

Deploying GitLab Enterprise Edition on Google Distributed Cloud air-gapped (VM-Based Approach)

Written by: GitLab

Last update: Oct 21, 2024

This guide provides a detailed guide for deploying GitLab Enterprise Edition on Google Distributed Cloud (GDC) in an air-gapped environment. GDC air-gapped is appropriate for environments with strict security requirements where the network is isolated from the internet. We assume that the reader has some knowledge of virtual machines, network configurations, and basic Linux command-line operations.

Overview

Deploying GitLab on GDC air-gapped environments involves several critical steps, from downloading the necessary software packages to configuring and running the GitLab application. This guide has easy-to-follow instructions to help make sure you use it correctly.

Prerequisites

- Access to Google Distributed Cloud account and permissions to create and manage virtual machines.
- Knowledge of how to access a remote server using SSH.
- Understanding of FTP (File Transfer Protocol) for transferring files in a secure network environment.
- Knowledge of RPM (Red Hat Package Manager) file handling.

Step-by-Step Guide

Step 1: Download the Required Software

1. Obtain RPM Files:

- Visit the [GitLab Enterprise Edition package page](#) to download the latest version of GitLab EE suitable for your system (select the RPM package).
- Use checksums to make sure the downloaded files haven't been tampered with during the download.

Step 2: Transfer Files to the Air-Gapped Environment

1. Prepare for Transfer:

- Ensure compliance with your organization's internal processes for transferring data between internet-connected and air-gapped environments. While it sometimes may involve using secure physical devices such as USB drives or other approved transfer methods, please consult your organization's requirements.
- Securely copy the downloaded RPM file to the transfer medium.

Step 3: Set Up the Virtual Machine

1. Create and Configure VM:

- Follow the guidelines provided in the [Google Distributed Cloud documentation](#) to create a new virtual machine ("VM"). Choose a setup that works well for GitLab. It usually needs 4 CPU cores and 8 GB of RAM.
- Make sure that the VM is configured with the appropriate security settings in accordance with your organization's policies.

Step 4: Access the Virtual Machine

1. Connect to Your VM:

- Use the Google Distributed Cloud console to initiate an SSH connection to your newly created VM. Read the [documentation](#) for more details.

Step 5: Transfer the GitLab Package to the VM

1. Transfer the RPM File:

- From the SSH session, use FTP to transfer the GitLab RPM from your local machine or transfer medium to the VM.

- Refer to [this guide](#) for detailed instructions on secure file transfer within air-gapped networks.

Step 6: Install GitLab

1. Installation Process:

- Follow the official GitLab [offline installation documentation](#) to install GitLab from the RPM file. This includes setting up dependencies, configuring GitLab, and initializing the system.
- Modify the `/etc/gitlab/gitlab.rb` file for necessary configurations like external URL and any other network or security settings.

Following these steps helps ensure a safe and efficient deployment of GitLab Enterprise Edition on Google Distributed Cloud. For more help, consult GitLab's documentation or contact GitLab support for advice specific to your environment.