Self Hosting with Nomad

My experiences of running and managing self hosted applications using Nomad.

Karan Sharma
mrkaran.dev

whoami

👨‍💻 Works at Zerodha

📓 Blogs about things I find interesting

📈 Interested in Self Hosting

Why (I) Self Host

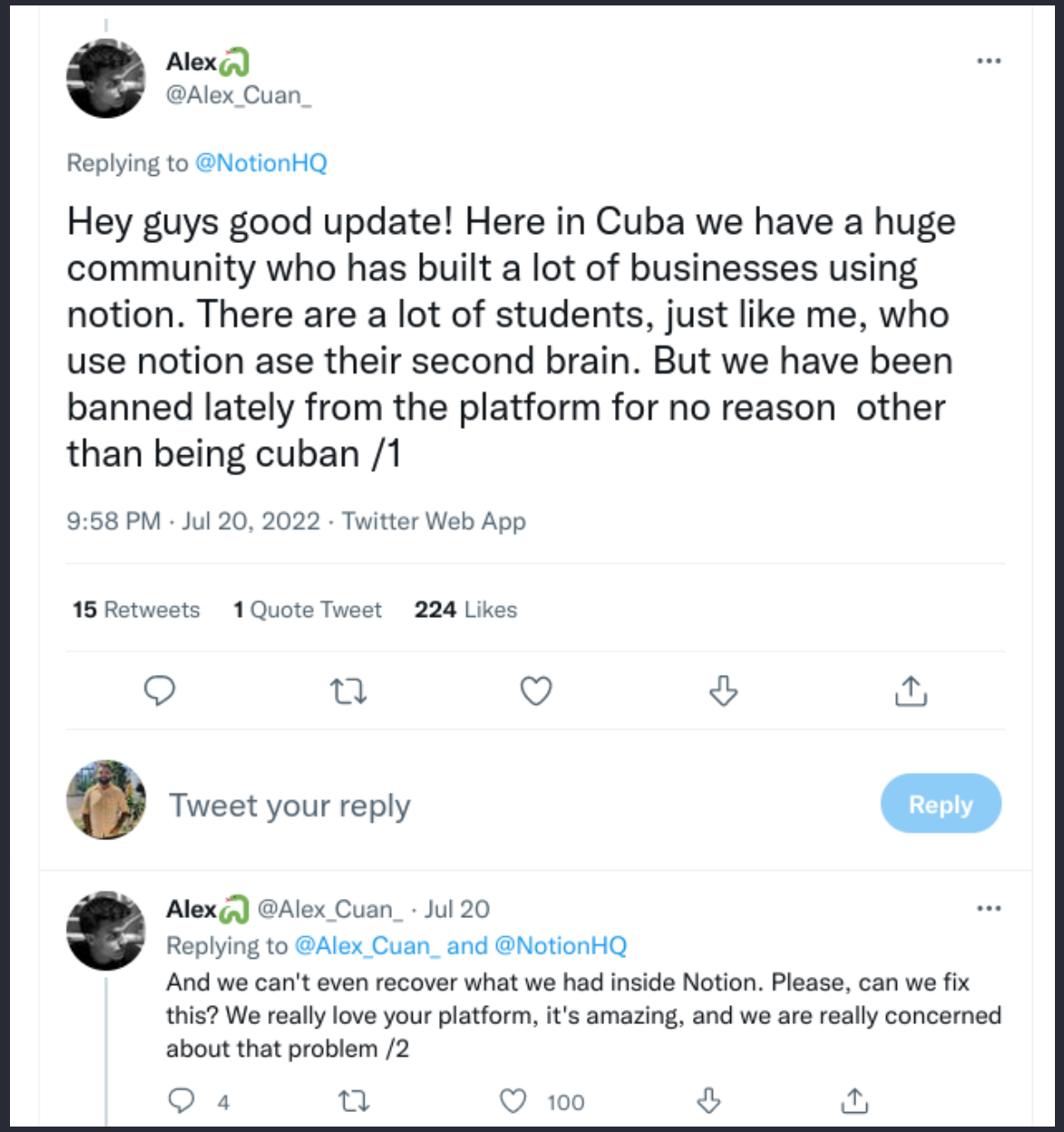
- Break from the Big Tech Co

Why (I) Self Host

- Break from the Big Tech Co
- Own your data

Why (I) Self Host

- Break from the Big Tech Co
- Own your data
- No lock ins for data which is critical



Alex
@Alex_Cuan_

Replying to @NotionHQ

Hey guys good update! Here in Cuba we have a huge community who has built a lot of businesses using notion. There are a lot of students, just like me, who use notion ase their second brain. But we have been banned lately from the platform for no reason other than being cuban /1

9:58 PM · Jul 20, 2022 · Twitter Web App

15 Retweets **1** Quote Tweet **224** Likes

Tweet your reply

Reply

Alex @Alex_Cuan_ · Jul 20

Replying to @Alex_Cuan_ and @NotionHQ

And we can't even recover what we had inside Notion. Please, can we fix this? We really love your platform, it's amazing, and we are really concerned about that problem /2

4 100

Why (I) Self Host

- Break from the Big Tech Co
- Own your data
- No lock ins for data which is critical
- Chance to contribute to OSS

Why (I) Self Host

- Break from the Big Tech Co
- Own your data
- No lock ins for data which is critical
- Chance to contribute to OSS
- Experiment and learn

My Setup

Servers

- DigitalOcean Droplet (2vCPU, 4GB RAM, blr1 Region)

Infra Tools

- Ansible
- Terraform
- Nomad

Ansible

- Boostrap the server
  - Harden SSH. User, Shell setups.
  - Install `node-exporter`, `docker`, `tailscale`.

Terraform

- DigitalOcean infra
  - Droplet
  - Firewalls
  - SSH Keys, Volumes, Floating IPs etc.
- Cloudflare DNS records

Nomad

- *Simple* workload orchestrator and scheduler
- Run workloads with multiple task drivers (not just docker containers)

Why Nomad

- Was using K8s before - went down the deep complexity hell
- Deployed my first app in a few minutes
- Single binary exectubale - with a UI

Nomad Agent

- Server takes the scheduling decisions
  - For HA, run 3/5/7 nodes. (Raft consensus)
- Client runs the actual task given by the server
  - Interacts with task plugins like docker etc

Ecosystem

Plugs into other Hashicorp tools very well

- Native integration with *Consul* Connect for ACLs
- Fetch secrets from *Vault*
- Deploy using *Waypoint*

Running Nomad

- Grab the binary
- `nomad agent -dev` -> starts in dev mode. Great for local testing
- Configure `server.hcl` / `client.hcl`

Jobspec

- A deployment file is called "Jobspec". Think of `docker-compose.yml`
- Specify all possible things to run that app in one file
  - Job -> Group -> Task
  - Artifact (S3/GitHub Releases/Remote config files)
  - Networking options
  - Volume mounts

Deploying Gitea

```
job "gitea" {
  datacenters = ["hydra"]
  type        = "service"
  group "app" {
    count = 1
    network {
      port "http" {
        to = 3000
      }
      port "ssh" {
        to           = 22
        static       = 4222
        host_network = "tailscale"
      }
    }
```

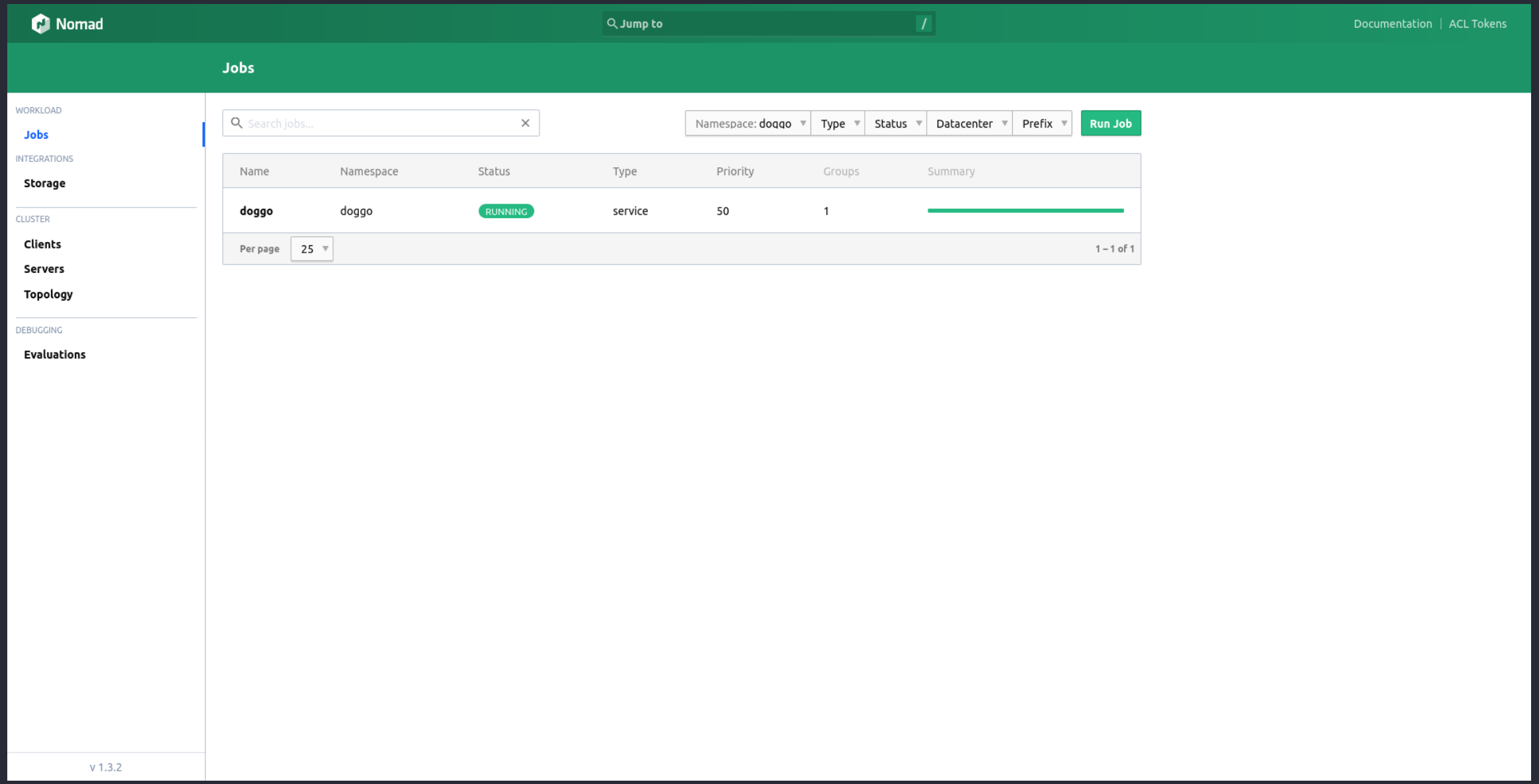
Deploying Gitea

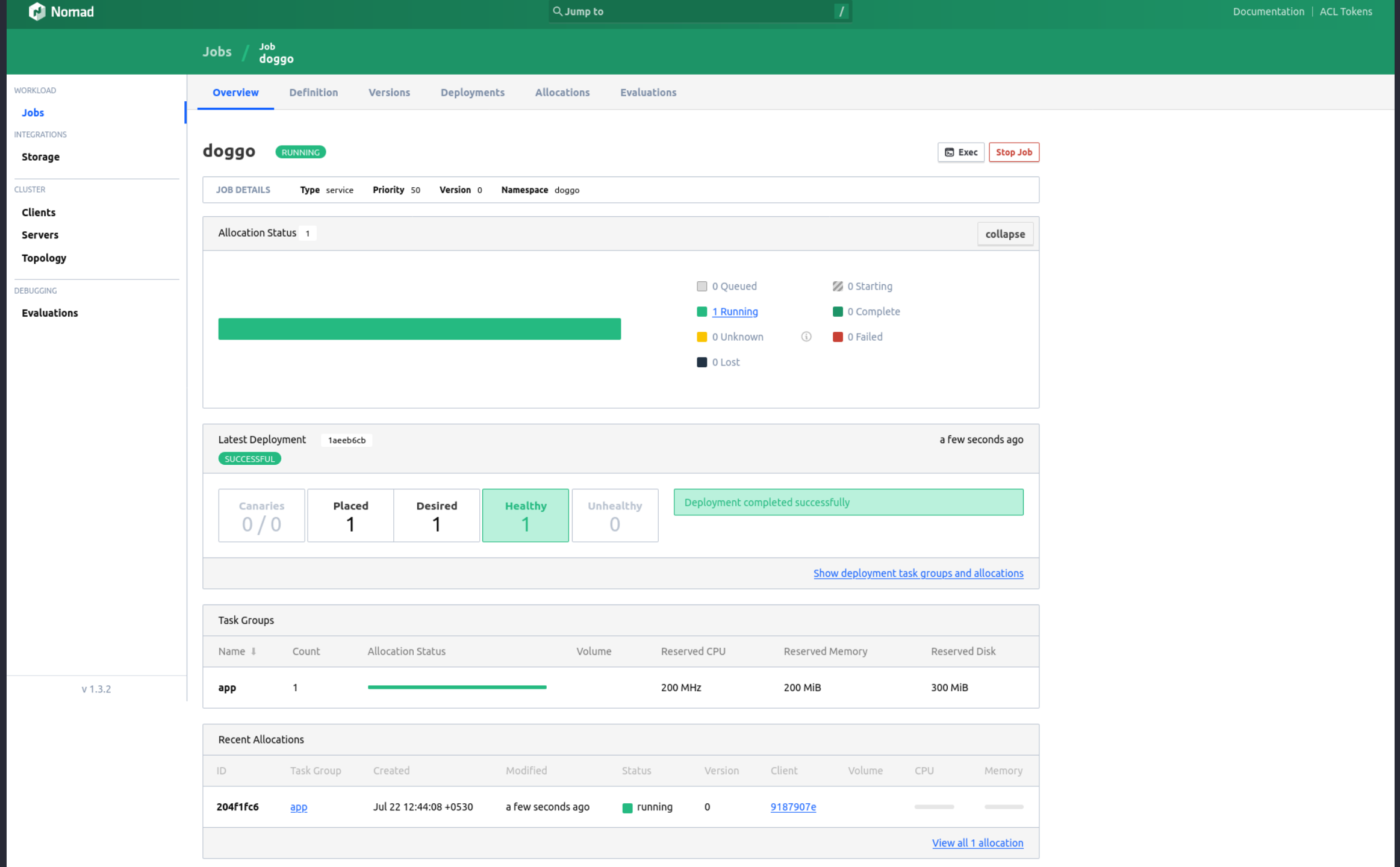
```
task "web" {
  driver = "docker"
  config {
    image = "gitea/gitea:latest"
    ports = ["http", "ssh"]
    mount {
      type   = "bind"
      source = "local/gitea.ini"
      target = "/data/gitea/conf/app.ini"
    }
  }
  resources {
    cpu    = 200
    memory = 300
  }
}
```

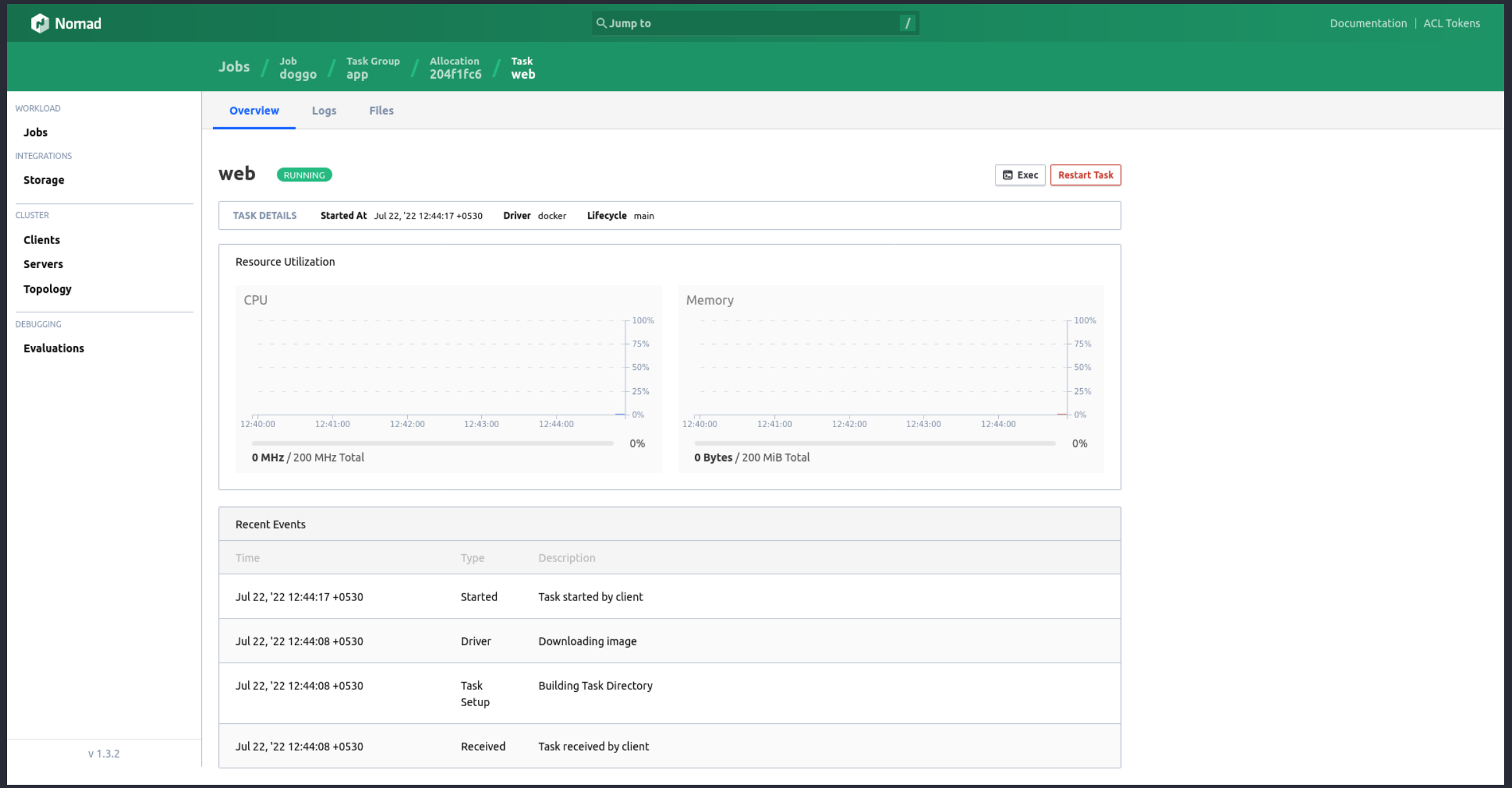
Deploying Gitea

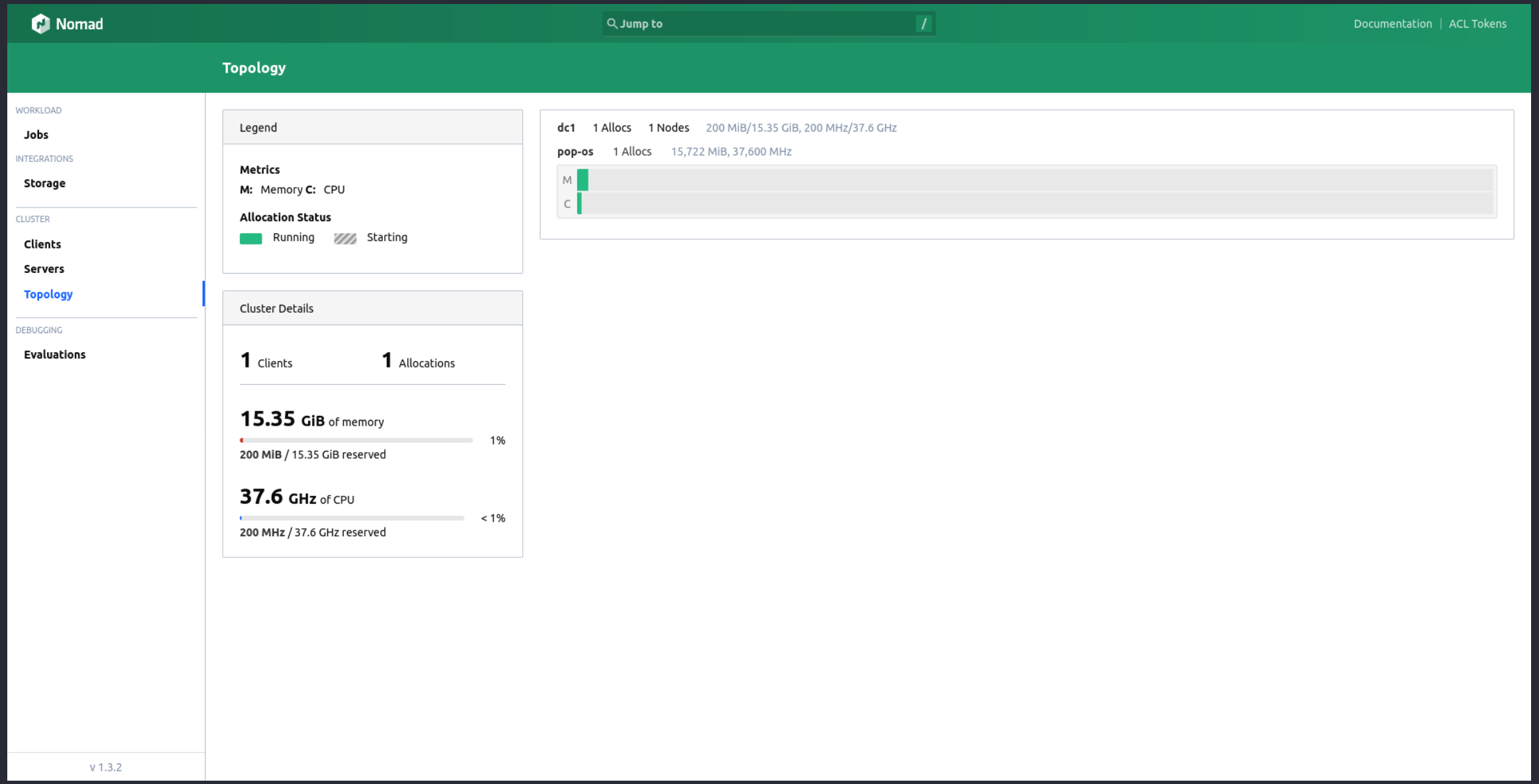
```
    service {
      provider = "nomad"
      name     = "gitea-web"
      tags     = ["gitea", "web"]
      port     = "http"
    }
    service {
      provider = "nomad"
      name     = "gitea-ssh"
      tags     = ["gitea", "ssh"]
      port     = "ssh"
    }
```

Exploring the UI


Nomad
Jump to
/
Documentation | ACL Tokens
Jobs
WORKLOAD
Jobs
INTEGRATIONS
Storage
CLUSTER
Clients
Servers
Topology
DEBUGGING
Evaluations
v 1.3.2
Search jobs...
Namespace: doggo
Type
Status
Datacenter
Prefix
Run Job
Name
Namespace
Status
Type
Priority
Groups
Summary
doggo
doggo
RUNNING
service
50
1
Per page
25
1 – 1 of 1

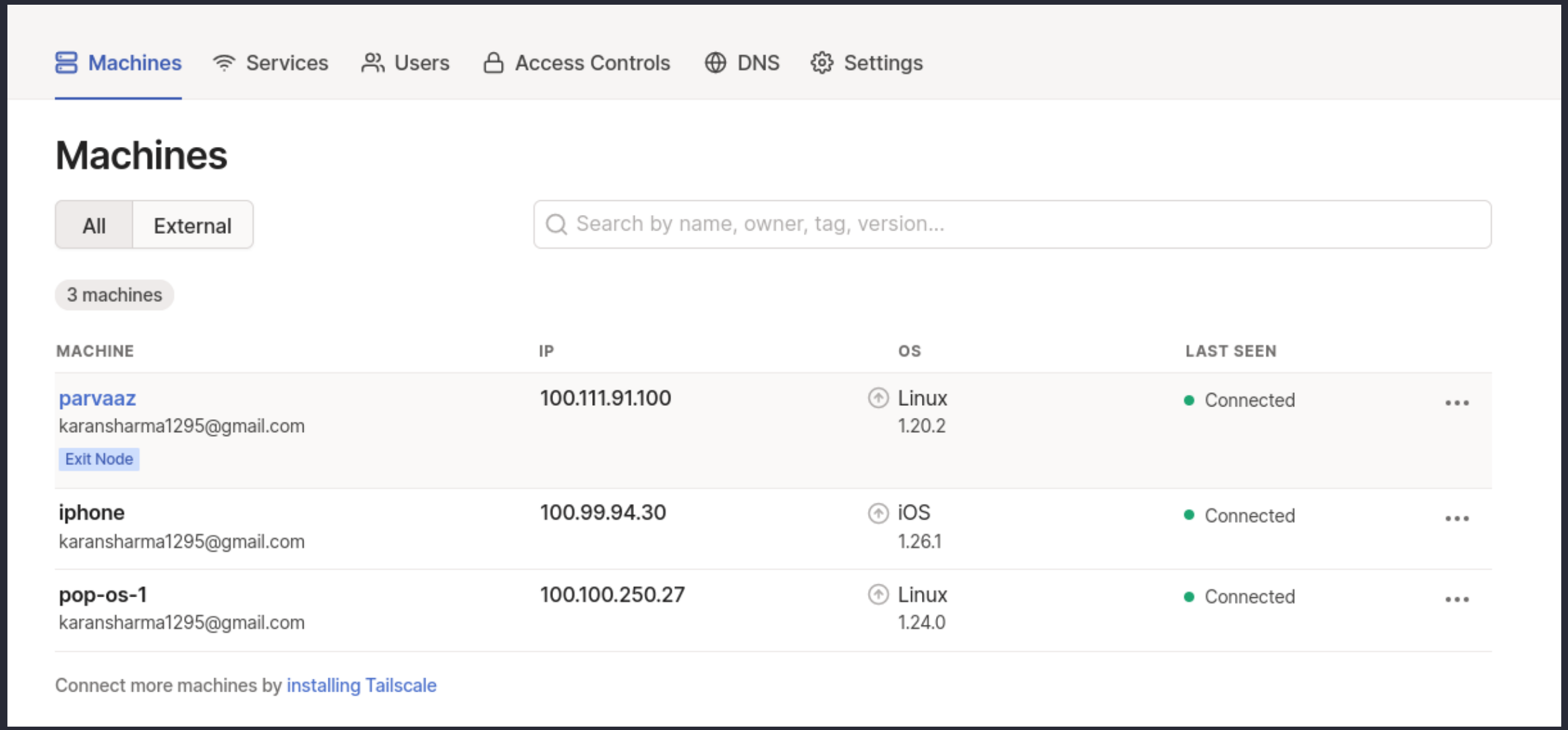

Nomad
Jump to
/
Documentation | ACL Tokens
Jobs / Job doggo
WORKLOAD
Jobs
INTEGRATIONS
Storage
CLUSTER
Clients
Servers
Topology
DEBUGGING
Evaluations
v 1.3.2
Overview
Definition
Versions
Deployments
Allocations
Evaluations
doggo
RUNNING
Exec
Stop Job
JOB DETAILS
Type service
Priority 50
Version 0
Namespace doggo
Allocation Status 1
collapse
0 Queued
0 Starting
1 Running
0 Complete
0 Unknown
0 Failed
0 Lost
Latest Deployment
1aeeb6cb
a few seconds ago
SUCCESSFUL
Canaries
0 / 0
Placed
1
Desired
1
Healthy
1
Unhealthy
0
Deployment completed successfully
Show deployment task groups and allocations
Task Groups
Name
Count
Allocation Status
Volume
Reserved CPU
Reserved Memory
Reserved Disk
app
1
200 MHz
200 MiB
300 MiB
Recent Allocations
ID
Task Group
Created
Modified
Status
Version
Client
Volume
CPU
Memory
204f1fc6
app
Jul 22 12:44:08 +0530
a few seconds ago
running
0
9187907e
View all 1 allocation


Nomad
Jump to
/
Documentation | ACL Tokens
Jobs / Job doggo / Task Group app / Allocation 204f1fc6 / Task web
WORKLOAD
Jobs
INTEGRATIONS
Storage
CLUSTER
Clients
Servers
Topology
DEBUGGING
Evaluations
v 1.3.2
Overview
Logs
Files
web
RUNNING
Exec
Restart Task
TASK DETAILS
Started At Jul 22, '22 12:44:17 +0530
Driver docker
Lifecycle main
Resource Utilization
CPU
100%
75%
50%
25%
0%
12:40:00
12:41:00
12:42:00
12:43:00
12:44:00
0%
0 MHz / 200 MHz Total
Memory
100%
75%
50%
25%
0%
12:40:00
12:41:00
12:42:00
12:43:00
12:44:00
0%
0 Bytes / 200 MiB Total
Recent Events
Time
Type
Description
Jul 22, '22 12:44:17 +0530
Started
Task started by client
Jul 22, '22 12:44:08 +0530
Driver
Downloading image
Jul 22, '22 12:44:08 +0530
Task Setup
Building Task Directory
Jul 22, '22 12:44:08 +0530
Received
Task received by client


Nomad
Jump to
/
Documentation | ACL Tokens
Topology
WORKLOAD
Jobs
INTEGRATIONS
Storage
CLUSTER
Clients
Servers
Topology
DEBUGGING
Evaluations
v 1.3.2
Legend
Metrics
M: Memory C: CPU
Allocation Status
Running
Starting
Cluster Details
1 Clients
1 Allocations
15.35 GiB of memory
1%
200 MiB / 15.35 GiB reserved
37.6 GHz of CPU
< 1%
200 MHz / 37.6 GHz reserved
dc1 1 Allocs 1 Nodes 200 MiB/15.35 GiB, 200 MHz/37.6 GHz
pop-os 1 Allocs 15,722 MiB, 37,600 MHz
M
C

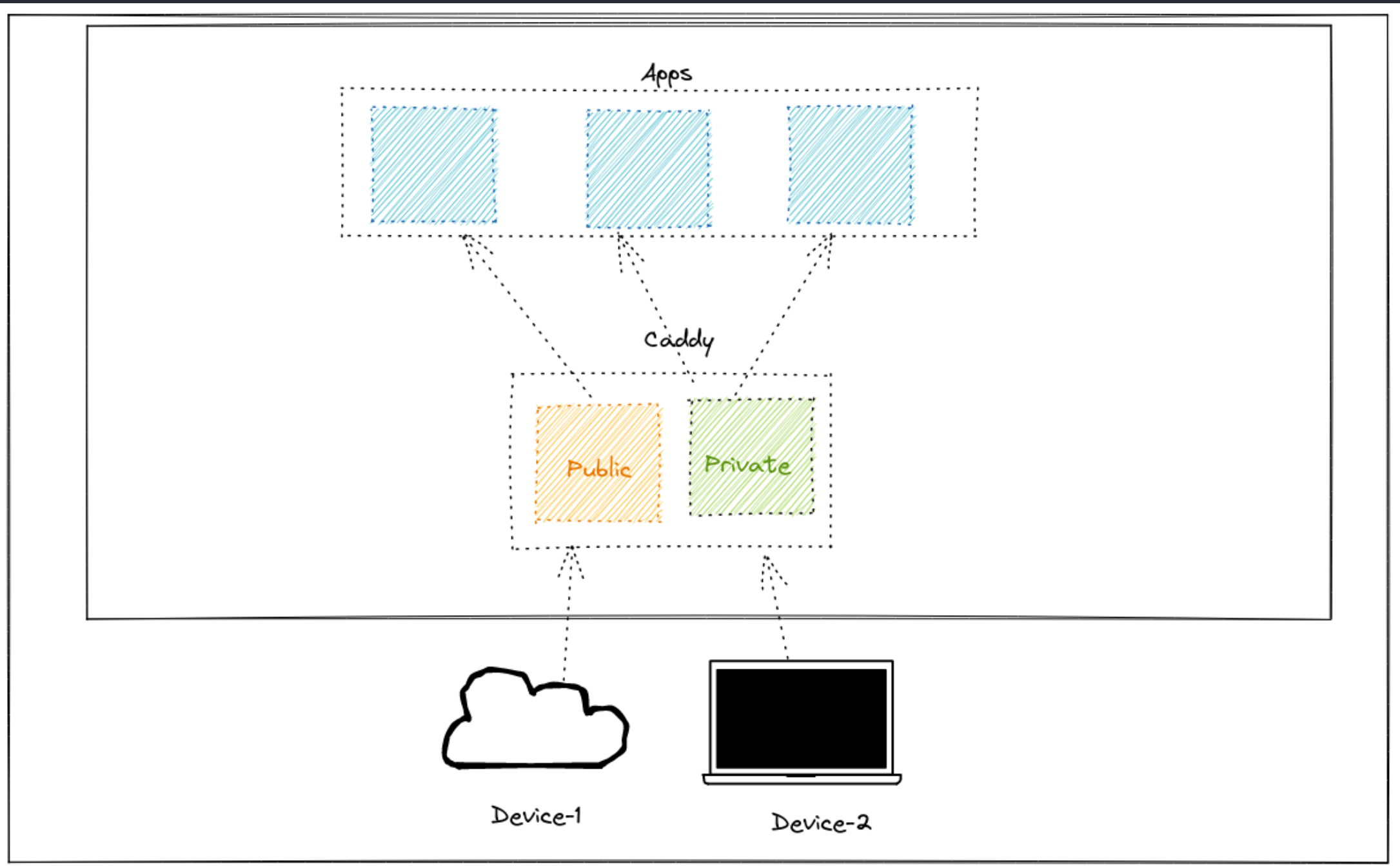
Networking

- Tailscale for "mesh network"
  - Based on Wireguard VPN.
  - Authenticated sessions only.
  - Expose services on an interbal network and access them on all devices
- Tailscale uses `100.64.0.0/10` subnet from the Carrier Grade NAT (CGNAT) space.


Machines
Services
Users
Access Controls
DNS
Settings
Machines
All
External
Search by name, owner, tag, version...
3 machines
MACHINE
IP
OS
LAST SEEN
parvaaz
karansharma1295@gmail.com
Exit Node
100.111.91.100
Linux
1.20.2
Connected
iphone
karansharma1295@gmail.com
100.99.94.30
iOS
1.26.1
Connected
pop-os-1
karansharma1295@gmail.com
100.100.250.27
Linux
1.24.0
Connected
Connect more machines by installing Tailscale

Networking

- Caddy as a proxy for all services.
  - Running 2 instances of Caddy.
    - Private: Listens on Tailscale Interface.
    - Public: Listens on DO's public IPv4 Interface.
  - Automatic SSL with ACME DNS challenge
    - Built my own image: https://github.com/mr-karan/caddy-plugins-docker



Self Hosting with Nomad/Caddy/Tailscale

Networking

Dont expose to the world

```
doggo adguard.mrkaran.dev
NAME                    TYPE    CLASS   TTL     ADDRESS          NAMESERVER
adguard.mrkaran.dev.    A       IN      23s     100.111.91.100   127.0.0.1:53
```

unless required

```
doggo doggo.mrkaran.dev
NAME                    TYPE    CLASS   TTL     ADDRESS          NAMESERVER
doggo.mrkaran.dev.      A       IN      25s     172.67.187.239   127.0.0.1:53
doggo.mrkaran.dev.      A       IN      25s     104.21.7.168     127.0.0.1:53
```

Storage

- Enable snapshots for volumes provided by cloud provider.
- Use separate DB instances for different applications.

Backups

- Restic
  - Periodic Job in Nomad.
  - Single vault with everything inside `/data`.
  - All applications mount inside `/data` folder.
  - Upload to Backblaze B2.

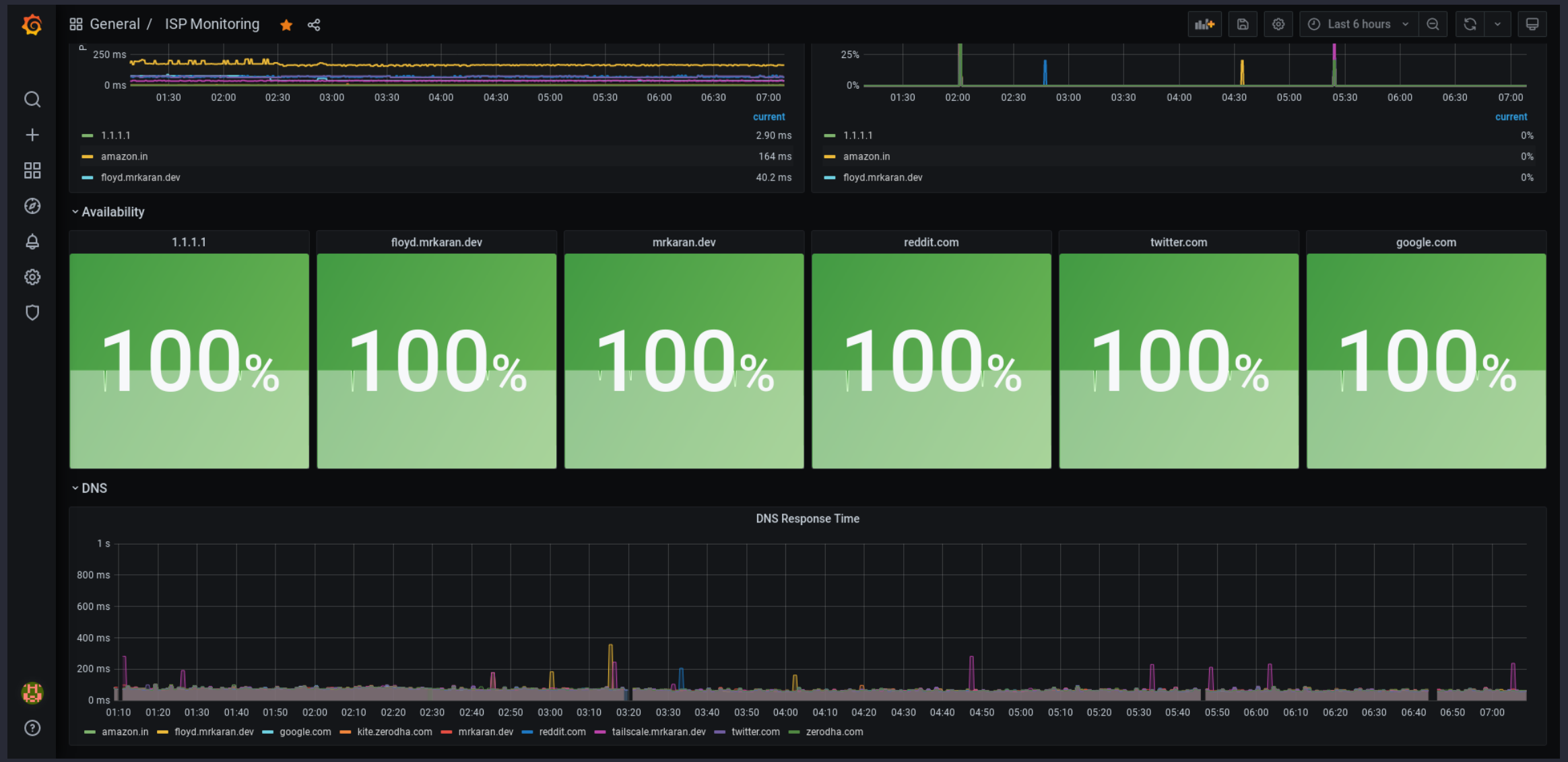
Services I run

- Adguard (DNS)
- Gitea (Git)
- Joplin Sync Server (Notes app)
- Miniflux (RSS reader)
- Plausible (Website Analytics)
- Grafana/Prometheus (Monitoring)
- Nextcloud (Documents/Photos)
- `doggo.mrkaran.dev` (DNS resolver)

Monitoring

- Grafana
- Prometheus
- Telegraf to collect home ISP stats
  - Ping Input plugin
  - DNS Input Plugin


General / ISP Monitoring
Last 6 hours
Avg Ping Response Time
1.1.1.1
2.90 ms
mrkaran.dev
68.4 ms
twitter.com
68.0 ms
reddit.com
36.6 ms
Upstream Health
BLR Droplet (Floyd)
Ping (ms)
Loss
1 s
750 ms
500 ms
250 ms
0 ms
100%
75%
50%
25%
0%
01:10 01:20 01:30 01:40 01:50 02:00 02:10 02:20 02:30 02:40 02:50 03:00 03:10 03:20 03:30 03:40 03:50 04:00 04:10 04:20 04:30 04:40 04:50 05:00 05:10 05:20 05:30 05:40 05:50 06:00 06:10 06:20 06:30 06:40 06:50 07:00
current
min 39.8 ms
avg 40.2 ms
max 40.7 ms
Avg Response Time
01:30 02:00 02:30 03:00 03:30 04:00 04:30 05:00 05:30 06:00 06:30 07:00
current
Packet Loss %
01:30 02:00 02:30 03:00 03:30 04:00 04:30 05:00 05:30 06:00 06:30 07:00
current


General / ISP Monitoring
Last 6 hours
250 ms
0 ms
01:30 02:00 02:30 03:00 03:30 04:00 04:30 05:00 05:30 06:00 06:30 07:00
current
1.1.1.1 2.90 ms
amazon.in 164 ms
floyd.mrkaran.dev 40.2 ms
25%
0%
01:30 02:00 02:30 03:00 03:30 04:00 04:30 05:00 05:30 06:00 06:30 07:00
current
1.1.1.1 0%
amazon.in 0%
floyd.mrkaran.dev 0%
Availability
1.1.1.1
100%
floyd.mrkaran.dev
100%
mrkaran.dev
100%
reddit.com
100%
twitter.com
100%
google.com
100%
DNS
DNS Response Time
1 s
800 ms
600 ms
400 ms
200 ms
0 ms
01:10 01:20 01:30 01:40 01:50 02:00 02:10 02:20 02:30 02:40 02:50 03:00 03:10 03:20 03:30 03:40 03:50 04:00 04:10 04:20 04:30 04:40 04:50 05:00 05:10 05:20 05:30 05:40 05:50 06:00 06:10 06:20 06:30 06:40 06:50 07:00
amazon.in floyd.mrkaran.dev google.com kite.zerodha.com mrkaran.dev reddit.com tailscale.mrkaran.dev twitter.com zerodha.com

Security

- If it should not be public facing, don't expose to WWW.
  - Prefer to use a VPN or mesh network instead of IP whitelists.
  - Tighter Firewall rules otherwise.
- Adguard, Gitea, etc Admin interfaces must always be protected with strong passwords.
- Periodic **updates** to App and OS.

Takeaways

- Don't get overwhelmed by choices. Pick something really simple (like Adguard/Pi-hole) and host it.
- Aim for simplicity

Resources

- Mon School - Self Hosting 101 Course
- r/selfhosted
  - Beginner friendly wiki: https://wiki.r-selfhosted.com/
- github.com/awesome-selfhosted
- learn.hashicorp.com/nomad

Thank You