

**ORACLE RDS SETUP GUIDE**

**The Below requirements are needed to establish a connection oracle rds .**

**Prerequisites**

Prerequisiteslink  
To connect your Oracle database to Fivetran, you need:  
  
Oracle 12c or above  
An AWS account with administrator access to the Oracle RDS database  
Access to your database server  
Your database host's IP (e.g., 1.2.3.4) or domain (your.server.com)  
Your database's port (usually 1521)  
Your database's system identifier (SID)/service name  
(If you want to connect using an SSH tunnel) An SSH server

**Setup Guide**

1. Choose connection method

* First, decide whether to connect Fivetran to your Oracle RDS database directly, using an SSH tunnel, or using AWS PrivateLink.

1. Connect directly (TLS required)

* IMPORTANT: You must have TLS enabled on your database to connect directly to Fivetran. Follow Amazon's TLS setup instructions for Oracle to enable TLS on your database.
* Fivetran connects directly to your Oracle RDS database. This is the simplest method.
* If you connect directly, you will create a rule in a security group that allows Fivetran access to your database instance.

1. Connect using an SSH tunnel (TLS optional)

* IMPORTANT: You must connect using an SSH tunnel if your Oracle database is version 12.1 or below.
* Fivetran connects to a separate server in your network that provides an SSH tunnel to your database. You must connect through an SSH tunnel if your database is in an inaccessible subnet.
* If you connect using an SSH tunnel, you will configure your SSH tunnel host's security group to allow Fivetran access and configure your database's security to allow access from the tunnel host.
* Before you proceed to the next step, you must follow our SSH tunnel connection instructions. If you want Fivetran to use end-to-end encryption using TLS, follow Amazon's TLS setup instructions for Oracle to enable TLS on your database.

1. Connect using AWS PrivateLink BETA

* IMPORTANT: You must have a Business Critical plan to use AWS PrivateLink.
* AWS PrivateLink allows VPCs and AWS-hosted or on-premises services to communicate with one another without exposing traffic to the public internet. Learn more in AWS’ PrivateLink documentation.
* Follow our AWS PrivateLink setup guide to configure PrivateLink for your database.

1. Enable database access

* Grant Fivetran's data processing servers access to your database server. How you grant access depends on whether or not your database instance is in a VPC.
* If your instance is in a VPC, you must configure the two mechanisms that control access: VPC security groups and network access control lists (ACLs). If your instance is not in a VPC, you only need to configure security groups.

1. Configure security group

* NOTE: These instructions assume that your database instance is in a VPC. If your database instance is not in a VPC, you can still use these instructions because configuring a non-VPC security group is an almost identical process.
* In your Amazon RDS dashboard, click on the database instance you want to connect to Fivetran.
* A panel of details for your read replica appears. In the Connectivity & security section, find the database's port number and make a note of it. You will need the port number to configure Fivetran.
* (If you're connecting directly) In the Security column, verify that the Public Accessibility value is Yes.
* IMPORTANT: If you're connecting using an SSH tunnel, skip this step. You do not have to make your database publicly accessible.

1. Click the to your database's security group.

* On the Security Groups page, click on the security group ID.
* On the Inbound tab, click Edit inbound rules.
* Click Add Rule. This creates a new Custom TCP Rule at the bottom of the list.
* Fill in the new Custom TCP Rule.
* In the Port Range field, enter your database's port number that you copied in step 2 of this section (usually 1521).
* What you enter in the Source Custom IP field depends on whether you're connecting directly or using an SSH tunnel.
* If you're connecting directly, enter Fivetran's IPs for your database's region.
* If you're connecting using an SSH tunnel, enter {your-ssh-tunnel-server-ip-address}/32.
* (Optional) Enter a brief description in the Description field.
* Click Save rules.

1. Configure Network ACLs (VPCs only)

* If your database is not in a VPC, skip ahead to Step 3.
* Return to the instance details page.

1. In the Connectivity and security section, click the to the VPC.

* On the Your VPCs page, click the VPC ID.

1. In the Details section, click the Main network ACL .

* Click the Network ACL ID.
* Select the Inbound Rules tab.
* If you have a default VPC that was automatically created by AWS, the settings already allow all incoming traffic. To verify that the settings allow incoming traffic, confirm that the Source value is 0.0.0.0/0 and that the ALLOW entry is listed above the DENY entry.
* If your inbound rules don't include an ALL - 0.0.0.0/0 - ALLOW entry, edit the rules to allow the Source to access the port number of your database instance. (The port will be 1521 for direct connections, unless you changed the default.) For additional help, see Amazon's Network ACL documentation.
* If you're connecting directly, enter Fivetran's IPs for your database's region.
* If you're connecting using an SSH tunnel, enter {your-ssh-tunnel-server-ip-address}/32.

1. Create user

* Create a database user for Fivetran's exclusive use.
* Connect to your Oracle database as an admin user.
* Create a user for Fivetran and grant it permission to connect to your database by executing the following SQL command. Replace <username> and password with a username and password of your choice.
* CREATE USER <username> IDENTIFIED BY <password>;
* GRANT CREATE SESSION TO "<username>";
* content\_copy
* NOTE: Usernames in Oracle are case sensitive. For example, fivetran is not the same user as FIVETRAN.

1. Grant read-only access

* Grant the Fivetran user read-only access to the data you want to sync.
* NOTE: Oracle database defaults to using upper case letters, unless the values are surrounded by double quotes.
* Grant the Fivetran user SELECT permission for each schema and tables you want to sync.
* GRANT SELECT ON "<schemaA>"."<tableA>" TO "<username>";
* GRANT SELECT ON "<schemaA>"."<tableB>" TO "<username>";
* GRANT SELECT ON "<schemaB>"."<tableC>" TO "<username>";
* content\_copy
* Alternatively, you can grant access to all tables:
* GRANT SELECT ANY TABLE TO "<username>";
* content\_copy
* Grant the Fivetran user access to the DBA\_EXTENTS, DBA\_TABLESPACES, and DBA\_SEGMENTS system views. We use these views to optimize our initial import queries.
* BEGIN
* rdsadmin.rdsadmin\_util.grant\_sys\_object('DBA\_EXTENTS','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('DBA\_TABLESPACES','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('DBA\_SEGMENTS','<username>','SELECT');
* END;
* content\_copy

1. Configure LogMiner

* To keep your data up to date after the initial sync, we use Oracle's built-in LogMiner technology. LogMiner uses archived redo log files to get a list of modified tables and determine which rows need to be updated, which allows Fivetran to update only the data that has changed since our last sync.
* To enable LogMiner, do the following:
* If ARCHIVELOG mode is not enabled on your database, enable ARCHIVELOG mode.
* NOTE: ARCHIVELOG mode is enabled automatically when automated backups are enabled by setting the backup retention period to a value greater than 0. For more information, see Amazon RDS's automated backups documentation.
* Retain redo log files for at least 24 hours. We recommend retaining data for seven days.
* BEGIN
* rdsadmin.rdsadmin\_util.set\_configuration('archivelog retention hours', 24);
* END;
* content\_copy
* Enable minimal supplemental logging by executing the following SQL statement. Minimal supplemental logging ensures that LogMiner has sufficient information to process the redo operations associated with DML changes.
* NOTE: To learn about minimal supplemental logging, read Oracle's database-level supplemental logging documentation.
* BEGIN
* rdsadmin.rdsadmin\_util.alter\_supplemental\_logging(p\_action => 'ADD');
* END;
* content\_copy
* Enable primary key supplemental logging if you expect a row containing a primary key to change.
* To enable supplemental logging on primary key columns, execute the following SQL statement:
* BEGIN
* rdsadmin.rdsadmin\_util.alter\_supplemental\_logging('ADD','PRIMARY KEY');
* END;
* content\_copy
* Grant the Fivetran user permission to run LogMiner.
* BEGIN
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$DATABASE','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$PARAMETER','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$ARCHIVED\_LOG','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$ARCHIVE\_DEST','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$LOGMNR\_CONTENTS','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('DBMS\_LOGMNR','<username>','EXECUTE');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('DBMS\_LOGMNR\_D','<username>','EXECUTE');
* END;
* content\_copy
* GRANT SELECT ANY TRANSACTION TO "<username>";
* content\_copy
* NOTE: To learn more about common DBA tasks for Oracle RDS databases, see Amazon's Oracle DB instance documentation.
* (Oracle version 12 and above) Grant logmining permissions to the Fivetran user.
* GRANT LOGMINING TO "<username>";
* content\_copy
* (Optional) Grant permissions to additional system tables.
* NOTE: We recommend granting these permissions to optimize the connector's performance and reliability.
* BEGIN
* rdsadmin.rdsadmin\_util.grant\_sys\_object('DBA\_FREE\_SPACE','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$LOG','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$TEMPFILE','<username>','SELECT');
* rdsadmin.rdsadmin\_util.grant\_sys\_object('V\_$DATAFILE','<username>','SELECT');
* END;
* content\_copy

1. Finish Fivetran configuration

* In your connector setup form, enter a destination schema prefix. This prefix applies to each replicated schema and cannot be changed once your connector is created.
* In the Host field, enter your database host's IP (for example, 1.2.3.4) or DNS name (for example, your.server.com).
* Enter your database's port number. The port number is usually 1521 or 2483 for unencrypted connections and 1521 or 2484 for encrypted connections using SSL/TLS.
* Enter the Fivetran-specific user that you created in Step 3.
* Enter the password for the Fivetran-specific user that you created in Step 3.
* Enter your database's SID/Service Name.
* Choose your connection method. If you selected Connect via an SSH tunnel, provide the following information:
* SSH hostname (do not use a load balancer's IP address/hostname)
* SSH port
* SSH user
* If you enabled TLS on your database in Step 1, set the Require TLS through tunnel toggle to ON.
* Click Save & Test. Fivetran tests and validates our connection to your Oracle RDS database. Upon successful completion of the setup tests, you can sync your data using Fivetran.

1. Setup tests

* Fivetran performs the following tests to ensure that we can connect to your Oracle RDS database and that it is properly configured:
* The Connecting to SSH Tunnel Test validates the SSH tunnel details you provided in the setup form. It then checks that we can connect to your database using the SSH tunnel. (We skip this test if you aren't connecting using an SSH tunnel.)
* The Connecting to Host Test validates the database credentials you provided in the setup form. It then verifies that the database host is not private and checks that we can connect to the host.
* The Validating Certificate Test generates a pop-up window only if the certificate is not already trusted or has not been used previously. In that pop-up window, you must choose which certificate you want Fivetran to use. The test then validates that certificate and checks that we can connect to your database using TLS. (We skip this test if you aren't connecting directly.)
* The Validating Database Version Test checks your database version. The test passes if your database version is 12.2 or above or if your database version is 12.1 or below and you are connecting using an SSH tunnel.
* The Validating System View Permission Test checks that we have permission to access the DBA\_EXTENTS, DBA\_TABLESPACES, and DBA\_SEGMENTS system views.
* The Validating Archive Log Access Test checks that we can access your archive log.
* The Validating Archive Log Retention Period Test verifies that your archive log is set to retain at least 24 hours' worth of changes.
* The Access to Database-Level Supplemental Logging Test verifies that supplemental logging is enabled on your database. If supplemental logging is not enabled, the test passes but generates a warning message in your dashboard.
* The Validating Speed Setup test checks how quickly Fivetran can fetch data from your source database. The test will show a warning if the speed is less than 5MB per second.
* NOTE: The tests may take a few minutes to finish running.

