
AWS
re:Invent
NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

OPN309

Secure and multi-tenant infrastructure as code with Crossplane and Argo

Vikram Sethi
Sr. Computer Scientist
Adobe

Nima Kaviani
Principal Solutions Architect
AWS

Gaurav Dhamija
Sr. Solutions Architect
AWS

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

- Adobe’s previous state
- Adobe’s Internal Developer Platform (IDP)
- Deployments using Argo projects
- Infrastructure provisioning using Crossplane + Argo
- Multi-tenancy and security – requirements and solutions
- Developer experience – before and after
- Challenges/Unknowns

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

The combo meal

I would like a Burger Combo Meal, please!

We only do burgers!

BYO Fries and Soda.







I would like a Burger Combo Meal, please!

Sure! Any sauces alongside?



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Adobe’s previous state – Pain points

Service team

- Provision cloud accounts
- Use GitOps for compute deployments
- Define infrastructure resources/templates
- Learn custom workflows/tools
- Use custom solution for infra provisioning
- Maintain/Manage provisioned non-compute resources
- Track non-compute resources separately than compute resources

Platform team

- Manage GitOps workflows
- Enforce multi-tenancy and security
- No visibility into non-compute resources
- Harder troubleshooting in real-world scenarios

Red – Pain point
Green – Working well
White – Neutral

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

The Ask – Internal Adobe survey

Survey results (237 responses)

89%
Need a templatized infrastructure provisioning solution

65%
Want infrastructure provisioning integrated with existing GitOps-based workflows

68%
Ready to use the solution starting in 2022



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Solution requirements

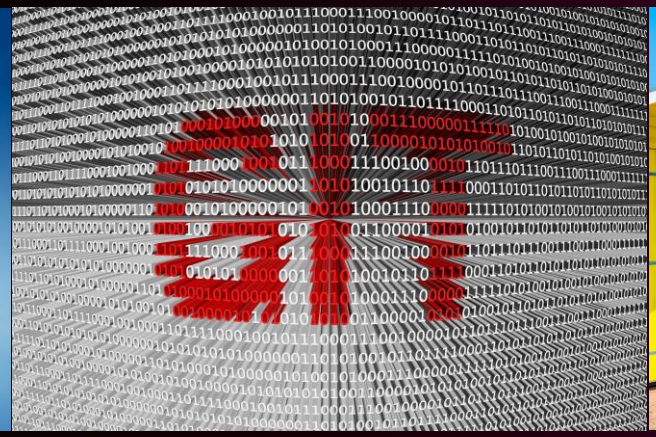
Standardization



Kubernetes-native



GitOps-friendly



Multi-tenant



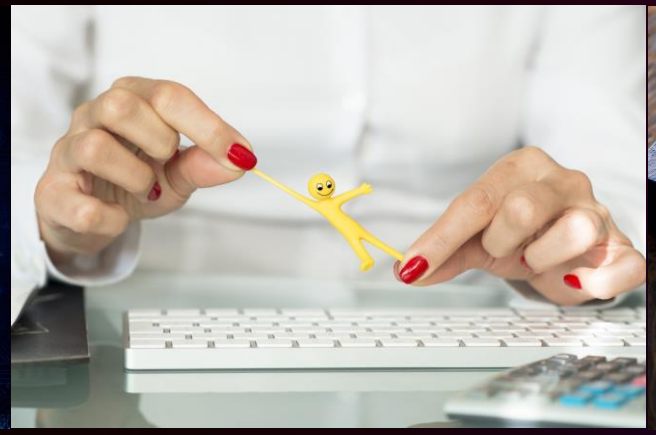
Secure



Multi-cloud



Extensible



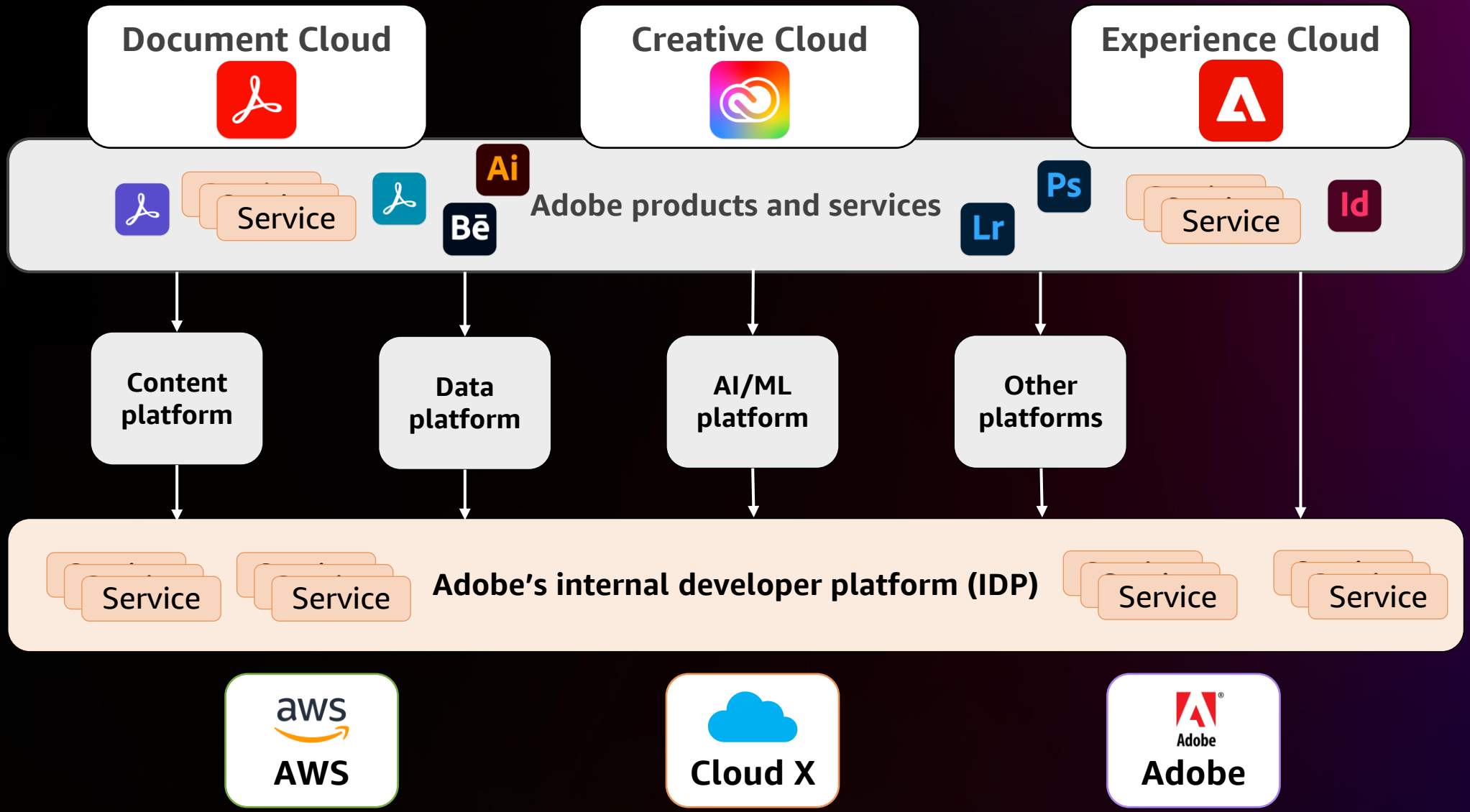
Industry alignment



aws

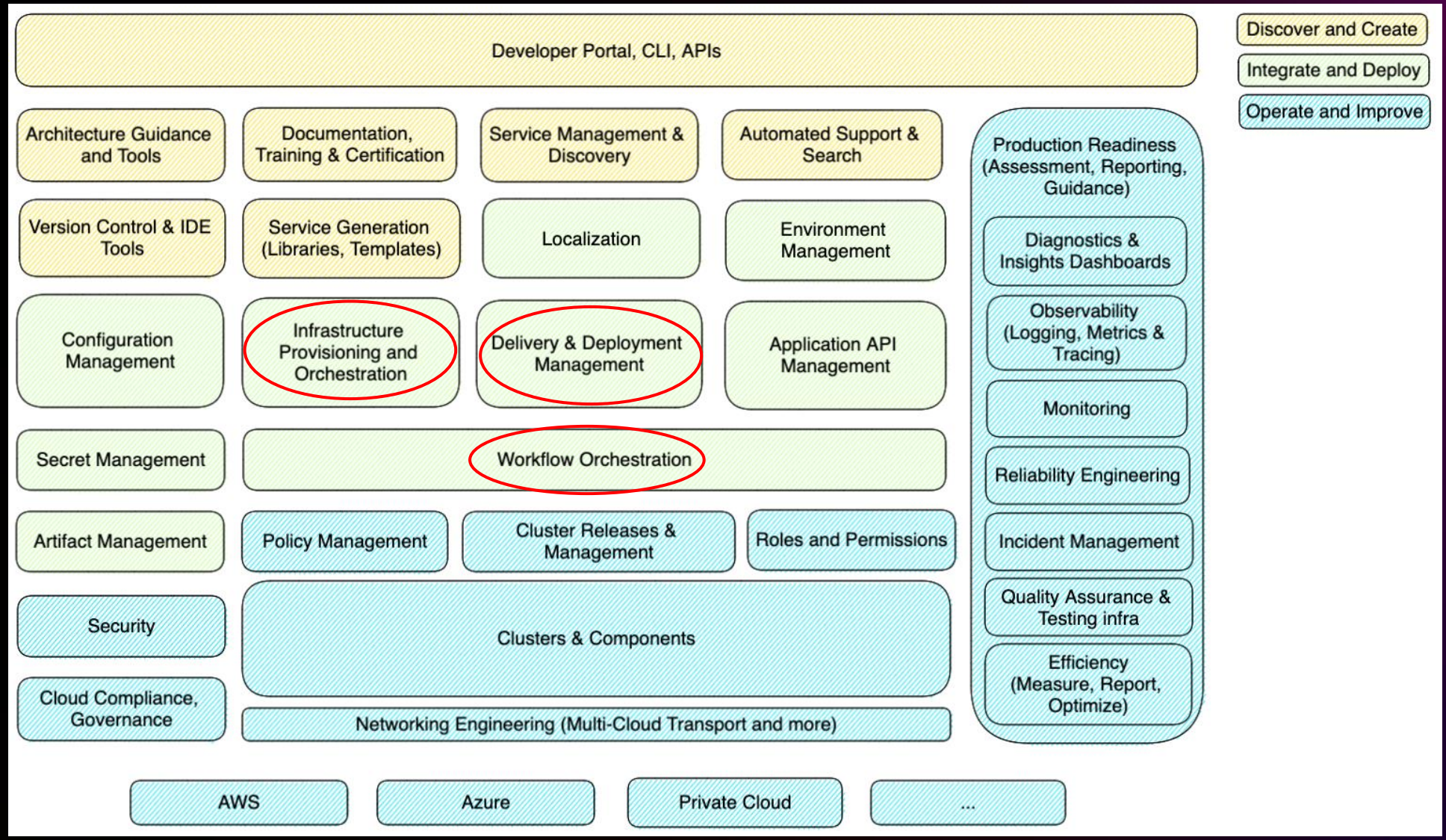
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Adobe’s services landscape



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Adobe's Internal Developer Platform (IDP)



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Introduction to Argo projects



Argo projects

Open-source Kubernetes-native tools, enabling GitOps-based production-grade application deployments and workflows

GitOps-based deployments

Paradigms and practices to use Git as the source of truth to manage deploying **application** and **infrastructure** resources

Argo CD

GitOps-based declarative continuous delivery

Alt: Flux

Argo Workflows

Workflow engine supporting DAGs and step-based workflows

Alt: Tekton Pipelines

Argo Events

Event-based dependency management and workflow automation

Alt: Knative Events

Argo Rollouts

Advanced deployment strategies

Alt: Spinnaker's Kayenta

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Introduction to Crossplane



Crossplane

Open-source Kubernetes-native tooling, extending Kubernetes APIs for cloud infrastructure resource provisioning as well as building higher-level abstractions

Infrastructure Resource Provisioning

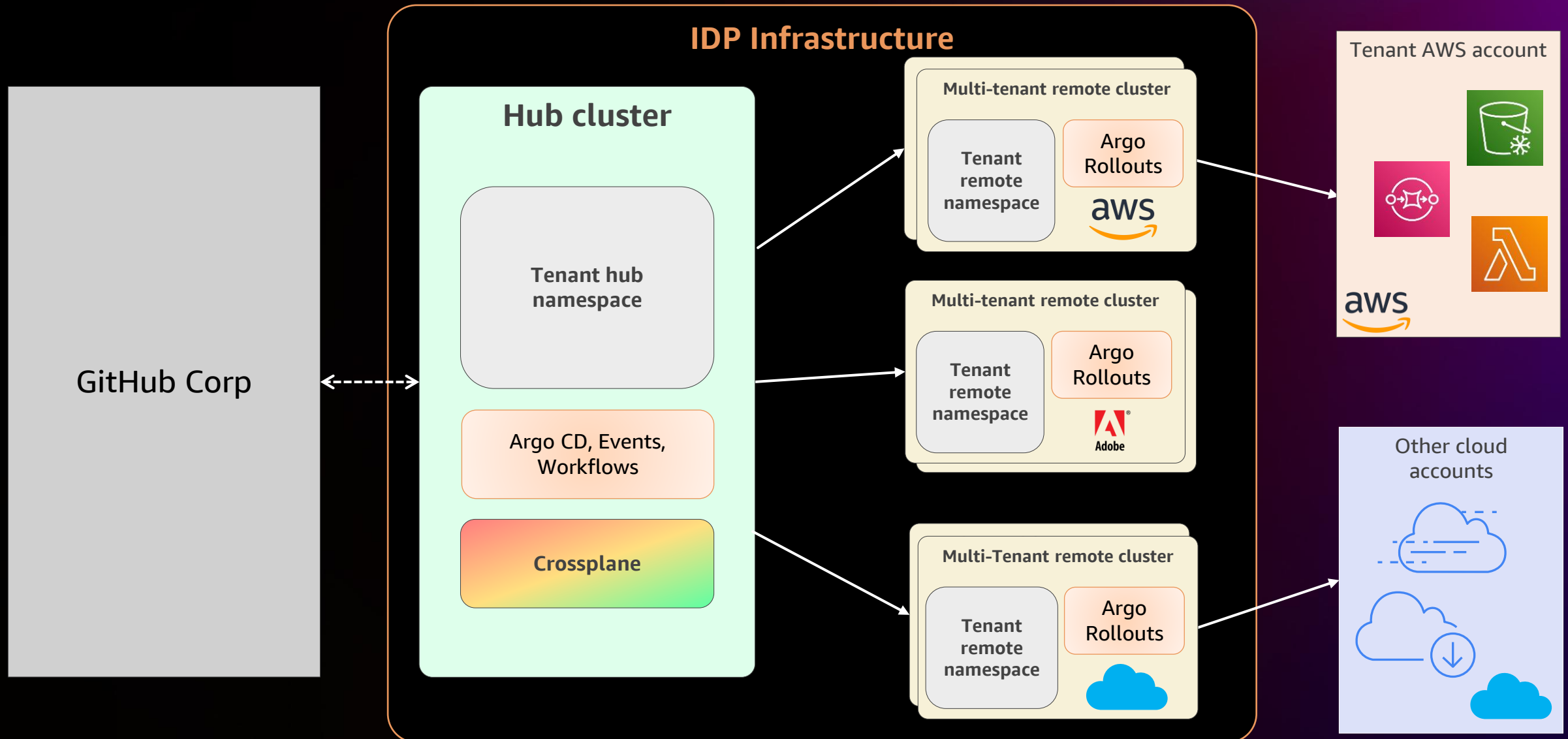
Process of creating, using, updating, and deleting of **cloud infrastructure resources**

Alternatives:
Terraform, Pulumi, Cloud Development Kit (CDK), Chef, Puppet . . .

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

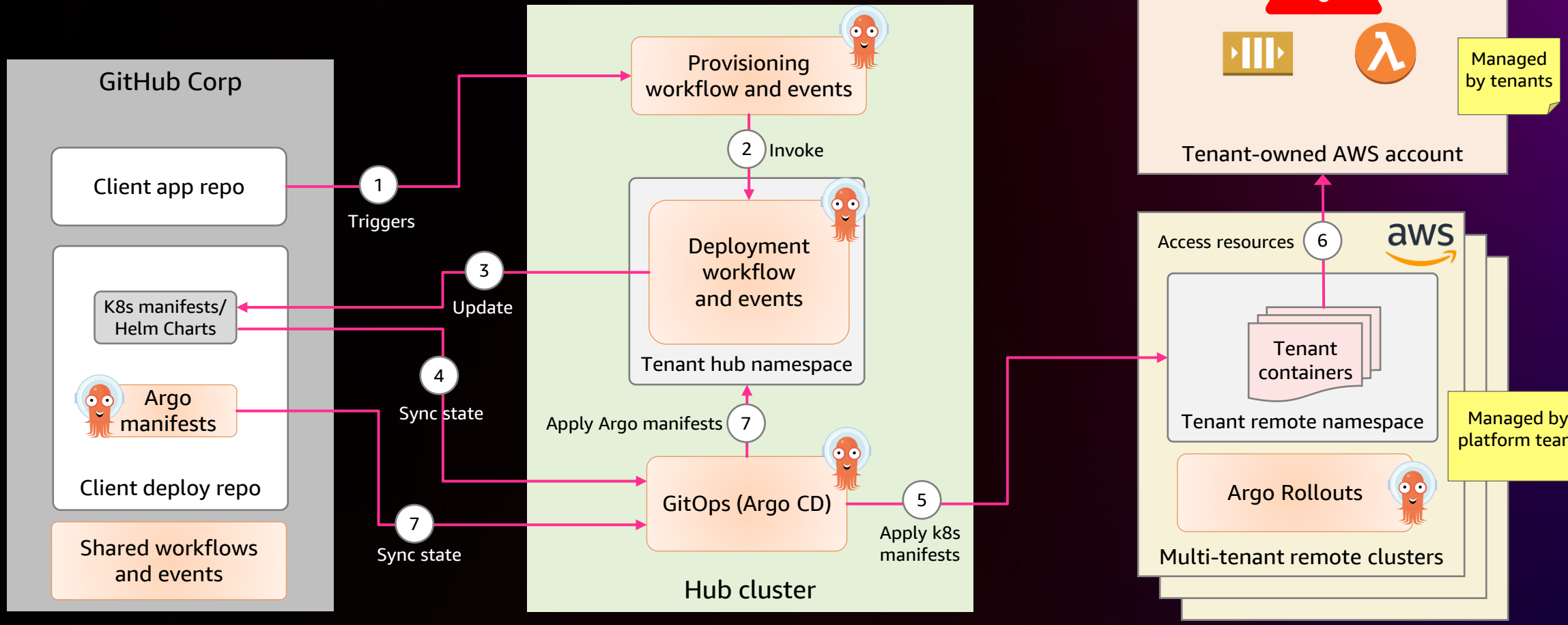
Hub-and-spoke model



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

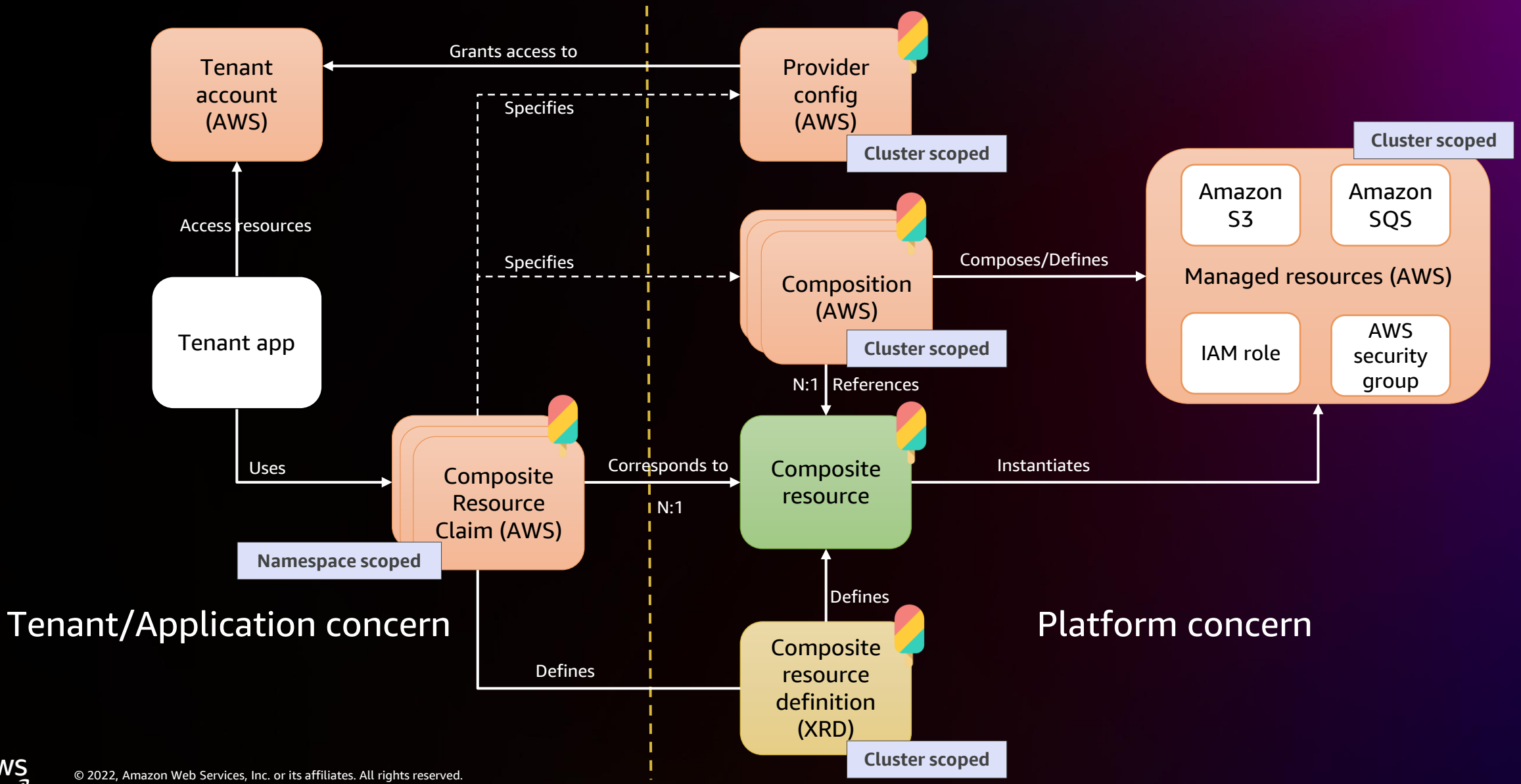
Deployments using Argo projects





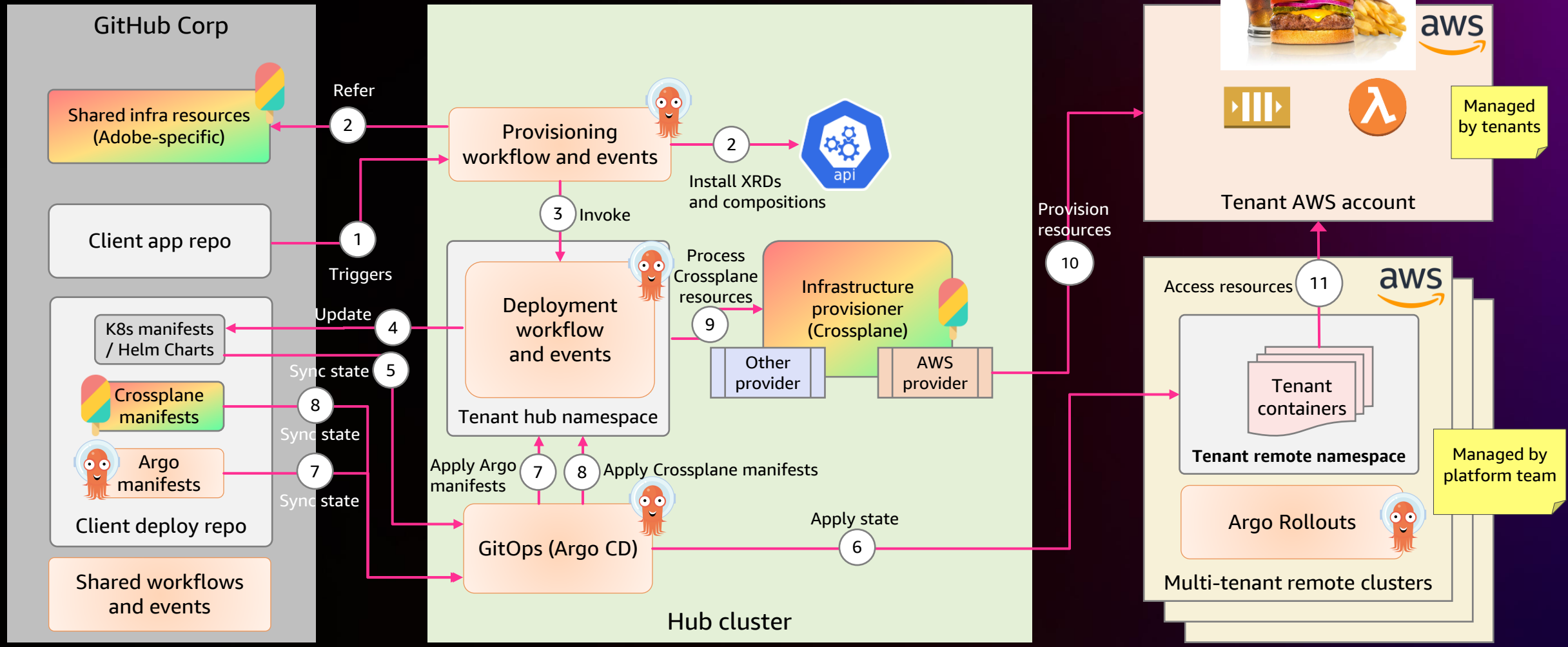
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Crossplane concepts



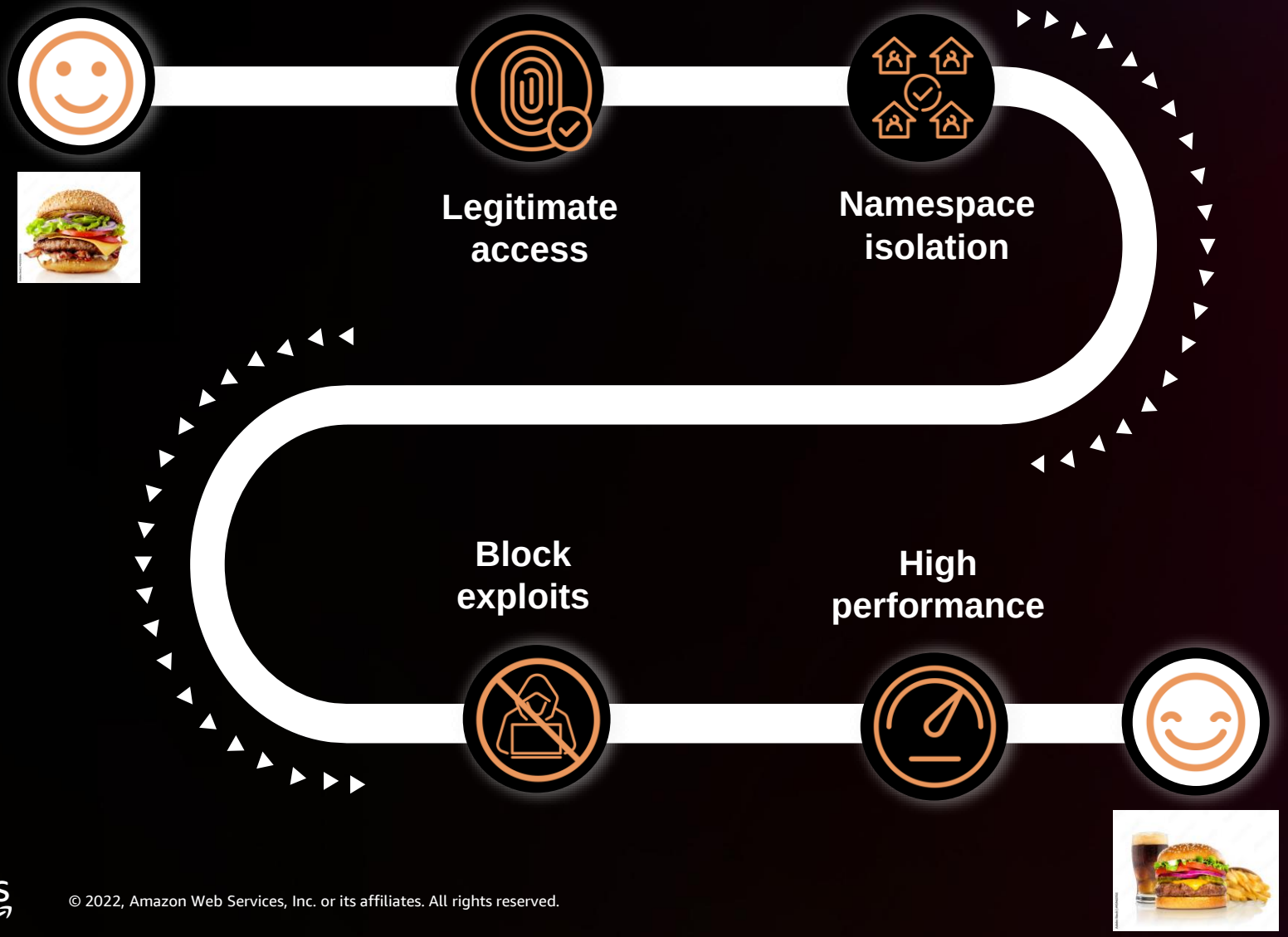
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Infra provisioning using Crossplane + Argo



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security requirements



- Restrict tenants to their own accounts and cloud resources
- Restrict resource access only at namespace level
- Rouge actors should not be able to exploit the system
- Existing and new workflows should be performant

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security requirements

```yaml
apiVersion: aws.crossplane.io/v1beta1
kind: ProviderConfig
metadata:
  name: aws-account-1
spec:
  assumeRole:
    roleARN: arn:aws:iam::<aws-account-1>:role/crossplane
```

≈

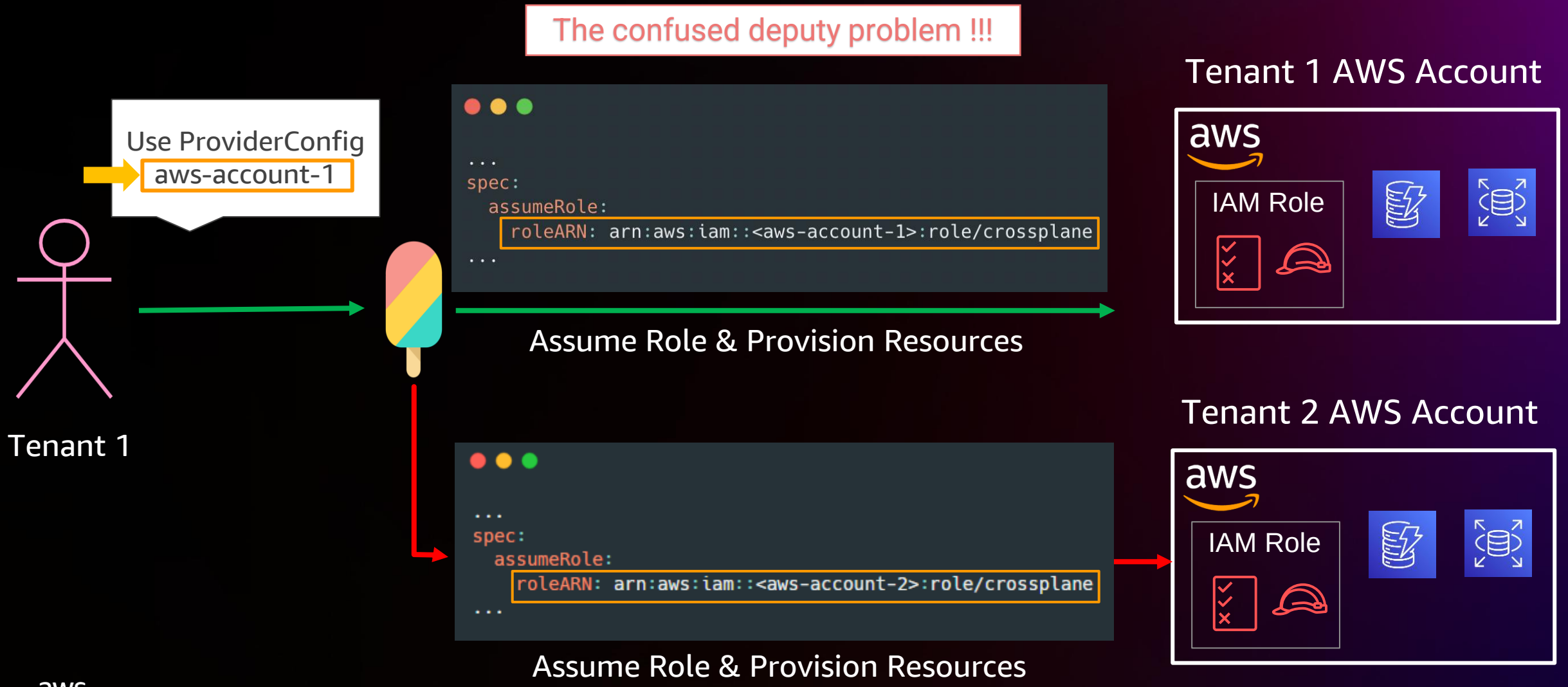


aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Legitimate access

#1. Tenants should be able to provision cloud resources in their own cloud accounts

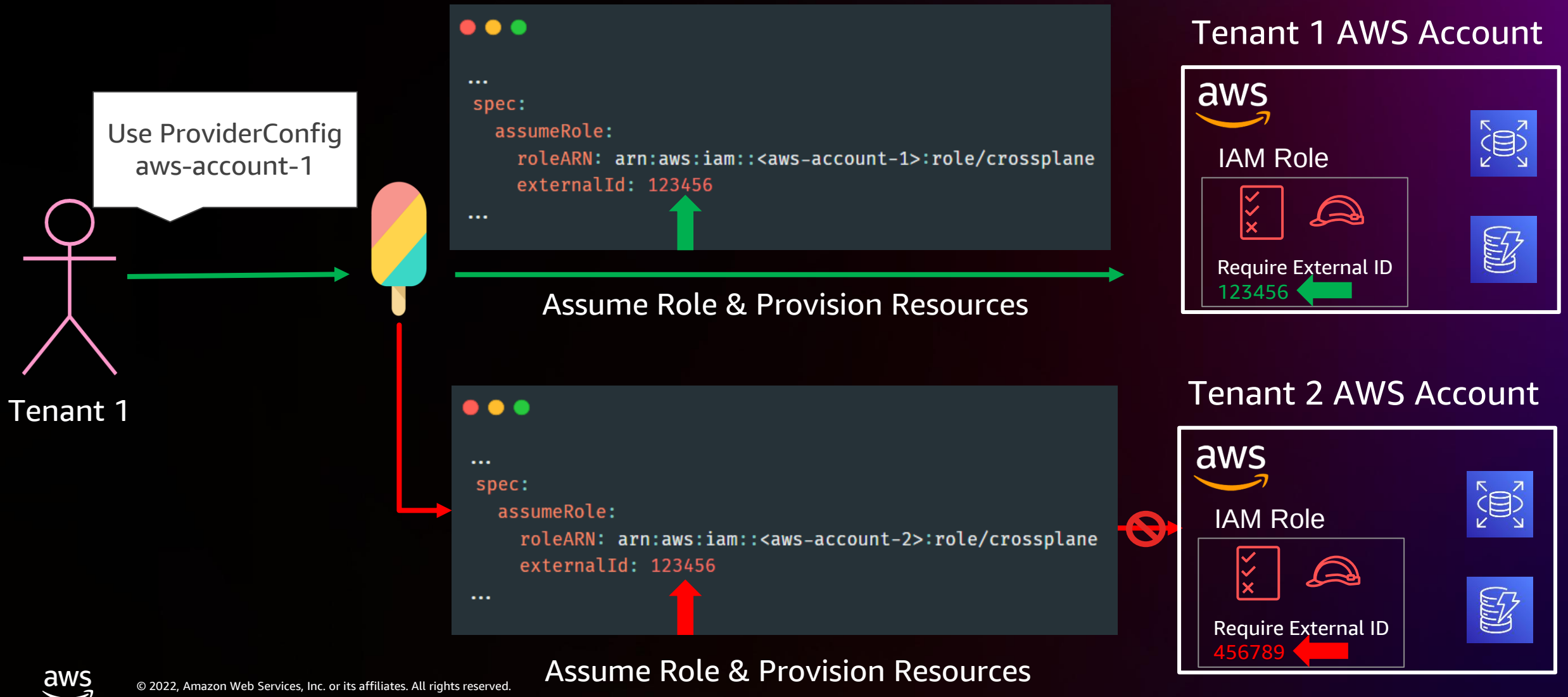
The confused deputy problem !!!



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Legitimate access

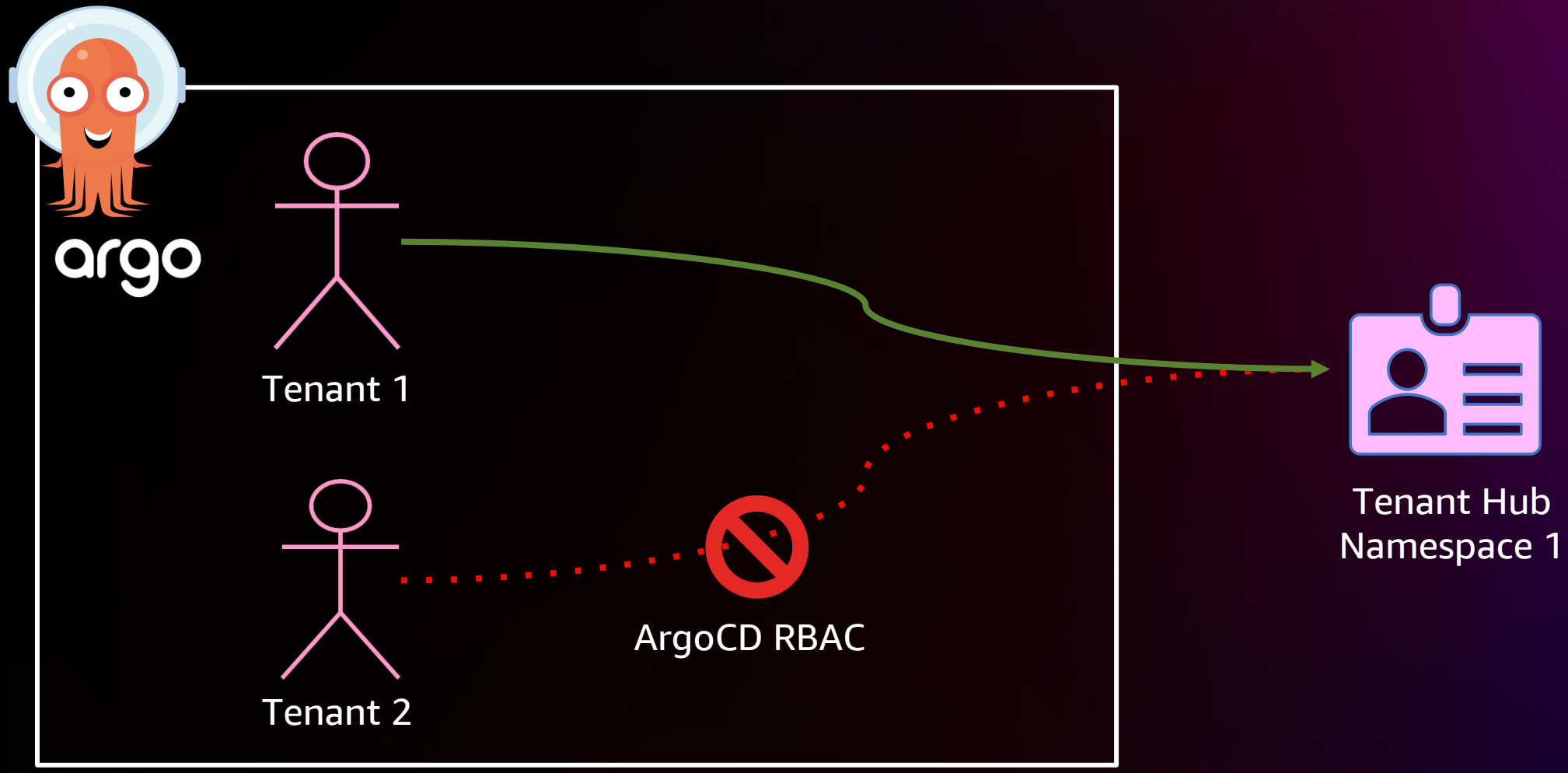
#2. Tenants should not have access to any other tenant's provisioned resources



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Legitimate access

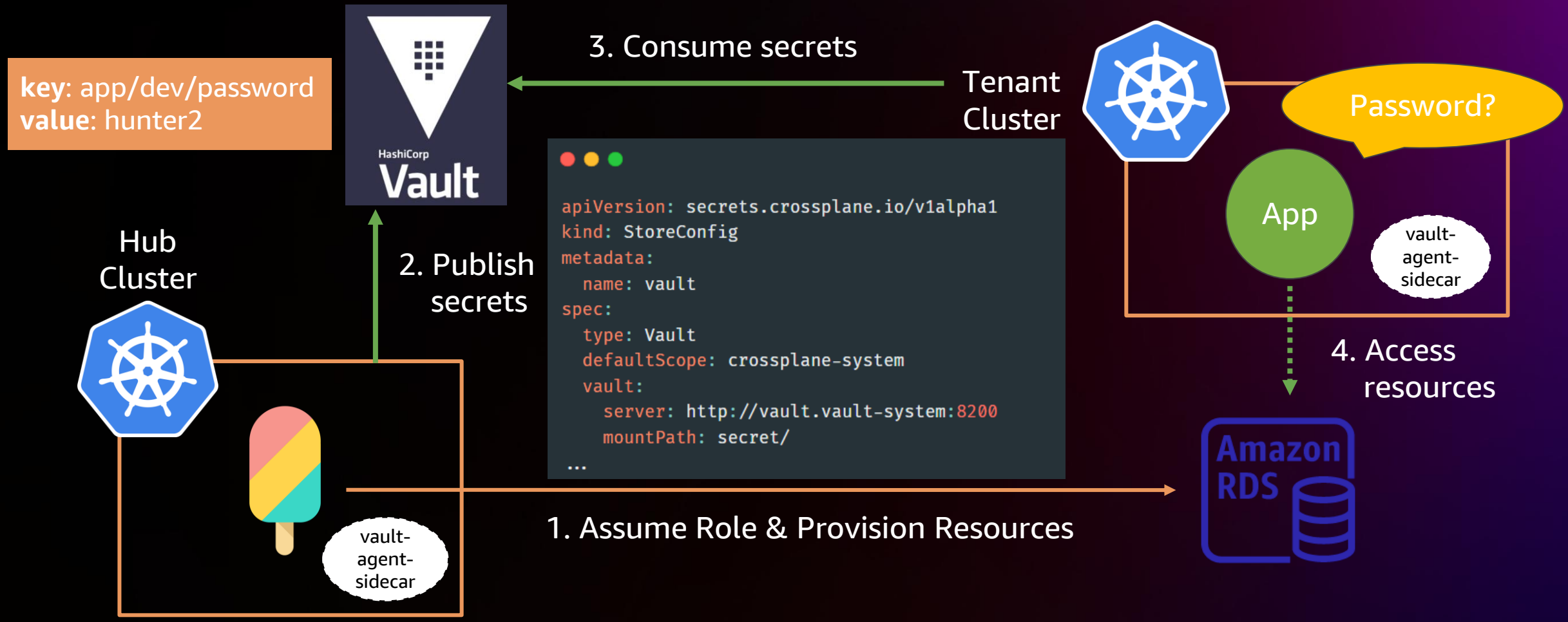
#2. Tenants should not have access to any other tenant's provisioned resources



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Legitimate access

#3. Secrets for created infra can be pushed/pulled dynamically to external vault



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

#4. ProviderConfig can only be used in context of a tenant namespace

```yaml
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: providerconfigs.aws.crossplane.io
spec:
  scope: Cluster
  names:
    kind: ProviderConfig
```

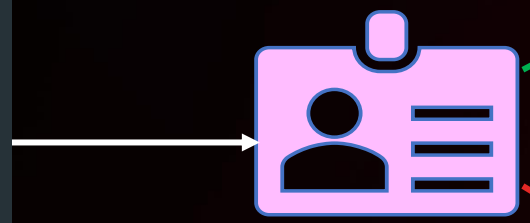
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

#4. ProviderConfig can only be used in context of a tenant namespace

```yaml
apiVersion: blob.crossplane.adobe.io/v1beta1
kind: ObjectStorage
metadata:
  name: test-object
  namespace: ns1
spec:
  resourceConfig:
    providerConfigName: ns1-provider-config
...
```

Composite Resource Claim (XRC)



Tenant Hub
Namespace (ns1)

Allowed

ns1-provider-config

Prohibited

Blocked

ns2-provider-config

aws

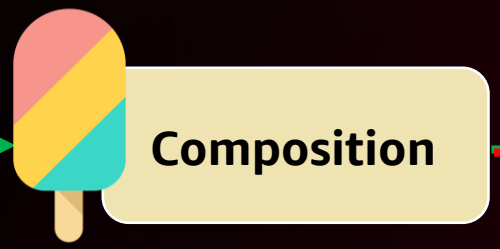
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

#4. ProviderConfig can only be used in context of a tenant namespace

```yaml
apiVersion: blob.crossplane.adobe.io/v1beta1
kind: ObjectStorage
metadata:
  name: test-object
  namespace: ns1
spec:
  resourceConfig:
    providerConfigName: ns2-provider-config
...
```

Composite Resource Claim (XRC)



Composition

ns1-provider-config

ns2-provider-config

Rewrite ProviderConfig name to “ns1-provider-config”

```yaml
fromFieldPath: metadata.labesl[crossplane.io/claim-namespace]
toFieldPath: spec.providerConfigRef.name
combine:
  string:
    fmt: "%s-provider-config"
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

#5. Tenant should be able to use any managed resource from providers, but with namespace scope

```yaml
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: buckets.s3.aws.crossplane.io
spec:
  scope: Cluster
```

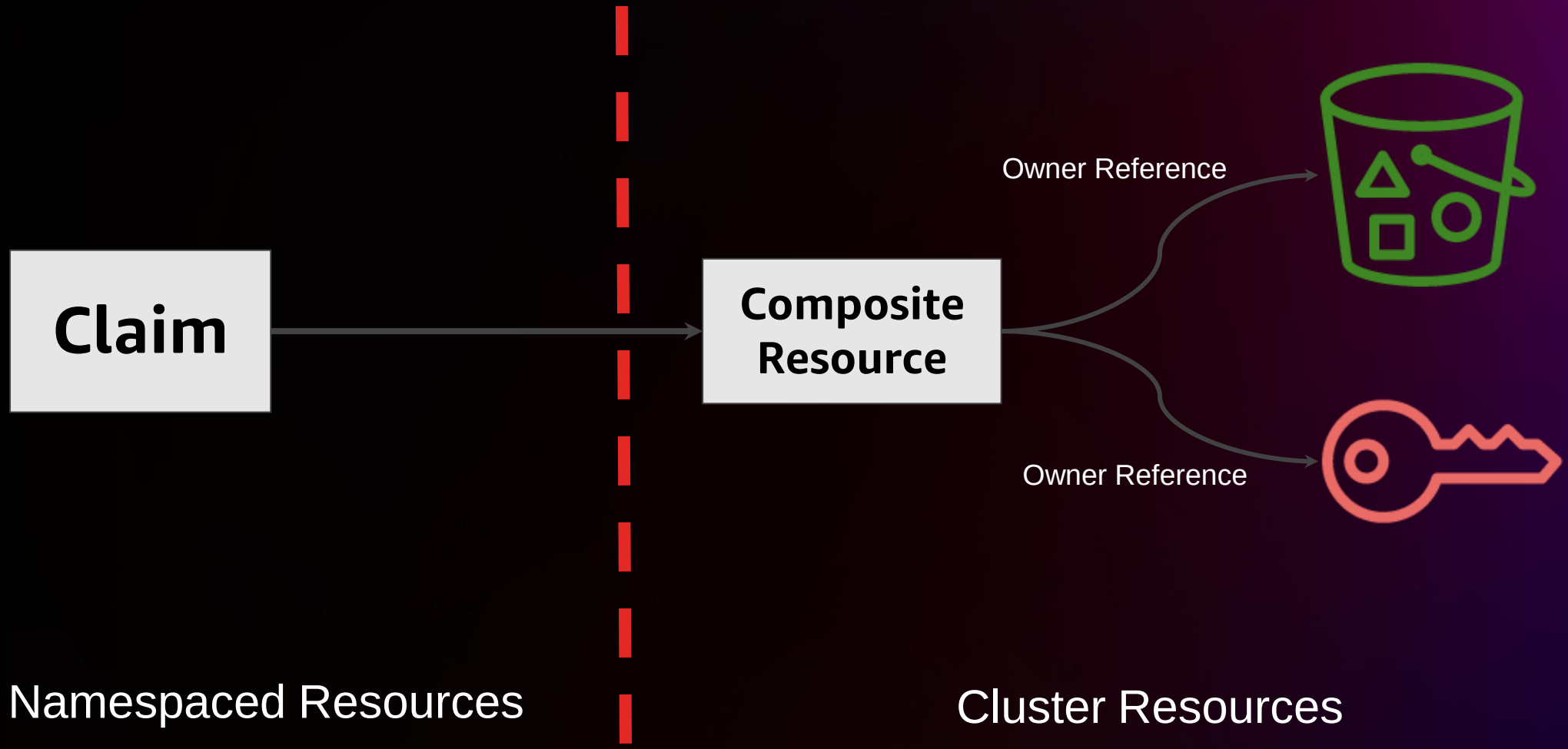
```yaml
apiVersion: s3.aws.crossplane.io/v1beta1
kind: Bucket
metadata:
  name: test-bucket
spec:
  forProvider:
    acl: private
    locationConstraint: us-east-1
  providerConfigRef:
    name: provider-config-1
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

#5. Tenant should be able to use any managed resource from providers, but with namespace scope

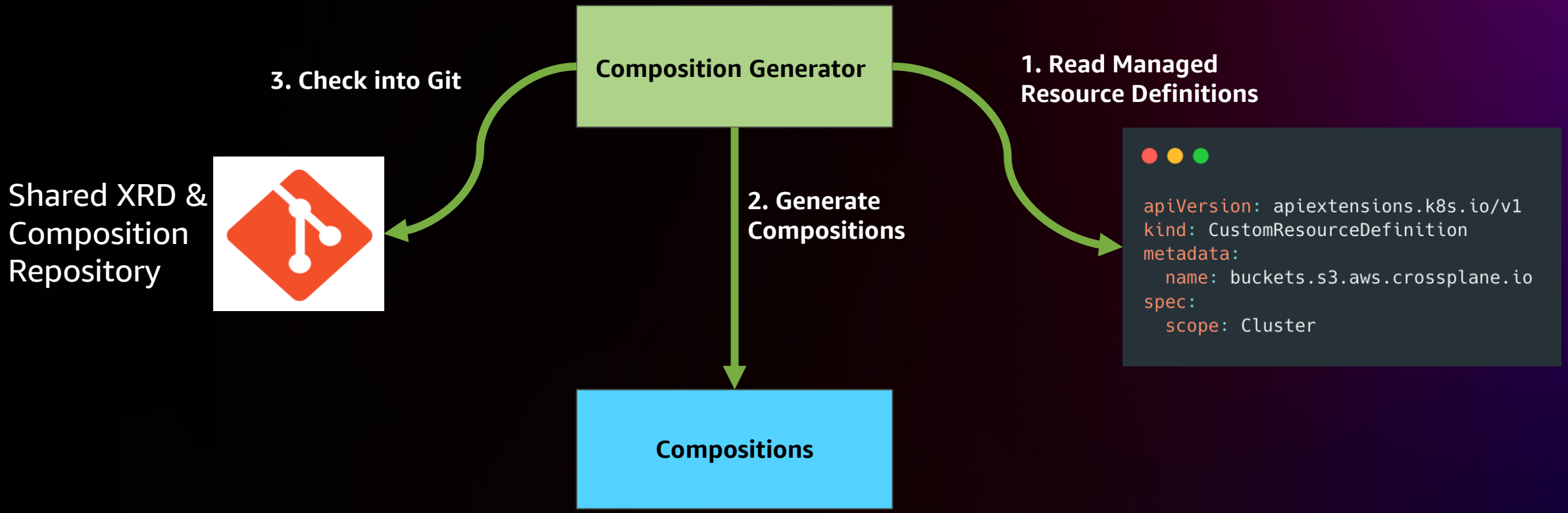


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

#5. Tenant should be able to use any managed resource from providers, but with namespace scope

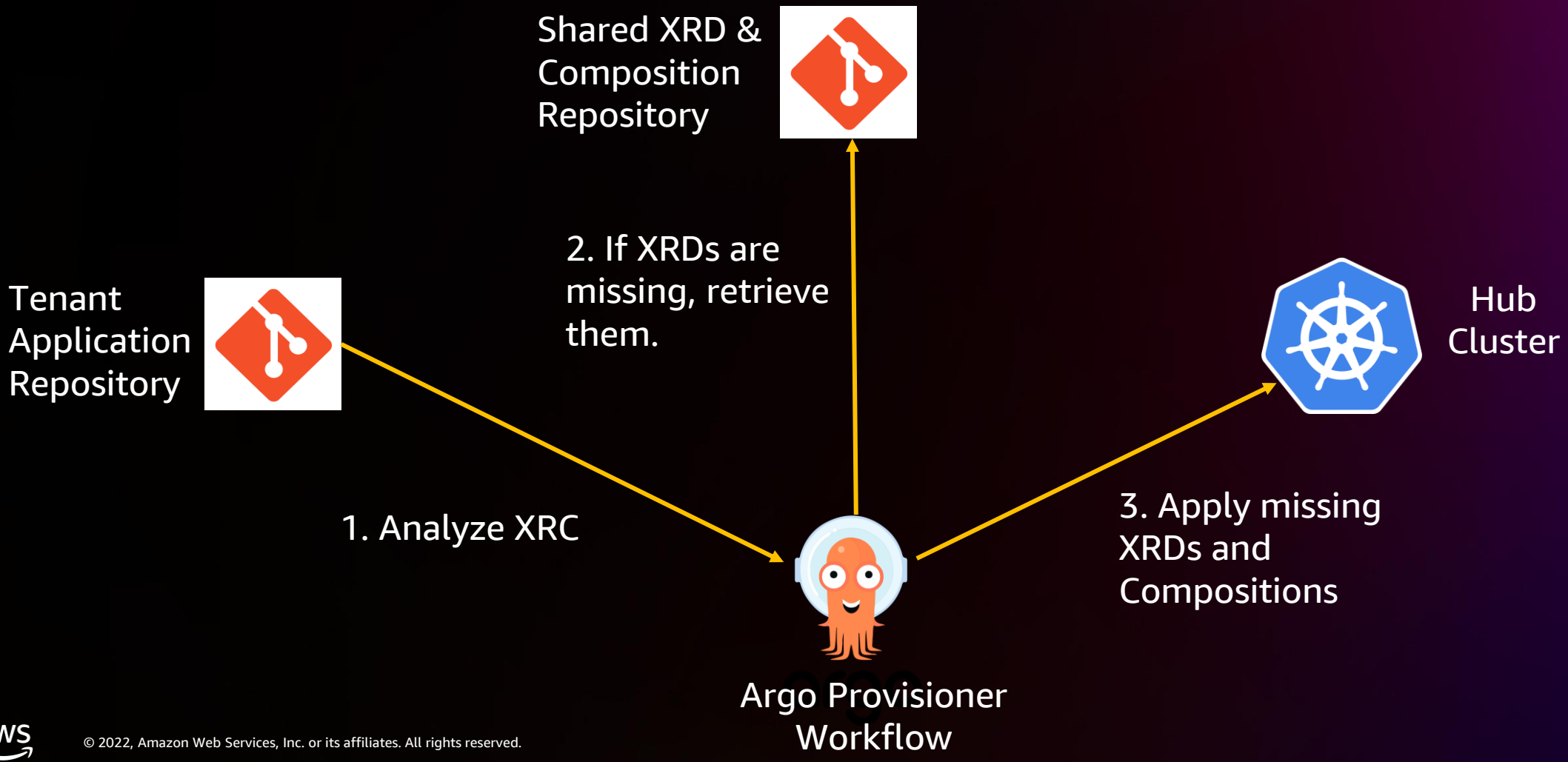
Thanks **Christopher Haar**! (https://github.com/crossplane-contrib/x-generation)



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

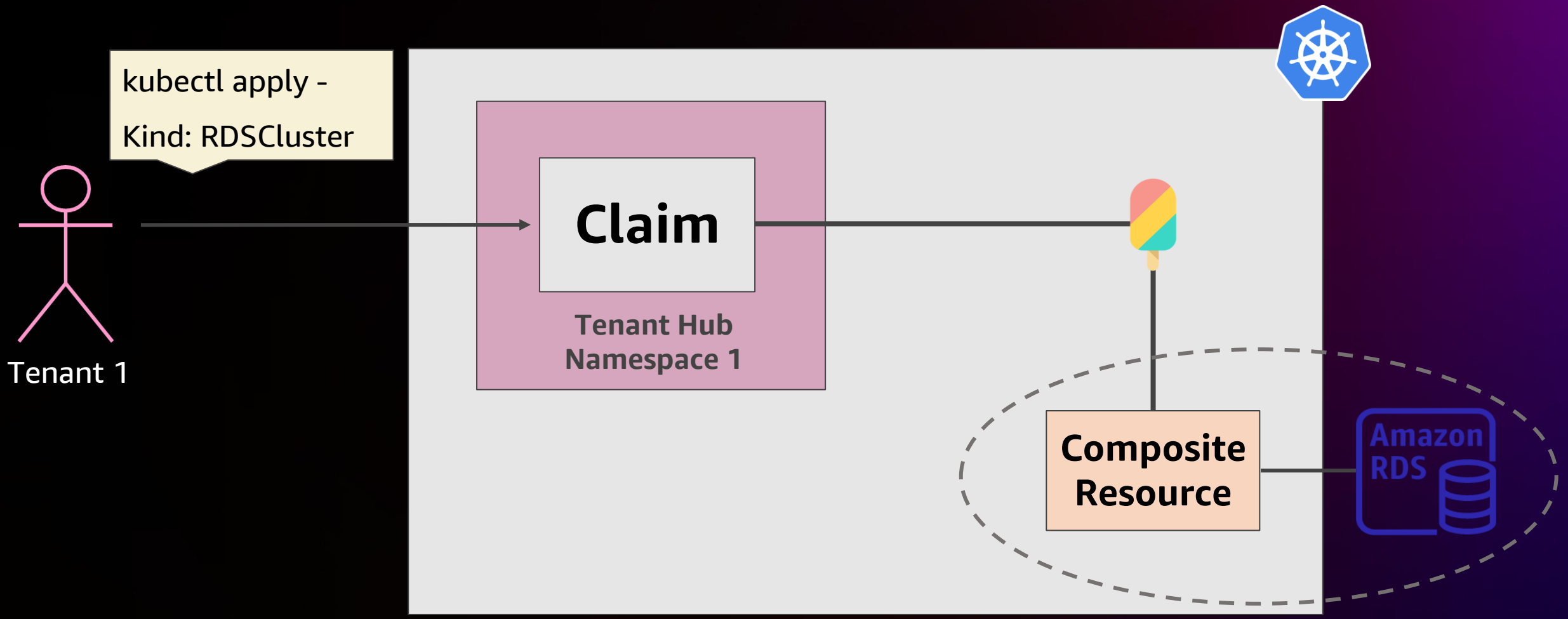
#5. Tenant should be able to use any managed resource from providers, but with namespace scope



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Namespace access

#5. Tenant should be able to use any managed resource from providers, but with namespace scope



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Block Exploits

#6. A rogue tenant should not be able to exploit another tenant's ProviderConfig

```yaml
apiVersion: s3.aws.crossplane.io/v1beta1
kind: Bucket
metadata:
  name: test-bucket
spec:
  forProvider:
    acl: private
    locationConstraint: us-east-1
  providerConfigRef:
    name: provider-config-1
```

Managed Resource



Open Policy Agent (OPA)



Tenant Hub Namespace 1

Allow

ns1-provider-config

Block

ns2-provider-config

```
violation[{"msg": msg}] {
  not input.review.object.metadata.labels["crossplane.io/composite"]
  msg := "crossplane composition not used"
}

violation[{"msg": msg}] {
  not input.review.object.metadata.ownerReferences
  msg := "ownerReferences field not present"
}

violation[{"msg": msg}] {
  ownerRef = input.review.object.metadata.ownerReferences[0]
  not re_match("^crossplane.adobe.io", ownerRef.apiVersion)
  msg := "owner reference is not valid"
}
```

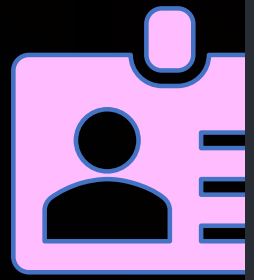
OPA Policies

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – Block Exploits

#6. A rogue tenant should not be able to exploit another tenant's ProviderConfig



Tenant H
Namespa

```
violation[{"msg": msg}] {
  not input.review.object.metadata.labels["crossplane.io/composite"]
  msg := "crossplane composition not used"
}

violation[{"msg": msg}] {
  not input.review.object.metadata.ownerReferences
  msg := "ownerReferences field not present"
}

violation[{"msg": msg}] {
  ownerRef = input.review.object.metadata.ownerReferences[0]
  not re_match("^crossplane.adobe.io", ownerRef.apiVersion)
  msg := "owner reference is not valid"
}
```

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – High performance

#7. Tenant workflows should not slow down with large number of CRDs

```
$ kubectl get crd | wc -l
Waited for 1.033772408s due to client-side throttling, not priority and fairness,
request: GET:https://api.ipa.org
Waited for 5.033772408s due to client-side throttling, not priority and fairness,
request: GET:https://api.ipa.org
     993
```



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Multi-tenancy and security – High performance

#7. Tenant workflows should not slow down with large number of CRDs

upbound



Client: **0.24.0+**

Server: **1.25.0+**

Nic Cope's blog post:

https://blog.upbound.io/scaling-kubernetes-to-thousands-of-crds/

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Before

Service team

- Provision cloud accounts
- Use GitOps for compute deployments
- Define infrastructure resources/templates
- Learn custom workflows/tools
- Use custom solution for infra resource provisioning
- Maintain/Manage provisioned non-compute resources
- Track provisioned non-compute resources separately than compute resources

Platform team

- Manage GitOps workflows
- Enforce multi-tenancy and security
- No visibility into non-compute resources
- Harder troubleshooting in real-world scenarios

Red – Pain Point
Green – Working well
White - Neutral

After

Service team

- Provision cloud accounts
- Use GitOps for compute deployments
- Provision infra resources via GitOps workflow
- Specify and track provisioned non-compute resources in Kubernetes-native way
- Maintain/Manage provisioned non-compute resources
- Consistent way to create/replicate environments

Platform team

- Manage GitOps workflows
- Enforce multi-tenancy and security
- Define “blessed” infrastructure resources/templates
- Maintain “standardized” infra provisioning workflow based on GitOps
- Improved auditability, observability
- Improved service architecture understanding
- Improved MTTR

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Challenges/Unknowns

Hub cluster, K8s – scale, performance



Other cloud provider support



Technology maturity



Tooling inertia



Ease of migration



Community support



aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How can the learnings of Adobe and AWS help you?

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Vikram Sethi
vsethi@adobe.com
@hellovsethi

Nima Kaviani
nkaviani@amazon.com
@nimak

Gaurav Dhamija
dhamijag@amazon.com
@gauravdhamija83



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

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.