Application Infrastructure & Security

Table of Contents

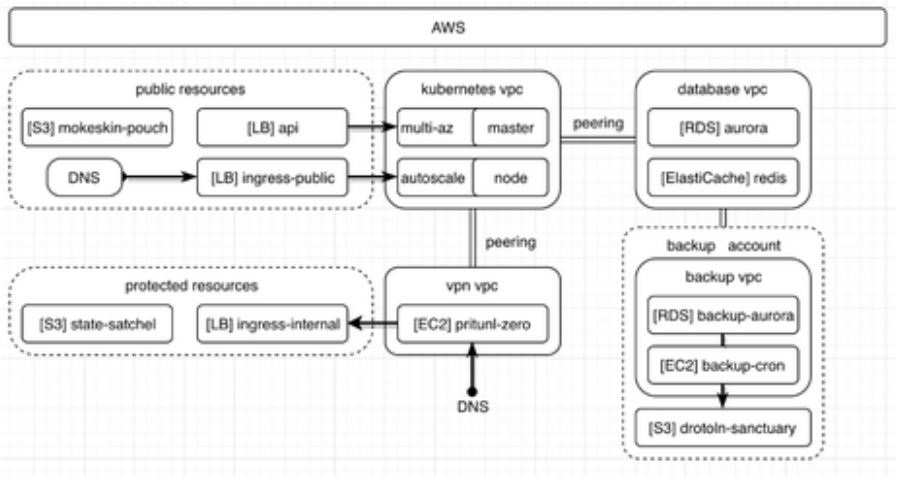
Infrastructure and Monitoring

We have employed the best practices to deploy and maintain cloud-based architecture. Our goals were:

- Effortless maintenance
- Scalability on every component
- Deployment and development best practice cultivation
- Failure tolerance and automated recovery

Design and Implementation

We have designed AWS infrastructure using Terraform automation. All changes in AWS are done via version controlled code, carefully reviewed and tested before applied to production. On top of standard AWS resources, we built a managed multi-master Kubernetes cluster with autoscaling node pool.



We control deployments on Kubernetes cluster via Helm, which helps to organize projects into maintainable bundles and handle their releases and rollbacks if the need arises.

Dictionary

Terraform	Terraform is a tool for building, changing, and combining infrastructure safely and efficiently.
Kubernetes	Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications.
Helm	Helm is a package manager for Kubernetes that allows to package, configure, and deploy applications and services onto Kubernetes clusters easier.
AWS	Amazon Web Services

Logging and Monitoring

We are using Fluentd to deliver logs to log collector of choice, which at time of writing was Loggly.com. We heavily use CloudWatch to setup basic alerts on system resources and Datadog to gather and aggregate runtime data from both AWS and internal Kubernetes sources.

Logs are kept in AWS S3 buckets (up to 30 days) and later retired to AWS Glacier for up to 1 year.

Dictionary

Fluentd	Fluentd is an open source data collector for unified logging layer.
Loggly	Loggly is a cloud-based log management and analytics service that helps to identify, troubleshoot and resolve issues quickly. Loggly is used for infrastructure and software errors and anomalies logging.
CloudWatch	Amazon CloudWatch is a monitoring and management service built for developers, system operators, site reliability engineers (SRE), and IT managers. CloudWatch is used to monitor infrastructure resources and usage.
Datadog	Datadog is a monitoring service for cloud-scale applications. It combines data from servers, databases, tools, and services to present a unified view of an entire stack.

Availability

We use AWS Balancers to maintain a stable entry point to our infrastructure, according to AWS SLA Balancer will remain active even during availability zone outages. Remaining infrastructure is also set up in a way to withstand a loss of availability zone.

In case of complete region loss, we are able to rebuild a complete copy of our infrastructure in another region within hours due to fully automated Terraform plans.

DRP

This is a high-level description of several test cases from our disaster recovery documentation.

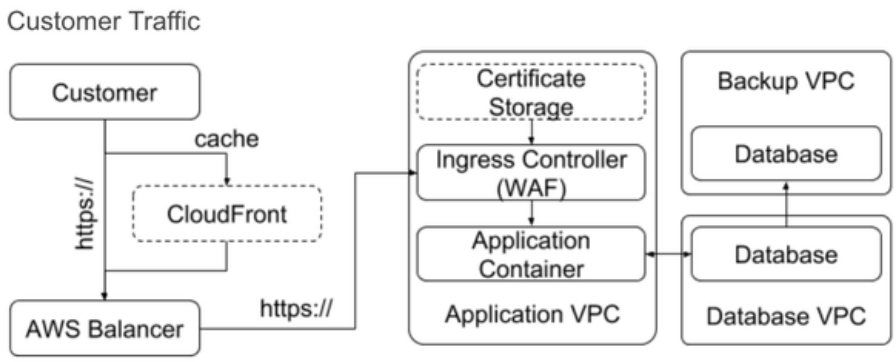
Test Case	Disaster Recovery	Time to recover

AWS region is down	1. Launch another kubernetes cluster using [core] Kubernetes Cluster described documentation
	2. Launch another database cluster using terraform files.
	  1. Ensure that you have valid credentials in those files
	  2. Ensure that you change target regions
	  3. Ensure that you start from "bucket.tf" which creates state bucket for other terraform files to run in
	  4. Ensure that new database cluster is up and running
	  5. Verify connectivity to kubernetes cluster. Review "vpn.tf" file for details
	  6. Upload data from the backup account into the database (see below for details)
	3. Use application terraform files and manifests to recover all applications.
	  1. Ensure that you are using valid *new* credentials and state storage location
	  2. Ensure that you are using valid database credentials
		2 - 5 Hrs

Corrupted database	In a case where data corruption has happened, login to a recovery AWS account, find a valid dump in recovery bucket (drotoln-sanctuary) Apply sequence: • ⚠ Dump corrupted database to a safe location • Destroy corrupted database • Use terraform from projects deployment catalog to recreate the database • Upload the latest non-corrupted dump • While the system is back and operational, perform a diff between corrupted and last-known-good dataset and figure out what to do with that	1 - 2 Hrs
AWS account is closed	Use secondary AWS account Perform full recovery as if the region went down.	2 - 5 Hrs

Dataflow and Backups

Customer Traffic



Customer traffic is always encrypted using TLS encryption. No data is stored in the plain text nor in local filesystems anywhere in the infrastructure. Databases are using AWS provided encryption to avoid any data leaks due to external breaches.

Backups

We are maintaining two tiers of backups:

1. Filesystem snapshots for fast migration to another instance in the same AWS regions.
2. Text dumps for recovery from corruption or migration to another region or different provider:
    1. Daily data backups are kept for up to 7 days old data.
    2. Weekly data backups are kept for up to 1-month-old data.
    3. Monthly data backups are kept for up to 1-year-old data.

Security

Our security framework contains different protection layers, located all around our infrastructure. Here's few of them:

Segmentation

As part of the standard practice, we separated databases and application runtime with internal AWS network segmentation and placed restrictive access rules to deny any unexpected traffic between these parts of the infrastructure.

Web application firewall (WAF)

Web application firewall (WAF) is an application firewall for HTTP/HTTPS applications. It applies a set of rules to an HTTP/HTTPS conversation. Generally, these rules cover common attacks vectors. While proxies generally protect clients, WAFs protect servers. Hence, a WAF protects a specific web application or set of web applications. Engineers consider a WAF a reverse proxy. WAFs may come in different forms, and the effort to perform this customization can be significant and as the application changes. Together with the NGINX Web Server, we use the OpenResty project. OpenResty is a full-fledged web platform

by integrating the standard Nginx core, LuaJIT, many carefully written Lua libraries, lots of high-quality 3rd-party Nginx modules, and most of their external dependencies.

Dictionary

WAF	Web Application Firewall
Nginx	Web Server
OpenResty	OpenResty is a web server which extends Nginx by bundling it with many useful Nginx modules and Lua libraries. OpenResty excels at scaling web applications and services.

Logging and monitoring

We're able to trace requests from end to end across distributed systems, track app performance with auto-generated service overviews, graph, and alert on error rates or latency percentiles (p95, p99, etc.). Also, we automatically collect logs from all services, applications, and platforms, are able to see log data in context with automated tagging and correlation, visualize and alert on log data.

Logs are kept in AWS S3 buckets (up to 30 days) and later retired to AWS Glacier for up to 1 year.

Dictionary

S3	Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance.
Glacier	Amazon Glacier is an online file storage web service that provides storage for data archiving and backup.

Docker

A big part of any organization's risk assessment process is to be aware of and gain visibility into vulnerabilities in the software being used. We use Clair as one of the core components to check our containers security. Clair is an open source project for the static analysis of vulnerabilities in application containers. In

regular intervals, Clair ingests vulnerability metadata from a configured set of sources and stores it in the database. We use the Clair API to index container images; this creates a list of features present in the image and stores them in the database. We use the Clair API to query the database for vulnerabilities of a particular image; correlating vulnerabilities and features is done for each request, avoiding the need to rescan images. When updates to vulnerability metadata occur, a notification is sent to alert systems that a change has occurred.

Our goal is to enable a more transparent view of the security of the container-based infrastructure.

Dictionary

Docker	Docker is a computer program that performs operating-system-level virtualization.
Clair	Clair is a tool to monitor the security of the app and Docker containers.