Save and close the `sqlnet.ora` file.

2. Start the Oracle Net Configuration Assistant by running the following command:

   ```
   %ORACLE_HOME%/BIN/NETCA
   ```

   This command assumes that you set the Oracle HOME and PATH to the following:

   set ORACLE_HOME=c:\oracle\product\[version]\db_1

   ```
   set PATH=%ORACLE_HOME%\bin:%PATH%
   ```

   Replace [version] with the Oracle 12c version you are running (either 12.1.0.2 or 12.2.0.1).
3. On the **Welcome** panel, select **Listener configuration** and click **Next**.
4. On the **Listener Configuration, Listener** panel, select **Add** and click **Next**.
5. On the **Listener Configuration, Listener Name** panel, select the default listener name, **LISTENER**, unless you must use a different name. Enter the password for your Oracle Home User, then click **Next**.
6. On the **Listener Configuration, Select Protocols** panel, select the **TCP** protocol and click **Next**.
7. On the **Listener Configuration, TCP/IP Protocol** panel, select **Use the standard port number of 1521** and click **Next**.
8. On the **Listener Configuration, More Listeners?** panel, select **No** and click **Next**.
9. On the **Listener Configuration Done** panel, click **Next**.
10. Configure the Local Net Service Name in the **Oracle Net Configuration Assistant**.

    Configuring the local net service name on Windows

    NOTE

    You must click **Finish** to exit the **Oracle Net Configuration Assistant** before continuing with this procedure.
11. On the computer that runs your Oracle database, open a command prompt. The command window must run as Administrator. (See your Microsoft Windows documentation.)
12. Run the following command:

    ```
    lsnrctl stop
    ```
13. Open the following file in a text editor:

    ```
    %ORACLE_HOME%\network\admin\listener.ora
    ```
14. Locate the following line:

    (ADDRESS = (PROTOCOL = IPC)(KEY = <key_value>))
15. Change key_value to `PROTECT`.
16. Add the following line to the end of the file:

    ```
    SECURE_REGISTER_LISTENER = (IPC)
    ```
17. Add the following lines for a multitenant database:

    - For 12.1.0.2:

      ```
      SID_LIST_LISTENER =
        (SID_LIST =
          (SID_DESC =
            (SID_NAME = CLRExtProc)
            (ORACLE_HOME = c:\oracle\product\12.1.0.2\db_1)
            (PROGRAM = extproc)
            (ENVS = "EXTPROC_DLLS=ONLY:c:\oracle\product\12.1.0.2\db_1\bin\oraclr12.dll")
      ```

```
        )
        (SID_DESC =
          (GLOBAL_DBNAME = DLPCDB)
          (SID_NAME = DLPCDB)
          (ORACLE_HOME = c:\oracle\product\12.1.0.2\db_1)
        )
        (SID_DESC =
          (GLOBAL_DBNAME = PROTECT)
          (SID_NAME = DLPCDB)
          (ORACLE_HOME = c:\oracle\product\12.1.0.2\db_1)
        )
      )
```

- For 12.2.0.1:

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = CLRExtProc)
      (ORACLE_HOME = c:\oracle\product\12.2.0.1\db_1)
      (PROGRAM = extproc)
      (ENVS = "EXTPROC_DLLS=ONLY:c:\oracle\product\12.2.0.1\db_1\bin\oraclr12.dll")
    )
    (SID_DESC =
      (GLOBAL_DBNAME = DLPCDB)
      (SID_NAME = DLPCDB)
      (ORACLE_HOME = c:\oracle\product\12.2.0.1\db_1)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = PROTECT)
      (SID_NAME = DLPCDB)
      (ORACLE_HOME = c:\oracle\product\12.2.0.1\db_1)
    )
  )
```

18. Save the file and exit the text editor.
19. Run the following command:

    ```
    lsnrctl start
    ```

20. Run the following commands to connect to the database using SQL Plus:

    ```
    sqlplus /nolog
    ```

    conn sys/<password> as sysdba

21. Run the following command:

    ```
    ALTER SYSTEM SET local_listener =
      '(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=PROTECT)))' SCOPE=both;
    ```

22. Run the following command to register the listener:

    ```
    ALTER SYSTEM REGISTER;
    ```

23. Exit SQL Plus by running the following command:

    ```
    exit
    ```

24. Run the following command to verify the change (if you are running a single tenant environment):

    ```
    lsnrctl services
    ```

    The command output should display a message similar to the following:

    ```
    Services Summary...
    Service "protect" has 1 instance(s).
    Instance "protect", status READY, has 1 handler(s) for this service...
        Handler(s):
         "DEDICATED" established:0 refused:0 state:ready
          LOCAL SERVER
    The command completed successfully
    ```

If you are upgrading from an older Oracle version, continue the upgrade process by referring to one of the following topics that apply to your configuration:

- Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation
- Upgrading from Oracle 11g SE1 or Oracle 11g SE on servers with two (or fewer) CPU sockets

Configuring the TNS Listener and Net Service Name

After you install the Oracle database, configure the TNS Listener and the Net Service Name.

1. (Optional) If you logged on as a domain user, you must set the `sqlnet.ora` file `SQLNET.AUTHENTICATION_SERVICES=()` value to `none`. Otherwise, proceed to step 2.

   To set the `sqlnet.ora` file `SQLNET.AUTHENTICATION_SERVICES=()` value, perform the following steps:

   a) Open `sqlnet.ora`, located in the `%ORACLE_HOME%\network\admin` folder, using a text editor.

   b) Change the `SQLNET.AUTHENTICATION_SERVICES=(NTS)` value to `none`.

      ```
      SQLNET.AUTHENTICATION_SERVICES=(none)
      ```

   c) Save and close the `sqlnet.ora` file.

2. Start the Oracle Net Configuration Assistant by running the following command:

   ```
   %ORACLE_HOME%\bin\netca
   ```

3. Create the TNS Listener.

   Refer to Table 5: Creating the TNS Listener for information on what to select and enter on each screen of the Database Configuration Assistant.

Table 5: Creating the TNS Listener

Screen	Action
Welcome	Select **Listener configuration** and click **Next**.
Listener Configuration, Listener	Select **Add** and click **Next**.
Listener Configuration, Listener Name	Enter a listener name and the password for your Oracle Home User, then click **Next**. **Note:** Use the default listener name, **LISTENER**, unless you must use a different name.
Listener Configuration, Select Protocols	Select the **TCP** protocol and click **Next**.
Listener Configuration, TCP/IP Protocol	Select **Use the standard port number of 1521** and click **Next**.

Screen	Action
Listener Configuration, More Listeners?	Select **No** and click **Next**.
Listener Configuration Done	Click **Next** and select **Local Net Service Name configuration**.

4. Configure the Net Service Name.

   Refer to Table 6: Configuring the Net Service Name for information on what to enter on each screen of the Database Configuration Assistant.

Table 6: Configuring the Net Service Name

Screen	Action
Net Service Name Configuration	Select **Add** and click **Next**.
Net Service Name Configuration, Service Name	Enter `protect` in the **Service Name** field and click **Next**.
Net Service Name Configuration, Select Protocols	Select **TCP** and click **Next**.
Net Service Name Configuration, TCP/IP Protocol	1. Enter the host name of the Oracle server computer in the **Host name** field. 2. Select **Use the standard port number of 1521** (the default value). 3. Click **Next**.
Net Service Name Configuration, Test	Select **No, do not test** and click **Next**. **Note:** Do not test the service configuration, because the listener has not yet started.
Net Service Name Configuration, Net Service Name	Select accept the default name of "protect" and click **Next**.
Net Service Name Configuration, Another Net Service Name?	Select **No** and click **Next**.
Net Service Name Configuration Done	Click **Next** and click **Finish**.

Verifying tnsnames.ora contents

Before you create the required Oracle user accounts, verify that the `tnsnames.ora` file contains entries for the `protect` database that you created.

1. Using a text editor, open the `tnsnames.ora` file, which is located in the `$ORACLE_HOME/network/admin` directory.
2. Verify that the following lines are present in the file:

```
PROTECT =
  (DESCRIPTION =
    (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = <ip_address>)(PORT = <port_number>))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = protect)
    )
  )
```

   If these lines do not exist, add them to the file, replacing *<ip_address>* and *<port_number>* with the correct values for your system.

NOTE

Do not copy and paste information to the `tnsnames.ora` file. Pasting can introduce hidden characters that cannot be parsed.

3. Add the following lines if you are installing a multitenant database, replacing *<host_name>* with the correct value for your system:

```
DLPCDB =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = <hostname>)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = DLPCDB)
    )
  )

PROTECT =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = <hostname>)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = PROTECT)
    )
  )
```

4. Save the `tnsnames.ora` file and exit the text editor.

Verifying that the PDB listener is created and registered on Windows

If you are running the database in a multitenant environment, verify that the PDB listener is created and registered.

1. Verify that you can log in to and exit SQL*Plus.
2. Run the following command using SQL*Plus to verify PDB accessibility:

```
sqlplus sys/<password>@protect as sysdba
```

3. Run the following commands to confirm that the PDB service is accessible:
   a) `sqlplus sys/<password> as sysdba`
   b) `show parameter service`

      The command output should display a message similar to the following message:

```
NAME TYPE VALUE
------------------------- ------- ------------------------------
service_names string dlpcdb
```

   c) `show parameter local_listener`

      The command output should display a message similar to the following message:

```
NAME TYPE VALUE
------------------------ ----------- ------------------------------
local_listener string (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=protect)))
```

   d) Run the following command to format output:

```
COLUMN SERVICE_ID FORMAT 9
COLUMN NAME FORMAT A20
```

```
COLUMN PDB FORMAT A20
```

e) `select service_id, name, pdb from v$services;`

Confirm that protect is listed in the output.

```
SERVICE_ID     NAME                           PDB
------------------ ------------------------------ ------------------------------
 1                    SYS$BACKGROUND   CDB$ROOT
 2                    SYS$USERS               CDB$ROOT
 3                    dlpcdbXDB               CDB$ROOT
 4                    dlpcdb                     CDB$ROOT
 7                    protect                     PROTECT
```

NOTE

The SERVICE_ID number may differ from those listed on your system.

4. Confirm the active services that are running under cdb$root by running the following command:

```
alter session set container=cdb$root;

select name from v$active_services;
```

The command output should display a message similar to the following message:

```
NAME
----------------------------------------------------------------
dlpcdb
SYS$BACKGROUND
SYS$USERS
protect
dlpcdbXDB
```

Confirm that the dlpcdb and protect services are listed in the output.

5. Run the following commands if the protect service is missing from the output in the preceding step:
   a) Run the following command:

```
Alter session set container=protect;
exec dbms_service.CREATE_SERVICE('PROTECT', 'PROTECT');
exec dbms_service.START_SERVICE(SERVICE_NAME=>'PROTECT');
```

   b) Run the following command to register the listener:

```
ALTER SYSTEM REGISTER;
```

   c) Exit SQL*Plus by running the following command:

```
exit
```

6. Restart the listener by running the following commands:

```
lsnrctl stop
lsnrctl start
```

The command output should display a message similar to the following message:

```
Service "DLPCDB" has 1 instance(s).
  Instance "dlpcdb", status READY, has 1 handler(s) for this service...
Service "PROTECT" has 1 instance(s).
  Instance "dlpcdb", status READY , has 1 handler(s) for this service...
```

7. Confirm that the PDB service is accessible by running the following commands:
   a) `sqlplus sys/<password>@protect as sysdba`
   b) `show con_name`

      to return the following message:

      ```
      CON_NAME
      ------------------------------
      PROTECT
      ```

   c) `show pdbs`

      to return the following message:

      ```
      CON_ID CON_NAME OPEN MODE RESTRICTED
      ---------- ------------------------------ ---------- ----------
      3  PROTECT READ WRITE NO
      ```

8. Run the following command if the `show pdbs` command returns *PROTECT* listed without **Read Write**:

   ```
   select inst_id, con_id, name, open_mode from gv$pdbs where name='PROTECT';
   ```

9. Exit SQL*Plus by running the following command:

   ```
   exit
   ```

Setting the protect PDB to autostart on Windows

If you are running the database in a multitenant environment, configure the protect PDB to auto start when the Oracle database restarts. You can set the PDB to auto start by saving the state of the PDB when it is open.

1. Open a command prompt as the Oracle user.
2. Start SQL*Plus by running the following command:

   ```
   sqlplus sys/<password> as sysdba
   ```

   where *<password>* is the SYS password.
3. Run the following commands:

   ```
   alter pluggable database protect open;
   ```

   ```
   alter pluggable database protect save state;
   ```

4. Exit SQL*Plus by running the following command:

   ```
   exit
   ```

Adding required tablespaces to the PDB database on Windows

If you are running the database in a multitenant environment, add tablespaces to the PDB database.

1. Navigate to the `C:\temp\Oracle\tools` folder.
2. Start SQL*Plus and run the `add_pdb_tablespace_WIN.sql` script.

   ```
   sqlplus /nolog @add_pdb_tablespace_WIN.sql
   ```

3. At the **Please enter the password for sys user** prompt, enter the password for the SYS user.
4. At the **Please enter Service Name** prompt, enter `protect`.
5. Confirm that all required tablespaces are added for the PDB by running the following command:

   ```
   sqlplus sys/<password>@protect as sysdba
   ```

```
SELECT tablespace_name FROM dba_tablespaces;
```

For example, if you are using Oracle 19.3.0.0, the output information should read:

```
TABLESPACE_NAME
------------------------------
SYSTEM
SYSAUX
UNDOTBS1
TEMP
USERS
LOB_TABLESPACE
```

6. Confirm the summary of tablespaces and that the data file paths are consistent by completing the following steps:
   a) Start SQL*Plus by running the following command:

```
sqlplus sys/<password>@protect as sysdba
```

   b) Run the following query:

```
COLUMN Tablespace_Name FORMAT A20
COLUMN File_Name FORMAT A50
COLUMN Size_Mb FORMAT 9999
SELECT substr(tablespace_name,1,20) as Tablespace_Name,
       substr(file_name,1,50) as File_Name,
       bytes/1024/1024 as Size_MB
FROM dba_data_files
union
SELECT 'TEMP' as Tablespace_Name,
       name as File_Name,
       bytes/1024/1024 as Size_MB
FROM v$tempfile;
```

   Confirm that the data file paths are consistently located in the same location under the `PROTECT` folder. For example, if you are using Oracle 19c, the output information should read:

```
TABLESPACE_NAME      FILE_NAME                                          SIZE_MB
-----------------  --------------------------------------------------    ----
LOB_TABLESPACE     C:\ORACLE\ORADATA\DLPCDB\PROTECT\LOB01.DBF            2048
LOB_TABLESPACE     C:\ORACLE\ORADATA\DLPCDB\PROTECT\LOB02.DBF            1024
LOB_TABLESPACE     C:\ORACLE\ORADATA\DLPCDB\PROTECT\LOB03.DBF            1024
SYSAUX             C:\ORACLE\ORADATA\DLPCDB\PROTECT\SYSAUX01.DBF         150
SYSTEM             C:\ORACLE\ORADATA\DLPCDB\PROTECT\SYSTEM01.DBF         169
TEMP               C:\ORACLE\ORADATA\DLPCDB\PROTECT\TEMP01.DBF           2048
UNDOTBS1           C:\ORACLE\ORADATA\DLPCDB\PROTECT\UNDOTBS01.DBF        2048
USERS              C:\ORACLE\ORADATA\DLPCDB\PROTECT\USERS01.DBF          2048
```

```
USERS                C:\ORACLE\ORADATA\DLPCDB\PROTECT\USERS02.DBF          2048

USERS                C:\ORACLE\ORADATA\DLPCDB\PROTECT\USERS03.DBF          2048
```

Creating the Oracle user account for Symantec Data Loss Prevention (Windows)

Perform the following procedure to create an Oracle user account and name it “protect.”

To create the new Oracle user account named "protect"

1. Navigate to the `C:\temp\Oracle\tools` folder.
2. Start SQL*Plus:

   ```
   sqlplus /nolog
   ```

3. Run the `oracle_create_user.sql` script:

   ```
   SQL> @oracle_create_user.sql
   ```

4. At the **Please enter the password for sys user** prompt, enter the password for the SYS user.
5. At the **Please enter Service Name** prompt, enter `protect`.
6. At the **Please enter required username to be created** prompt, enter `protect` for the user name.
7. At the **Please enter a password for the new username** prompt, enter a new password.

   Follow these guidelines to create acceptable passwords:

   - Passwords cannot contain more than 30 characters.
   - Passwords cannot contain double quotation marks, commas, or backslashes.
   - Avoid using the `&` character.
   - Passwords are case-sensitive by default. You can change the case sensitivity through an Oracle configuration setting.
   - If your password uses special characters other than `_`, `#`, or `$`, or if your password begins with a number, you must enclose the password in double quotes when you configure it.

   Store the password in a secure location for future use. You must use this password to install Symantec Data Loss Prevention. If you need to change the password after you install Symantec Data Loss Prevention, see the Symantec Data Loss Prevention Help Center.

8. Confirm that tablespaces are available for the Oracle user you created by running the following commands in the listed order:

   - `sqlplus protect/protect@protect`
   - `SQL> SELECT tablespace_name FROM user_tablespaces;`

     The command returns the following:

     ```
     TABLESPACE_NAME
     ------------------------------
     SYSTEM
     SYSAUX
     UNDOTBS1
     TEMP
     USERS
     DRSYS
     LOB_TABLESPACE
     ```

Verifying the Symantec Data Loss Prevention database

After you create the Symantec Data Loss Prevention database, verify that it was created correctly.

To verify that the database was created correctly

1. Open a new command prompt and start SQL*Plus:

   ```
   sqlplus /nolog
   ```

2. Log on as the SYS user:

   ```
   SQL> connect sys/password@protect as sysdba
   ```

   Where password represents the SYS password.

3. Run the following query:

   ```
   SQL> SELECT * FROM v$version;
   ```

4. Make sure that the output from the query contains the following information, which identifies the software components as version 12.2.0.1.

   The output information should read:

   ```
   BANNER
   --------------------------------------------------------------------------------

   Oracle Database 12c Release 12.2.0.1.0 - 64-bit Production
   PL/SQL Release 12.2.0.1.0 - Production
   CORE    12.2.0.1.0      Production
   TNS for 64-bit Windows: Version 12.2.0.1.0 - Production
   NLSRTL Version 12.2.0.1.0 - Production
   ```

5. Exit SQL*Plus:

   ```
   SQL> exit
   ```

Installing Oracle 12c Standard Edition 2 on Linux

Steps to install Oracle 12c Release 1 Standard Edition 2 on Linux

Table 7: Oracle 12c Release 1 Standard Edition 2 installation steps provides the Oracle 12c installation process. You can find additional detail for each step of the process as indicated in the table.

Table 7: Oracle 12c Release 1 Standard Edition 2 installation steps

Step	Action	More information
1	Perform the preinstallation steps.	Preparing the Linux environment
2	Install Oracle 12c Release 1 Standard Edition 2.	Preparing to install Oracle 12c Release 1 Standard Edition 2 on Linux
3	Create the Symantec Data Loss Prevention database.	Creating the Symantec Data Loss Prevention database on Linux
4	Set the protect PDB to autostart if you are installing the database to a multitenant environment.	Verifying the PDB database on Linux
5	Verify the Complete additional steps to verify the database if you are installing the database to a multitenant environment.	Verifying the PDB database on Linux
6	Create the database listener.	Creating the TNS Listener on Linux
7	Verify tnsnames.ora contents.	Verifying tnsnames.ora contents
8	Configure the local net service name.	Configuring the local net service name on Linux
9	Complete the following steps if you are installing the database to a multitenant environment: • Verify that the PDB listener is created and registered. • Add tablespaces to the PDB database. • Set the protect PDB to autostart.	Verifying that the PDB listener is created and registered on Linux Adding required tablespaces to the PDB database on Linux Setting the protect PDB to autostart on Linux
10	Verify the Oracle database.	Verifying the Symantec Data Loss Prevention database
11	Create the Symantec Data Loss Prevention database user.	Creating the Oracle user account for Symantec Data Loss Prevention (Linux)
12	Configure your system to start Oracle when the server computer boots.	Configuring automatic startup and shutdown of the database

Performing the Linux preinstallation steps

Perform the following procedure to prepare your Linux environment for installation. The preinstallation requires Python. You can use any Python version 2.4.6 through 3.6.3.

Preparing the Linux environment

Follow this procedure to prepare the Linux environment.

To prepare the Linux environment

1. Log on as the root user.
2. Locate and copy the ZIP file for your Oracle 12c SE2 version to the Linux server:
   - `12.1.0.2_64_bit_Installation_Tools.tar.gz` for Oracle 12c SE2 Release 1
   - `12.2.0.1_64_bit_Installation_Tools.tar.gz` for Oracle 12c SE2 Release 2
3. Extract the file contents into the temporary directory (`/tmp`). For example, to extract the Oracle 12.2.0.1 files:

   ```
   tar xvfz 12.2.0.1_64_bit_Installation_Tools.tar.gz -C /tmp
   ```

   Extracting creates a subdirectory named `oracle_install` in the `/tmp` directory and extracts the files into that subdirectory.
4. In the `oracle_install` directory, run the Oracle preparation script:

   ```
   cd /tmp/oracle_install
   ./scripts/oracle_prepare.sh
   ```
5. Enter the Oracle User password when prompted.
6. After the preparation script has run to completion, switch to the `tmp/oracle_install/scripts` directory and run the verification script:

   ```
   cd /tmp/oracle_install/scripts
   ./oracle_verify.py
   ```

   The verification script displays settings (such as RAM, swap space, shared memory, `/tmp` disc space) that do not meet the requirements for Oracle. Adjust any settings to the required values.

   If you have mismatched values between kernel parameters and resource limits, run the `oracle_config_kernel_parameters.py` script in the `/tmp/oracle_install/scripts` directory. This script sets the kernel parameters to the required settings.
7. Restart the server so that the updated kernel parameters take effect.
8. Verify that there is enough space under `/var`. For a small to medium enterprise, `/var` should have at least 15 GB. For a large enterprise, `/var` should have at least 30 GB. For a very large enterprise, `/var` should have at least 45 GB of free space. As your organization's traffic expands, these figures should increase, and you must allocate more free space.
9. Verify that the `/opt` and `/boot` file systems have the required free space for your Symantec Data Loss Prevention installation. See the Symantec Data Loss Prevention Help Center for more information.

Preparing to install Oracle 12c Release 1 Standard Edition 2 on Linux

The Enforce Server uses the Oracle thin driver and the Oracle Instant Client. Symantec Data Loss Prevention packages the JAR files for the Oracle thin driver with the Symantec Data Loss Prevention software.

You must install the Oracle Instant Client using the `Admin` option if you implement a three-tier system. The Symantec Data Loss Prevention installer needs SQL*Plus to create tables and views on the Enforce Server. Therefore, the Linux user account that is used to install Symantec Data Loss Prevention must be able to access to SQL*Plus.

About installing or upgrading to Oracle 12c Standard Edition 2

The instructions in this section assume that you are logged on locally to the Linux server and running the X Window System. It also assumes that you have the `xorg-x11-apps.x86_64 package` installed. If you connect to the server remotely, you need a terminal emulator. You also need to set the location where the GUI tools can display their output; you use the export display command to do that. For example:

```
export DISPLAY=ip_address:display_number
```

NOTE

Refer to the configuration information in the X server management program for the IP address and display number. Typically, the display number is 0.

As you run the GUI tools later, you might get a response similar to the following:

```
X connection to localhost:10.0 broken (explicit kill or server shutdown)
```

Run the export display command again.

Installing Oracle 12c SE2 on Linux systems

For Symantec Data Loss Prevention installation on Linux systems, follow this procedure to install Oracle 12c SE2 Release 1 or SE2 Release 2.

To install Oracle SE on Linux systems

1. Log in to the terminal as the root user, then execute the following command:

   ```
   su -l root
   xhost +SI:localuser:oracle
   ```

2. Switch to the Oracle user terminal.
3. Copy the required software installation file or files to `/home/oracle`.
4. From `/home/oracle`, unzip the ZIP files you copied. You must run the `unzip` command as the Oracle user. If you run it as the root user, then the Oracle user is not able to view the extracted files unless you change the permissions. However, changing the permissions is not advisable from a security standpoint.
5. Put the contents of the **database** directory from the ZIP file you extracted to `/home/oracle` into a directory titled **database**. You should now have a directory named `/home/oracle/database`.
6. Change directory to:

   ```
   cd /home/oracle/database/stage/cvu/cv/admin
   ```

7. Back up the `cvu_config` file using this command:

   ```
   cp cvu_config backup_cvu_config
   ```

8. Edit the original `cvu_config` file as follows:

   Set `CV_ASSUME_DISTID=OEL6` if you are using Linux 6.x

   Set `CV_ASSUME_DISTID=OEL7` if you are using Linux 7.x

   Save the edited `cvu_config` file.

9. Navigate to the `/tmp/oracle_install` directory where you extracted the file for your Oracle 12c SE2 version:
   - `12.1.0.2_64_bit_Installation_Tools.tar.gz` for Oracle 12c SE2 Release 1
   - `12.2.0.1_64_bit_Installation_Tools.tar.gz` for Oracle 12c SE2 Release 2
10. Copy the response files based on your Oracle 12c SE2 version:
    - Oracle 12c SE2 Release 1
      - Single tenant: `Oracle_12.1.0.2_DBCA_Linux.rsp` and `Oracle_12.1.0.2_Standard_Edition_Installation_Linux.rsp` from `/tmp/oracle_install/responsefiles` to a temporary folder such as `/home/oracle/oracle_install/responsefiles`.
      - Multi tenant: `Oracle_12.1.0.2_DBCA_PDB_Linux.rsp` and `Oracle_12.1.0.2_Standard_Edition_CDB_Installation_Linux.rsp` to a temporary folder such as `/home/oracle/oracle_install/responsefiles`.

- Oracle 12c SE2 Release 2
  - Single tenant: `Oracle_12.2.0.1_DBCA_Linux.rsp` and `Oracle_12.2.0.1_Standard_Edition_Installation_Linux.rsp` from `/tmp/oracle_install/responsefiles` to a temporary folder such as `/home/oracle/oracle_install/responsefiles`.
  - Multi tenant: `Oracle_12.2.0.1_DBCA_PDB_Linux.rsp` and `Oracle_12.2.0.1_Standard_Edition_CDB_Installation_Linux.rsp` to a temporary folder such as `/home/oracle/oracle_install/responsefiles`.

11. Provide read and write access to the `/opt` directory for the Oracle user.
12. In the Oracle user terminal execute the following command based on your Oracle 12c SE2 version:

    NOTE

    Line breaks added for legibility.

    - Oracle 12c SE2 Release 1
      - Run the following command for a single tenant database installation:
        ```
        /home/oracle/database/runInstaller -noconfig -responseFile
        /home/oracle/oracle_install/responsefiles
        /Oracle_12.1.0.2_Standard_Edition_Installation_Linux.rsp
        ```
      - Run the following command for a multitenant database installation:
        ```
        /home/oracle/database/runInstaller -noconfig -responseFile
        /home/oracle/oracle_install/responsefiles/
        mutitenant/Oracle_12.2.0.1_Standard_Edition_CDB_Installation_Linux.rsp
        ```
    - Oracle 12c SE2 Release 2
      - Run the following command for a single tenant database installation:
        ```
        /home/oracle/database/runInstaller -noconfig -responseFile
        /home/oracle/oracle_install/responsefiles
        /Oracle_12.2.0.1_Standard_Edition_Installation_Linux.rsp
        ```
      - Run the following command for a multitenant database installation:
        ```
        /home/oracle/database/runInstaller -noconfig -responseFile
        /home/oracle/oracle_install/responsefiles/
        mutitenant/Oracle_12.1.0.2_Standard_Edition_CDB_Installation_Linux.rsp
        ```
13. On the **Configure Security Updates** panel, **I wish to receive security updates via My Oracle Support** is selected. Click **Next**.
14. Click **Yes** to confirm that you have not provided an email address.
15. On the **Select Installation Option** panel, **Install database software only** is selected. Click **Next**.
16. On the **Grid Installation Options** panel, **Single instance database installation** is selected. Click **Next**.
17. On the **Select Database Edition** panel, **Standard Edition** is selected. Click **Next**.
18. On the **Specify Installation Location** panel, enter the following paths are specified. Click **Next**:
    - **Oracle Base**: `/opt/oracle`
    - **Software Location**: `/opt/oracle/product/[version]/db_1`
      Replace [version] with the Oracle 12c version you are running (either 12.1.0.2 or 12.2.0.1).

19. If this is the first Oracle installation on the server computer, the installer application displays the **Create Inventory** panel. The inventory path is entered as `/opt/oracle/oraInventory` and the group name is entered as `oinstall`. Click **Next**.

    The installer may display a warning message that you placed the central inventory location inside of the Oracle base directory. You can safely ignore this message for Symantec Data Loss Prevention database installations.

20. On the **Privileged Operating System Groups** panel, click **Next** to grant the Database Administrator and Database Operator privileges to the default DBA group.

    The installer application performs a prerequisite check and displays the results. Click the **Fix and Check Again** button to correct any warnings or installation errors.

21. On the **Summary** panel, click **Install** to begin the installation.

    The installer application installs the Oracle 12c software on your computer.

22. The installer displays the **Execute Configuration scripts** window, which instructs you to execute two scripts as the root user. From the root xterm window, run the following two scripts:

    ```
    /opt/oracle/oraInventory/orainstRoot.sh
    /opt/oracle/product/[version]/db_1/root.sh
    ```

    Replace [version] with the Oracle version.

    After you run the script, you are prompted to enter the full pathname to the local binary directory. Accept the default `/usr/local/bin` directory and press **Enter**. Enter `Y` if the script asks for confirmation to overwrite the following files: `dbhome`, `oraenv` and `coraenv`.

    The script displays `Finished product-specific root actions` when it is finished.

23. Enter Y after the script displays `Do you want to setup oracle trace analyzer`. Entering Y finishes the process to create the Oracle trace analyzer.

24. Return to the **Execute Configuration scripts** screen and click **OK**.

25. On the **Finish** panel, click **Close** to exit the installer application. You can safely ignore the configuration note displayed on this panel.

Creating the Symantec Data Loss Prevention database on Linux

Follow this procedure to create the Symantec Data Loss Prevention database on a Linux system.

To create the Symantec Data Loss Prevention database on Linux systems

1. Set the ORACLE_HOME and ORACLE_SID environment variables for your new installation. Open a command prompt as the `Oracle` user and enter:

    ```
    export ORACLE_HOME=/opt/oracle/product/[version]/db_1
    export ORACLE_SID=protect
    ```

    Replace [version] with the Oracle version, either 12.2.0.1 or 12.1.0.2.

    If you installed Oracle 12c into a different location, substitute the correct directory in this command.

    You may want to add these commands to your user profile configuration so that the ORACLE_HOME and ORACLE_SID environment variables are defined each time you log on. See your Linux documentation for details about setting environment variables.

2. Navigate to `/tmp/oracle_install` where you extracted the `12.1.0.2_64_bit_Installation_Tools.tar.gz` or the `12.2.0.1_64_bit_Installation_Tools.tar.gz` file.
3. Copy one of the following database template files corresponding with your Oracle version to the `$ORACLE_HOME/assistants/dbca/templates` directory:
   - Single tenant database: `Oracle_[version]_Template_for_64_bit_Linux.dbt` from `/tmp/oracle_install/templates/singleinstance`
   - Multi tenant database: `Oracle_[version]_Template_for_64_bit_PDB_Linux.dbt` from `/tmp/oracle_install/templates/multitenant`
     Replace [version] with your Oracle version.
4. At the command prompt, execute one of the following commands based on your Oracle version and tenant type.

   NOTE

   Line breaks added for legibility.

   - Oracle 12.1.0.2:
     – Run the following command for a single tenant environment:
     ```
     $ORACLE_HOME/bin/dbca
     -progressOnly
     -responseFile /home/oracle/oracle_install/responsefiles
     /Oracle_12.1.0.2_DBCA_Linux.rsp
     ```
     – Run the following command for a multitenant environment:
     ```
     $ORACLE_HOME/bin/dbca
     -progressOnly
     -responseFile /home/oracle/oracle_install/responsefiles
     /Oracle_12.1.0.2_DBCA_Linux.rsp
     ```
   - For Oracle 12.2.0.1:
     – Run the following command for a single tenant environment:
     ```
     $ORACLE_HOME/bin/dbca
     -createDatabase
     -progressOnly
     -responseFile /home/oracle/oracle_install/responsefiles
     /Oracle_12.2.0.1_DBCA_Linux.rsp
     ```
     – Run the following command for a multitenant environment:
     ```
     $ORACLE_HOME/bin/dbca
     -createDatabase
     -progressOnly
     -responseFile /home/oracle/oracle_install/responsefiles/multitenant
     /Oracle_12.2.0.2_DBCA_PDB_Linux.rsp
     ```
5. Enter the SYS password when you are prompted.
6. Create the SYSTEM password when you are prompted.

   Follow these guidelines to create acceptable passwords:

- Passwords cannot contain more than 30 characters.
- Passwords cannot contain double quotation marks, commas, or backslashes.
- Avoid using the `&` character
- Passwords are case-sensitive by default. You can change the case sensitivity through an Oracle configuration setting.
- If your password uses special characters other than `_`, `#`, or `$`, or if your password begins with a number, you must enclose the password in double quotes when you configure it.

7. If you are creating the database in a multitenant environment, a dialog displays that prompts you to enter the PDBAdmin user and password. Enter the user account and password you used when you created the PDB.
8. The progress of the Symantec Data Loss Prevention database creation displays on the terminal window.

Verifying the PDB database on Linux

After you complete the CDB and PDB database installation on RAC, you verify components of the installation. Specifically, you confirm that the CDB name is dlpcdb and that the PDB name is protect.

1. Set environment variables by running the following command:

```
export ORACLE_HOME=/opt/oracle/product/19.3.0.0/db_1
export ORACLE_SID=dlpcdb
```

2. Open a command prompt as the Oracle user and start SQL*Plus:

```
sqlplus sys/<password> as sysdba
```

3. Confirm the CON_NAME by running the following command.

```
show con_name
```

The command output should display a message similar to the following message:

```
CON_NAME
------------------------------
CDB$ROOT
```

4. Confirm the PDBS name by running the following command:

```
show pdbs
```

The command output should display a message similar to the following message:

```
CON_ID CON_NAME   OPEN MODE RESTRICTED
---------- ------------------------------ ---------- ----------
2 PDB$SEED                READ ONLY NO
3 PROTECT                 READ WRITE NO
```

Creating the TNS Listener on Linux

Perform the following procedure to create a TNS listener for the Symantec Data Loss Prevention database.

> **NOTE**
>
> To use the the commands referenced in this procedure, ensure that your working directory is `$ORACLE_HOME/bin`. If SQL*Plus does not work while following this procedure, set your `$PATH` variable to point to `$ORACLE_HOME/bin`.

Before you create the TNS listener, confirm that the local host name can be resolved using the DNS server name or a hosts file. If no DNS server resolution exists, the Net Configuration Assistant (NETCA) does not start. If you use a host

file (at `/etc/hosts`), it must contain IP-address-to-host-name mappings that point to the DNS server name. Add two entries to the `/etc/hosts` file, one that resolves the static IP and one that resolves the local host IP. For example, use the following:

```
[IP address or DNS] myhost.mydomain.com myhost
127.0.0.1 myhost.mydomain.com myhost
```

Replace myhost with the actual host name.

To create the TNS Listener

1. As the `Oracle` user, confirm that the following environment variables are set before starting **Oracle Net Configuration Assistant**:

   Set Oracle HOME to the following:

   - export ORACLE_HOME= /opt/oracle/product/[version]/db_1
     Replace [version] with the Oracle 12c version you are running (either 12.1.0.2 or 12.2.0.1).
   - Set the Oracle database service name to the following:
     `export ORACLE_SID=protect`
   - Set the path to the following
     `export PATH=$ORACLE_HOME\bin:$PATH$`

2. As the `Oracle` user, start the **Oracle Net Configuration Assistant**:

   ```
   $ORACLE_HOME/bin/netca
   ```

3. On the **Welcome** panel, select **Listener configuration** and click **Next**.
4. On the **Listener Configuration, Listener** panel, select **Add** and click **Next**.
5. On the **Listener Configuration, Listener Name** panel, enter a listener name and click **Next**.

   NOTE

   Use the default listener name, **LISTENER**, unless you must use a different name.

6. On the **Listener Configuration, Select Protocols** panel, select the **TCP** protocol and click **Next**.
7. On the **Listener Configuration, TCP/IP Protocol** panel, select **Use the standard port number of 1521** and click **Next**.
8. On the **Listener Configuration, More Listeners?** panel, select **No** and click **Next**.
9. On the **Listener Configuration Done** panel, click **Next**.
10. Configure the Local Net Service Name in the **Oracle Net Configuration Assistant**.

    Configuring the local net service name on Linux

    NOTE

    You must click **Finish** to exit the **Oracle Net Configuration Assistant** before continuing with this procedure.

11. Log on to the Oracle host computer as the Oracle user.

    ```
    su - oracle
    ```

12. Confirm that the ORACLE_SID is set to the following:

    ```
    export ORACLE_SID=protect
    ```

13. Run the following command:

    ```
    lsnrctl stop
    ```

14. Open the following file in a text editor:

    ```
    $ORACLE_HOME/network/admin/listener.ora
    ```

15. Locate the following line:

    (ADDRESS = (PROTOCOL = IPC)(KEY = <key_value>))

16. Change key_value to `PROTECT`.

17. Add the following line to the end of the file:

    ```
    SECURE_REGISTER_LISTENER = (IPC)
    ```

18. Save the file and exit the text editor.

19. Run the following command as a root user:

    ```
    lsnrctl start
    ```

20. Run the following commands to connect to the database using SQL*Plus:

    ```
    sqlplus /nolog
    ```

    conn sys/<password> as sysdba

21. Run the following command:

    ```
    ALTER SYSTEM SET local_listener =
      '(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=PROTECT)))' SCOPE=both;
    ```

22. Run the following command to register the listener:

    ```
    ALTER SYSTEM REGISTER;
    ```

23. Exit SQL Plus by running the following command:

    ```
    exit
    ```

24. Run the following command to verify the change (if you are running a single tenant environment):

    ```
    lsnrctl services
    ```

    The command output should display a message similar to the following:

    ```
    Services Summary...
    Service "protect" has 1 instance(s).
    Instance "protect", status READY, has 1 handler(s) for this service...
        Handler(s):
         "DEDICATED" established:0 refused:0 state:ready
          LOCAL SERVER
    The command completed successfully
    ```

If you are upgrading from an older Oracle version, continue the upgrade process by referring to one of the following topics that apply to your configuration:

- Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation
- Upgrading from Oracle 11g SE1 or Oracle 11g SE on servers with two (or fewer) CPU sockets

Configuring the local net service name on Linux

Perform the following procedure to configure the Local Net Service Name for the Symantec Data Loss Prevention database.

To configure the local net service name

1. If the Oracle Net Configuration Assistant is not already running, log on as the `Oracle` user and start it:

   ```
   $ORACLE_HOME/bin/netca
   ```

2. On the **Welcome** panel, select **Local Net Service Name configuration** and click **Next**.
3. On the **Net Service Name Configuration** panel, select **Add** and click **Next**.
4. On the **Net Service Name Configuration, Service Name** panel, enter "protect" in the **Service Name** field and click **Next**.
5. On the **Net Service Name Configuration, Select Protocols** panel, select **TCP** and click **Next**.
6. On the **Net Service Name Configuration, TCP/IP Protocol** panel:
   - Enter the IP address of the Oracle server computer in the **Host name** field.
   - Select **Use the standard port number of 1521** (the default value).
   - Click **Next**.
7. On the **Net Service Name Configuration, Test** panel, select **No, do not test** and click **Next**.

   Do not test the service configuration, because the listener has not yet started.
8. On the **Net Service Name Configuration, Net Service Name** panel, select accept the default name of "protect" and click **Next**.
9. On the **Net Service Name Configuration, Another Net Service Name?** panel, select **No** and click **Next**.
10. On the **Net Service Name Configuration Done** panel, select **Next**.
11. Click **Finish** to exit the **Oracle Net Configuration Assistant**.

Verifying tnsnames.ora contents

Before you create the required Oracle user accounts, verify that the `tnsnames.ora` file contains entries for the `protect` database that you created.

If you are preparing the database for a multitenant environment, you modify the `tnsnames.ora` file contents.

1. Using a text editor, open the `tnsnames.ora` file, which is located in the `$ORACLE_HOME/network/admin` directory.
2. Verify that the following lines are present in the file:

   ```
   PROTECT =
     (DESCRIPTION =
       (ADDRESS_LIST =
       (ADDRESS = (PROTOCOL = TCP)(HOST = host_name)(PORT = port_number))
       )
       (CONNECT_DATA =
         (SERVICE_NAME = protect)
       )
     )
   ```

   If these lines do not exist, add them to the file, replacing *`host_name`* and *`port_number`* with the correct values for your system.

   NOTE

   Do not copy and paste information to the `tnsnames.ora` file. Pasting information to the file can introduce hidden characters that cannot be parsed.

3. Add the following lines if you are installing a multitenant database, replacing *<host_name>* with the correct value for your system:

   - ```
     DLPCDB =
       (DESCRIPTION =
         (ADDRESS = (PROTOCOL = TCP)(HOST = <host_name>)(PORT = 1521))
         (CONNECT_DATA =
           (SERVER = DEDICATED)
           (SERVICE_NAME = DLPCDB)
         )
       )
     ```
   - ```
     PROTECT =
       (DESCRIPTION =
         (ADDRESS = (PROTOCOL = TCP)(HOST = <host_name>)(PORT = 1521))
         (CONNECT_DATA =
           (SERVER = DEDICATED)
           (SERVICE_NAME = PROTECT)
         )
       )
     ```

4. Save the `tnsnames.ora` file and exit the text editor.

Verifying that the PDB listener is created and registered on Linux

If you are running the database in a multitenant environment, verify that the PDB listener is created and registered.

1. Run the following command in SQL*Plus to verify PDB accessibility:

   ```
   sqlplus sys/<password>@protect as sysdba
   ```

2. Exit SQL*Plus after you run the command.
3. Run the following commands to confirm that the PDB service is accessible:
   a) `sqlplus sys/<password> as sysdba`
   b) `show parameter service`

      The command output should display a message similar to the following message:

      ```
      NAME TYPE VALUE
      ------------------------- ------- ------------------------------
      service_names string dlpcdb
      ```

   c) `show parameter local_listener`

      The command output should display a message similar to the following message:

      ```
      NAME TYPE VALUE
      ------------------------ ----------- ------------------------------
      local_listener string (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=protect)))
      ```

   d) Run the following command to format the output:

      ```
      COLUMN SERVICE_ID FORMAT 9
      COLUMN NAME FORMAT A20
      COLUMN PDB FORMAT A20
      ```

   e) `select service_id, name, pdb from v$services;`

      The command output should display a message similar to the following message:

      ```
      SERVICE_ID     NAME                           PDB
      ------------------ ------------------------------ ------------------------------
      ```

```
1                    SYS$BACKGROUND   CDB$ROOT
2                    SYS$USERS               CDB$ROOT
3                    dlpcdbXDB               CDB$ROOT
4                    dlpcdb                     CDB$ROOT
7                    protect                     PROTECT
```

NOTE

The SERVICE_ID number may differ from what is listed on your system.

4. Confirm the active services that are running under cdb$root by running the following command:

   ```
   alter session set container=cdb$root;
   ```

   ```
   select name from v$active_services;
   ```

   The command output should display a message similar to the following message:

   ```
   NAME
   ----------------------------------------------------------------
   dlpcdb
   SYS$BACKGROUND
   SYS$USERS
   protect
   dlpcdbXDB
   ```

   Confirm that the dlpcdb and protect services are listed in the output.

5. Complete the following steps if the protect service is missing from the output in step 4:
   a) Run the following commands:

      ```
      Alter session set container=protect;
      ```

      ```
      exec dbms_service.CREATE_SERVICE('PROTECT', 'PROTECT');
      ```

      ```
      exec dbms_service.START_SERVICE(SERVICE_NAME=>'PROTECT');
      ```

      ```
      ALTER SYSTEM REGISTER;
      ```

   b) Restart the listener by running the following command:

      ```
      lsnrctl stop;
      ```

      ```
      lsnrctl start;
      ```

      ```
      lsnrctl status
      ```

      The command output should display a message similar to the following message:

      ```
      Service "DLPCDB" has 1 instance(s).

        Instance "dlpcdb", status READY, has 1 handler(s) for this service...
      Service "PROTECT" has 1 instance(s).

        Instance "dlpcdb", status READY , has 1 handler(s) for this service...
      ```

6. Confirm that the PDB service is accessible by running the following commands:
   a) `sqlplus sys/<password>@protect as sysdba`
   b) `show con_name`

   Returns the following message:

   ```
   CON_NAME
   ------------------------------
   ```

```
PROTECT
```

c) `show pdbs`

Returns the following message:

```
CON_ID CON_NAME OPEN MODE RESTRICTED
---------- ------------------------------ ---------- ----------
3  PROTECT READ WRITE NO
```

7. Run the following command if the `show pdbs` command returns protect listed without **Read Write**:

```
sqlplus sys/<password>@protect as sysdba

select inst_id, con_id, name, open_mode from gv$pdbs where name='PROTECT';
```

Setting the protect PDB to autostart on Linux

If you are running the database in a multitenant environment, configure the protect PDB to autostart when the Oracle database restarts. You can set the PDB to autostart by saving the state of the PDB when it is open.

1. Open a command prompt as the Oracle user.
2. Start SQL*Plus by running the following command:

```
sqlplus sys/<password> as sysdba
```

3. Run the following command:

```
alter pluggable database protect open;

alter pluggable database protect save state;
```

Adding required tablespaces to the PDB database on Linux

If you are running the database in a multitenant environment, add tablespaces to the PDB database.

1. Navigate to the `/home/oracle/oracle_install/scripts` folder.
2. Start SQL*Plus and run the add_pdb_tablespace_WIN.sql script:

```
sqlplus /nolog @add_pdb_tablespace_Linux.sql
```

3. At the **Please enter the password for sys user** prompt, enter the password for the SYS user.
4. At the **Please enter Service Name** prompt, enter `protect`.
5. Confirm that all required tablespaces are added for the PDB by running the following script:

```
sqlplus sys/<password>@protect as sysdba
SELECT tablespace_name FROM dba_tablespaces;
```

For example, if you are using Oracle 19.3.0.0, the output information should read:

```
TABLESPACE_NAME
------------------------------
SYSTEM
SYSAUX
UNDOTBS1
TEMP
USERS
LOB_TABLESPACE
```

6. Confirm the summary of tablespaces and that the data file paths are consistent by running the following steps:
   a) Run the following query:

```
sqlplus sys/<password>@protect as sysdba
```

   b) Run the following commands:

```
COLUMN Tablespace_Name FORMAT A20
COLUMN File_Name FORMAT A50
COLUMN Size_Mb FORMAT 9999
SELECT substr(tablespace_name,1,20) as Tablespace_Name,
       substr(file_name,1,50) as File_Name,
       bytes/1024/1024 as Size_MB
FROM dba_data_files
union
SELECT 'TEMP' as Tablespace_Name,
       name as File_Name,
       bytes/1024/1024 as Size_MB
FROM v$tempfile;
```

   c) Confirm that the data file paths are consistently located in the same location under the `PROTECT` folder. For example, if you are using Oracle 19c, the output information should read:

```
TABLESPACE_NAME      FILE_NAME                                          SIZE_MB

-------------------- -------------------------------------------------- -------

LOB_TABLESPACE       /opt/oracle/oradata/dlpcdb/protect/LOB01.DBF           2048

LOB_TABLESPACE       /opt/oracle/oradata/dlpcdb/protect/LOB02.DBF           1024

LOB_TABLESPACE       /opt/oracle/oradata/dlpcdb/protect/LOB03.DBF           1024

SYSAUX               /opt/oracle/oradata/dlpcdb/protect/SYSAUX01.DBF         150

SYSTEM               /opt/oracle/oradata/dlpcdb/protect/SYSTEM01.DBF         169

TEMP                 /opt/oracle/oradata/dlpcdb/protect/TEMP01.DBF          2048

UNDOTBS1             /opt/oracle/oradata/dlpcdb/protect/UNDOTBS01.DBF       2048

USERS                /opt/oracle/oradata/dlpcdb/protect/USERS01.DBF         2048

USERS                /opt/oracle/oradata/dlpcdb/protect/USERS02.DBF         2048

USERS                /opt/oracle/oradata/dlpcdb/protect/USERS03.DBF         2048
```

Verifying the Symantec Data Loss Prevention database

After you create the Symantec Data Loss Prevention database, verify that it was created correctly.

To verify that the database was created correctly

1. Open a command prompt as the `Oracle` user and start SQL*Plus:

```
$ORACLE_HOME/bin/sqlplus /nolog
```

2. Log on as the SYS user:

   ```
   SQL> connect sys/password@protect as sysdba
   ```

   Where password represents the SYS password.

3. Run the following query:

   ```
   SQL> SELECT * FROM v$version;
   ```

4. Make sure that the output from the query contains the following information, which identifies the correct software components based on the version.

   For example, the output information for version 12.2.0.1 should read:

   ```
   BANNER
   --------------------------------------------------------------------------------

   Oracle Database 12c Release 12.2.0.1.0 - 64bit Production
   PL/SQL Release 12.2.0.1.0 - Production
   CORE    12.2.0.1.0      Production
   TNS for Linux: Version 12.2.0.1.0 - Production
   NLSRTL Version 12.2.0.1.0 - Production
   ```

5. Exit SQL*Plus:

   ```
   SQL> exit
   ```

Creating the Oracle user account for Symantec Data Loss Prevention (Linux)

Perform the following procedure to create an Oracle user account and name it “protect.”

1. Copy the `oracle_create_user.sql` file from `/tmp/oracle_install/scripts` to a local directory.
2. Open a command prompt as the `Oracle` user and go to the directory where you copied the `oracle_create_user.sql` file.
3. Start SQL*Plus:

   ```
   sqlplus /nolog
   ```

4. Run the `oracle_create_user.sql` script:

   ```
   SQL> @oracle_create_user.sql
   ```

5. At the **Please enter the password for sys user** prompt, enter the password for the SYS user.
6. At the **Please enter Service Name** prompt, enter `protect`.
7. At the **Please enter required username to be created** prompt, enter `protect`.
8. At the **Please enter a password for the new username** prompt, enter a new password.

   Follow these guidelines to create acceptable passwords:

- Passwords cannot contain more than 30 characters.
- Passwords cannot contain double quotation marks, commas, or backslashes.
- Avoid using the `&` character.
- Passwords are case-sensitive by default. You can change the case sensitivity through an Oracle configuration setting.
- If your password uses special characters other than `_`, `#`, or `$`, or if your password begins with a number, you must enclose the password in double quotes when you configure it.

Store the password in a secure location for future use. You use this password to install Symantec Data Loss Prevention. If you need to change the password after you install Symantec Data Loss Prevention, see the Symantec Data Loss Prevention Help Center.

9. Confirm that tablespaces are available for the Oracle user you created by running the following commands in the listed order:

   - `sqlplus protect/protect@protect`
   - `SQL> SELECT tablespace_name FROM user_tablespaces;`

     The command returns the following:

     ```
     TABLESPACE_NAME
     ------------------------------
     SYSTEM
     SYSAUX
     UNDOTBS1
     TEMP
     USERS
     DRSYS
     LOB_TABLESPACE
     ```

Configuring automatic startup and shutdown of the database

To configure automatic startup and shutdown of the database, follow this procedure:

1. Switch to the root user.
2. Go to the `oracle_install` directory.

   ```
   cd /home/oracle/oracle_install
   ```

3. Run the `oracle_post.sh` script from the `oracle_install` directory.

   ```
   ./scripts/oracle_post.sh
   ```

4. Verify that the script completed successfully by confirming that the last line of the output is:

   ```
   dbora 0:off 1:off 2:off 3:on 4:on 5:on 6:off
   ```

   You may see errors before the last line (for example, `cannot access /var/log/dbora`). You can ignore these errors.

   Validate that the settings were applied by viewing the file `/etc/oratab` and confirming that **Y** appears in the final line: `protect:/opt/oracle/product/19.3.0.0[version]/db_1:Y`. If *N* appears, change it to `Y` and save your changes.

Upgrading to Oracle 12c Standard Edition 2

Upgrading from Oracle 11g SE1 or Oracle 11g SE on servers with two (or fewer) CPU sockets

Upgrade path from Oracle 11g SE1 or Oracle 11g SE on servers with two (or fewer) CPU sockets describes the steps you complete to upgrade to Oracle 12c SE2 Release 1 or Release 2 from Oracle 11g SE or SE1 on servers with two (or fewer) CPU sockets.

Table 8: Upgrade path from Oracle 11g SE1 or Oracle 11g SE on servers with two (or fewer) CPU sockets

Step	Action	Information
1	Stop all Symantec Data Loss Prevention services on the Enforce Server except the Symantec DLP Update Service.	For information on stopping services, see the *Symantec Data Loss Prevention Upgrade Guide*.
2	Upgrade the database to Oracle 12c SE2 Release 1 or Release 2.	Upgrading to Oracle 12.1.0.2 Upgrading to Oracle 12.2.0.1
3	Re-create the TNS listener.	Creating the TNS Listener on Windows Creating the TNS Listener on Linux
4	Complete the following steps if you are installing the database to a multitenant environment: • Verify that the PDB listener is created and registered. • Set the protect PDB to autostart. • Add tablespaces to the PDB database.	Verifying that the PDB listener is created and registered (on Windows) Setting the protect PDB to autostart (on Windows) Adding required tablespaces to the PDB database (on Windows) Verifying that the PDB listener is created and registered (on Linux) Setting the protect PDB to autostart (on Linux) Adding required tablespaces to the PDB database (on Linux)
5	Re-create the local net services.	Configuring the local net service name on Windows Configuring the local net service name on Linux
5	Connect Symantec Data Loss Prevention to the Oracle 12c SE2 Release 1 or Release 2 database.	Connect Symantec Data Loss Prevention to the Oracle 12c SE2 database
6	Restart all Symantec Data Loss Prevention services.	For information on starting services, see the *Symantec Data Loss Prevention Upgrade Guide* available at Related Documents.

Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a two-tier installation

Upgrade path for Oracle 11g SE on servers with more than two CPU sockets on a two-tier installation describes the steps you complete to upgrade to Oracle 12c SE2 Release 1 or Release 2 from Oracle 11g SE on servers with more than two CPU sockets on a two-tier installation.

It is likely that your license for Oracle 12c SE2 limits CPU sockets to two. You may be running an Oracle 11g database on computers with more than two CPU sockets. If this is the case, you must migrate your database to a new computer with two CPU sockets.

Table 9: Upgrade path for Oracle 11g SE on servers with more than two CPU sockets on a two-tier installation

Step	Action	Information
1	Migrate the database to new hardware with two CPU sockets.	About migrating the Oracle database to supported hardware
2	Stop all Symantec Data Loss Prevention services on the Enforce Server except the Symantec DLP Update Service.	For information on stopping services, see the *Symantec Data Loss Prevention Upgrade Guide*.
3	Install the Oracle 12c SE2 on new two CPU socket hardware.	Steps to install Oracle 12c SE2 on Windows Steps to install Oracle 12c SE2 on Linux
4	Re-create the TNS listener.	Creating the TNS Listener on Windows Creating the TNS Listener on Linux
5	Re-create the local net services.	Configuring the local net service name on Windows Configuring the local net service name on Linux
6	Connect Symantec Data Loss Prevention to the Oracle 12c SE2 database.	Connect Symantec Data Loss Prevention to the Oracle 12c SE2 database
7	Restart all Symantec Data Loss Prevention services.	For information on starting services, see the *Symantec Data Loss Prevention Upgrade Guide*.

Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation

Upgrade path for Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation describes the steps you complete to upgrade to Oracle 12c SE2 from Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation.

It is likely that your license for Oracle 12c SE2 limits CPU sockets to two. You may be running an Oracle 11g database on computers with more than two CPU sockets. If this is the case, you must migrate your database to a new computer with two CPU sockets.

Table 10: Upgrade path for Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation

Step	Action	Information
1	Migrate the database to new hardware with two CPU sockets.	About migrating the Oracle database to supported hardware
2	On your Symantec Data Loss Prevention 14.6 Enforce Server, stop all Symantec Data Loss Prevention services except the Symantec DLP Update Service.	For information on stopping services, see chapter 5 in the *Symantec Data Loss Prevention Upgrade Guide*.
3	Install the Oracle 12c SE2 on new two CPU socket hardware.	Steps to install Oracle 12c SE2 on Windows Steps to install Oracle 12c SE2 on Linux
4	Migrate the Enforce Server to supported hardware.	See the *Symantec Symantec Data Loss Prevention System Maintenance Guide* at Related Documents.

Upgrading to Oracle 12.2.0.1Upgrading to Oracle 19c

You use the following steps to upgrade your Oracle 11g SE1 or Oracle 11g SE database running on servers with two (or fewer) CPU sockets. The following steps include details for both Windows and Linux.

NOTE

If you upgrade the Oracle database on Linux, you must complete prerequisite steps. Preparing the Linux environment

You use the steps in this section to upgrade your Oracle database (either 11g or 12c) to Oracle 19c.

NOTE

If you are installing the database to a Linux environment, you must export display settings. The instructions in this section assume that you are logged on locally to the Linux server and running the X Window System. See Installing the Oracle 19c software on Linux for more details.

1. Obtain and review the Oracle 19c installation software.

   About installing Oracle 19c on Windows

   About installing Oracle 19c on Linux

2. Prepare the database environment.

   Preparing the Windows environment

   Performing the Linux preinstallation steps

3. Install the Oracle 19c software.

   Installing the Oracle 19c software on Windows

   Installing the Oracle 19c software on Linux

4. Obtain the Oracle 12.2.0.1 installation software.

   Go to MySymantec to download the installation media.

5. Install Oracle 12.2.0.1 under the same folder as Oracle 11g. For example:

   `c:\oracle\product\12.2.0.1\db1` for Windows

   `/opt/oracle/product/12.2.0.1/db1` for Linux

   Refer to the install steps for your particular database server OS to install the Oracle 12.2.0.1 database:

   Preparing to install Oracle 12c SE2 on Windows

   Preparing to install Oracle 12c SE2 on Linux

6. Set ORACLE_HOME depending on your database server OS:

   `set ORACLE_HOME=c:\oracle\product\12.2.0.1\db_1` for Windows

   `export ORACLE_HOME=/opt/oracle/product/12.2.0.1/db_1` for Linux

   NOTE

   Clear any errors before starting the Database Upgrade Assistant.

7. Set the database service name variable:

   `set SERVICE_NAME=protect` for Windows

   `export SERVICE_NAME=protect` for Linux

8. Set the display variable if you upgrade on Linux by running the following command:

   export DISPLAY=ip_address:display_number

   Where ip_address is the local host.

9. Start the Database Upgrade Assistant by running the following command:

   `%ORACLE_HOME%/bin/dbua` for Windows

   `$ORACLE_HOME/bin/dbua` for Linux

   If the Database Upgrade Assistant does not launch and an error message displays, complete the following items in order:

   1. Open the command prompt window.
   2. Set ORACLE_HOME depending on your database server OS:
      `set ORACLE_HOME=c:\oracle\product\12.2.0.119.3.0.0\db_1` for Windows
      `export ORACLE_HOME=/opt/oracle/product/12.2.0.119.3.0.0/db1` for Linux
   3. Set the path:
      `set PATH=%PATH%:%ORACLE_HOME%\bin` for Windows
      `export PATH=$PATH:$ORACLE_HOME/bin` for Linux
   4. Restart the Database Upgrade Assistant:
      `%ORACLE_HOME%\bin\dbua` for Windows
      `$ORACLE_HOME/bin/dbua` for Linux

10. Confirm that the OracleServicePROTECT service is running.

    If the service is not running, an error message displays and the upgrade process cannot finish.

11. Refer to the following table for information on what to enter on each screen of the Database Upgrade Assistant.

Screen	Description
Select Database	Enter the sysdba user name and password.
Prerequisite Checks	Resolve any warnings or errors that display. Sometimes, you must drop packages from previous Symantec Data Loss Prevention versions to clear errors. For example, to drop Symantec Data Loss Prevention 14.6 packages, you run the following SQL command: `SQL> drop package UPGRADESCEHEME_PRELOAD_V14_6_0`
Select Upgrade Options	Leave the settings as default.
Select Recover Options	Select **I have my own backup and restore strategy**.
Configure Network	Clear the selected listener that displays on the **Listener Selection** tab. You re-create the listener in a later step. Leave the remaining settings default.
Configure Management	Clear the **Configure Enterprise Manager (EM) database express** selection.
Summary	The **Summary** screen lists the settings that are used during the database upgrade. Click **Finish**.
Progress	The **Progress** screen displays the details about the upgrade. The upgrade can take around 30 minutes to complete.
Results	The **Results** screen displays when the upgrade completes.

12. Re-create the TNS listener.

    Creating the TNS Listener on Windows

    Creating the TNS Listener on Linux

13. Re-create the local net services.

    Configuring the local net service name on Windows

    Configuring the local net service name on Linux

14. Re-create the TNS Listener and Net Service Name.

    Configuring the TNS Listener and Net Service Name for Windows

    Configuring TNS Listener and Net Service Name for Linux

15. Restart Symantec Data Loss Prevention services.

16. Log on to the Enforce Server administration platform.

    If the Enforce Server logon page does not load and instead displays a 'GLOBAL NOT_FOUND' message, restart all Symantec Data Loss Prevention services again.

Upgrading to Oracle 12.1.0.2

You use the following steps to upgrade your Oracle 11g SE1 or Oracle 11g SE database running on servers with two (or fewer) CPU sockets. The following steps include details for both Windows and Linux.

NOTE

If you upgrade the Oracle database on Linux, you must complete prerequisite steps. Preparing the Linux environment

Complete the following steps to upgrade your version to Oracle 12.1.0.2:Complete the following steps to upgrade your version to Oracle 12c Enterprise Release 2 (12.1.0.2):

1. Obtain the Oracle 12.1.0.2 installation software.

   Go to MySymantec to download the installation media.

2. Install Oracle 12.1.0.2 under the same folder as Oracle 11g. For example:

   `c:\oracle\product\12.1.0.2\db1` for Windows

   `/opt/oracle/product/12.1.0.2/db1` for Linux

   Refer to the install steps for your particular database server OS to install the Oracle 12.1.0.2 database:

   Preparing to install Oracle 12c SE2 on Windows

   Preparing to install Oracle 12c SE2 on Linux

3. Set ORACLE_HOME depending on your database server OS:

   `set ORACLE_HOME=c:\oracle\product\12.1.0.2\db_1` for Windows

   `export ORACLE_HOME=/opt/oracle/product/12.1.0.2/db_1` for Linux

   NOTE

   Clear any errors before starting the Database Upgrade Assistant.

4. Set the ORACLE_SID variable:

   `set ORACLE_SID=protect` for Windows

   `export ORACLE_SID=protect` for Linux

5. Set the display variable if you upgrade on Linux by running the following command:

   export DISPLAY=ip_address:display_number

   Where ip_address is the local host.

6. Start the Database Upgrade Assistant by running the following command:

   `%ORACLE_HOME%/bin/dbua` for Windows

`$ORACLE_HOME/bin/dbua` for Linux

If the Database Upgrade Assistant does not launch and an error message displays, complete the following items in order:

- Open the command prompt window.
- Set ORACLE_HOME depending on your database server OS:
  `set ORACLE_HOME=c:\oracle\product\12.1.0.2\db_1` for Windows
  `export ORACLE_HOME= /opt/oracle/product/12.1.0.2/db1` for Linux
- Set the path:
  `set PATH=%PATH%:%ORACLE_HOME%\bin` for Windows
  `export PATH=$PATH:$ORACLE_HOME\bin` for Linux
- Restart the Database Upgrade Assistant:
  `%ORACLE_HOME%/bin/dbua` for Windows
  `$ORACLE_HOME/bin/dbua` for Linux

7. Confirm that the OracleServicePROTECT service is running.

   If the service is not running, an error message displays and the upgrade process cannot finish.

8. Refer to the following table for information on what to enter on each screen of the Database Upgrade Assistant.

Screen	Description
Select Database	Do not enter the sysdba user name and password. You connect to the existing database at a later part of the upgrade.
Prerequisite Checks	Resolve any warnings or errors that display. In some cases, you must drop packages from previous Symantec Data Loss Prevention versions to clear errors. For example, to drop Symantec Data Loss Prevention 14.0 packages, you run the following SQL command: `SQL> drop package UPGRADESCEHEME_PRELOAD_V14_0_0`
Select Upgrade Options	Leave the settings as default.
Select Recover Options	Select **I have my own backup and restore strategy**.
Configure Network	Clear the selected listener that displays on the **Listener Selection** tab. You re-create the listener in a later step. Leave the remaining settings default.
Configure Management	Clear the **Configure Enterprise Manager (EM) database express** selection.
Summary	The **Summary** screen lists the settings that are used during the database upgrade. Click **Finish**.
Progress	The **Progress** screen displays the details about the upgrade. The upgrade can take around 30 minutes to complete.
Results	The **Results** screen displays when the upgrade completes.

9. Re-create the TNS listener.

   Creating the TNS Listener on Windows

   Creating the TNS Listener on Linux

   Creating the TNS Listener on Windows

   Creating the TNS Listener on Linux

10. Re-create the local net services.

    Configuring the local net service name on Windows

Configuring the local net service name on Linux

Configuring the local net service name

Configuring the local net service name

11. Restart Symantec Data Loss Prevention services.
12. Log on to the Enforce Server administration platform.

    If the Enforce Server logon page does not load and instead displays a 'GLOBAL NOT_FOUND' message, restart all Symantec Data Loss Prevention services again.

Migrating the Oracle database

About migrating the Oracle database to supported hardware

Your license requires that you run Oracle 12c SE2 on a two CPU socket system. If you are running an Oracle 11g database on computers with more than two CPU sockets you must migrate your database to a new computer with two CPU sockets running Oracle 12c SE2.

NOTE

If you are migrating to a two-CPU socket system and you use a single-tier or a two-tier Symantec Data Loss Prevention implementation, you must migrate the Enforce Server and detection servers to the two-CPU system. See chapter 6 for Windows or chapter 7 for Linux in the *Symantec Data Loss Prevention System Maintenance Guide* available here:

Related Documents

Workflow for migrating the Oracle database to supported hardware

Steps to migrate the Oracle database to a server with two CPU sockets describes the process to migrate the Oracle 12c SE2 database to a server with two CPU sockets.

These steps assume that you have updated Symantec Data Loss Prevention to version 14.6, 15.0, 15.1, 15.5, or 15.7.

Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a two-tier installation

Upgrading from Oracle 11g SE on servers with more than two CPU sockets on a single-tier installation

NOTE

You can use these steps to run the Oracle 12c SE2 database on the same server as the Enforce Server (a single-tier system).

Table 11: Steps to migrate the Oracle database to a server with two CPU sockets

Step	Action	Description
1	Stop Symantec Data Loss Prevention services on the Enforce Server.	For information on stopping services, see the *Symantec Data Loss Prevention Upgrade Guide* available at Related Documents.
2	Install Oracle 12c SE2 on the two CPU socket system.	Steps to install Oracle 12c SE2 on Windows Steps to install Oracle 12c SE2 on Linux
3	Confirm the database row count.	Confirm the schema row count before the export (Windows) Confirm the schema row count before the export (Linux)
4	Export the existing database.	Exporting a database schema (Windows) Exporting a database schema (Linux)
4	Create the Oracle user account.	Creating the Oracle user account for Symantec Data Loss Prevention (Windows) Creating the Oracle user account for Symantec Data Loss Prevention (Linux)

Step	Action	Description
5	Import the database.	Copy the Oracle `export.dmp` file from the location on the previous server computer to the same locations on the two CPU socket server computer. Importing a database backup schema on Windows Import the database backup schema on Linux
6	Confirm the schema row count in the imported database.	Confirm the schema row count after the import (Windows) Confirm the schema row count after the import (Linux)
7	Create the TNS listener on the Oracle 12c SE2 server.	Creating the TNS Listener on Windows Creating the TNS Listener on Linux
8	Modify the `jdbc.properties` file on the Enforce Server to point to the Oracle 12c SE2 database.	Connect Symantec Data Loss Prevention to the Oracle 12c SE2 database

Confirm the schema row count before the export (Windows)

Confirm the schema row count before you begin the database export. You use the database row count to compare to the count after you complete the export.

To confirm the database object count

1. Run the command:

```
sqlplus protect/<password>@protect
```

2. Run the following command to create a PL\SQL function to generate the row count:

```
SQL>create or replace function

row_count (p_tablename in varchar2)

return number

as

l_count number;

begin execute immediate

'select count(*)

from ' || p_tablename

into l_count;

return l_count;

end;

/
```

3. Run the following query to generate the row count for each table in your schema:

```
SQL>spool rowCount_before_export.txt

SQL>select table_name, row_count(table_name) num_of_rows from user_tables;

SQL>spool off
```

The `rowCount_before_export.txt` file is generated in the execution directory.

4. Save the `rowCount_before_export.txt` file for future use.

Exporting a database schema on Windows

Complete the following steps to migrate the necessary Oracle database schemas from an unsupported Oracle installation on Windows.

NOTE

Before you begin the process to export database schemas, back up the database. For more information, see the *Symantec Data Loss Prevention System Maintenance Guide*, available at Related Documents.

To export the Oracle database on Windows

1. Set the ORACLE_HOME and ORACLE_SID variables using the following commands:

   set ORACLE_HOME=c:\oracle\product\oracle_version\db_1 where oracle_version is the existing Oracle database version. For example, enter `11.2.0.4` if you are running Oracle 11g Release 2.

   ```
   set ORACLE_SID = protect
   ```

2. Log on as the SYS user:

   ```
   sqlplus /nolog
   ```

   SQL> connect sys/`password`@protect as sysdba

   Where `password` represents the SYS password.

3. Run the following command:

   ```
   Select * from dba_directories;
   ```

   Running this command identifies the location of the `DATA_PUMP_DIR` where the `export.dmp` file is created at the end of the database export procedure.

4. Run the following command:

   ```
   Grant read,write on directory DATA_PUMP_DIR to protect;
   ```

5. Run the following command:

   ```
   Grant exp_full_database to protect;
   ```

6. Exit SQL*Plus:

   ```
   SQL> exit
   ```

7. Export the database schema by running the following command from a command prompt:

   ```
   Expdp protect/<protect schema password> DUMPFILE=export.dmp schemas=protect
   DIRECTORY=DATA_PUMP_DIR EXCLUDE=STATISTICS
   ```

8. Verify that the `export.dmp` file is created in the `DATA_PUMP_DIR` location.

Confirm the schema row count before the export (Linux)

Confirm the schema row count before you begin the database export. You use the schema row count to compare to the count after you complete the export.

To confirm the database object count

1. Run the command:

   ```
   sqlplus protect/<password>@protect
   ```

2. Run the following command to create a PL\SQL function to generate the row count:

```
SQL>create or replace function
row_count (p_tablename in varchar2)
return number
as
l_count number;
begin execute immediate
'select count(*)
from ' || p_tablename
into l_count;
return l_count;
end;
/
```

3. Run the following query to generate the row count for each table in your schema:

```
SQL>spool rowCount_before_export.txt
SQL>select table_name, row_count(table_name) num_of_rows from user_tables;
SQL>spool off
```

   The `rowCount_before_export.txt` file is generated in the execution directory.

4. Save the `rowCount_before_export.txt` file for future use.

Exporting a database schema on Linux

Complete the following steps to export the necessary Oracle database schemas from an unsupported Oracle installation on Linux.

NOTE

Before you begin the process to export database schemas, back up the database. For more information, see the *Symantec Data Loss Prevention System Maintenance Guide*, available at Related Documents.

To export the Oracle database on Linux

1. Set the ORACLE_HOME and ORACLE_SID variables using the following commands:

   export ORACLE_HOME=/opt/oracle/product/oracle_version/db_1 where oracle_version is the existing Oracle database version. For example, enter `11.2.0.4` if you are running Oracle 11g Release 2.

```
export ORACLE_SID = protect
```

2. Log on as the SYS user:

```
sqlplus /nolog
```

   SQL> connect sys/`password`@protect as sysdba

   Where `password` represents the SYS password.

3. Run the following command:

```
Select * from dba_directories;
```

Running this command identifies the location of the `DATA_PUMP_DIR` where the `export.dmp` file is created at the end of the database export procedure.

4. Run the following command:

   ```
   Grant read,write on directory DATA_PUMP_DIR to protect;
   ```

5. Run the following command:

   ```
   Grant exp_full_database to protect;
   ```

6. Exit SQL*Plus:

   ```
   SQL> exit
   ```

7. Switch to the Oracle user by running the following command:

   ```
   su - oracle
   ```

8. Export the database schema by running the following command from Terminal:

   ```
   Expdp protect/<protect schema password> DUMPFILE=export.dmp schemas=protect
   DIRECTORY=DATA_PUMP_DIR EXCLUDE=INDEX, STATISTICS
   ```

9. Verify that the `export.dmp` file is created in the `DATA_PUMP_DIR` location.

Importing a database backup schema on Windows

Complete the following steps on the two CPU socket Windows server computer.

Import the database backup schema to Windows

1. Set the ORACLE_HOME and ORACLE_SID variables as follows:

   ```
   set ORACLE_HOME=c:\oracle\product\12.2.0.1\db_1
   ```

   ```
   set ORACLE_SID = protect
   ```

2. Log on as the SYS user:

   ```
   sqlplus /nolog
   ```

   SQL> connect sys/`<password>`@protect as sysdba

   Where `password` represents the SYS password.

3. Run the following command:

   ```
   Grant read,write on directory DATA_PUMP_DIR to protect;
   ```

4. Run the following command:

   Grant imp_full_database to protect;

5. Exit SQL*Plus:

   ```
   SQL> exit
   ```

6. Import the database schema by running the following command from a command prompt:

   Impdp 'sys/`<password>` as sysdba' DUMPFILE=export.dmp schemas=protect DIRECTORY=DATA_PUMP_DIR

7. Regenerate statistics and rebuild indexes after the import completes.

8. Verify that the import process was successful by running the following command:

   sqlplus protect/<`<password>`>@protect

   Verify that data is present in the schema.

Confirm the schema row count after the import

After you import the database schema, you generate a row count of each table in the schema. You compare the data that you generate with the data you generated before the schema export.

1. Run the following command:

   ```
   sqlplus protect/<password>@protect
   ```

2. Run the following command to create a PL\SQL function to generate the row count:

   ```
   SQL>create or replace function

   row_count (p_tablename in varchar2)

   return number

   as

   l_count number;

   begin

   execute immediate

   'select count(*)

   from ' || p_tablename

   into l_count;

   return l_count;

   end;

   /
   ```

3. Run the following query to generate a row count for each table in the schema:

   ```
   SQL>spool rowCount_after_import.txt

   SQL>select table_name, row_count(table_name) num_of_rows

   from user_tables;

   SQL>spool off
   ```

   Confirm the schema row count before the export

   The `rowCount_after_import.txt` is created in the execution directory.

4. Compare the data in `rowCount_after_import.txt` with the `rowCount_before_export.txt` file you created before the export operation.

Confirm the schema row count before the export

Import the database backup schema on Linux

Complete the following steps on the two CPU socket Linux server computer.

Import the database backup schema to Linux

1. Set the ORACLE_HOME and ORACLE_SID variables as follows:

   ```
   export ORACLE_HOME=/opt/oracle/product/12.2.0.1/db_1

   export ORACLE_SID = protect
   ```

2. Log on as the SYS user:

   ```
   sqlplus /nolog
   ```

   SQL> connect sys/password@protect as sysdba

   Where password represents the SYS password.

3. Run the following command:

   ```
   Grant read,write on directory DATA_PUMP_DIR to protect;
   ```

4. Run the following command:

   ```
   Grant imp_full_database to protect;
   ```

5. Exit SQL*Plus:

   ```
   SQL> exit
   ```

6. Switch to the Oracle user using the following command:

   ```
   su - oracle
   ```

7. Import the database schema by running the following command from the Terminal:

   ```
   impdp \'sys/<password> as sysdba\' dumpfile=export.dmp schemas=protect DIRECTORY=DATA_PUMP_DIR
   ```

8. Regenerate statistics and rebuild indexes after the import completes.
9. Verify that import was successful by running the following command:

   sqlplus protect/<password>@protect

   Verify that data is present in the schema.

Confirm the schema row count after the import on Linux

After importing the database schema, you generate a row count of each table in the schema. You compare the data you generate with the data you generated before the schema export.

1. Run the following command:

   ```
   sqlplus protect/<password>@protect
   ```

2. Run the following command to create a PL\SQL function to generate the row count:

   ```
   SQL>create or replace function
   row_count (p_tablename in varchar2)
   return number
   as
   l_count number;
   begin
   execute immediate
   'select count(*)
   from ' || p_tablename
   into l_count;
   return l_count;
   ```

```
end;
/
```

3. Run the following query to generate a row count for each table in the schema:

```
SQL>spool rowCount_ater_import.txt
SQL>select table_name, row_count(table_name) num_of_rows
from user_tables;
SQL>spool off
```

The `rowCount_after_import.txt` is created in the execution directory.

4. Compare the data in `rowCount_after_import.txt` with the `rowCount_before_export.txt` file you created before the export operation.

Confirm the schema row count before the export for Linux

Connect Symantec Data Loss Prevention to the Oracle 12c SE2 database

You update the configuration file `jdbc.properties` on the existing Enforce Server file system to reference the Oracle 12c SE2 database.

Complete the following steps to update the `jdbc.properties`:

1. Locate the `jdbc.properties` file on the Enforce Server. Refer to the following list to locate the file on your particular platform and version:
   - Windows
     - Version 14.6 through 15.0: `\SymantecDLP\Protect\config`
     - Version 15.1: `\Program Files\Symantec\Data Loss Prevention\Enforce Server\15.1\Protect \config`
     - Version 15.5 and later: `\Program Files\Symantec\DataLossPrevention\EnforceServer \vv.u\Protect\config`
       Replace vv.u with the version number.
   - Linux
     - Version 14.6 through 15.0: `/opt/SymantecDLP/Protect/config`
     - Version 15.1: `/opt/Symantec/DataLossPrevention/Enforce Server/15.1/Protect/config`
     - Version 15.5 and later: `/opt/Symantec/DataLossPrevention/EnforceServer/vv.u/Protect/ config`
       Replace vv.u with the version number.
2. Open the file and locate the **host** line where the **jdbc.dbalias.oracle-thin** value displays the default IP.
3. Enter the DNS or IP of the two CPU server computer.
4. Save the file.
5. Restart the Symantec DLP Manager service on the Enforce Server.
6. Log on to the Enforce Server administration console to confirm that the Enforce Server is connected to the database. If you cannot log on, restart all Symantec DLP services and log on again.

Migrating to an Oracle multitenant environment on Linux

Table 12: Steps to set up Oracle multitenant environment on Linux lists the process to install Oracle CDP/PDB on Linux systems.

Table 12: Steps to set up Oracle multitenant environment on Linux

Step	Action	More info
1	Complete preinstallation steps.	About installing or upgrading to Oracle 12c Standard Edition 2 Performing the Linux preinstallation steps
2	Install the Oracle database.	Installing the Oracle 19c software on Linux
3	Create the CDB and PDB database.	Creating the Symantec Data Loss Prevention database on Linux
4	Confirm the following: • Confirm that the Container Database name is 'dlpcdb'. • Confirm that the Pluggable Database name is 'protect'.	Verifying the PDB database on Linux
5	Configure the Oracle listener.	Configuring the database connection on Linux
6	Verify that the PDB listener is created and registered.	Verifying that the PDB listener is created and registered on Linux
7	Set the protect PDB to autostart.	Setting the protect PDB to autostart on Linux
8	Create the Oracle user account.	Creating the Oracle user account for Symantec Data Loss Prevention on Linux

Migrating to an Oracle multitenant environment on Windows

Table 13: Steps to set up an Oracle multitenant environment on Windows lists the process to install Oracle CDB/PDB on Windows systems.

Table 13: Steps to set up an Oracle multitenant environment on Windows

Step	Action	More info
1	Install the Oracle database.	Installing the Oracle 19c software on Windows
2	Create the PDB database.	Creating the Symantec Data Loss Prevention database on Windows
3	Confirm the following: • Confirm that the Container Database name is 'dlpcdb'. • Confirm that the Pluggable Database name is 'protect'.	Verifying and PDB database for RAC on Windows
4	Verify that the CDB/PDB is created.	Verifying and PDB database for RAC on Windows
5	Configure the Oracle listeners.	Configuring the database connection on Windows
6	Verify that the PDB listener is created and registered.	Verifying that the PDB listener is created and registered on Windows
7	Set the PDB to autostart (for Windows only).	Setting the protect PDB to autostart on Windows
8	Add required tablespaces to the PDB database.	Adding required tablespaces to the PDB database on Windows
9	Create the Oracle user account.	Creating the Oracle user account for Symantec Data Loss Prevention on Windows

Copyright statement


Symantec
A Division of Broadcom