Hosting your own Git Infrastructure

By Kishan Takoordyal - MSCC Monthly Meetup September 2025

About Me

I'm Kishan Takoordyal. Professionally, I'm a Senior DevOps Engineer but during my free time, I also do a lot of web and game development. I have been an author in the past, and have participated in a couple of contests and hackathons.

Agenda

01. **Features of a Git / DevOps platform**

02. **Why self-host your own Git Infra?**

03. **Choosing the best option**

04. **Demo Time**

05. **Next Steps**

06. **Resources**

Features of a Git / DevOps Platform

01. **Version Control using Git**

02. **Project Management**

03. **Collaboration through Issues and PRs**

04. **Integrated CI/CD Pipelines**

05. **Container Registry**

06. **Security & support for Policies**

Why self-host your own Git Infra?

Enhanced Security & Data Control

Having the complete control over the data and code eliminates reliance on third-party providers. This is crucial for sensitive projects or compliance requirements.

Customization & Flexibility

The environment can be tailored to your exact specifications, including hardware, operating system and integrated tools. This allows for optimized performance and the integration of specific workflows not available in hosted solutions.

Cost Management

There are initial setup and maintenance costs but in the long run, self-hosting can be more cost-effective, especially for large teams and extensive usage.

Performance & Local Access

For teams working primarily on-site, a Git server on the local network can offer significantly faster data transfer speeds. CI/CD pipelines can also run faster, depending on the hardware provided.

Choosing the best option

Product	Features	Resource Requirements	Target Audience
GitLab	Git hosting, CI/CD, issue tracking, project management, security scanning, registries, etc	Very Demanding	Enterprise / very large teams looking for an all-in-one solution
Gitea	Git hosting, issue tracking, project management, basic CI/CD system, registries, etc	Very efficient and requires minimal resources	Individuals / small teams prioritizing simplicity and resource efficiency
Gogs	Gitea was forked from Gogs. It has less features and less active development than the latter.	Extremely low resource footprint	Users seeking the absolute simplest and most resource-efficient solution
OneDev	Git hosting, CI/CD, issue tracking, project management, integrated kanban boards, registries, etc	Relatively lightweight compared to Gitlab but more demanding than Gitea	Teams looking for a good balance of performance and integrated features such as a CI/CD system & Kanban boards

Demo Time

Next Steps

01.

Define, deploy and enforce secure settings regarding user access..

02.

Implement proper policies on the organization and project levels.

03.

Ensure to always keep the Git server up-to-date and put a backup system in place.

04.

Continuously monitor and adapt.

Resources

Here are the slides from this session as well as all the code demonstrated.

Slides

https://kinesis.world/go/mscc-slides-0925

Ansible Code

https://kinesis.world/go/mscc-ansible-0925

Demo site code

https://kinesis.world/go/mscc-react-0925

Questions?

Thank you!