Nomad Workload orchestrator

CONCEPTS

You submit a job; Nomad evaluates it and places allocations on client nodes. Status commands walk that chain.

Job your spec file
↓ evaluation
Eval scheduler run
↓ places
Allocation group on a node
↓ runs
Task container, process
↓ logs, exec

job plan exits 0 for no change, 1 when allocations would change, 255 on error. Handy in CI.

Job, alloc and node IDs can be shortened to a unique prefix.

CONNECT

export NOMAD_ADDR=<*url*>
Set the server address

export NOMAD_TOKEN=<*secret-id*>
Authenticate with an ACL token

export NOMAD_NAMESPACE=<*ns*>
Work in a namespace

nomad server members
List servers and leader

nomad version
Print the CLI version

RUN JOBS

nomad job init
Write example.nomad.hcl

nomad job plan <*file*>
Dry run with a diff

nomad job run <*file*>
Submit or update a job

nomad job run -var <*key*>=<*value*> <*file*>
Set a job variable

INSPECT JOBS

nomad status
List all jobs

INSPECT JOBS (cont.)

nomad job status <*job*>
Summary, allocations, deployment

CHANGE AND STOP

nomad job stop <*job*>
Stop a job

nomad job scale <*job*> <*group*> <*count*>
Set a task group count

nomad job revert <*job*> <*version*>
Roll back to an older version

ALLOCATIONS

nomad alloc status <*alloc*>
Task states and events

nomad alloc logs <*alloc*> <*task*>
Print task stdout

nomad alloc logs -f <*alloc*> <*task*>
Follow logs

nomad alloc exec <*alloc*> <*cmd*>
Run a command in a task

NODES

nomad node status
List client nodes

nomad node status <*node*>
Resources and allocations of a node

nomad node drain -enable <*node*>
Migrate allocations away

nomad node eligibility -disable <*node*>
Stop new placements

EVALS AND DEPLOYMENTS

nomad eval status <*eval*>
Why a placement failed

VARIABLES AND NAMESPACES

nomad var put <*path*> <*key*>=<*value*>
Write a variable

nomad var get <*path*>
Read a variable