JamLab Nomad

JamLab Nomad Architecture

JamLab was the first, now mostly deprecated experiment to fool around with IaC concepts and Hashicorp Nomad. This page describes the JamLab Nomad architecture.

You can also find a mediocre written report of the Nomad laboratory and a better report on monitoring the Nomad system.

Abstract overview

JamLab is a hardware installed at JamFox's home, the lab is behind switch that is connected to the NAT gateway router with a dynamic IP and is managed by one or more Proxmox Virtual Environment bare metal hypervisor hosts with a heap memory and CPU resources for running virtual machines. Internal DNS is provided by another low-power always-on bare metal host, in this case a Raspberry Pi. All bare metal hosts are configured using Red Hat Ansible. Secrets are handled by Ansible Vault. Hypervisor host runs virtual machines. Configured virtual machine templates are built with Hashicorp Packer and provisioned using Hashicorp Terraform and configured by Ansible post-provision. Virtual machines fall into two groups: base infrastructure nodes (called `vb` nodes) and service infrastructure nodes (called `vs` nodes). Base infrastructure nodes run Hashicorp Consul service discovery, Hashicorp Nomad orchestration servers. Service infrastructure nodes use Nomad clients to run containerized services.

Features:

- Dynamically parsed hosts list: adding a new `vs` and `vb` nodes is as easy as adding a new entry to their respective host groups.
- Reproducible VM provisioning: Packer creates VM template and Terraform provisions VMs.
- Internal DNS: reach nodes via subdomain instead of trying to remember IPs.
- External DNS: internet-exposed services available with DDNS pointing to router or by using Cloudflare Tunnels.
- Load balancing dynamic services: HAProxy as the external load balancer forwards traffic to live internal load balancers integrated with Consul service discovery that

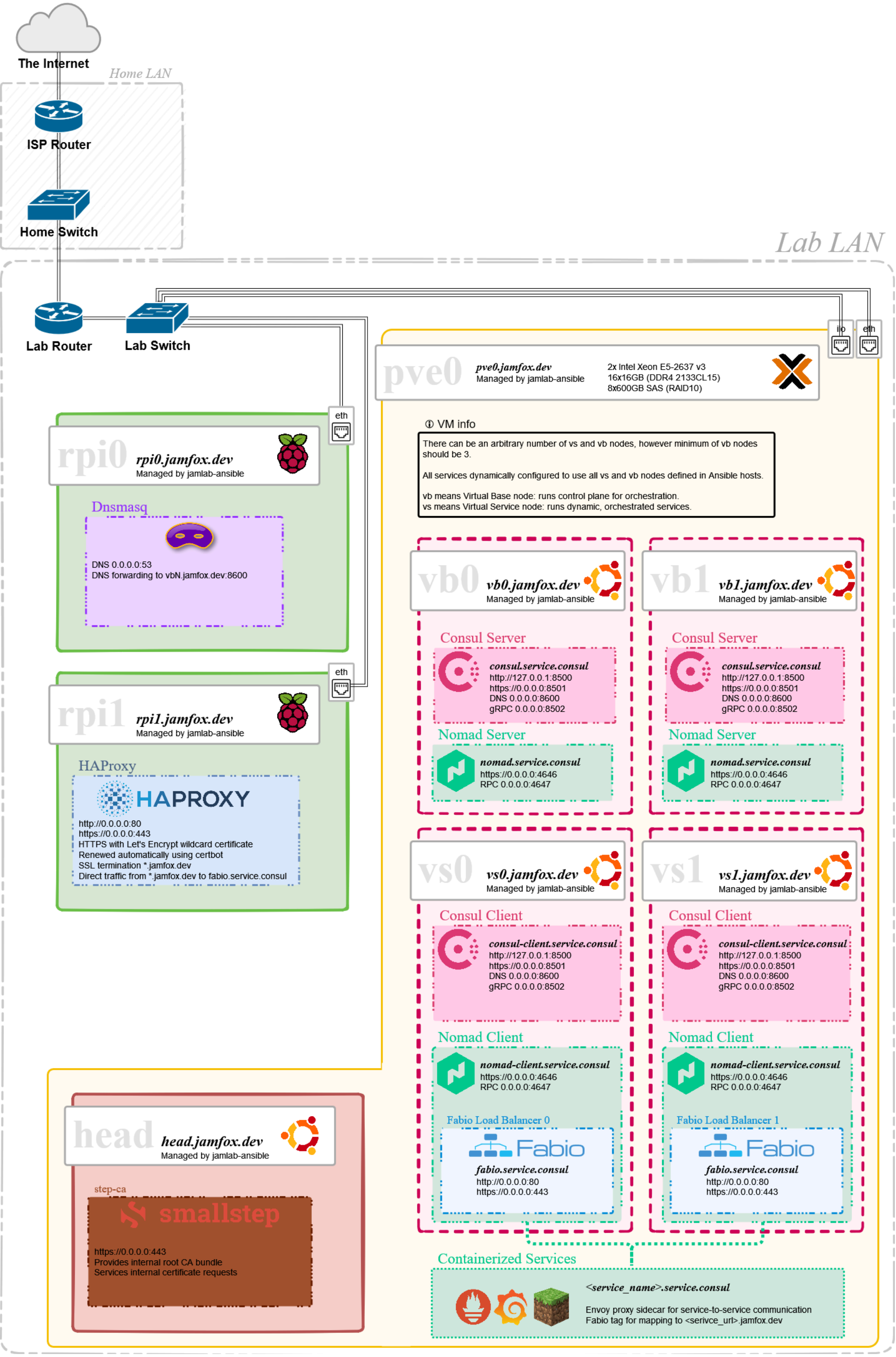
forward to appropriate dynamic service hosts and ports.

- Nomad orchestration: host any service and orchestrate it.
- Consul service discovery: services with multiple instances discoverable from `<service name>.service.consul` address.
- Consul Connect service mesh: Nomad services are able to securely communicate with each other with no manual plumbing configuration.
- NFS shared storage: for general shared storage and Nomad stateful storage.

JamLab repositories:

- jamlab-ansible: Homelab bootstrap and pull-mode configuration management with Ansible and bash.
- jamlab-packer: Packer configurations for building homelab images.
- jamlab-terraform: Terraform configurations for provisioning homelab VMs.

Overview diagram



pve0
pve0.jamfox.dev
Managed by jamlab-ansible
2x Intel Xeon E5-2637 v3
16x16GB (DDR4 2133CL15)
8x600GB SAS (RAID10)

ⓘ VM info

There can be an arbitrary number of vs and vb nodes, however minimum of vb nodes should be 3.

All services dynamically configured to use all vs and vb nodes defined in Ansible hosts.

vb means Virtual Base node: runs control plane for orchestration.
vs means Virtual Service node: runs dynamic, orchestrated services.

rpi0
rpi0.jamfox.dev
Managed by jamlab-ansible

Dnsmasq
DNS 0.0.0.0:53
DNS forwarding to vbN.jamfox.dev:8600

rpi1
rpi1.jamfox.dev
Managed by jamlab-ansible

HAProxy
HAPROXY
http://0.0.0.0:80
https://0.0.0.0:443
HTTPS with Let's Encrypt wildcard certificate
Renewed automatically using certbot
SSL termination *.jamfox.dev
Direct traffic from *.jamfox.dev to fabio.service.consul

vb0
vb0.jamfox.dev
Managed by jamlab-ansible

Consul Server
consul.service.consul
http://127.0.0.1:8500
https://0.0.0.0:8501
DNS 0.0.0.0:8600
gRPC 0.0.0.0:8502

Nomad Server
nomad.service.consul
https://0.0.0.0:4646
RPC 0.0.0.0:4647

vb1
vb1.jamfox.dev
Managed by jamlab-ansible

Consul Server
consul.service.consul
http://127.0.0.1:8500
https://0.0.0.0:8501
DNS 0.0.0.0:8600
gRPC 0.0.0.0:8502

Nomad Server
nomad.service.consul
https://0.0.0.0:4646
RPC 0.0.0.0:4647

vs0
vs0.jamfox.dev
Managed by jamlab-ansible

Consul Client
consul-client.service.consul
http://127.0.0.1:8500
https://0.0.0.0:8501
DNS 0.0.0.0:8600
gRPC 0.0.0.0:8502

Nomad Client
nomad-client.service.consul
https://0.0.0.0:4646
RPC 0.0.0.0:4647

Fabio Load Balancer 0
Fabio
fabio.service.consul
http://0.0.0.0:80
https://0.0.0.0:443

vs1
vs1.jamfox.dev
Managed by jamlab-ansible

Consul Client
consul-client.service.consul
http://127.0.0.1:8500
https://0.0.0.0:8501
DNS 0.0.0.0:8600
gRPC 0.0.0.0:8502

Nomad Client
nomad-client.service.consul
https://0.0.0.0:4646
RPC 0.0.0.0:4647

Fabio Load Balancer 1
Fabio
fabio.service.consul
http://0.0.0.0:80
https://0.0.0.0:443

head
head.jamfox.dev
Managed by jamlab-ansible

step-ca
smallstep
https://0.0.0.0:443
Provides internal root CA bundle
Services internal certificate requests

Containerized Services
<service_name>.service.consul
Envoy proxy sidecar for service-to-service communication
Fabio tag for mapping to <serivce_url>.jamfox.dev