



AWS Infrastructure Roadmap

October 6, 2023

DRAFT

DRAFT

Section 01

The Current State

DRAFT

A Survey Of Resources In The AWS Account

A Survey Of Resources In The AWS Account

- ❖ No sandbox ... development, customer implementation, performance/scalability testing, making safe mistakes.
- ❖ Not yet fully set up for isolation of customer tenant operating environments.
- ❖ Unified costing structure.
- ❖ Unnecessary use of two Availability Zones.
- ❖ Open network rules.
- ❖ S3 buckets not locked down.
- ❖ Spare/Junk resources (subnetworks).
- ❖ Unencrypted disk snapshots.
- ❖ No shared disk resources for common artifacts.
- ❖ No Cross-Tenant awareness (e.g. common SnapCor metrics db)
- ❖ Unmonitored

DRAFT

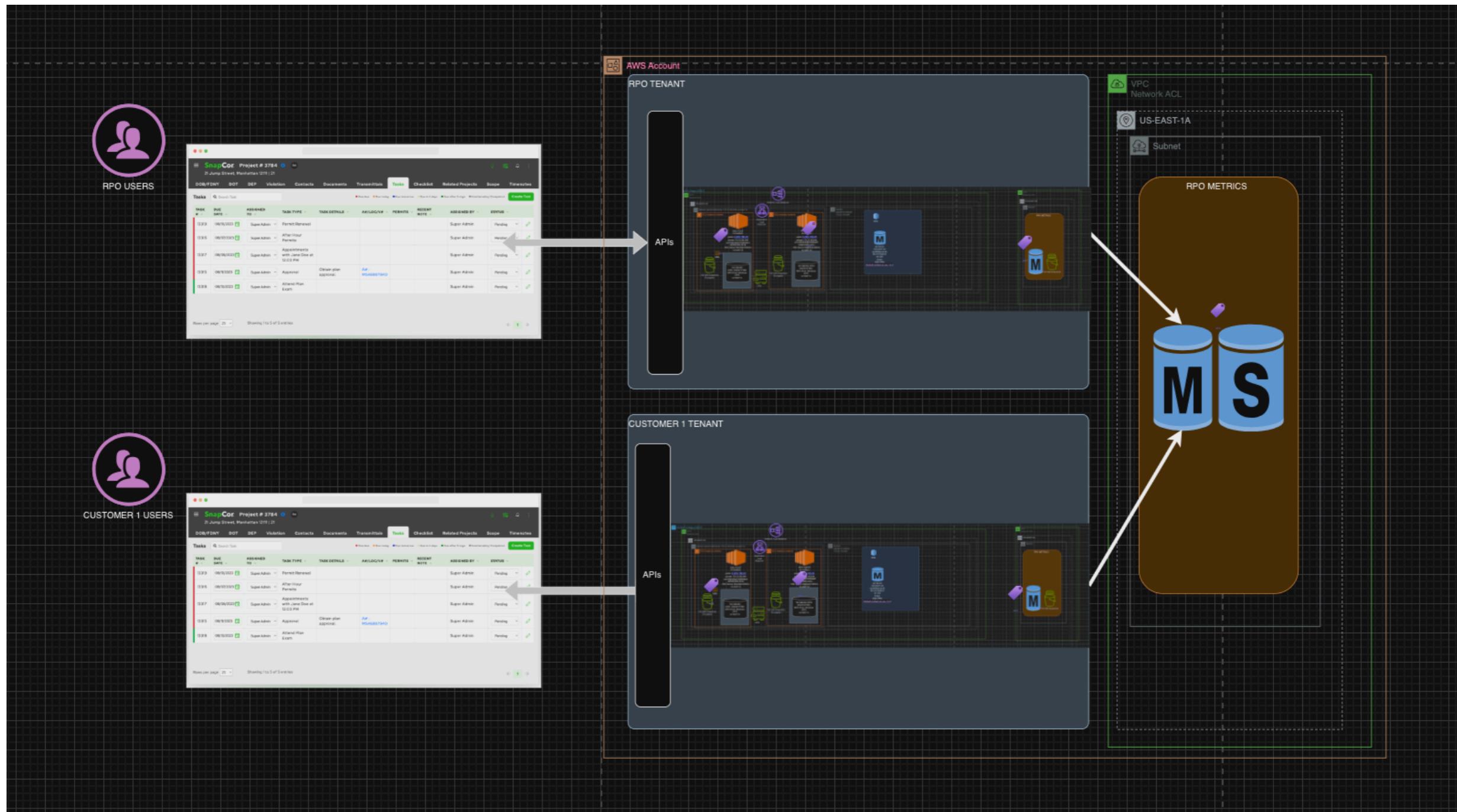
Section 02

AWS Infrastructure

Vision & Roadmap

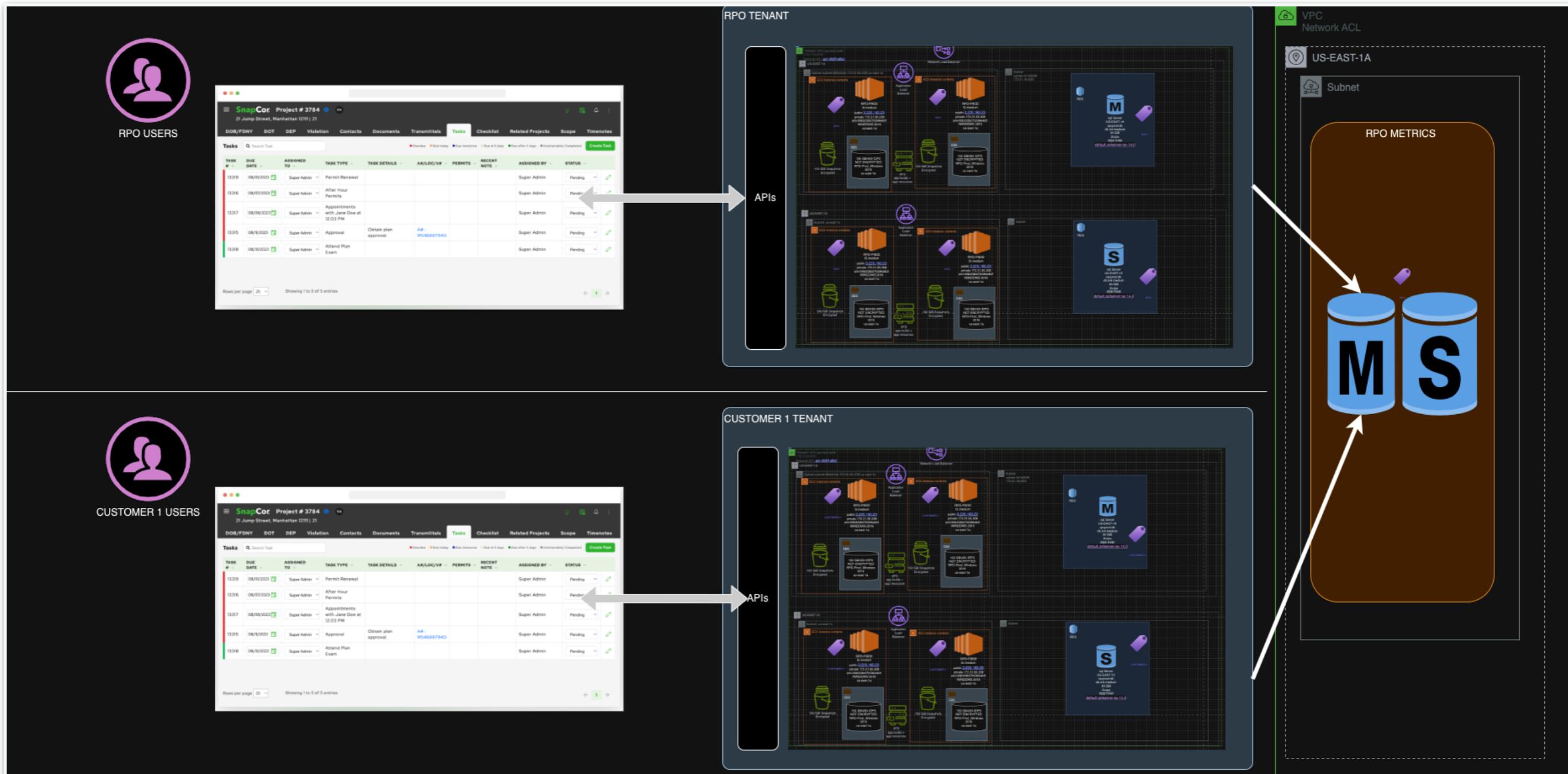
DRAFT

MULTI-TENANCY - BASIC RESILIENCY



DRAFT

MULTI-TENANCY - HIGHER RESILIENCY



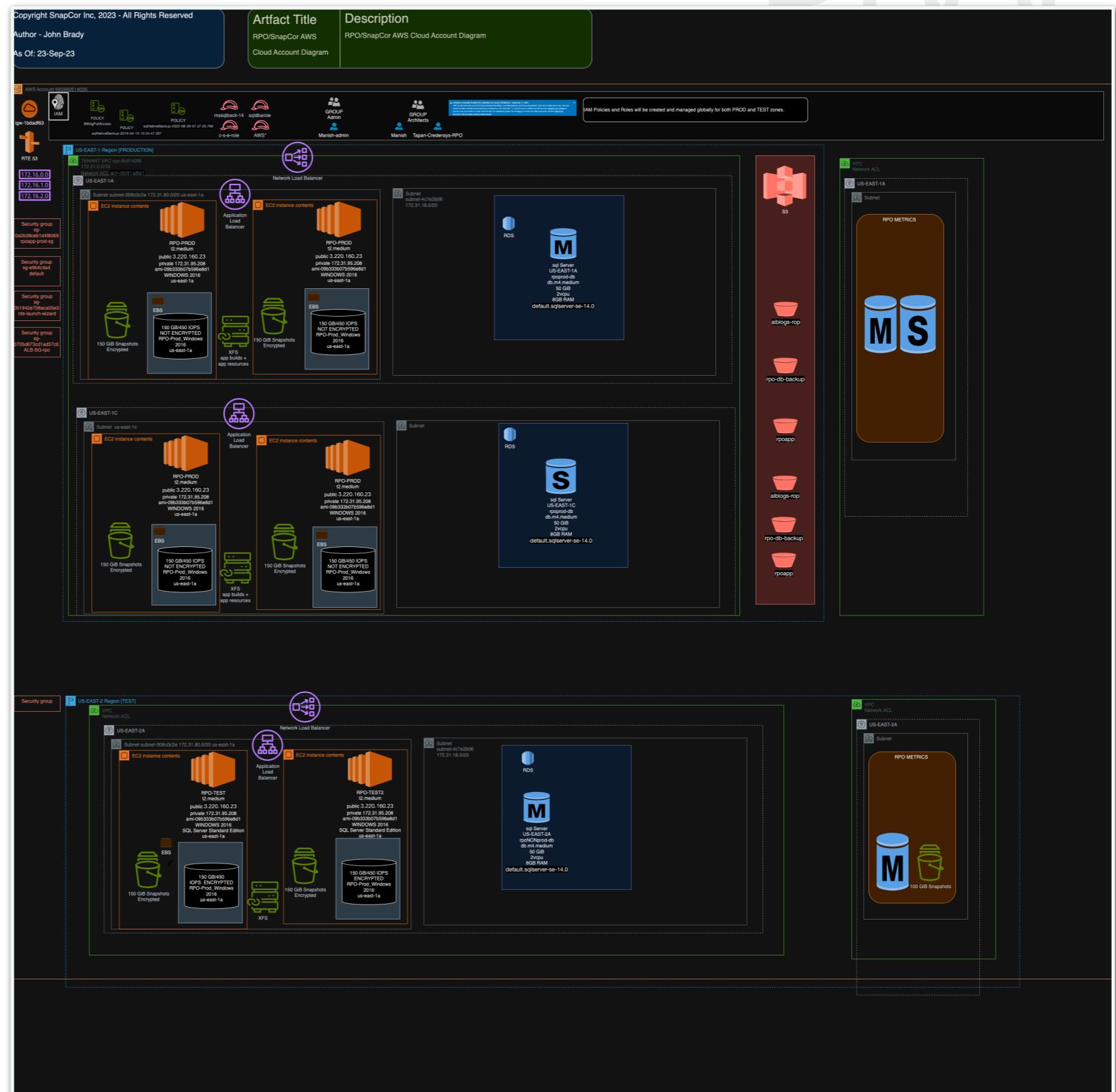
MULTI-TENANCY CAPABILITIES NEEDED

- ❖ Safe sandbox for operations, customer implementation, performance/scalability testing.
- ❖ Robust isolation of production customer tenant operating environments.
- ❖ Manageable isolation of cost by customer tenant.
- ❖ Customizable resource provisioning by customer tenant.
- ❖ Infrastructure security locked down:
 - ❖ Network ports, ACLs set for minimal access.
 - ❖ Administrator access to database using best practices
 - ❖ Deployment through automation

MULTI-TENANCY CAPABILITIES OPTIONAL

- ❖ Some redundant resources for higher resiliency.
- ❖ Multiple Availability Zone strategy with some redundant resources for higher resiliency.

INFRASTRUCTURE

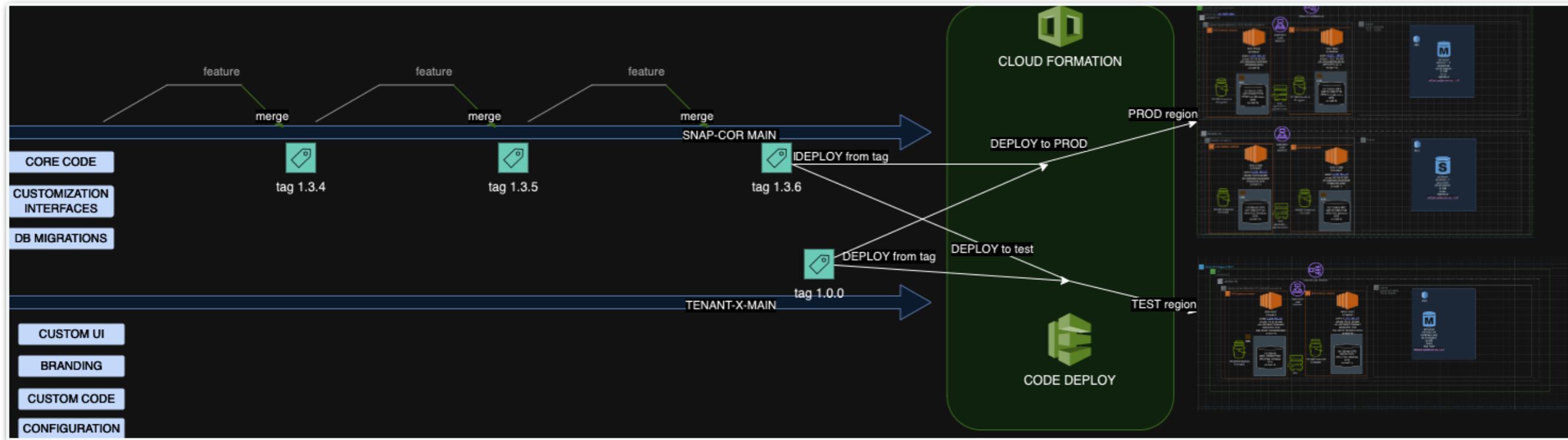


DRAFT

INFRASTRUCTURE CAPABILITIES NEEDED

- ❖ Safe sandbox for development, customer implementation, performance/scalability testing.
- ❖ Security locked down: best practices implemented and monitored.
- ❖ Code endpoints reviewed for authentication/authorization (see references, Activate Account allows anonymous access)
- ❖ Resilient infrastructure (survives EC2 failure, Availability Zone failure)
- ❖ Robust isolation of customer tenant operating environments.
 - ❖ Load Balancers, VPC, Subnetworks, Disk Instances, Database instances.
- ❖ Manageable isolation of cost by customer tenant.
- ❖ Main/Secondary architecture for RDS Databases to increase resiliency in case of AZ failure.
- ❖ Shared disk resources for common code for simple common framework updates to all tenants.
- ❖ Common meta-store for valuable operational metrics that are collected from all tenant workflows.
- ❖ Monitoring/Alerting.
- ❖ Comprehensive network security.
- ❖ Best practice use of encryption.
- ❖ Billing isolation per tenant.
- ❖ Infrastructure as code, templating, through Cloud Formation and possibly other technologies.
- ❖ Automation where possible. e.g. software deployment, testing.

CODE INTEGRATION + DEPLOYMENT



DRAFT

CODE INTEGRATION + DEPLOYMENT CAPABILITIES NEEDED

- ❖ Industry best practice efficient source control management system and practices.
- ❖ Independence enabling capability for developers with quality checks, automation hooks.
- ❖ Enable independent product version/custom capability roadmap per tenant.
- ❖ Robust isolation of customer tenant customizations and configurations.
- ❖ Manageable isolation of cost by customer tenant.
- ❖ Customization enabling code written into the common core code repository. Customizations implemented by customer in tenant specific repositories.

DRAFT

Section 03

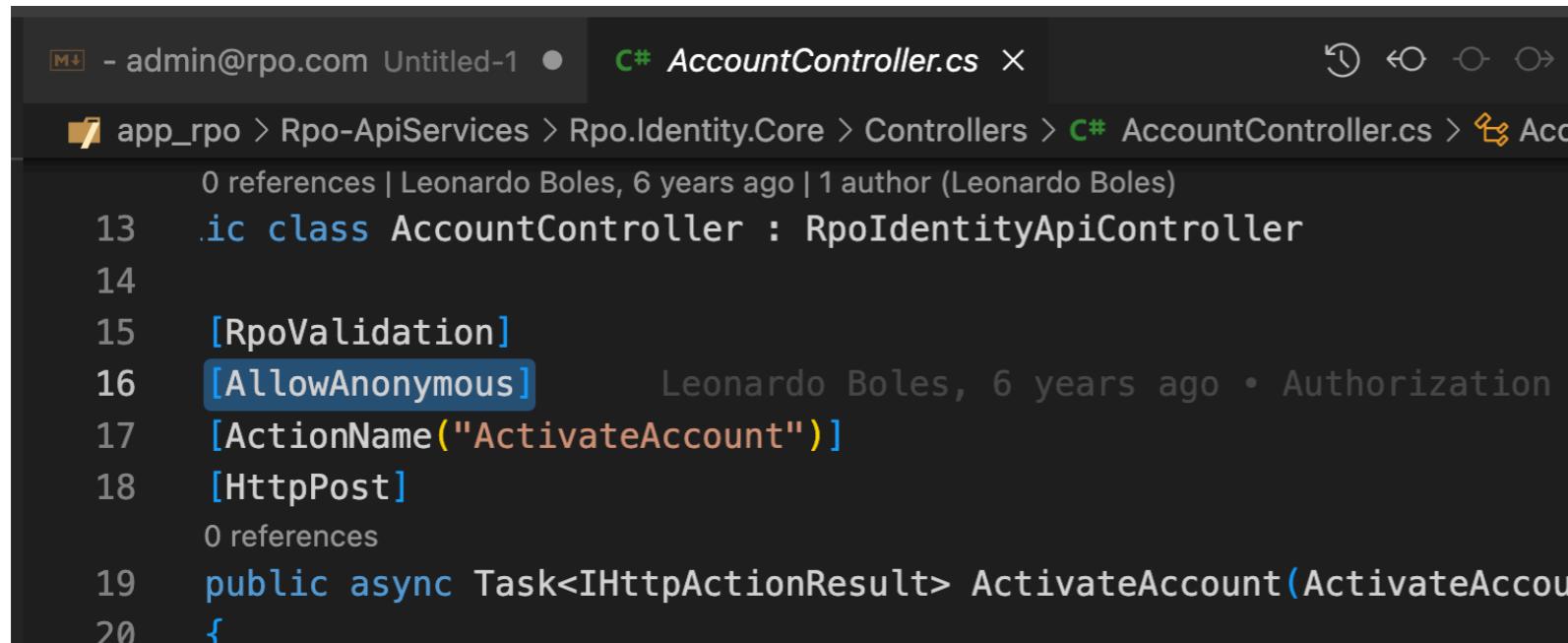
REFERENCES

DRAFT

SQL Server Licensing - <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/SQLServer.Concepts.General.Licensing.html>

FSx shared storage - <https://aws.amazon.com/fsx/>

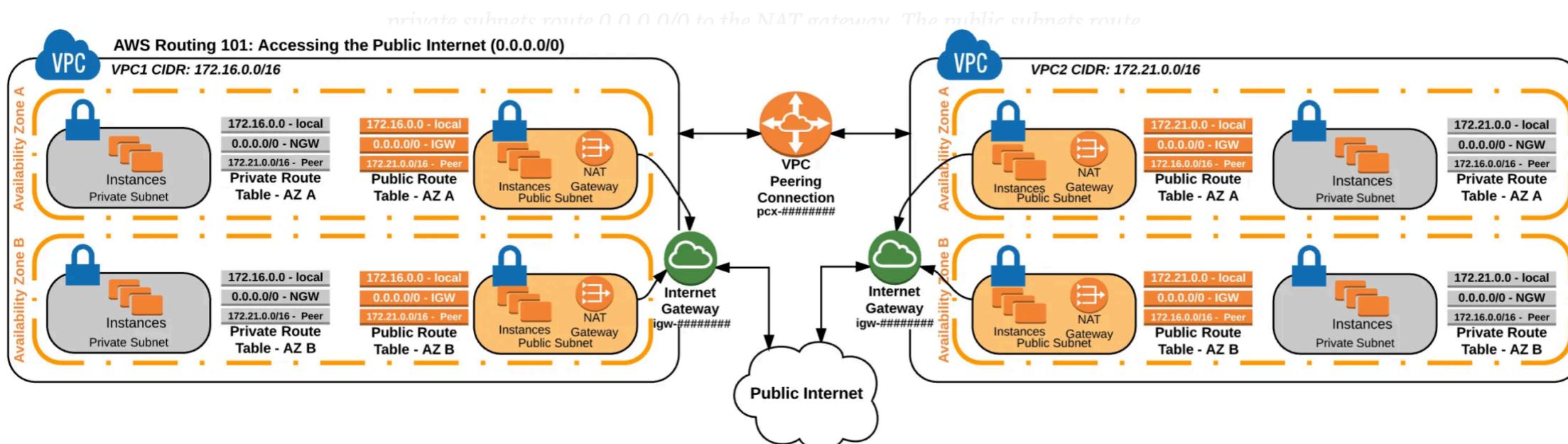
Networking - [LINK](#)



```

M+ - admin@rpo.com Untitled-1 ● C# AccountController.cs ×
7 app_rpo > Rpo-ApiServices > Rpo.Identity.Core > Controllers > C# AccountController.cs > 📁 Acco
0 references | Leonardo Boles, 6 years ago | 1 author (Leonardo Boles)
13 ic class AccountController : RpoIdentityApiController
14
15 [RpoValidation]
16 [AllowAnonymous] Leonardo Boles, 6 years ago • Authorization a
17 [ActionName("ActivateAccount")]
18 [HttpPost]
0 references
19 public async Task<IHttpActionResult> ActivateAccount(ActivateAccou
20 {

```



address on the VPN device within your on-premise datacenter.