Shared Automation Initiative

We believe that nothing and no one should be left out of this new path, for that reason we want you to be part of the change…

Learn, Test, Automate and Share

$ whoami

Jonathan Prado

Cloud Architect Lead. Nerd & Rider

Ninja skils

- *Python*
- *Bash*

Ninja Tools

- *Kubernetes*
- *Docker & compose*
- *Terraform*
- *Chaos Engineering*

AWS:

EC2, RDS, EKS, ELB, S3, R53, EBS

Objectives

Our objective is **Improve the “Customer Lifetime Value”**. To do it we need some actions:

- Reduce manual intervention.
- Automate as many manual tasks as possible.
- Share knowledge between areas.

Common problems:

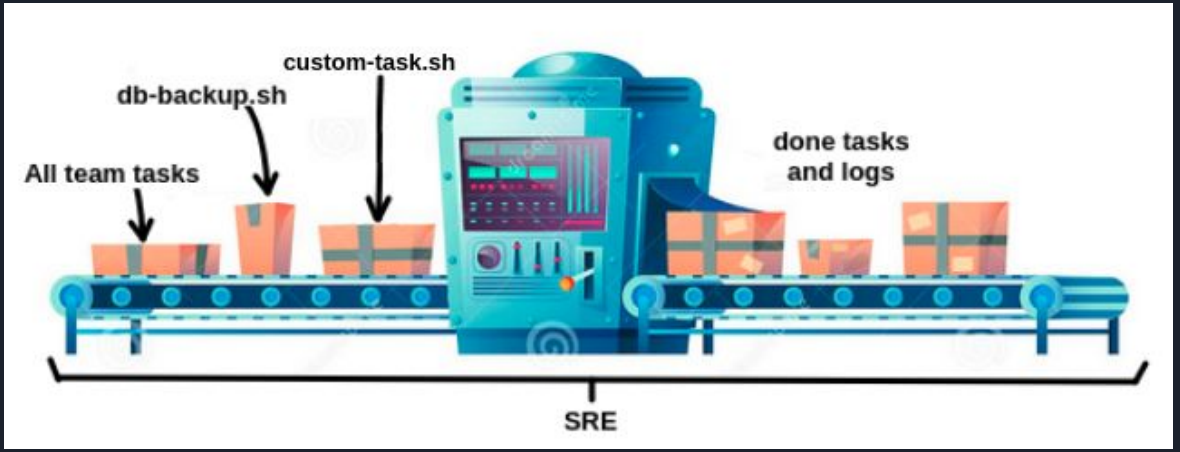
- **Security risks:**
  We have a lot users and access points to secure and maintain, this goes against the security best practices *(Least privilege)*.

- **Administrative tasks overhead:**
  We see a lot of same and recurrently jobs, scripts executions that are executed by humans.

- **Auditory and traceability:**
  We can't see and/or is so complicated follow all execution flows.

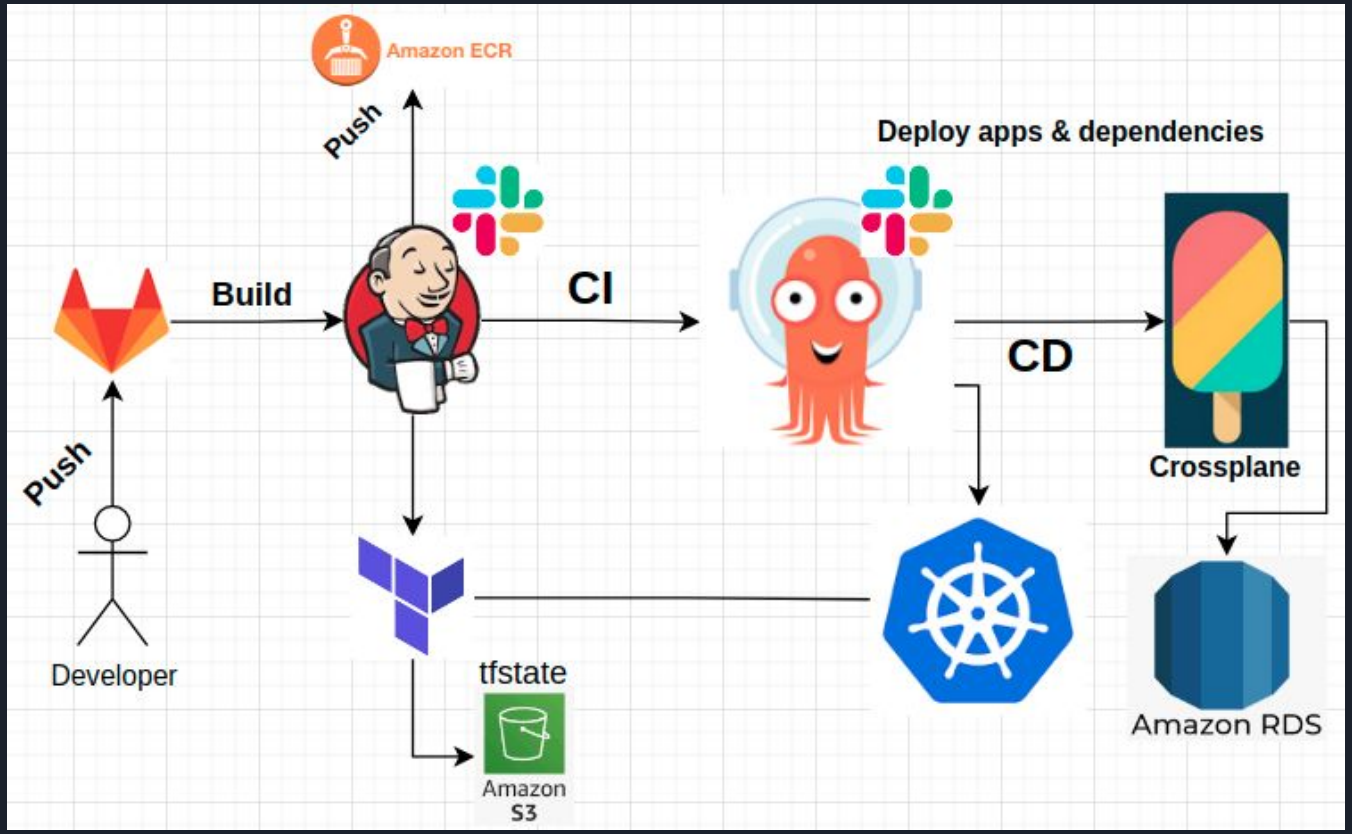
- **Knowledge is not public:**
  We depend on a particular human to do some things, this is a big problem because if it person can't continue doing it no one else can. (Need to use "No code Owner" philosophy)

Our proposal



- **Shared automaton:** sharing information everyone can participate in the improvement of the platform.
- **Centralized jobs/tasks:** Each team can create and maintain our own code and scripts in a centralized repo.
- **Security risks:** Create a only one access point to do all tasks.
- **Administrative task overload:** splitting tasks, there will be no dependencies between teams (we will use a new concept: "put it here and execute it where and when you need").
- **Auditory and traceability:** Centralizing logins we can eliminate risks of shared credentials. And we can do the reverse way (rollback) if something was wrong, because we can see all executed steps/jobs
- **Knowledge is not public:** We don't have secrets, our code is shared with all and it can be maintained by all
- **AaaC:** All as a code, all jobs or tasks need to be code, because it can be improved, managed and save & restore easily and can be checked in a control version platform (like git)

High level design



Demo

Doubts or questions