Information security and data processing

INSIDERLOG

INSIDERLOG

Technical setup

General description

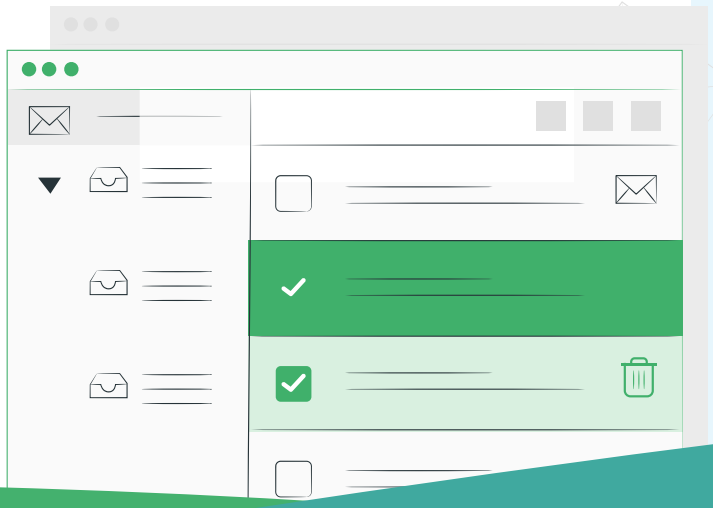
InsiderLog uses Amazon Web Services (AWS) to ensure a safe and reliable environment. The two base components are an EC2-instance, based on an Amazon Linux AMI, and an AWS RDS MySQL Aurora database. These two components are both running within a Virtual Private Cloud (VPC), which makes sure that all traffic within the Virtual Private Cloud is encrypted and inaccessible.

An internet gateway (AWS Tool) makes sure that the traffic to and from the VPC can be controlled. During the setup we make sure that only HTTPS and SSH (optional) traffic is allowed. All incoming traffic to HTTP is redirected to HTTPS.

The InsiderLog infrastructure is installed using CloudFormation templates, which automatically installs everything that is needed. The infrastructure is divided and installed using two stacks; database stack and application stack. During the setup we set up Security groups and assign them rules so that only the necessary resources are able to access them.

Database stack

All data is stored in AWS RDS MySQL Aurora database. This database is solely used by your InsiderLog application, and the setup ensures that only your application is allowed to read and write data from here. By using AWS RDS, InsiderLog takes advantage of AWS services which makes it possible to restore the database to a given point back in time, and ensures that your database receives the required security patches regularly.



Application stack

The application stacks set up an AWS EC2 t3.micro instance that hosts the InsiderLog application, which consists of a backend and a frontend module, served by Docker. By using Docker, the application environment can be controlled, and it doesn't matter what the underlying infrastructure is.

The backend application is written in Java and use frameworks as Spring Boot and Hibernate. The frontend is built using JavaScript framework Angular 6.

Logging

The application stack writes log from the backend and frontend component to Amazons CloudWatch, which is a tool built for handling logging. By default, the logs are deleted automatically after 60 days but this can be changed to any desired timeframe.

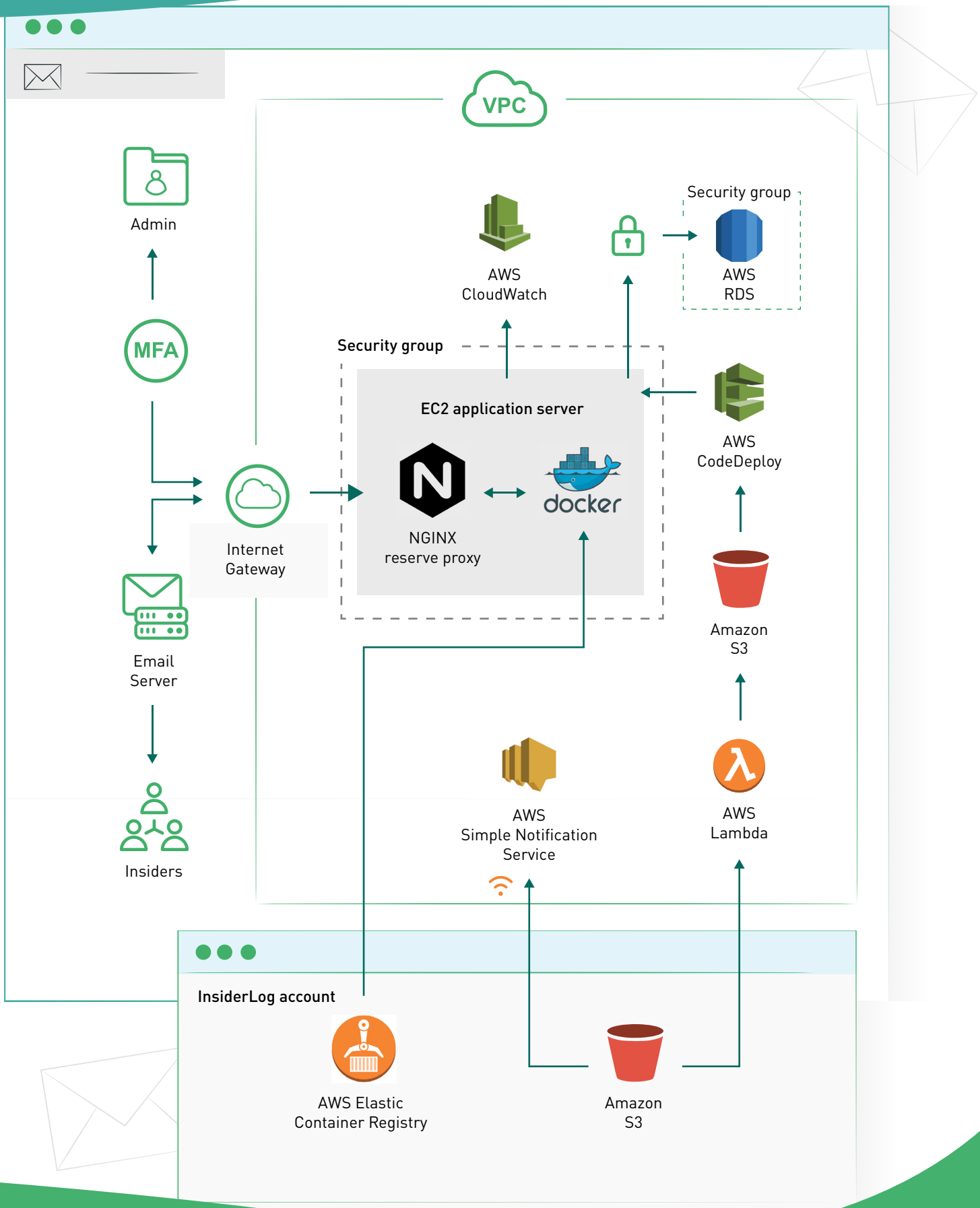
Installation and updates

Frontend and backend deployments are made automatically when we push updates to our central AWS Elastic Container Registry.

We are also able to update the underlying infrastructure of the application. These updates mean changes on the server configuration and version of the tools running the application. When we publish an update for this, your setup gets noticed using AWS Simple Notification Service. Whenever there is an update available, Lambda downloads the new configuration package and AWS CodeDeploy performs the update.

INSIDERLOG

Technical setup



www.complylog.com

INSIDERLOG