SVS315

Building multi-tenant applications with AWS Lambda and AWS Fargate

Tod Golding

Senior Principal Partner Solutions Architect
AWS SaaS Factory

Anton Aleksandrov

Principal Serverless Solutions Architect
AWS

Picking a compute model

Does it fit your scaling profile?

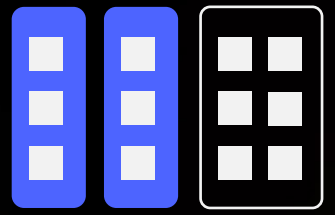
How will it influence your cost profile?

How does it align with the profile of your team?

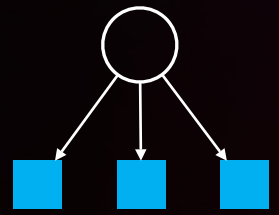
What kind of workloads will you need to support?

What is your starting point?

Key architecture considerations

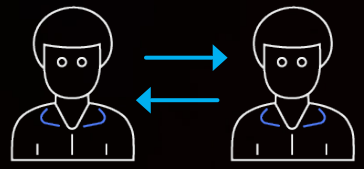


Which deployment models will you support?

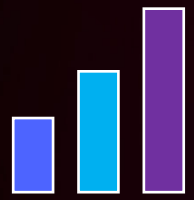


How will you route tenant loads?

How do you automate application configuration/deployment?

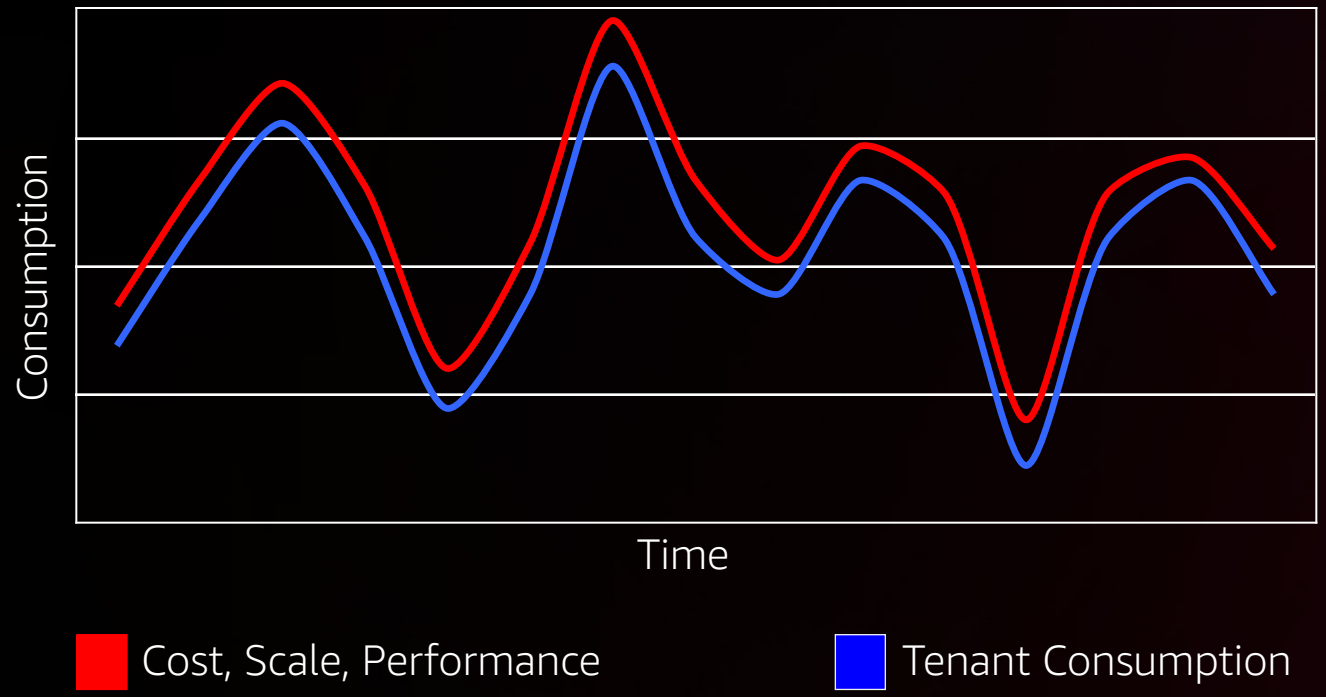


How will you isolate tenant resources?



How will you support tiered tenant experiences?

The SaaS and serverless fit



- Agility
- Cost optimization
- Operational efficiency
- Blast radius
- Focus on IP

Compute trade-offs

AWS Fargate

Pros

- Community tooling
- Selection (Amazon EKS, Amazon ECS, Fargate)
- Developer learning curve
- Deployment/management tooling
- Built-in multi-tenant constructs

Cons

- Isolation considerations
- Scaling tuning
- Idle compute resources

Serverless

Pros

- Fine-grained deployment
- No idle resources/scaling
- Third-party extensibility
- Amazon API Gateway integration
- Layers for shared concepts

Cons

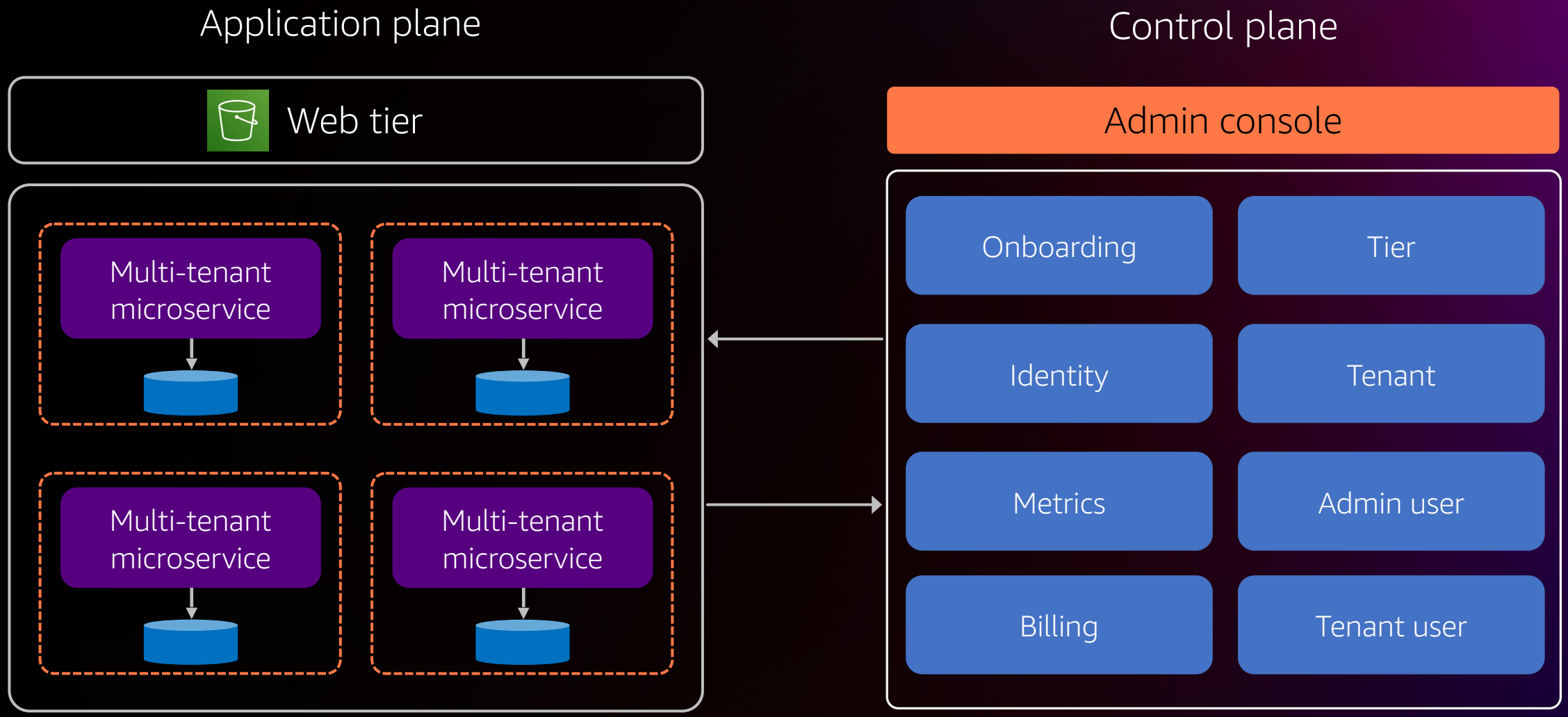
- Developer learning curve
- Concurrency management
- Silo proliferation of functions

Two models, one mindset

- Deployment
- Isolation
- Noisy neighbor
- Tiering
- Routing

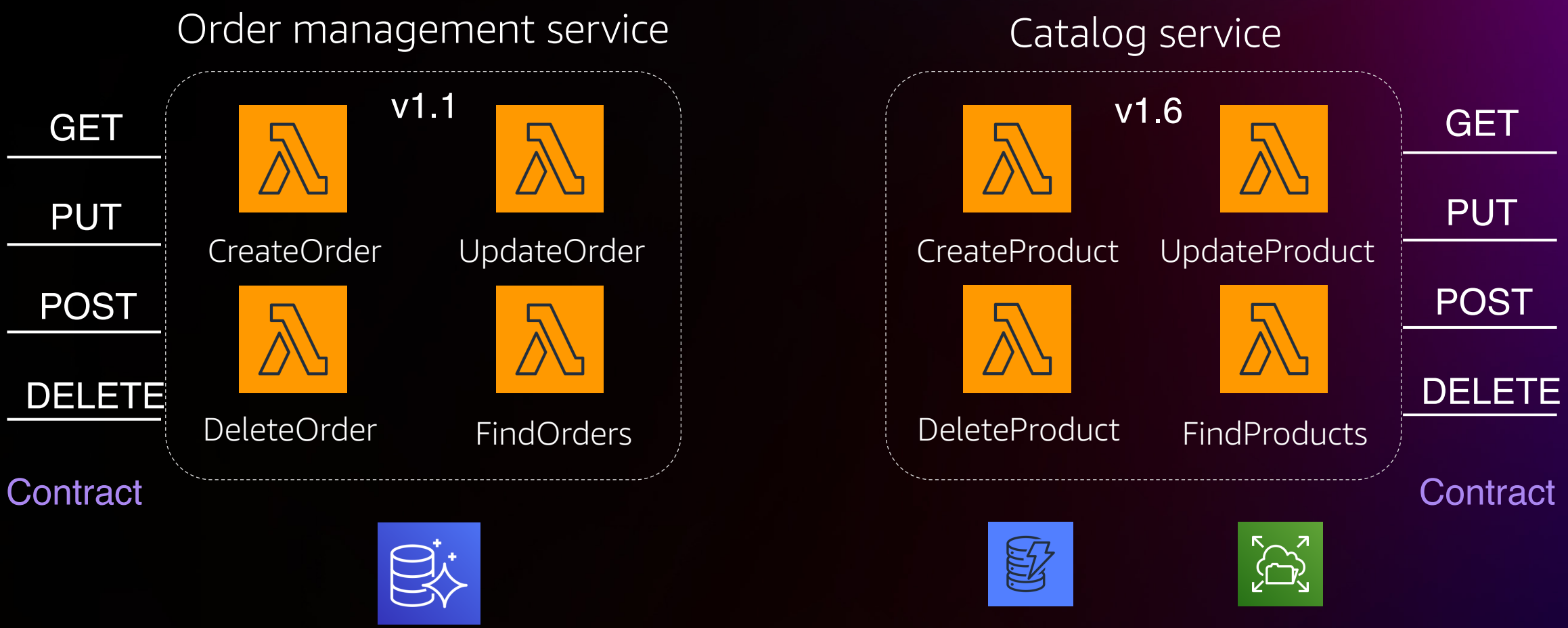
Fully managed execution and scale

Common to all serverless models

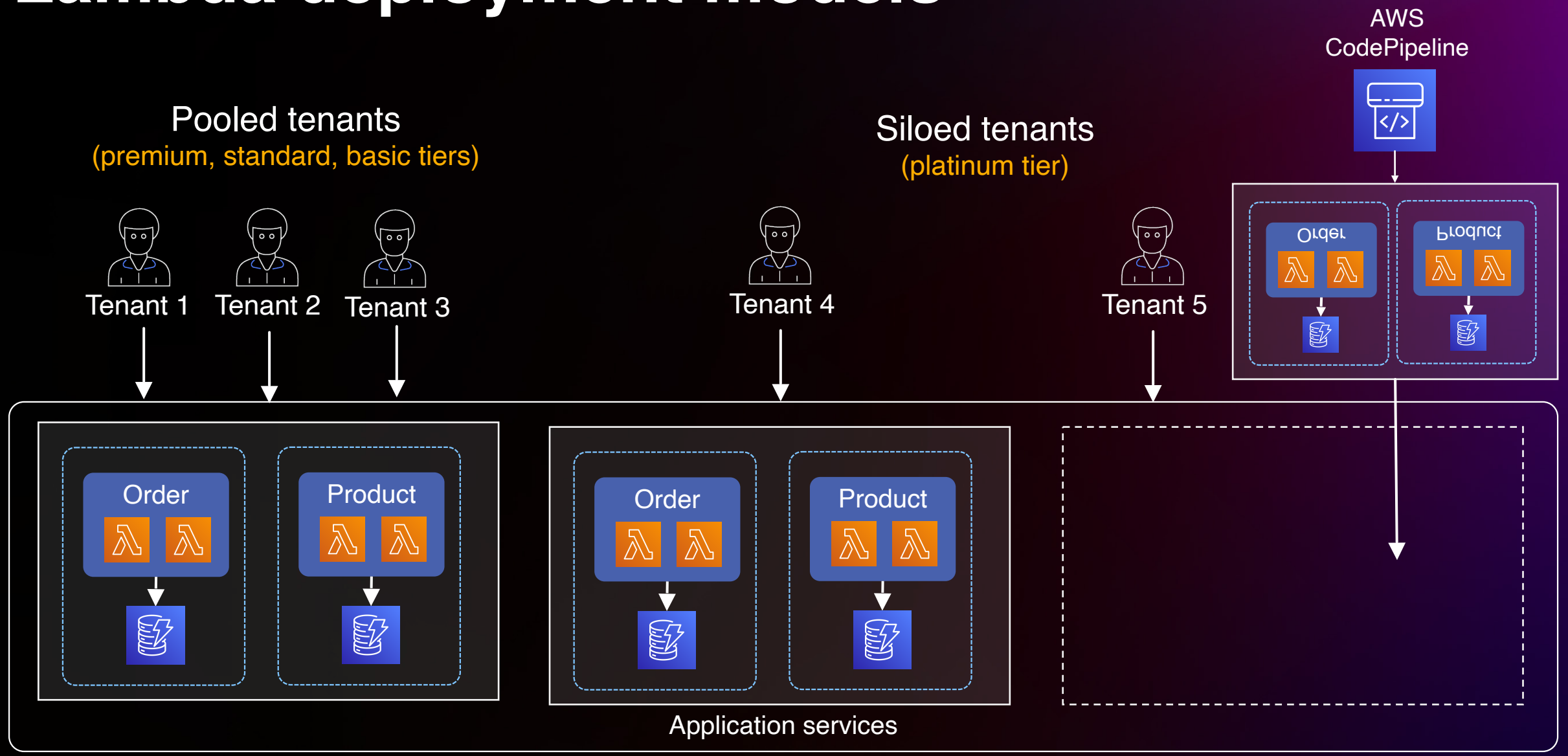


The multi-tenant Lambda model

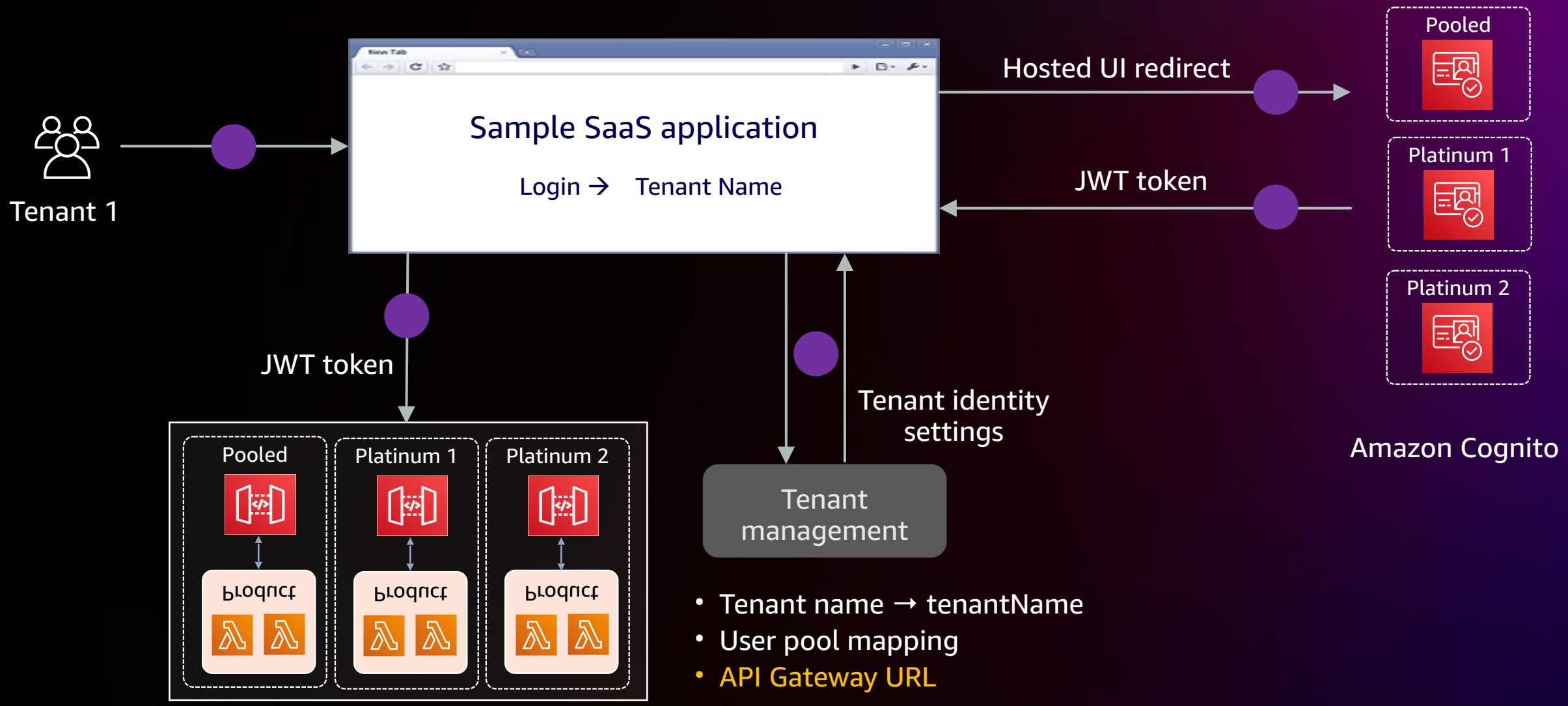
Composing microservices from functions



Lambda deployment models

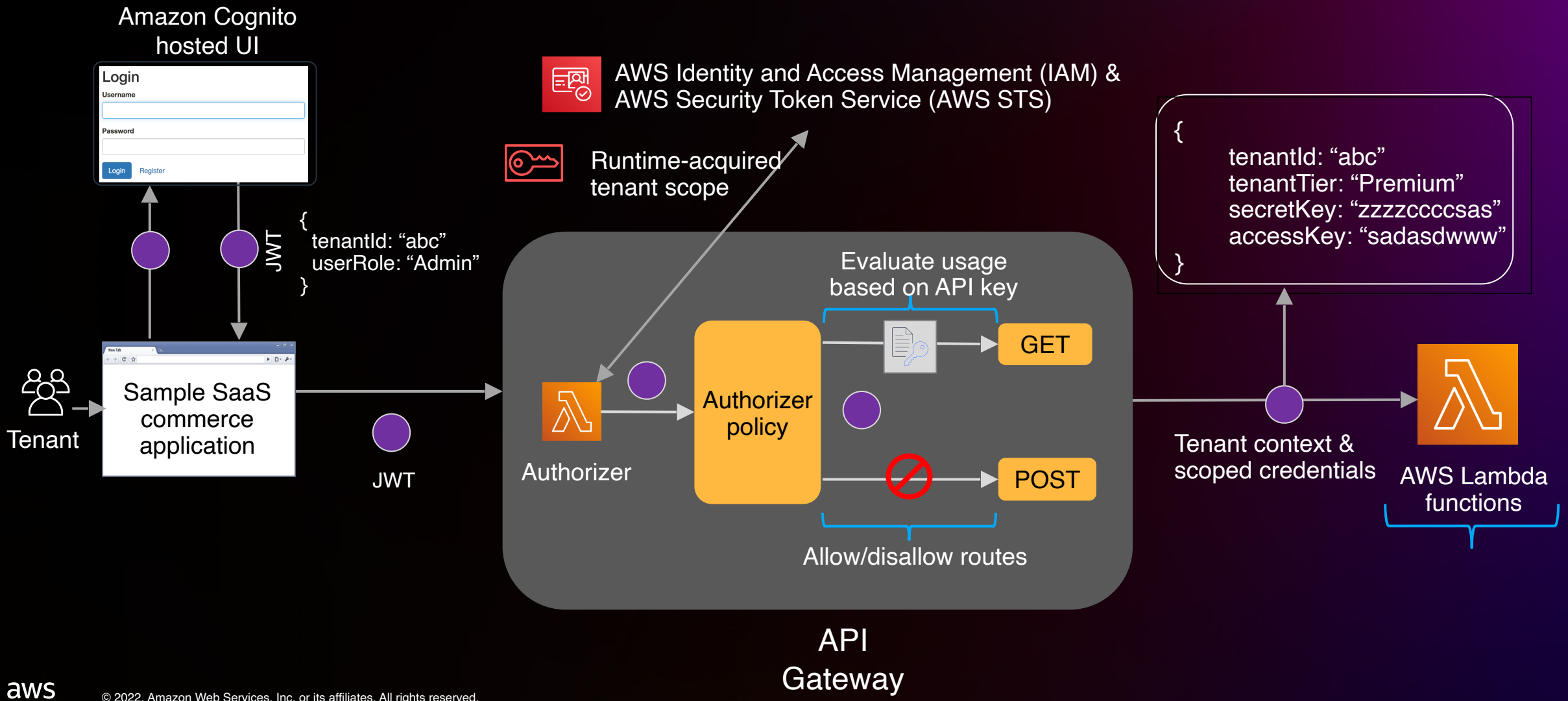


Lambda routing models

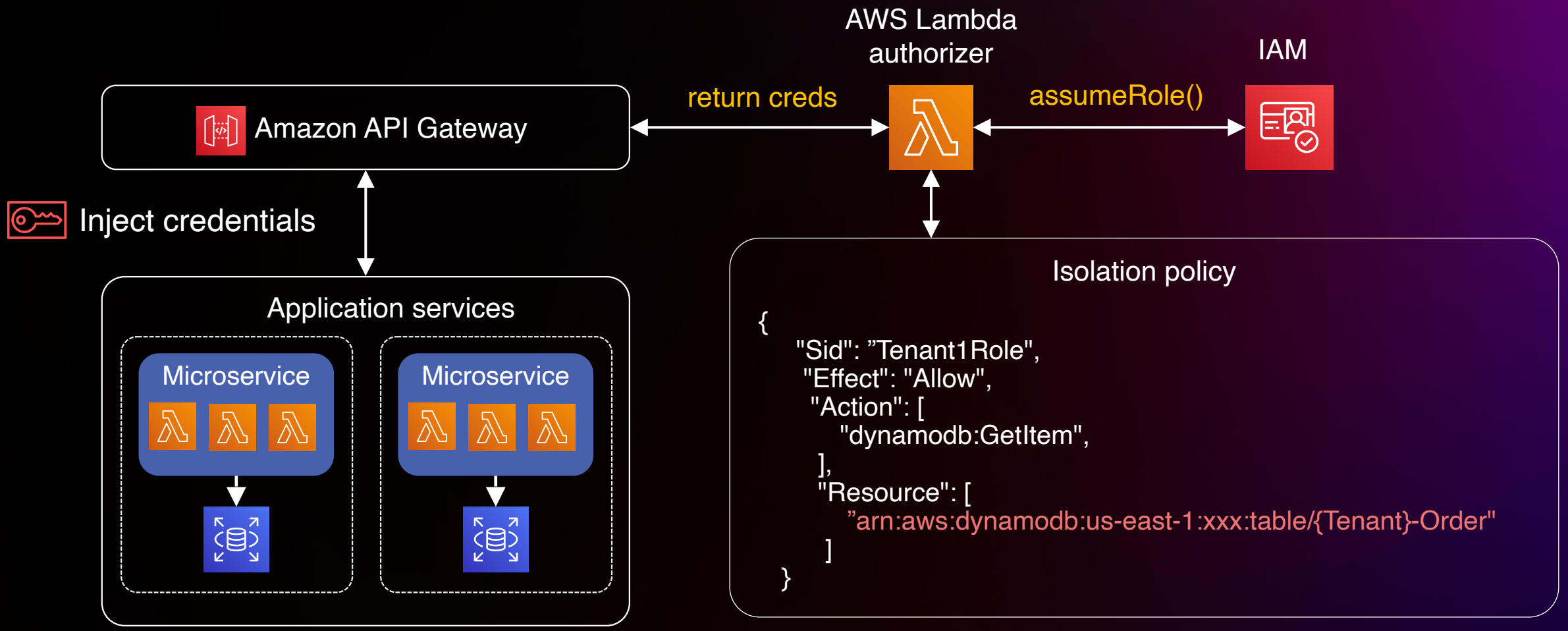


- Tenant name → tenantName
- User pool mapping
- API Gateway URL

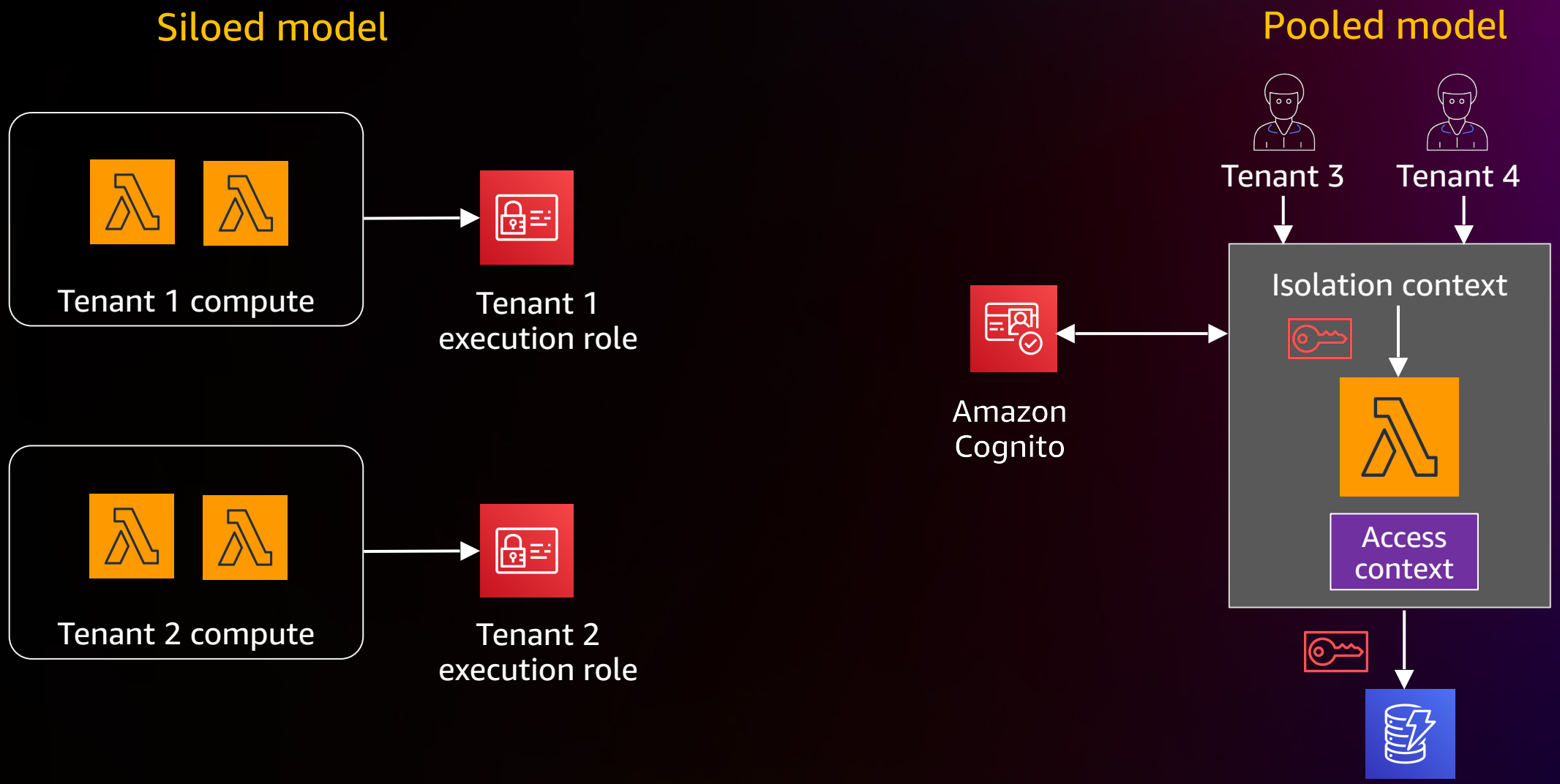
Lambda authentication/authorization



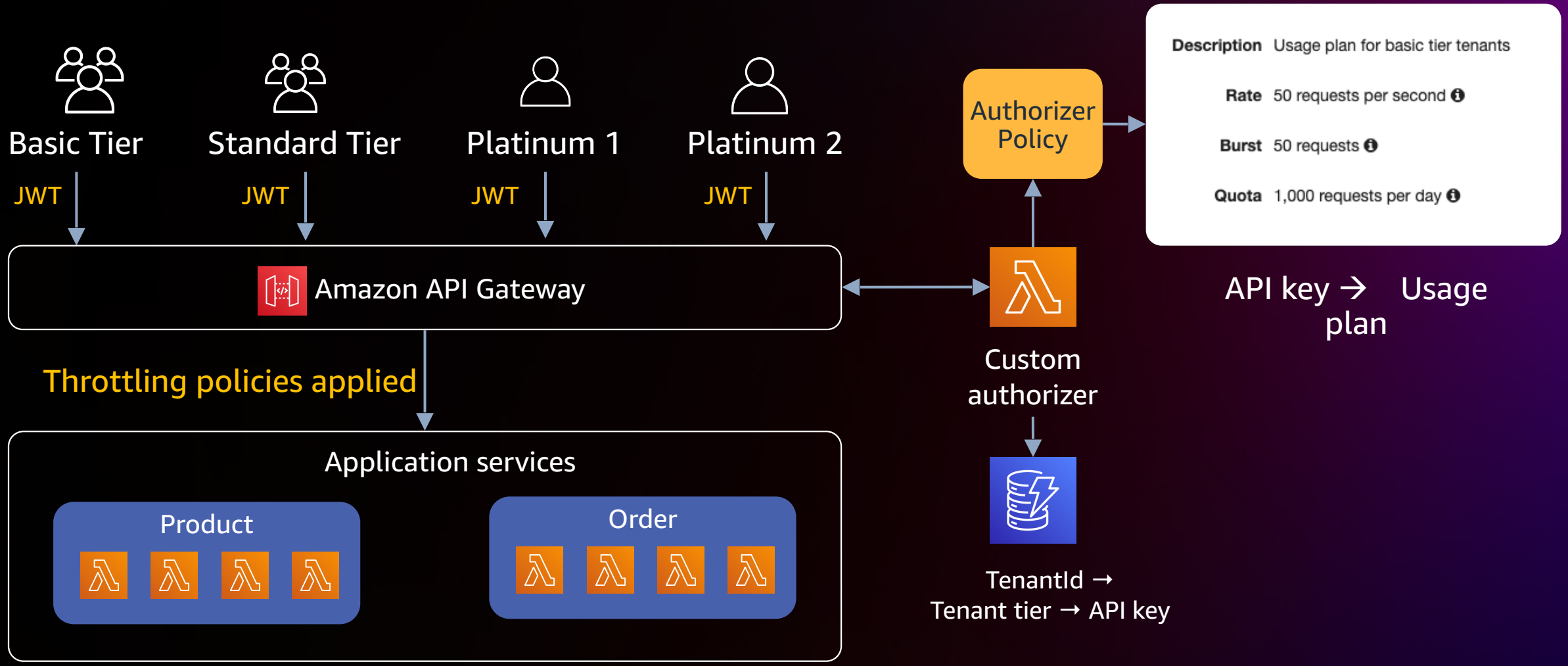
Authorizer injected credentials



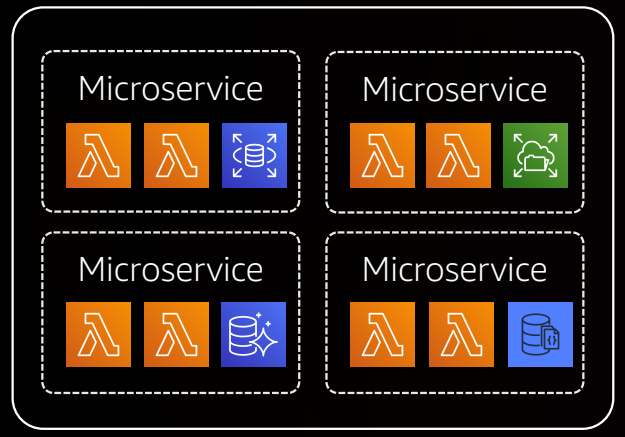
Lambda isolation models



Tier-based throttling with Lambda

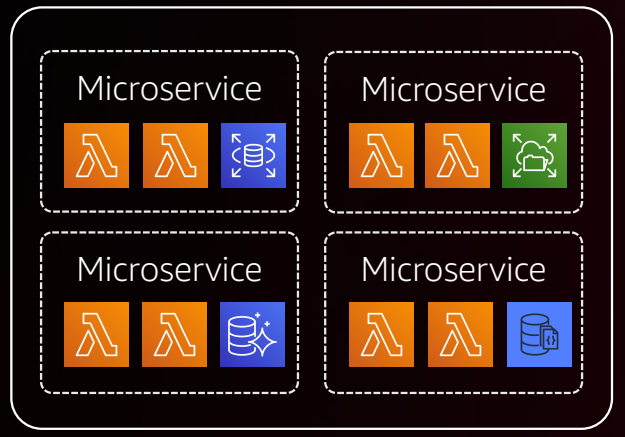


Noisy neighbor and Lambda concurrency



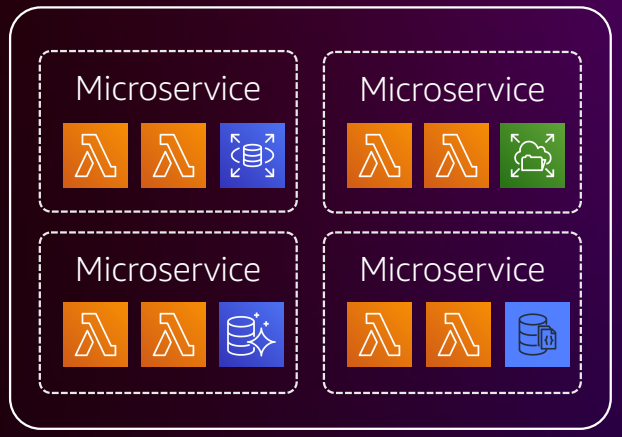
Basic tier

Reserve concurrency = 100



Advanced tier

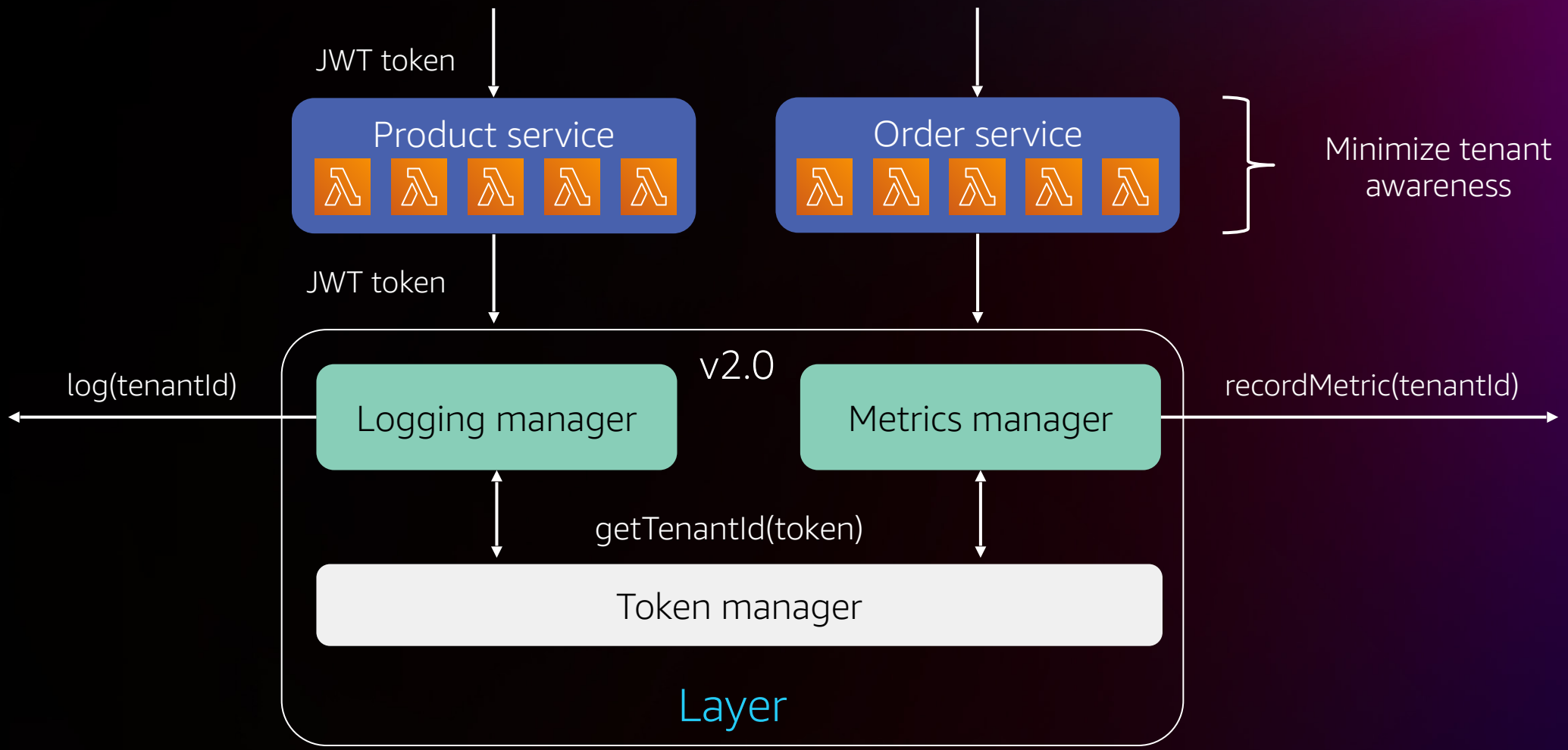
Reserve concurrency = 300



Premium tier

Reserve concurrency = All unreserved

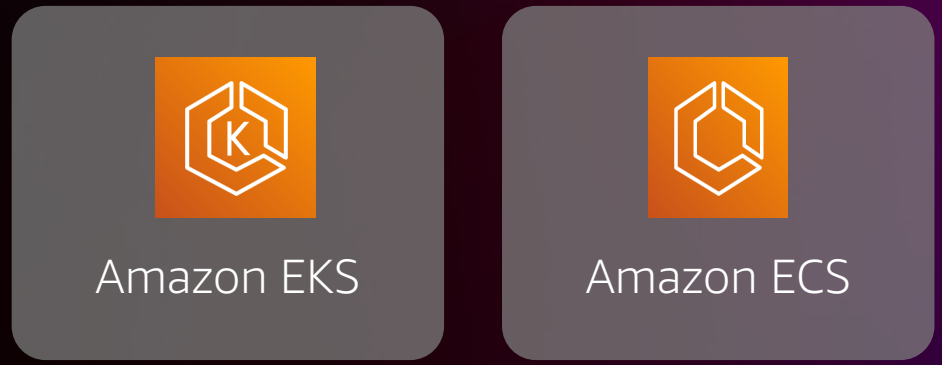
Lambda layers for shared constructs



Layers are deployed and versioned separately

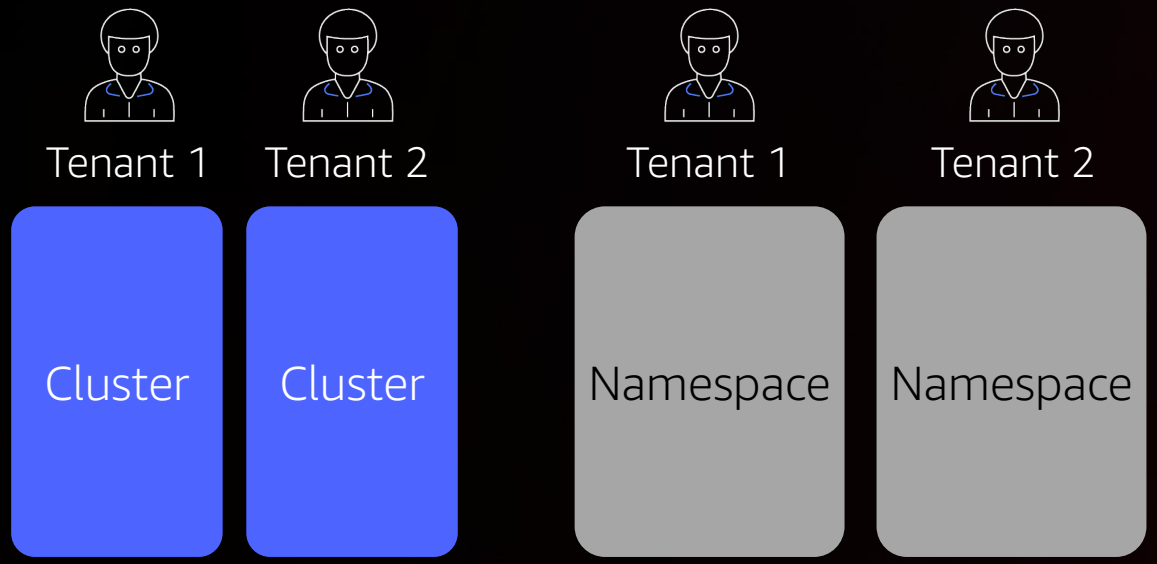
The multi-tenant Fargate model

EKS and ECS on Fargate



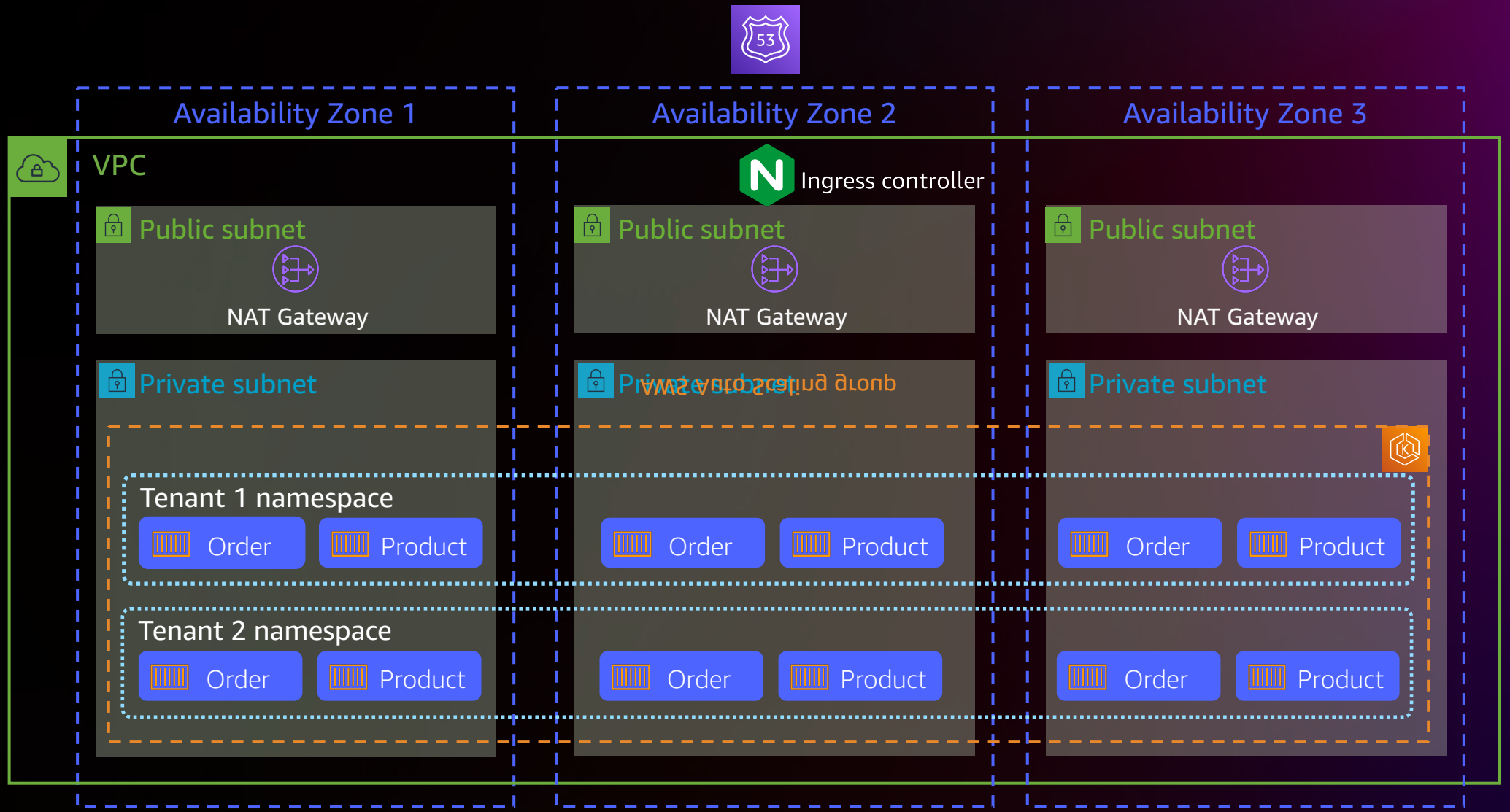
Container deployment models

Siloed deployments

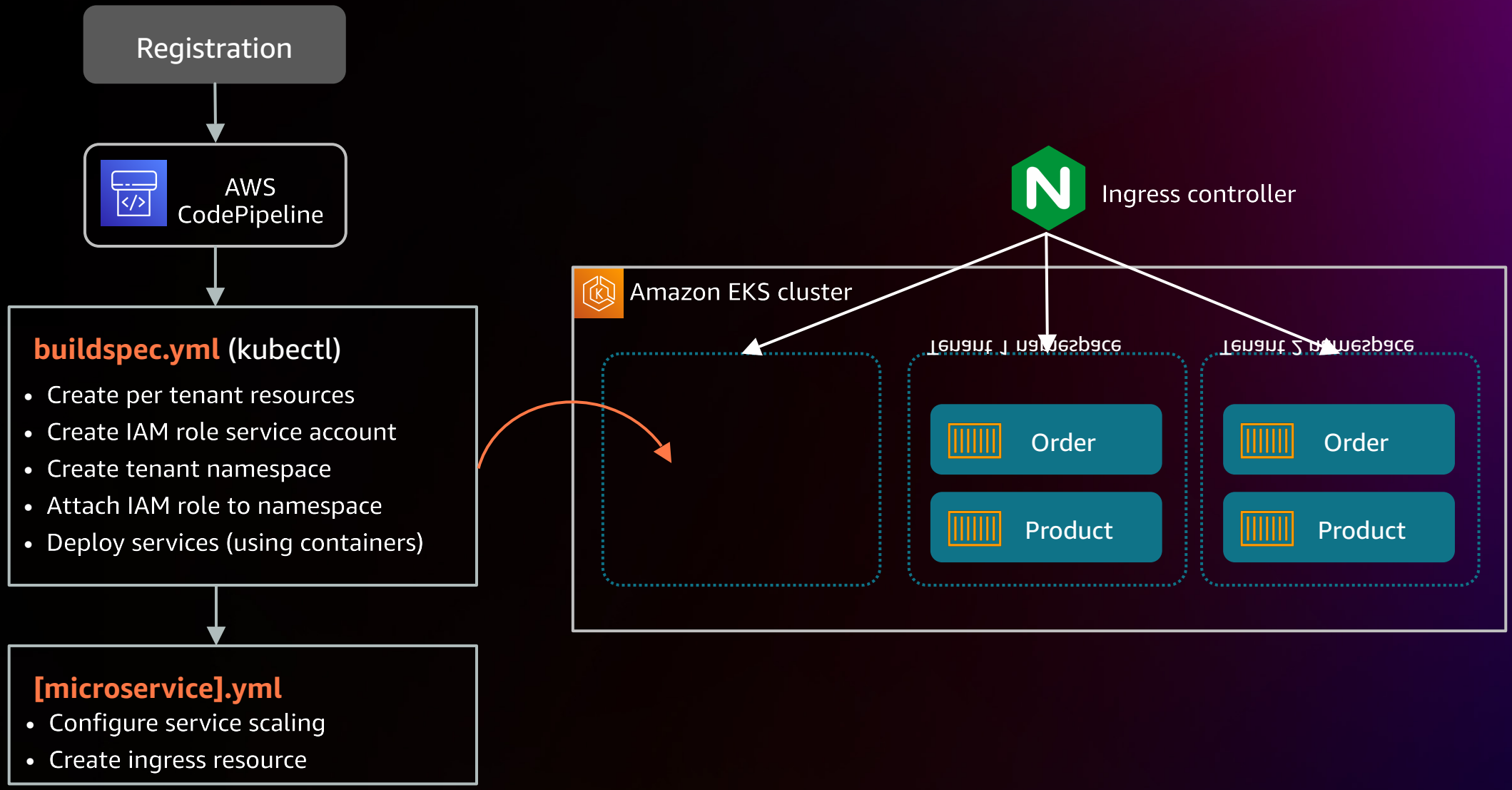


Pooled deployments

Sample EKS deployment: Namespace per tenant



Namespace provisioning

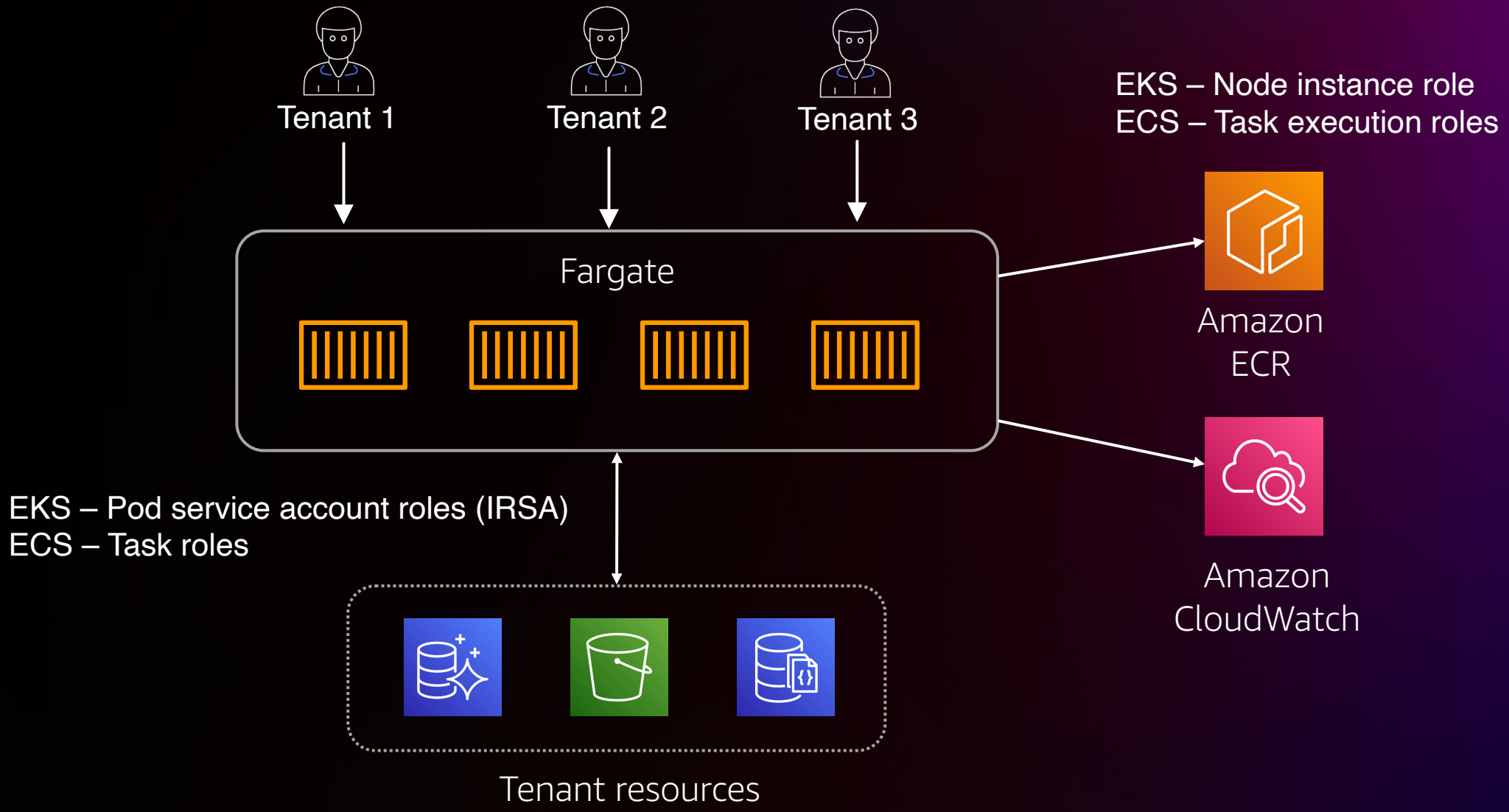


Tenant isolation: A layered model

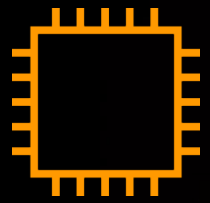
EKS and ECS constructs
(Cluster and namespace based)

Fargate constructs

Tenant isolation terminology/constructs



Fargate and tenant isolation



Hardware virtualization

Tasks/pods get isolated compute

Network isolation

Tasks get dedicated ENIs

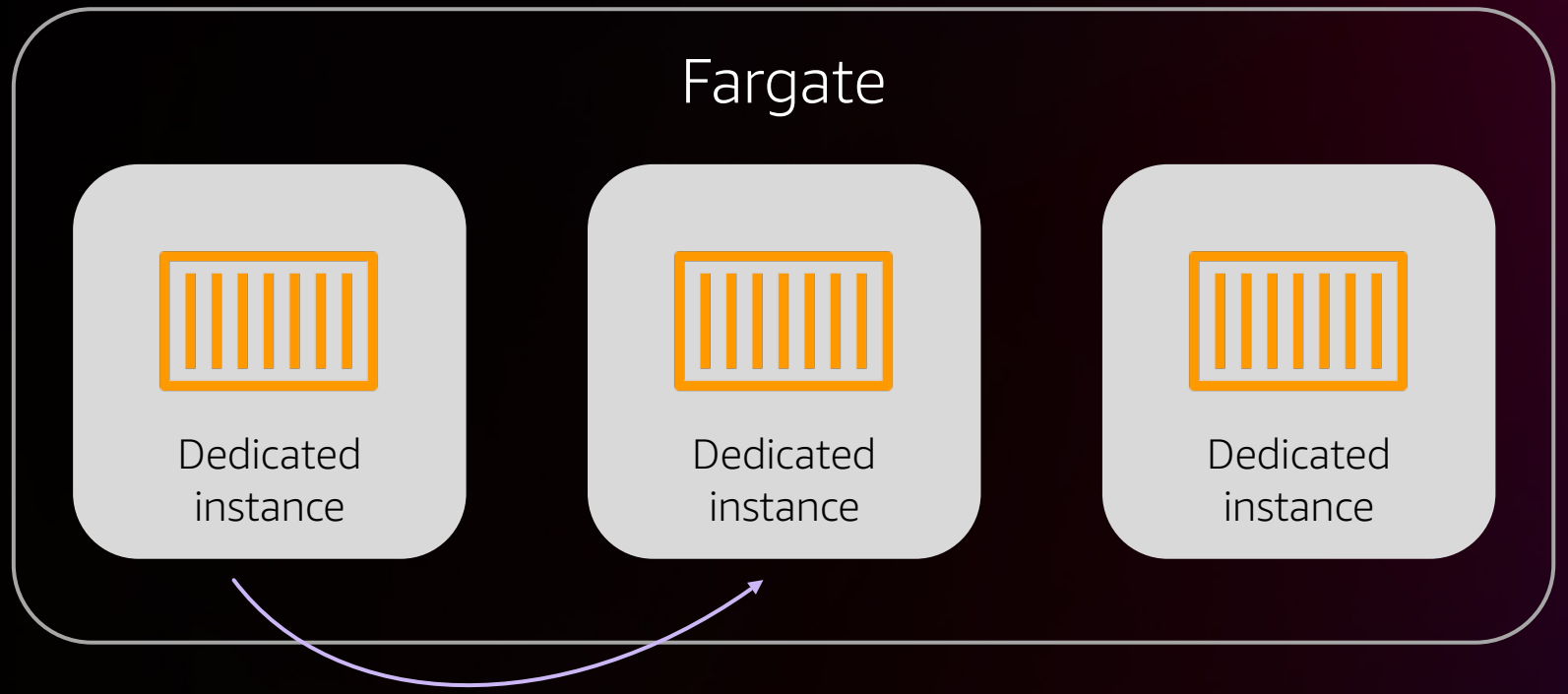
Storage isolation

Tasks get dedicated storage

Credential isolation

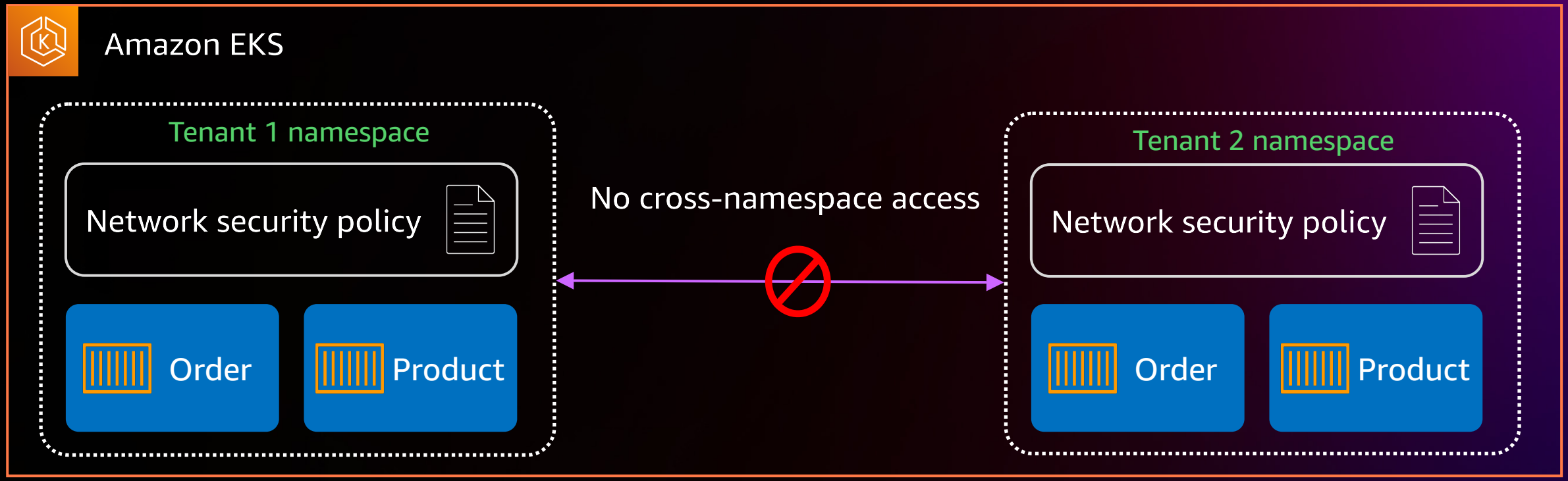
Tasks get credentials scoped to applications

Fargate enhances the isolation model

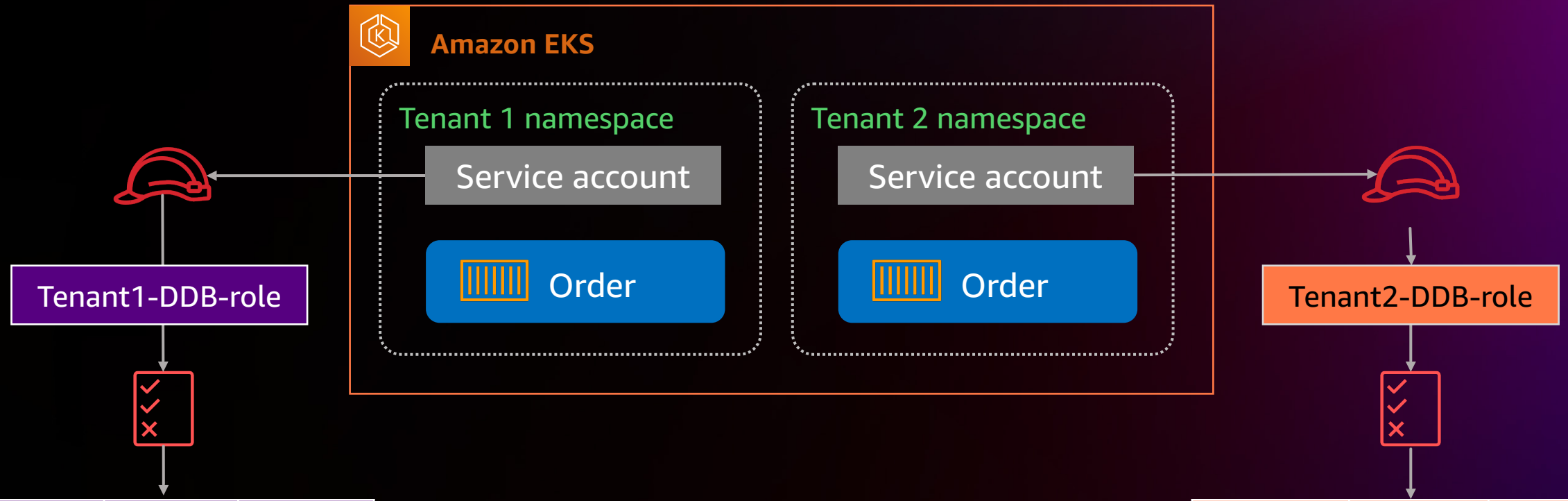


Limits concerns about escapability

EKS isolation with namespaces



Applying IAM roles for service accounts (IRSA)



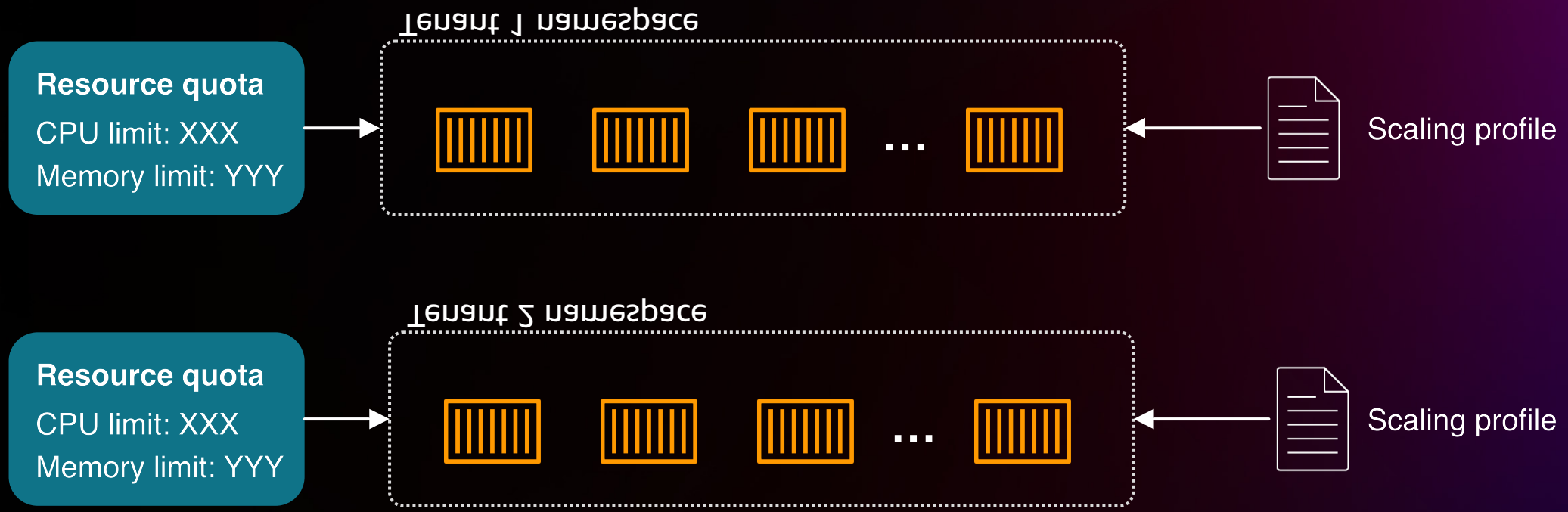
OrderId	Price	Quantity
20001	$500.00	20
20002	$299.99	24
20003	$204.12	30

Order-Tenant1 table

OrderId	Price	Quantity
33919	$13.55	1
98194	$11.66	34
73919	$532.33	2

Order-Tenant2 table

Tiering and throttling



Takeaways

- Serverless aligns naturally to the SaaS value proposition
- Developer experience may drive your preferences
- Cost and operational complexity can vary based on your domain
- Factor onboarding and DevOps into your thought process
- Consider using Lambda and Fargate together

Additional resources

1

Subscribe to AWS SaaS Insights

Get monthly emails with bite-size advice and the latest updates

2

Explore the SaaS on AWS hub

Check out the SaaS on AWS page for more resources and insights

3

Discover resources for builders

Access our curated list of SaaS reference solutions, demos, tech events, and more

Thank you!

Tod Golding
todg@amazon.com

Anton Aleksandrov
antonaws@amazon.com

Please complete the session survey in the **mobile app**