Engineer Data

Wrote a scope rule after a credential leak

Wrote a scope rule into the repository after a restructure carried another company's inventory, firewall allowances and prose into it — and kept the quarantined residue under the secret scanner rather than excluding it.

September 2026

a scope rule, after another company's estate arrived

Situation. Work done for a different company rode along through a repository restructure and stayed. What came with it was not abstract: a scratch log holding live plaintext credentials, which sat in the tree past every guard for most of the repository's history; a stale inventory naming that company's host; and a firewall task file opening ports to sixteen of its client networks. None of it ran. That is exactly why it survived — nothing that runs nowhere is ever reviewed again.

Task. The boundary had to become a rule with a test attached, rather than an intention, and the residue had to be dealt with in a way that did not simply hide it.

Action. The rule is now the opening section of the repository's instruction set: this repository manages one company's infrastructure and only that, and another party's host, inventory, firewall allowance, DNS zone or credential does not belong in it — not even disabled, commented out, or parked in a file no playbook imports. The practical test is written next to it: would this company still be responsible for this if the relationship ended. The residue was quarantined into a clearly-named directory rather than deleted, so the history stays legible, and the quarantine is deliberately partial — the two structural linters skip it, and the secret scanner, the vault guard and the emoji check deliberately still read it, because those are the three that would catch the thing that got in. The leak itself produced a linter rule: a custom pattern that flags a credential passed as a command-line flag, which is the shape the leaked one had and which the standard rule set did not match.

Result. The scope boundary is a written rule with a stated test, the residue is visible rather than buried, and the specific shape of credential that got through now fails a commit. The lesson recorded alongside it is the general one: the dangerous artefact is not the one that runs, it is the one that does not, because that is the one nobody reads again.

Bring this experience to your project

Use “Wrote a scope rule after a credential leak” as a starting point for a similar challenge in your organisation. The services behind this work are available for your project. Rules and checks that catch bad data on the way in, and a clear answer to who owns which figure and where it came from.

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. Wrote a scope rule after a credential leak — a scope rule, after another company's estate arrived

Explore our services Get in touch →

Could one project's secrets end up in another's?

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: Security Engineer. It draws on Data Governance, Infrastructure, Automation & CI/CD, Security and Documentation. And while you are here, we are rather good at Data Governance & Quality, Security & Access Management, AI Agent Security and Technical Documentation.

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 a representative dataset, the questions it must answer and the current bottleneck. We can agree how to measure quality, throughput and recovery.

See what it costs Get in touch →

Could one project's secrets end up in another's?

Ask about the decisions behind “Wrote a scope rule after a credential leak”. 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://data.engineer.company/portfolio/wrote-a-scope-rule-after-a-credential-leak/