Ronak (RB)

Contact Me • US Born • **Github** • **SO** • **LinkedIn** • **Blog** • • Resume (**HTML**, **PDF**, **Slides**)

Engineer with architect, frontend, backend, AWS, remote, consultant, and security experience and worked in adtech, security, gambling, and IoT industries (details available upon request)

Work

Your company - Remote

- I would love to solve problems for you. **Hire me**.

Geolocation - SRE / DevSecOps / DevOps (2022 - Present) - Remote

- Lead a migration project from CloudFront (usage cost) to CloudFlare (flat cost) savings a lot of money and decreasing latency for clients and maintaining IaC in terraform
- Lead a Single Point of Failure project to compile a list of all known vendors, billing, across all teams, assign owners, and create action items to reduce SPOFs across the org
- Lead a migration project from EC2 to EKS by working with teams, reducing blockers, forecasting, presenting in front of VPs, and coming up with scalable solutions to ease the transition
- Lead a project for IaC only AWS accounts to introduce best practices and security out of the box for 3rd party contractors which served as an example for future AWS accounts
- Mentored multiple members internally in ops, infrastructure as code, xy problem, and overall scalable, secure, solutions
- Rearchitected atlantis (terraform CICD) to be a pleasant experience and thus eliminating a separate project to recreate those workflows in a bespoke system that would need to be internally maintained
- Introduced architectural design records (ADRs) to solidify decisions per repo and participated in org-wide ADRs for security and infrastructure
- Introduced policy-as-code using tools like OpenPolicyAgent (OPA), Conftest, Gatekeeper and applied in terraform code and in kubernetes
- SAST tools (tflint, tfsec / trivy) for terraform-cicd pipeline to resolve security issues prior to merge
- Reduced Critical and High security vulnerabilities across the organization using a combination of best practices documentation, trainings, cloud custodian policies, and Service Control Policies (SCPs) across the org
- Introduced renovatebot for automatically keeping dependencies up to date
- Enabled colocation of terraform resources by introducing an internal account-map terraform module
- Created an internal FAQ that has reduced common questions in many slack channels and is now maintained by many teams
- Noted that devs did not take advantage of AWS support and setup AWS Chat Bot and AWS Support Chat Bot to make it easier for users
- Worked with IT, security, internal auditors, external auditors to get to root issues to come up with solutions to streamline auditing
- Removed all persistent administrators from many production accounts by creating new least-privilege roles and migrating users
- Automatically tagging resources helped with forecasting costs per teams
- Setup AWS Budgets (cost and usage) for expensive services to notify slack/email via SNS if costs are rising faster than initially forecasted

RB Consulting LLC - CEO / Principal Staff (2021 - Present) - Remote

- Consulted for Fortune 500 clients, midcap, and small companies
- Helped manage 100s of open source terraform modules in terraform-aws-modules and cloudposse

- Everything, absolutely everything in terraform (infrastructure as code)
- Experience getting into Amazon's Marketplace by using AWS Control Tower Account Factory for Terraform
- Setup from the ground up: SAML, AWS SSO with Okta, Gravitational's Teleport, AWS Client VPN, Spacelift (terraform automation), Datadog, OpsGenie, private/public R53 split-horizon, Cloudtrail, AWS WAF, delegated IAM roles, AWS Shield, GuardDuty
- Configured and maintained EKS (with and without bottlerocket) and ECS Fargate
- All helm releases with IRSA roles built with terraform using the helm provider
- Used tooling such as tflint, tfsec, checkov, infracost
- Tested terraform using terratest and Open Policy Agent rego policies
- Spearheaded architectural design decisions (ADRs) in repos and related mermaidjs diagrams

Security & Adtech - SRE / DevSecOps / DevOps (2019 - 2021) - Remote

- Lead a team to implement a zero trust solution to replace a costly VPN
- Greatly reduced manual work by automating items by first documenting manual work, creating tickets, and tackling items
- Automated packer Golden AMIs using Buildkite pipelines
- Worked with IT to setup and migrate employees to AWS SSO with Okta integration
- Reduced costs and added governance using **CloudCustodian**
- Installed **Atlantis** on fargate for terraform pull request automation
- Secured static jekyll company blog using cloudfront, private s3, with lambdas to set secure headers
- Gave presentations across teams to improve infrastructure best practices
- Modularized terraform and leveraged the registry to follow DRY principles across the organization
- Expanded documentation and least privilege to reduce Ops requests from 10/day to 10/week
- Mitigated several high level security incidents
- Deployed services in Kubernetes using Helm charts
- Built services in typescript / nodejs using Hapi

DFS - SRE / DevSecOps / DevOps (2018 - 2019) - In Office and Remote

- Brought **CloudCustodian** as a governance tool
- Used Akamai CDN and AWS WWAF to protect servers from bot traffic and DDoS attacks
- Built microservices using AWS's ALB, TG, and lambdas using python and terraform
- Coded GAppsScript to grab deployment history from OctopusDeploy using a Subscription and webhook
- Advised dev teams on testing locally, decreasing feedback loop, improving services, improving pipelines
- Wrote a lambda to forward Slack requests to another lambda allowing creation of many slack commands
- Improved patching process of AMIs by automating Rapid7 InsightVM scans
- Created VM templates for Windows and Linux in datacenters using Packer
- Monitored microservices in ASGs and ECS docker containers using Datadog and ELK stack dashboards
- 3rd party upgrades, Chef, reviewing PRs, daily requests from devs, blue/green and rolling deploys

Adtech & Sales CMS - DevOps / DevSecOps / Full Stack Engineer (2016 - 2018) - In Office and Remote

- Architected and built a multi-million dollar, highly scalable, and concurrent **Node.js** app using **koa** and **pm2** in only 8 weeks with 30% test coverage. The app handles over 150M req/mo and allowed our team and product to rapidly expand.
- Designed and helped build a proxy API for Sales written in Java **Springboot** with a TypeScript frontend
- Uncovered over 60 high severity issues in legacy apps by manual and automated testing using **Burp**, **w3af**, **sqlmap**, and custom scripts
- Containerized our apps in **docker** and in **Vagrant** VMs using Ansible to provision
- Setup local pre-commit hooks, built **Jenkins** CI pipelines to run unit tests and **static code analysis** tools to auto merge and auto deploy
- Load tested apps using apache bench, wrk, loader.io, jmeter, and blazemeter
- Designed and created a pre-buy app in jQuery, bootstrap, font awesome, **flask**, with docker deployment

- Maintained and improved upon a once complicated ETL left behind by a previous team
- Gave presentations on TDD (Test Driven Development) and CICD principles

Networking & Firewall - DevOps / DevSecOps / Automation Engineer (2013 - 2016) - In Office and Remote

- Implemented a scalable solution to test **ExtJs** frontend code using **Jasmine** tests via **Siesta**
- Built a remote python agent to act as a test runner and integrated results into a common database. Ran tests in Jenkins based on a commit trigger.
- Won an award for cost savings by creating a self correcting Smartsheet using **Smartsheet-Data-Tracker**
- Setup VMs using Packer using provisioners, veewee for iso automation, and deployed on vSphere and created local development environments using Vagrant
- Managed and monitored LDAP and Radius servers for network firewall authentication
- Wrote packet manipulation scripts using **Scapy** to circumvent IDS/IPS firewall software
- Scripted reproduction scenarios for **Heartbleed** and other security issues

IoT - Engineer (2011 - 2013) - In Office

- Created self healing mesh network for buoys that sends real time data and field tested in a boat
- Programmed custom **Arduino** boards in C++ to work with GPS and Digi modules
- Developed functions around **Yorick** legacy code written by **USGS** for the **EAARL** (Experimental Advanced Airborne Research LIDAR) project
- Trained USGS personnel via presentations on how to develop and build upon the EAARL project
- Read **LIDAR** (Light Detection and Ranging) papers under a PhD professor's tutelage to derive important variables for a radiometric calibration to generate light reflectance graphs

Skills

Open Source

- Won a company award for fixing a spreadsheet sync failure in **Smartsheet-Data-Tracker**
- npm package **terraform-plan-applier** to apply terraform with criteria
- Fixed a gzip issue with **pm2-logrotate** and added a time zone feature
- Device support for specific switches in **netmiko**, a **Paramiko** based SSH library
- Added a mobile friendly **pandoc_resume** and made its `Makefile` more modular
- Homebrew, choco, scoop support for tools I like such as **iam-policy-json-to-terraform**
- Contributed the `SourceAMICreationDate` to **Packer**
- Contributed to the terraform AWS provider

Personal Projects

- Built **Chrome extension manager** to install browser extensions using the cli
- Wrote **curltopy** to convert browser tools' network curl calls to python requests
- Completed a dockerized **weather app** challenge using Django
- Completed a **questionnaire app** challenge in Flask and jQuery

Languages

Python: I have worked with Python for years. I've created APIs in Flask and Falcon, written many scripts, lambdas, and built a Django app.

Golang: Contributed to Packer, Terraform AWS Provider, and atlantis projects.

Node.js & TypeScript: Built a scalable API using koa and **pm2** utilizing es7 features like block scoping and async / await and used mocha style jasmine tests to unit and integration test the app. Also lambdas and google scripts.

Java: Built a microservice using Springboot.

Dabbled: **D3**, Golang, Ruby, Bash, Kotlin and personal cloud (AWS and Azure) accounts. Envoy/Istio. Podman instead of Docker. Used gRPC and swagger for fun. I try to stay language agnostic / polyglot as much as possible.

Education

- Bachelor of Science
- AWS labs on **quiklabs** and **acloud.guru**
- Learned Kubernetes and helm using courses on **katacoda.com**