Engineer Platform

Infrastructure defined in a repository

Built the company's own infrastructure as 19 Ansible playbooks and 34 roles across 12,065 lines of YAML, converging a live host to a declared state with every play idempotent.

August 2026

How the platform configuration is applied

Implementation overview of provisioning; rebuilding depends on credentials and provider prerequisites.

1.

Declared configuration

Host tiers and group variables state which configuration each machine should receive.

2.

Ansible roles

Ansible roles apply the declared configuration to the selected hosts.

3.

Service definitions

Systemd units and service definitions describe how the provisioned services run.

19

playbooks converging a live host to a written state

Situation. Engineer ApS maintains its own infrastructure as code. The repository covers deployed public services and the configuration of other components, including backup and mail automation whose operational activation is a separate step. Capturing that scope in code makes the setup inspectable rather than dependent on memory.

Task. The whole estate had to be described in a repository rather than in a head, and described in a form that converges a real host rather than documenting one.

Action. The recorded implementation comprises 19 playbooks and 34 roles across 12,065 lines of YAML; these figures describe that recorded stage, not the current repository size. The shape matters more than the size. Composition is data rather than flags: a host sits in a tier group whose variables declare which roles it runs, so provisioning with no arguments converges every host to its declared state. Idempotence is a contract rather than an aspiration — a converged host reports zero changes, and a play that cannot say that is not finished. Four command namespaces keep the promises apart: a gate that touches no host, reports that read one and never change it, provisioning that changes a host to match the repository, and verification that changes a host on purpose and returns a verdict.

Result. The repository supplies the configuration for rebuilding the declared environment, and the parts of it that were only ever true because somebody remembered them are now assertions that fail a run. The cost is real: every change is slower to make than editing a file on the server would be, and a converge that half-applies is worse than one that refuses, which is why a preflight play was later added in front of it. That trade was made deliberately and it has held.

Bring this experience to your project

Use “Infrastructure defined in a repository” as a starting point for a similar challenge in your organisation. The services behind this work are available for your project. Servers sized for what you actually use, moved into the cloud or back out of it, with the monthly bill worked out first.

Explore our services Get in touch →

Look behind the result

Read the situation, decisions and result in this example. We can discuss which parts transfer to your constraints and which need another approach. Infrastructure defined in a repository — 19 playbooks converging a live host to a written state

Explore our services Get in touch →

Want your servers described in code, not in someone's head?

Talk through your project with the engineer responsible for delivery. We agree the scope, price basis and handover in writing before work begins.

Role context: Infrastructure Automation Engineer. It draws on Platform Architecture, DevOps, Infrastructure, Automation & CI/CD, Reliability & Backups, Linux & Servers, System Administration and Documentation. And while you are here, we are rather good at Cloud Infrastructure & Migration, Infrastructure as Code, DevOps & CI/CD Automation and System Administration.

Get in touch Explore our services

One piece of **a longer record** — and it is all on this site.

Every entry is written the same way: the situation, the task, what we did and what changed.

See the whole record

Define the improvement you need

Bring the deployment path, the operational risks and the people on call. We can scope a change with checks, a recovery route and clear ownership.

See what it costs Get in touch →

Want your servers described in code, not in someone's head?

Ask about the decisions behind “Infrastructure defined in a repository”. You will speak with the people who can explain the work and assess your own project.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://platform.engineer.company/portfolio/servers-you-can-rebuild-from-scratch/