

Flexing your Security Governance with Azure Policy as Code

Jesse Loudon

Principal Consultant (LAB³) // Microsoft MVP (Azure)



Speaker Bio

Allow us to introduce... *LAB*

PROFESSIONAL SERVICES

Richard Verkade, Leif Eriksen, Jacky Huang, Dylan Longworth, Jesse Loudon, Chris Pleasants, Kerrin D'Arcy

Peter Goreta, Vinod Ralh, Hamed Monafred, Ted Aritao, Sue Coates, Magid Shehab

Mark Robertson, Emma Nguyen-Huu, David Barakat, Gowri Kandasamy, Neal Stirton, Eddie El-Leissy, Matt Ratcliffe

- **Background:** Infra, SysAdmin, PreSales, Consulting
- **Pre 2018 Tech:** AD, Exchange, VMware, Citrix
- **Post 2018 Tech:** Microsoft Azure, IaC, DevOps, Git




Microsoft Reactor Sydney

Session Abstract



David O'Brien (he/him)

@david_obrien 

Aside from "putting RDP on the internet", what is the most common security misconfiguration you see people do in the cloud?
Don't care if AWS or Azure.

What **common security misconfigurations** exist?



Why **Azure Policy can help to govern** your environments.



How **Azure Policy as Code** fits.



What can go wrong? – Identity & Access



AAD role assignments with Contributor rights when not needed

Admins without MFA enabled

Guest accounts with Owner/Global Administrator rights

User Administrator role overuse



What can go wrong? - Network



Resources with VNET integration/private endpoints but public access still open e.g. no firewall config.

NSGs allowing any-any for source/destination

Storage Account blob containers with public access enabled

Public IPs on domain controllers



What can go wrong? – Encryption



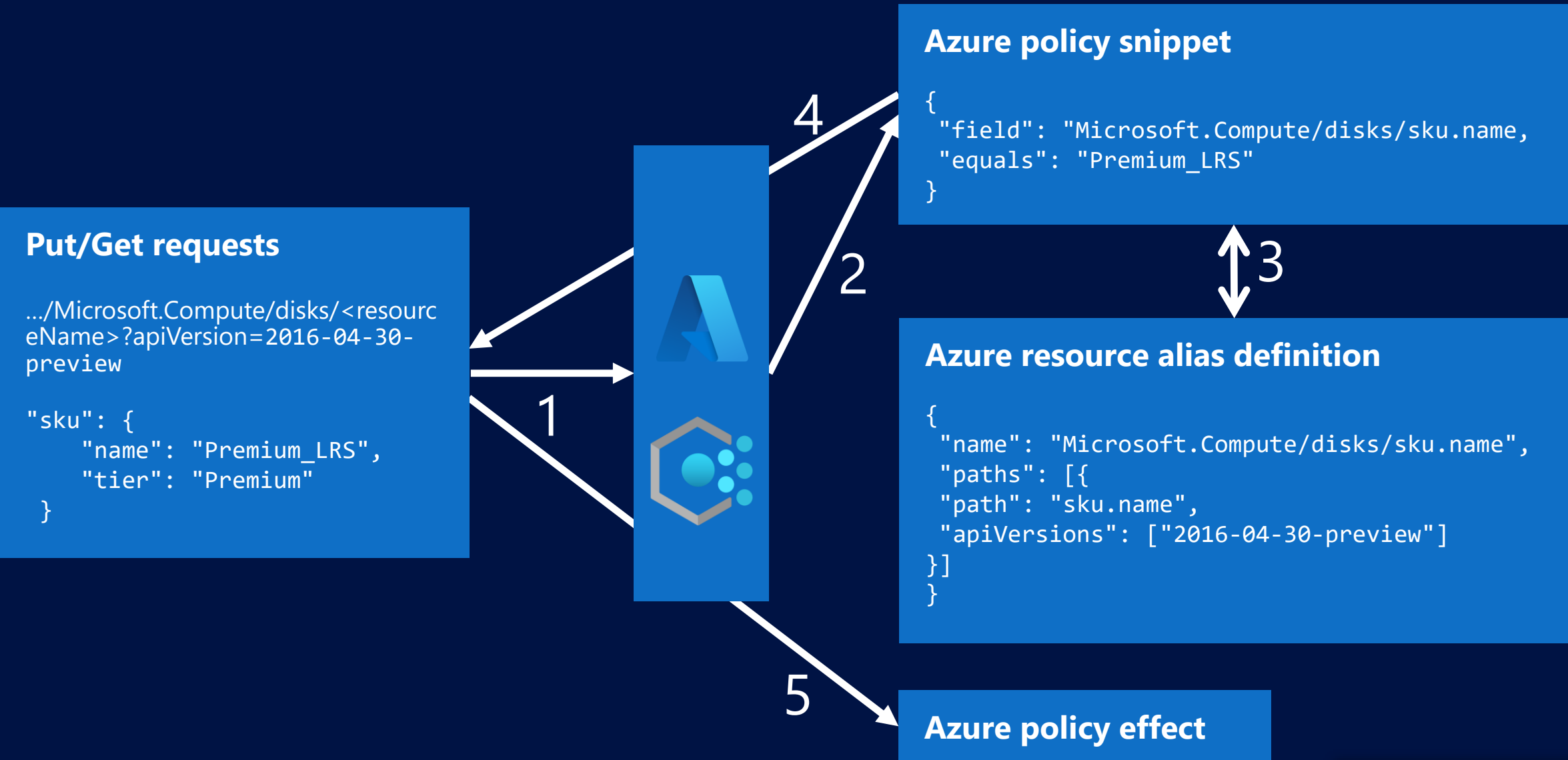
Lack of disk encryption within OS
(Bitlocker/DMcrypt)

Databases lacking encryption

TLS 1.0/1.1 enabled



AzPolicy – Resource Aliases



Put/Get requests

```
.../Microsoft.Compute/disks/<resourceName>?apiVersion=2016-04-30-preview
```

```
"sku": {  
  "name": "Premium_LRS",  
  "tier": "Premium"  
}
```

Azure policy snippet

```
{  
  "field": "Microsoft.Compute/disks/sku.name",  
  "equals": "Premium_LRS"  
}
```

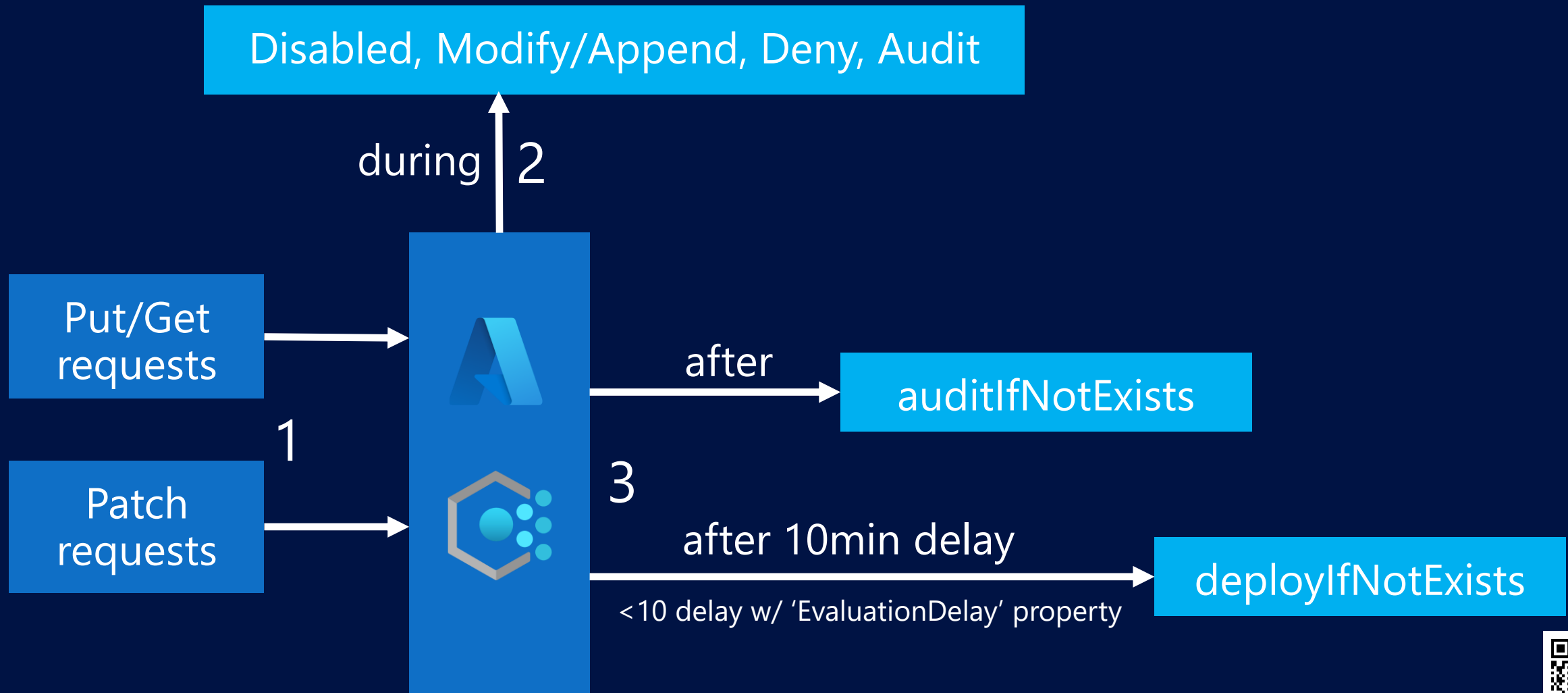
Azure resource alias definition

```
{  
  "name": "Microsoft.Compute/disks/sku.name",  
  "paths": [{  
    "path": "sku.name",  
    "apiVersions": ["2016-04-30-preview"]  
  }]  
}
```

Azure policy effect



AzPolicy - Effects



AzPolicy - Demo



IDE



Security
Governance



Cloud
Environment



IaC Language



Source Control



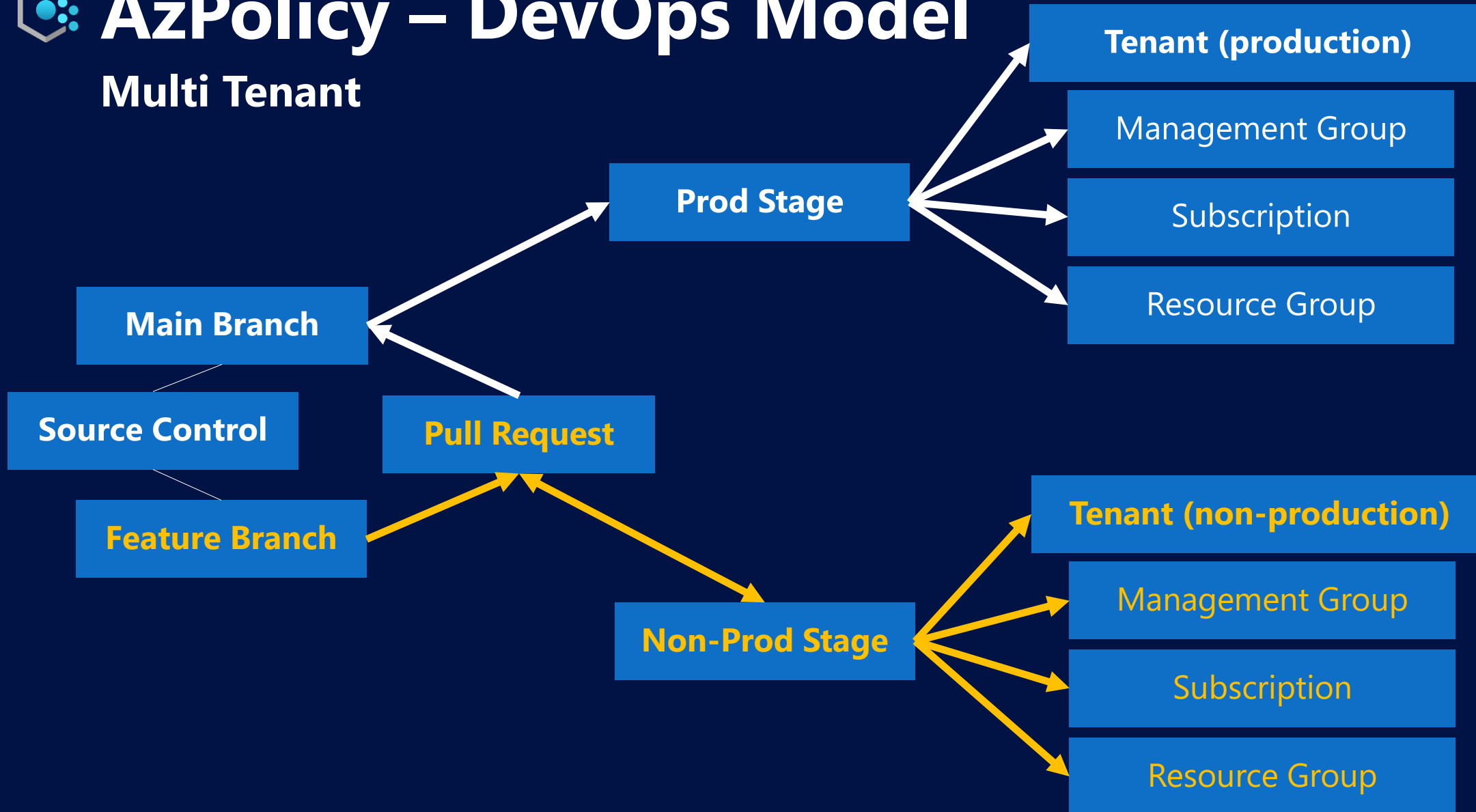
AzPolicy - RBAC Design

Identity/AAD Group	AAD Built-in Roles	AAD Role Assignment Scope
Service Principals	<ul style="list-style-type: none">Resource Policy ContributorUser Access AdministratorContributor (bicep deployments)	<ul style="list-style-type: none">TenantManagement Group
Policy Developers	<ul style="list-style-type: none">Resource Policy ContributorContributor	<ul style="list-style-type: none">Management GroupSubscription
Policy Users	<ul style="list-style-type: none">Resource Policy Contributor	<ul style="list-style-type: none">Management GroupSubscription
Policy Readers	<ul style="list-style-type: none">Reader	<ul style="list-style-type: none">TenantManagement Group



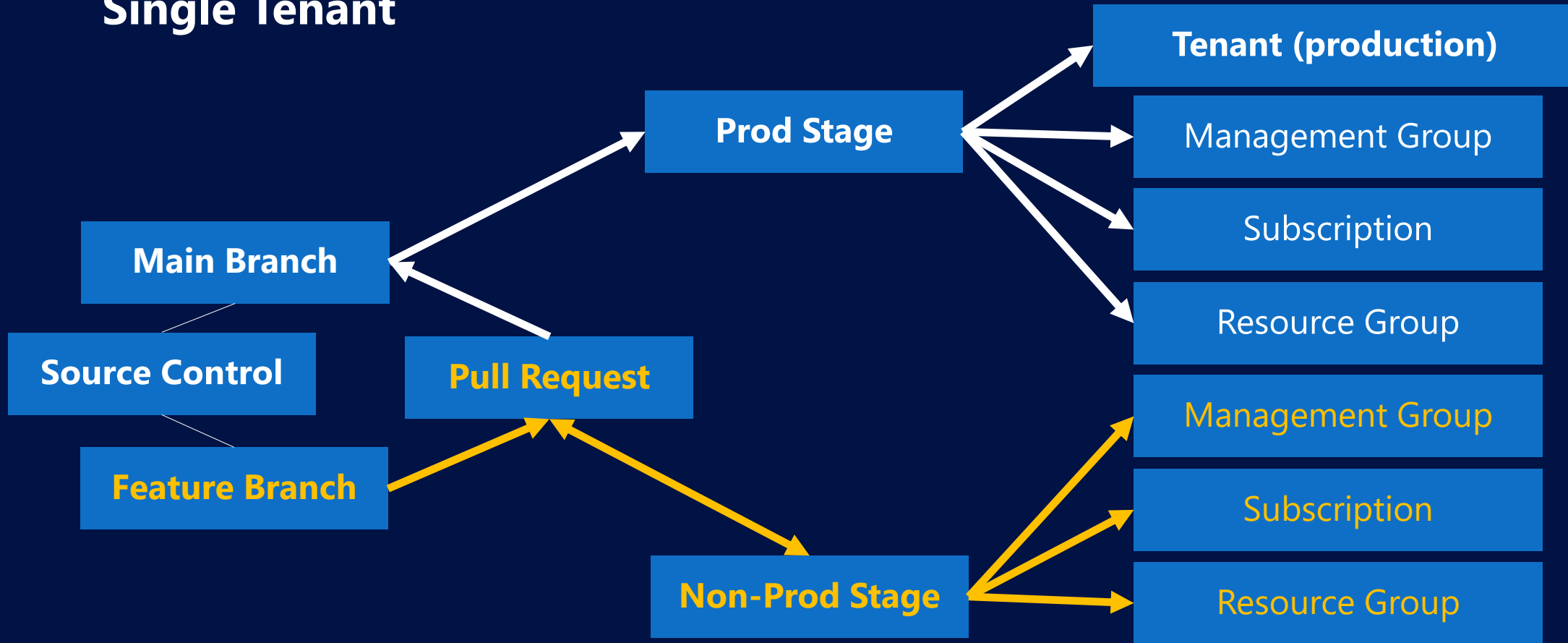
AzPolicy – DevOps Model

Multi Tenant



AzPolicy – DevOps Model

Single Tenant



AzPolicy - Ecosystem

Official Docs	aka.ms/azurepolicy bicep, arm templates, api references
Official Code	github/azure-policy github/community-policy
Official Tooling	GitHub, Azure DevOps, Azure Policy extension (VS Code)
Languages	Azure CLI, ARM Templates, PowerShell, Bicep language, HashiCorp Language, Pulumi
Community	azadvertizer.net github/awesome-azure-policy

- **773** GA policy definitions
- **61** categories
- **49771** aliases
- **104** namespaces
- **883** resource types
- **17691** network aliases



Your Next Steps

{ Investigate Security gaps within your environment }

{ Review existing policy repo code/pipeline for improvements }

{ Setup policy non-compliance alerting and remediation workflows }

{ Group common policies into initiatives for assignment }

{ Enhance your policy-as-code workflow with CI/CD stages }

{ Implement policy assignment custom deny messages }

{ Find an existing built-in policy to customise or create your own }

{ Setup your policy development environment }

{ Create, deploy, and test your 1st custom policy }



Thanks For Watching!

Connect w/ Jesse

[

 @coder_au,

 jloudon.com,

 @jesseloudon

]

