AWS re:Invent

NET335-R1

# Cloud networking for the on-premises network engineer

**Sidhartha Chauhan**
Solutions Architect
Amazon Web Services

**Nick Sciarrilli**
Solutions Architect
Amazon Web Services

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

# How is the cloud changing traditional networking?

# Translate on-premises networking into cloud

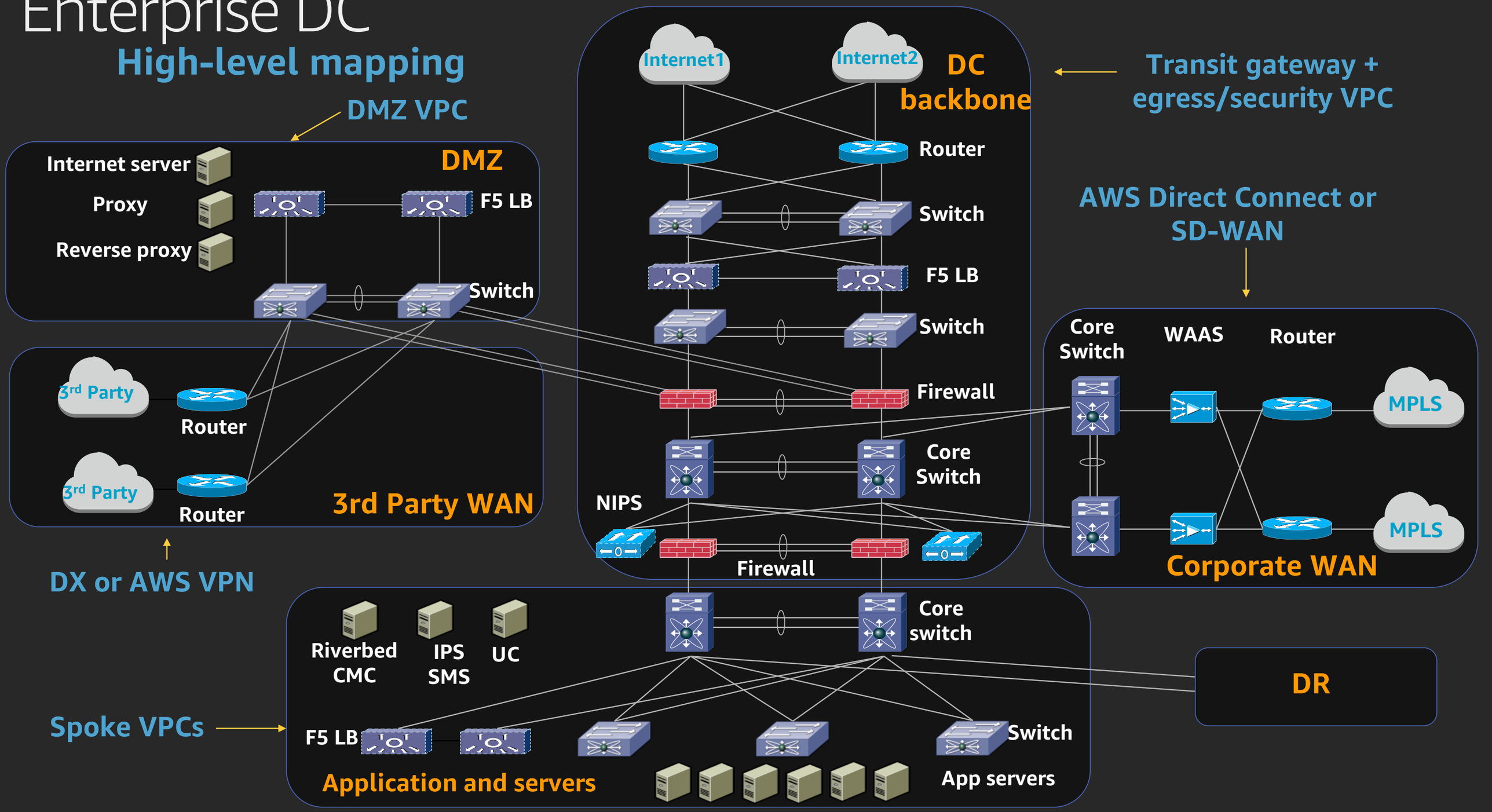
AWS re:Invent

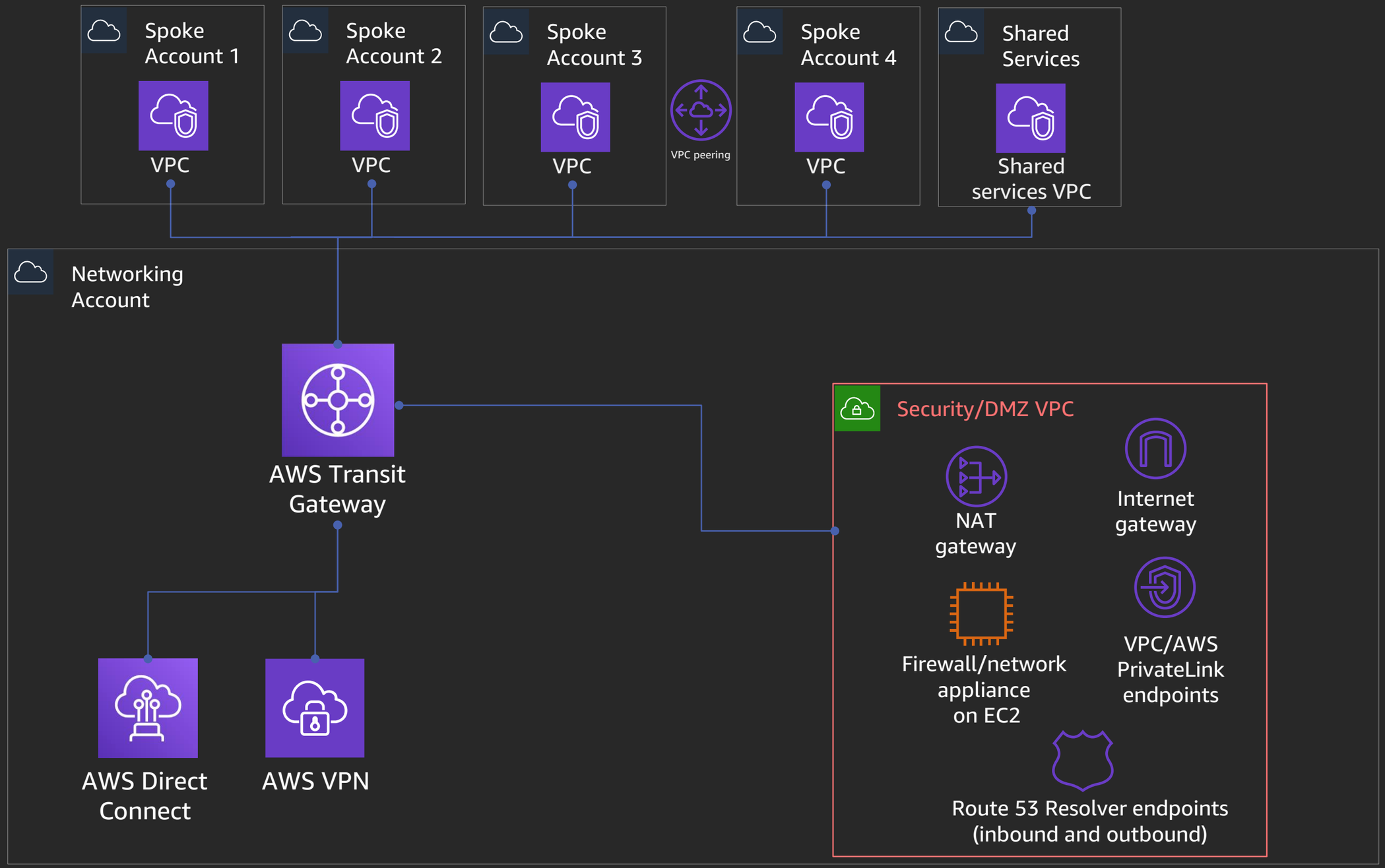
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

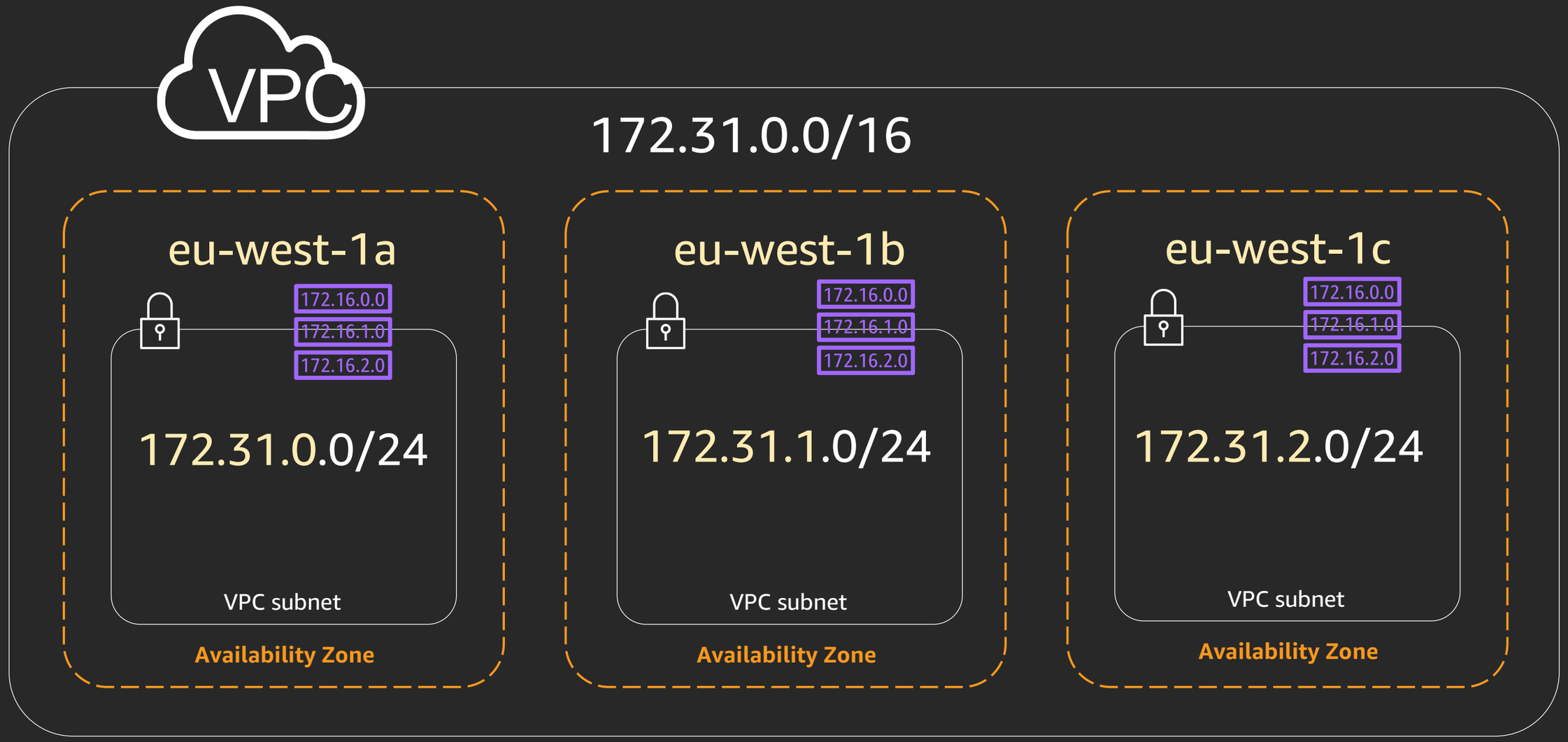
# Enterprise DC

## High-level mapping

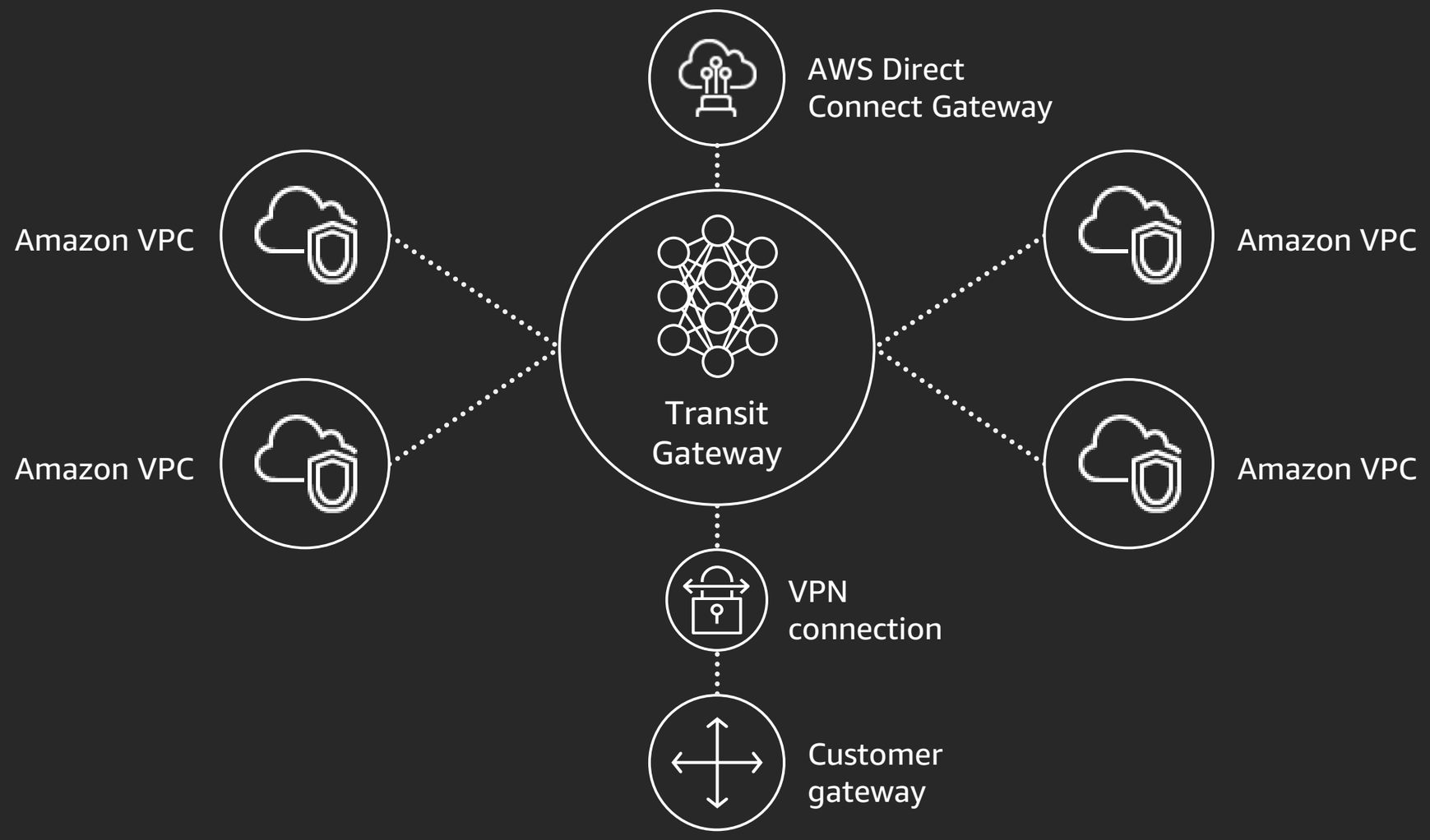


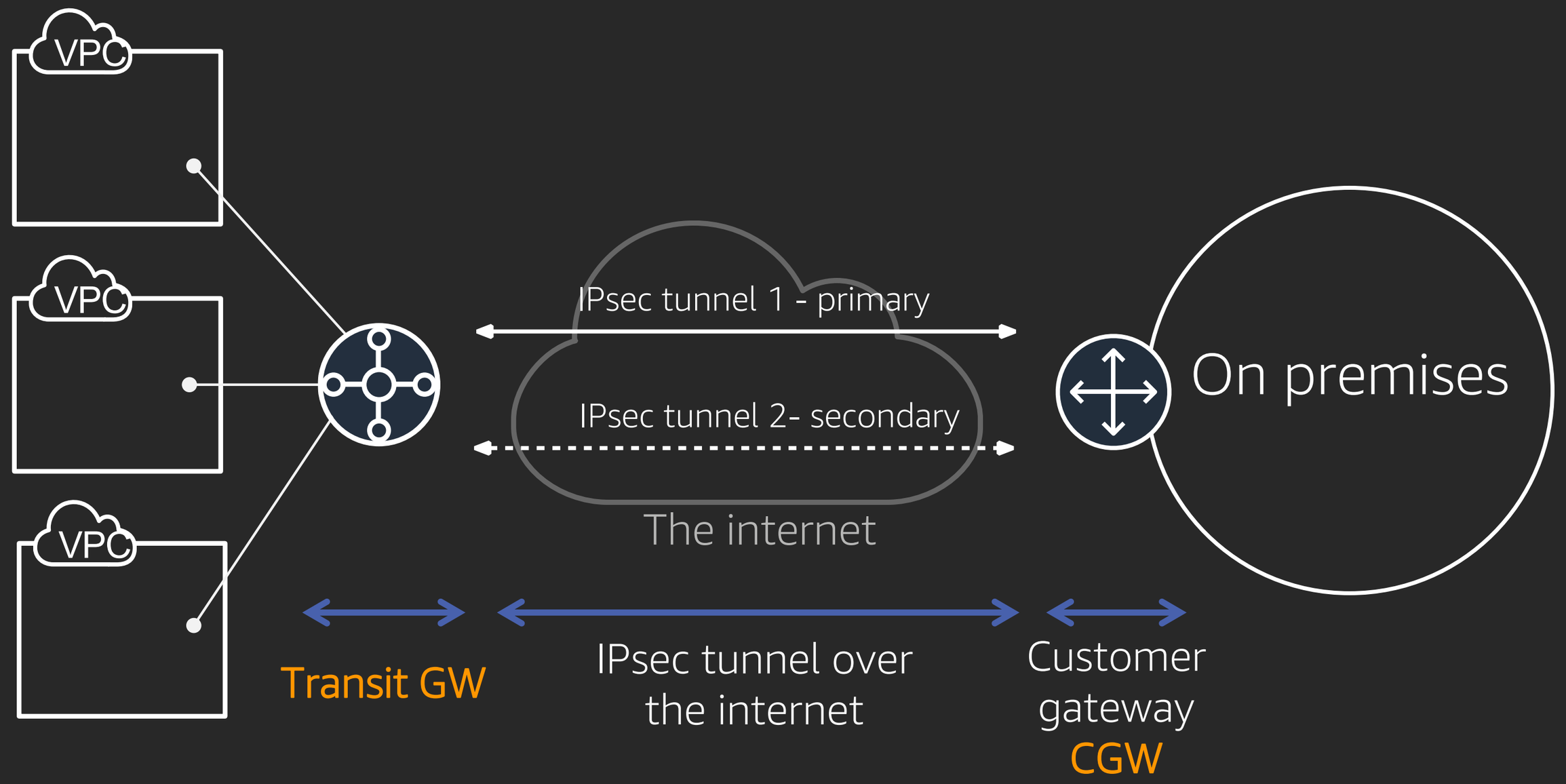

Spoke Account 1
VPC
Spoke Account 2
VPC
Spoke Account 3
VPC
VPC peering
Spoke Account 4
VPC
Shared Services
Shared services VPC
Networking Account
AWS Transit Gateway
AWS Direct Connect
AWS VPN
Security/DMZ VPC
NAT gateway
Internet gateway
Firewall/network appliance on EC2
VPC/AWS PrivateLink endpoints
Route 53 Resolver endpoints (inbound and outbound)

# VPC subnets and Availability Zones



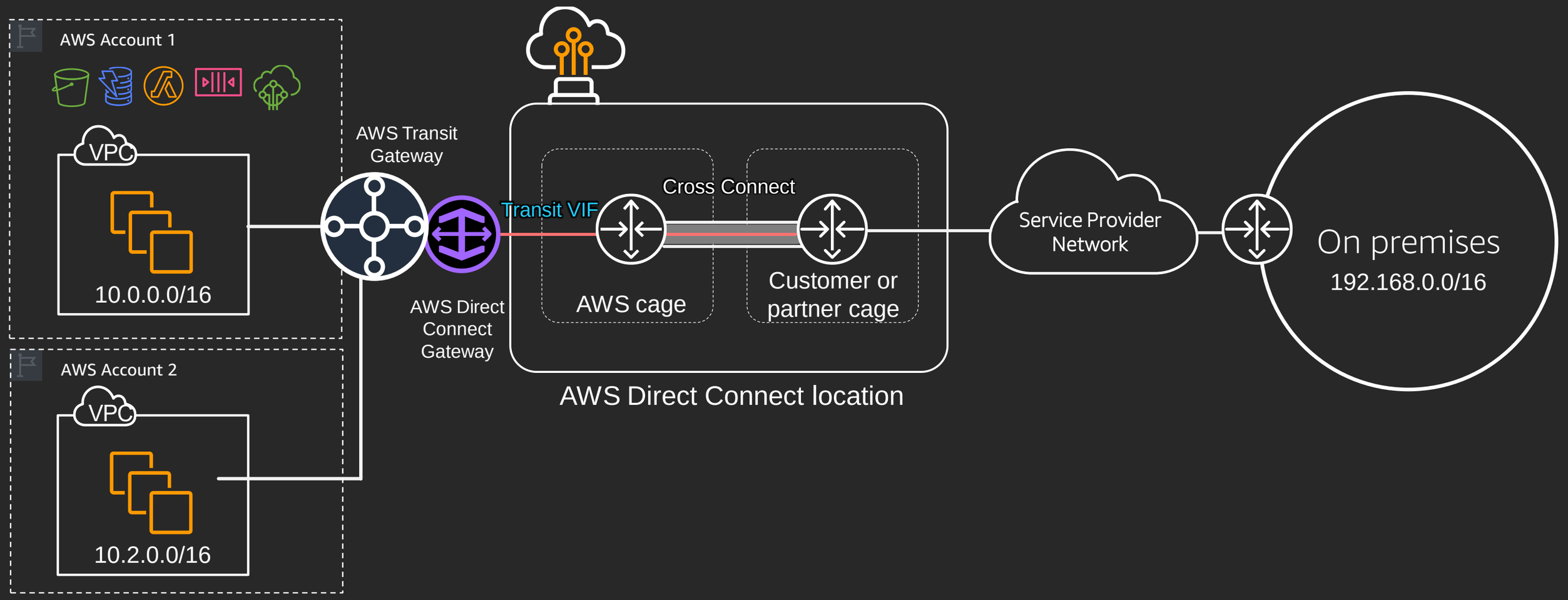
# Transit Gateway

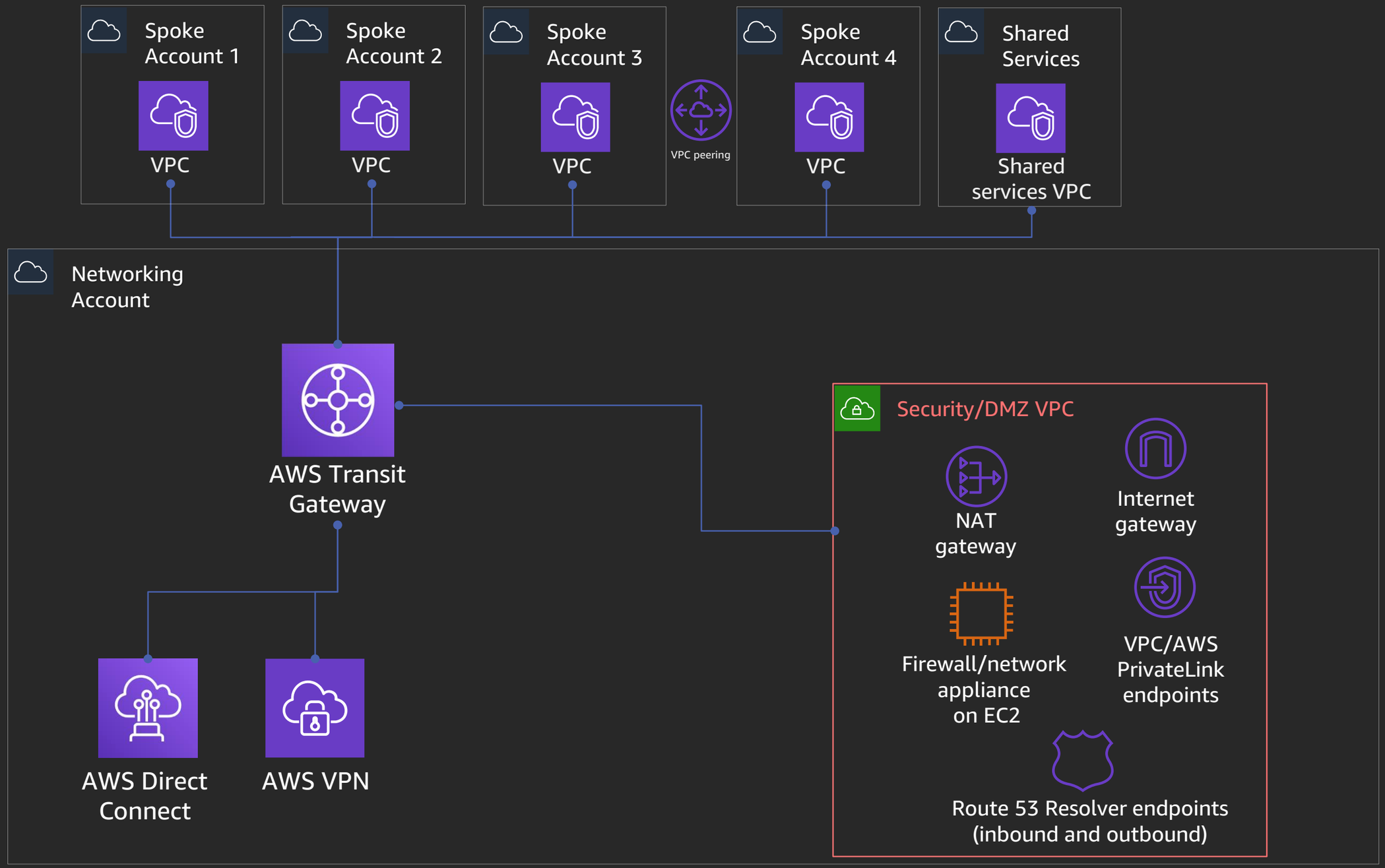



VPC
VPC
VPC
IPsec tunnel 1 - primary
IPsec tunnel 2- secondary
The internet
On premises
Transit GW
IPsec tunnel over the internet
Customer gateway
CGW

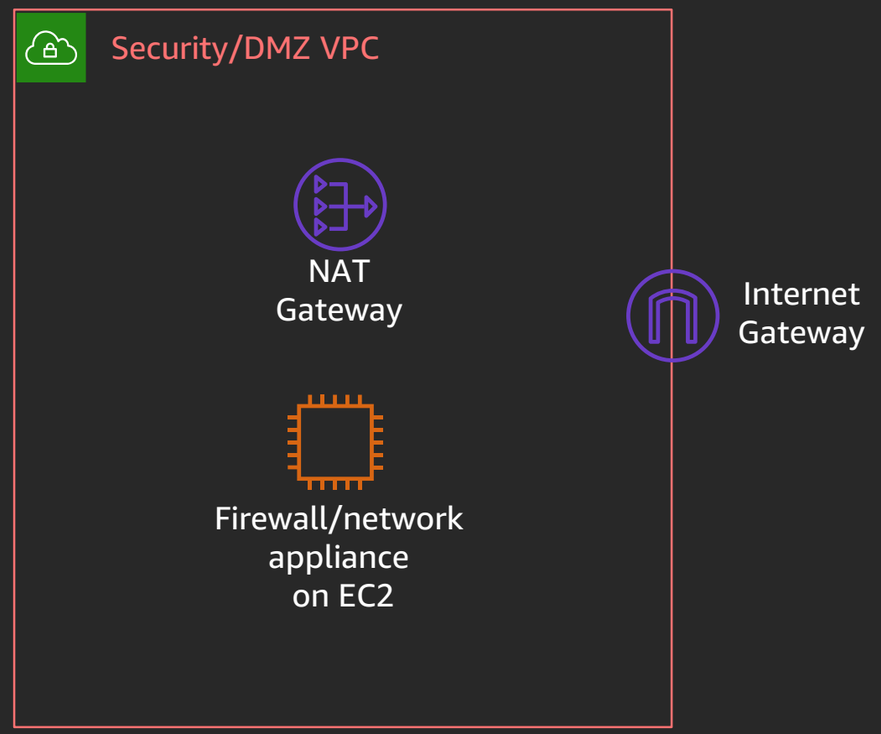
# AWS Direct Connect

## One Transit VIF → Many VPCs




Spoke Account 1
VPC
Spoke Account 2
VPC
Spoke Account 3
VPC
VPC peering
Spoke Account 4
VPC
Shared Services
Shared services VPC
Networking Account
AWS Transit Gateway
AWS Direct Connect
AWS VPN
Security/DMZ VPC
NAT gateway
Internet gateway
Firewall/network appliance on EC2
VPC/AWS PrivateLink endpoints
Route 53 Resolver endpoints (inbound and outbound)

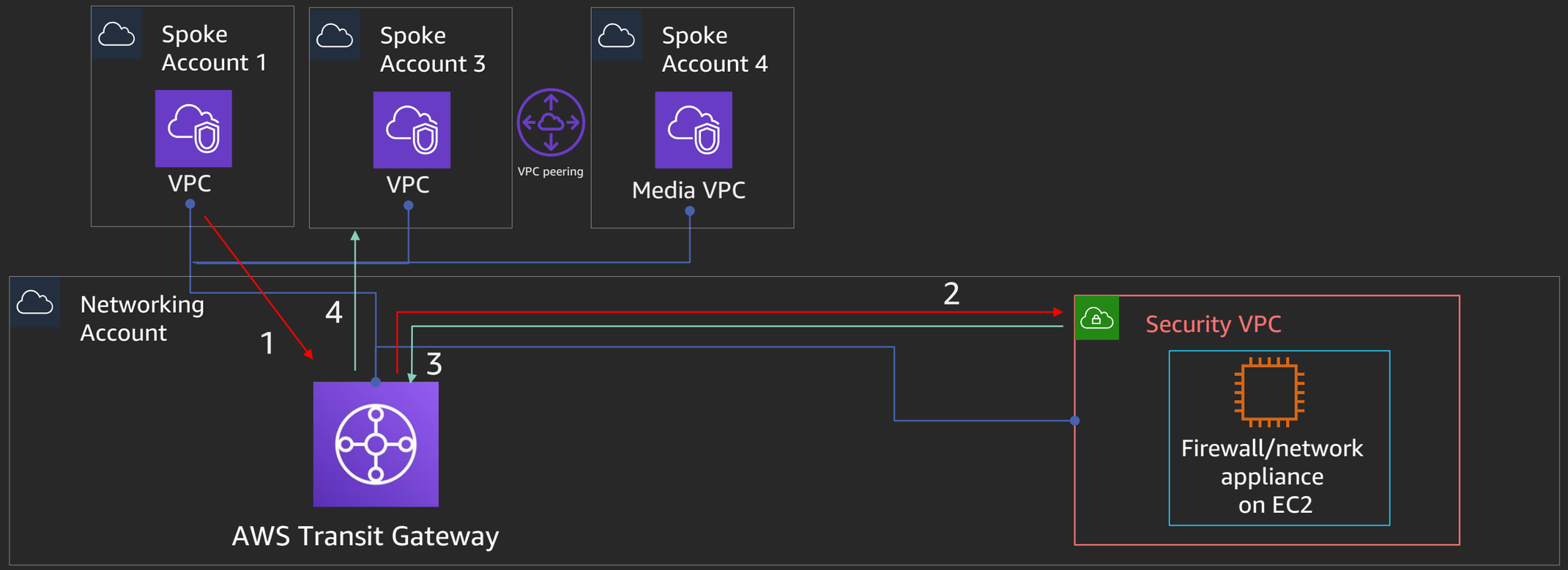
# Security

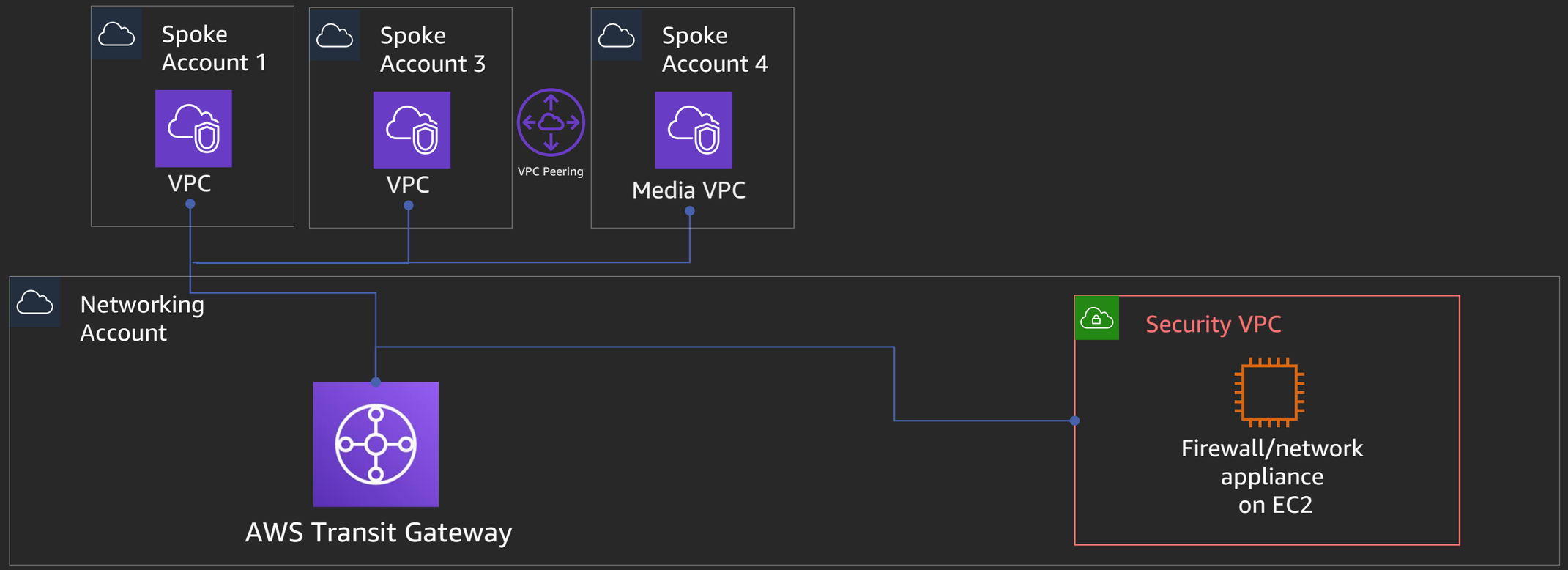


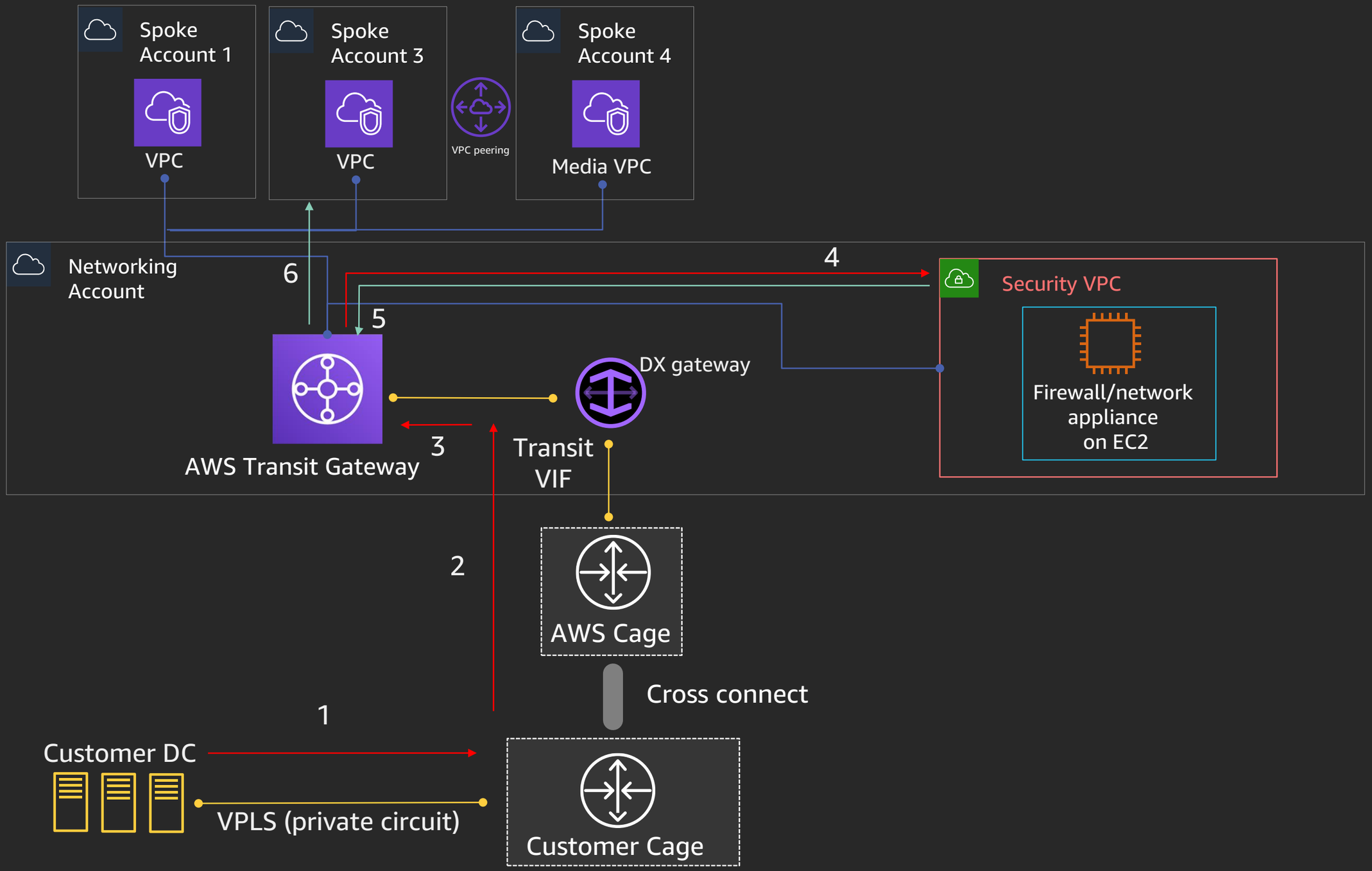

vs

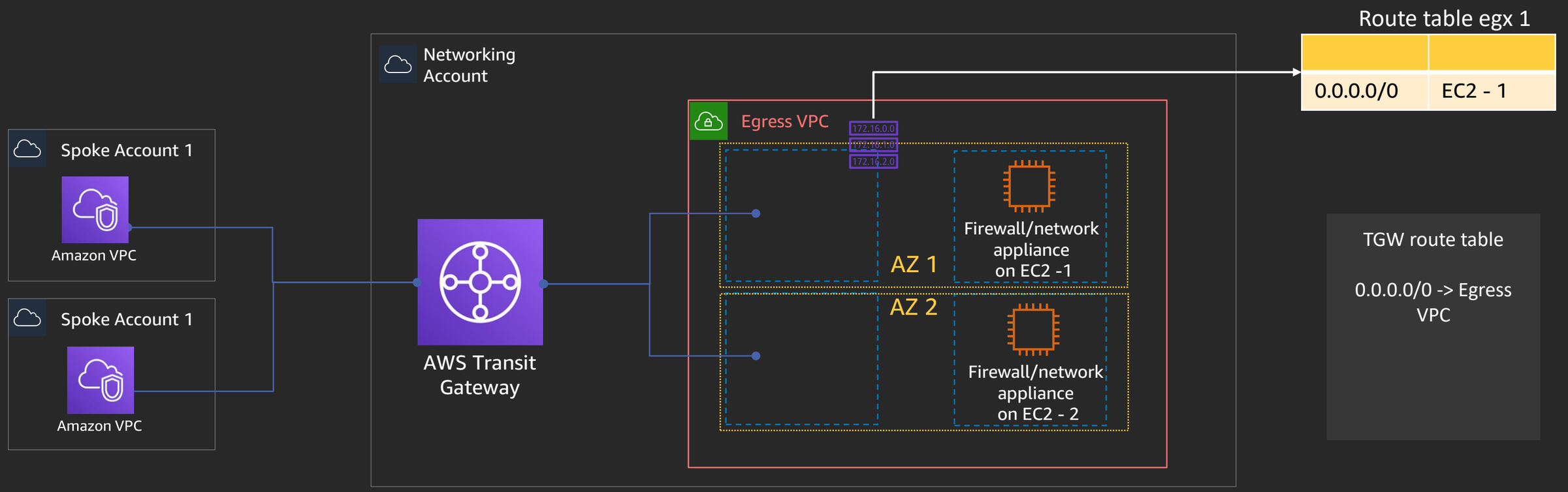
- Security groups
- Network ACLs
- AWS Security Hub, Amazon GuardDuty
- Amazon Inspector
- Service level policies, IAM policies

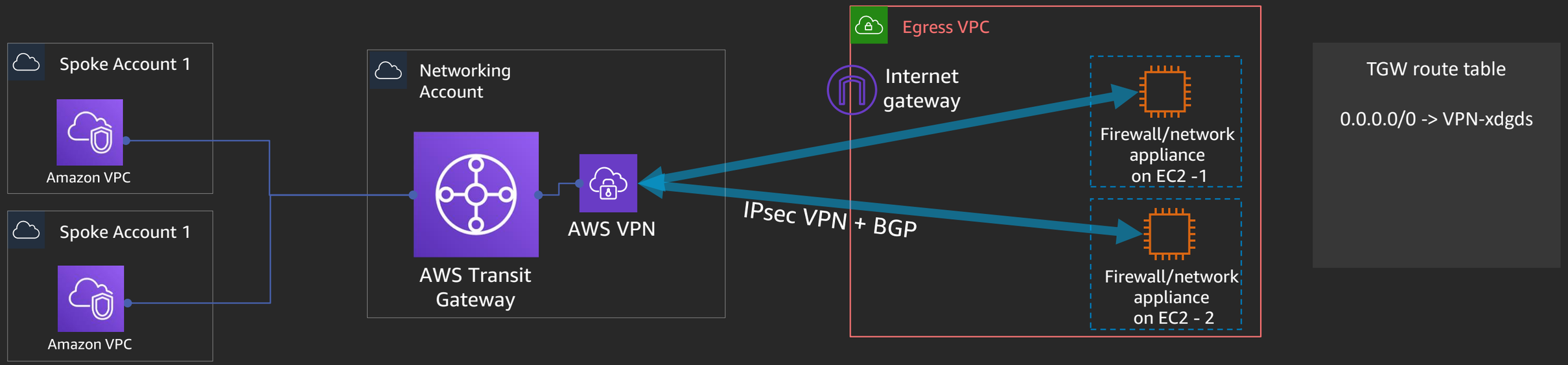
Centralized DMZ VPC


Spoke Account 1
VPC
Spoke Account 3
VPC
VPC peering
Spoke Account 4
Media VPC
Networking Account
1
2
3
4
AWS Transit Gateway
Security VPC
Firewall/network appliance on EC2


Spoke Account 1
VPC
Spoke Account 3
VPC
VPC Peering
Spoke Account 4
Media VPC
Networking Account
AWS Transit Gateway
Security VPC
Firewall/network appliance on EC2


Spoke Account 1
VPC
Spoke Account 3
VPC
VPC peering
Spoke Account 4
Media VPC
Networking Account
6
4
5
Security VPC
Firewall/network appliance on EC2
DX gateway
AWS Transit Gateway
3
Transit VIF
2
AWS Cage
Cross connect
1
Customer DC
VPLS (private circuit)
Customer Cage


Route table egx 1
0.0.0.0/0
EC2 - 1
Networking Account
Egress VPC
172.16.0.0
172.16.1.0
172.16.2.0
Spoke Account 1
Amazon VPC
Spoke Account 1
Amazon VPC
AWS Transit Gateway
AZ 1
AZ 2
Firewall/network appliance on EC2 -1
Firewall/network appliance on EC2 - 2
TGW route table
0.0.0.0/0 -> Egress VPC


Spoke Account 1
Amazon VPC
Spoke Account 1
Amazon VPC
Networking Account
AWS Transit Gateway
AWS VPN
IPsec VPN + BGP
Egress VPC
Internet gateway
Firewall/network appliance on EC2 -1
Firewall/network appliance on EC2 - 2
TGW route table
0.0.0.0/0 -> VPN-xdgds

# Questions?



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# DevOps
# +
# Networking



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# What is DevOps?

a software development process
a set of practices
tooling

# Why use DevOps?

**1. Agility**

Make more changes in a shorter period of time with Infrastructure as Code

**2. Version Control**

All histories are tracked and can be rolled back

**3. Reliability**

Testing can be automated and use separate environments

**4. Operations**

Failure time from recovery is reduced

# DevOps Tools

## Orchestration

AWS CloudFormation

## Git repository

AWS CodeCommit

## CI/CD Pipeline

AWS CodePipeline

# AWS CloudFormation VPC and subnet

```yaml
AWSTemplateFormatVersion: '2010-09-09'
Description: 'NET335 re:Invent 2019’

Resources:
  myVPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: 10.0.0.0/16
      Tags:
        - Key: Name
          Value: cloudformation-vpc

  mySUBNET:
    Type: AWS::EC2::Subnet
    Properties:
      AvailabilityZone: us-east-1a
      CidrBlock: 10.0.0.0/24
      VpcId: !Ref myVPC
```

# AWS CloudFormation Transit Gateway

```yaml
myTGW:
  Type: "AWS::EC2::TransitGateway"
  Properties:
    AmazonSideAsn: 65000
    Description: "TGW NET335"
    AutoAcceptSharedAttachments: "disable"
    DefaultRouteTableAssociation: "enable"
    DnsSupport: "enable"
    VpnEcmpSupport: "enable"
    Tags:
    - Key: Name
      Value: TGW-NET335

myATTACHMENT:
  Type: AWS::EC2::TransitGatewayAttachment
  Properties:
    SubnetIds:
      - !Ref mySUBNET
    TransitGatewayId: !Ref myTGW
    VpcId: !Ref myVPC
```

# Git cheat sheet

GIT BASICS

```
git init                    create empty repository
git clone <repo>            clone existing repository
git add .                   stage all changes
git commit -m ”<msg>”       commit changes with message
git status                  list staged/unchanged/untracked changes
git log                     display commit history
git diff                    show unstaged changes
```

EXAMPLE

```
<create git repository>
git clone https://git-codecommit.us-east-1.amazonaws.com/v1/repos/my-project
cd my-project/

<add some cloudformation templates>

git add .
git commit -m “updated cfn template”
git push
```

# Example CI pipeline

```yaml
image: sciarrilli/awscli

stages:
  - test
  - deploy
  - destroy

test:
  script:
  - aws cloudformation validate-template --template-body file://NET335-cfn.yaml
  # any return other than 0 from $? in bash stops the next stage

deploy:
  script:
  - aws cloudformation deploy --stack-name NET335 --template-file NET335-cfn.yaml --no-
fail-on-empty-changeset

destroy:
  script:
  - aws cloudformation delete-stack --stack-name NET335
  when: manual
```

# Q&A/whiteboarding



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Learn networking with AWS Training and Certification

## Resources created by the experts at AWS to help you build and validate networking skills

Free digital courses cover topics related to networking and content delivery, including Introduction to Amazon CloudFront and Introduction to Amazon VPC

Validate expertise with the
**AWS Certified Advanced Networking - Specialty** exam

Visit aws.amazon.com/training/paths-specialty

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws training and certification

# Thank you!

**Sidhartha Chauhan**
sidhartc@amazon.com

**Nick Sciarrilli**
nsci@amazon.com

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

Please complete the session survey in the mobile app.

AWS re:Invent

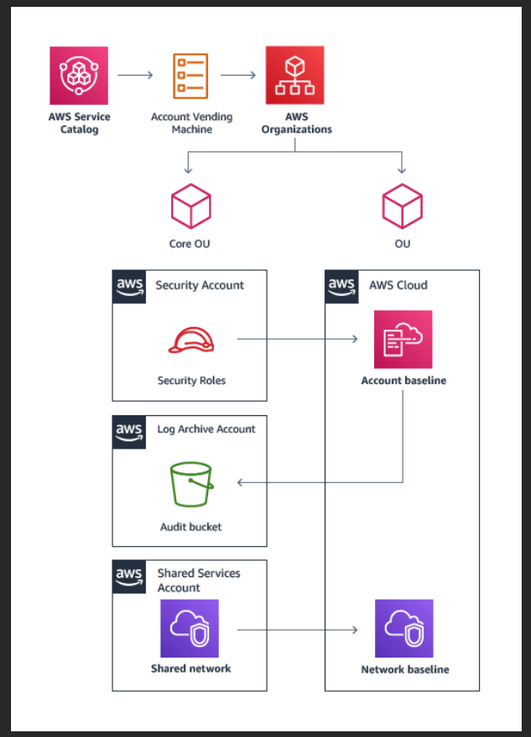
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

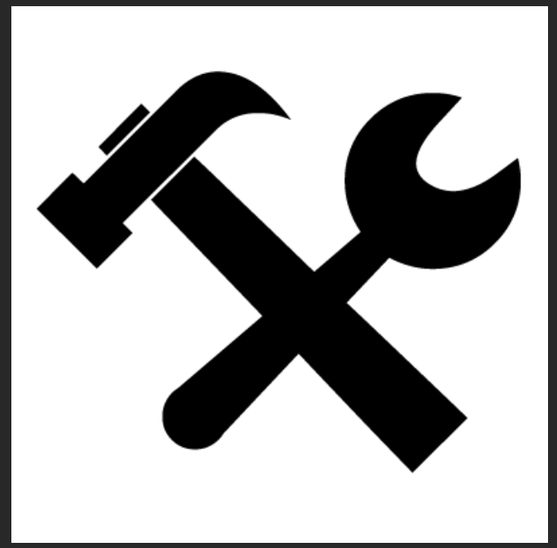
# Sample AWS CloudFormation IaC and GitLab pipeline

https://gitlab.com/sciarrilli/net335

# Setting up a landing zone

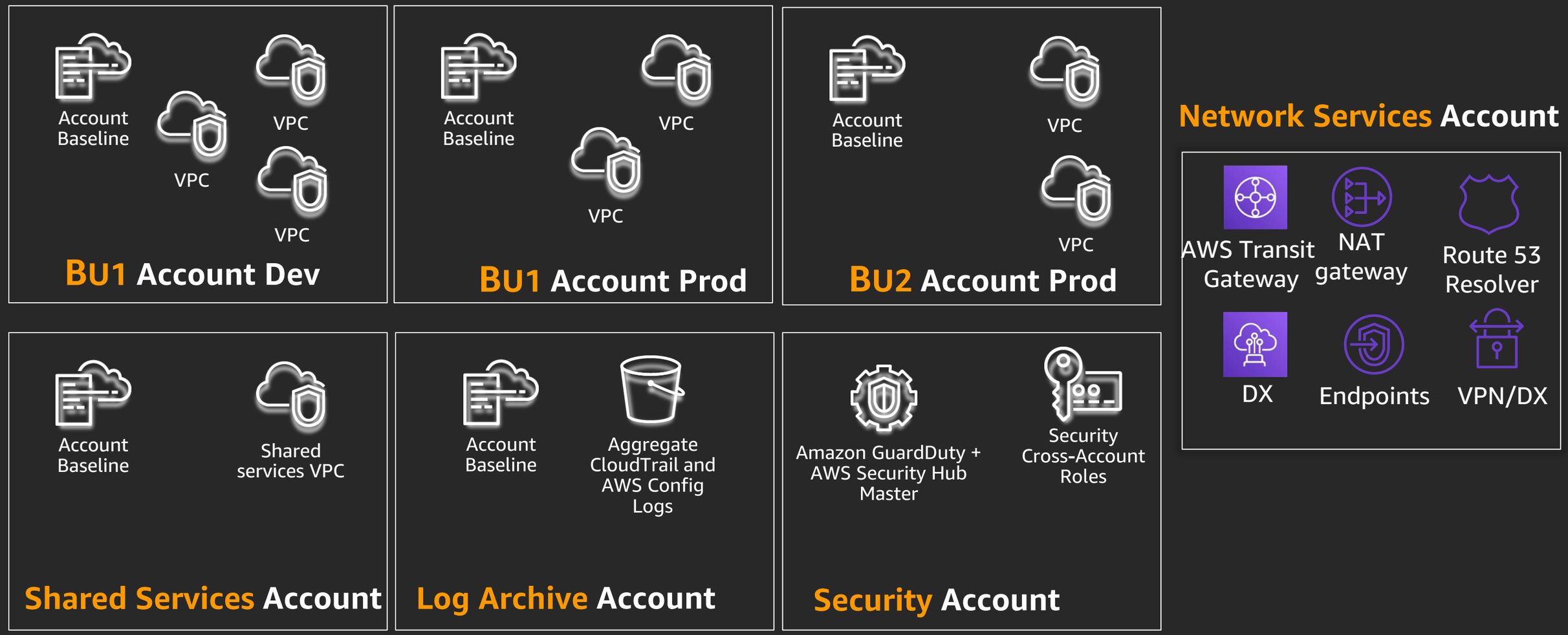


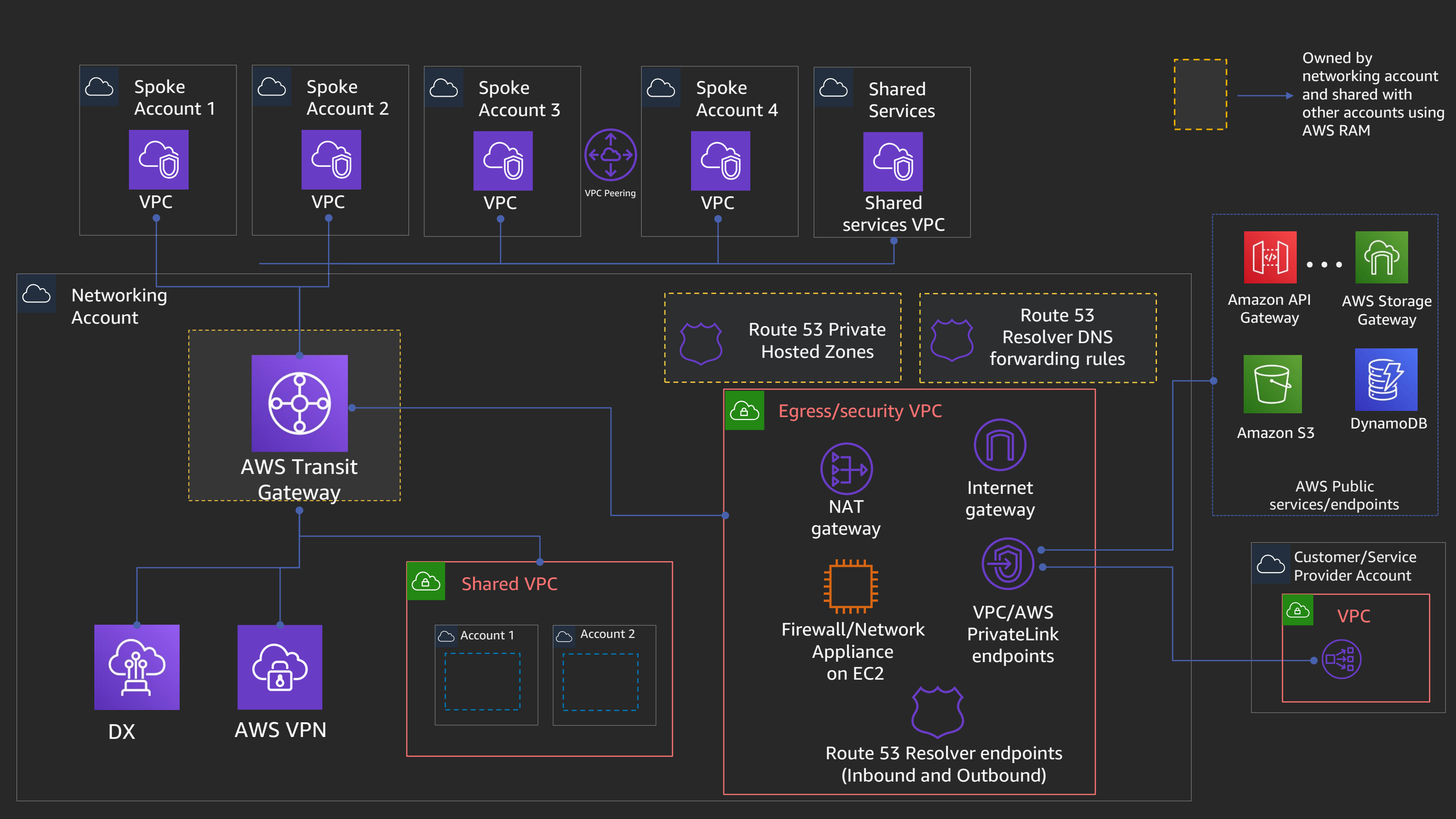

AWS Landing Zone solution



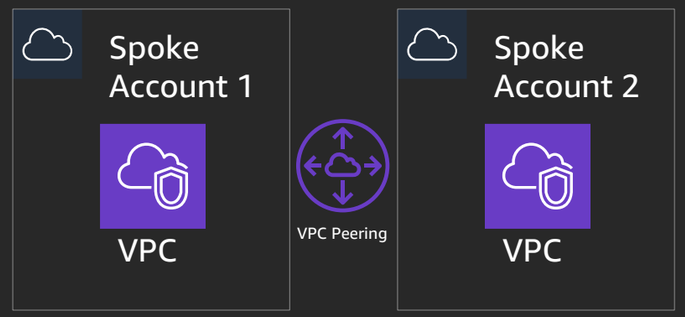
3rd party tool or build your own

# AWS Landing Zone sample reference architecture

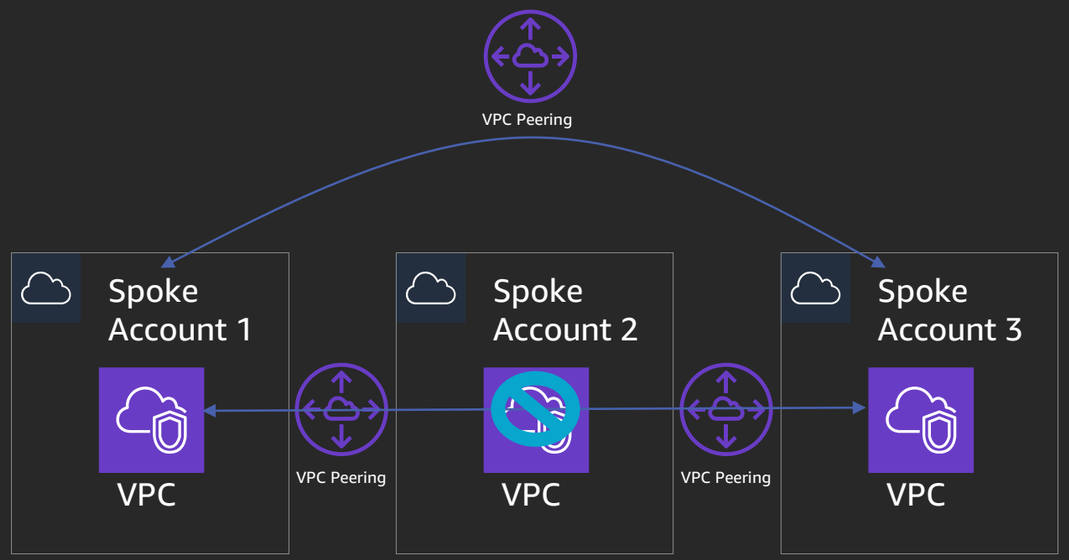



Spoke Account 1
VPC
Spoke Account 2
VPC
Spoke Account 3
VPC
VPC Peering
Spoke Account 4
VPC
Shared Services
Shared services VPC
Owned by networking account and shared with other accounts using AWS RAM
Networking Account
AWS Transit Gateway
Route 53 Private Hosted Zones
Route 53 Resolver DNS forwarding rules
Egress/security VPC
NAT gateway
Internet gateway
Firewall/Network Appliance on EC2
VPC/AWS PrivateLink endpoints
Route 53 Resolver endpoints (Inbound and Outbound)
Shared VPC
Account 1
Account 2
DX
AWS VPN
Amazon API Gateway
AWS Storage Gateway
Amazon S3
DynamoDB
AWS Public services/endpoints
Customer/Service Provider Account
VPC

# Dissecting the reference architecture—VPC-to-VPC connectivity

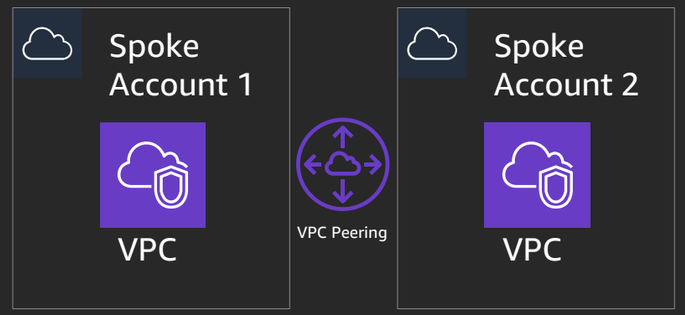



VPC peering




- No transitive routing
- Full mesh design – difficult to manage at scale

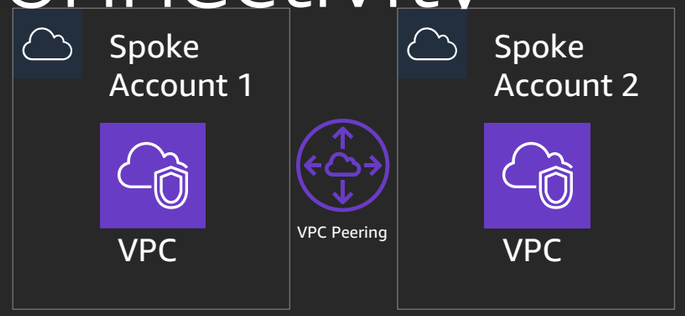
# Dissecting the reference architecture—VPC-to-VPC connectivity



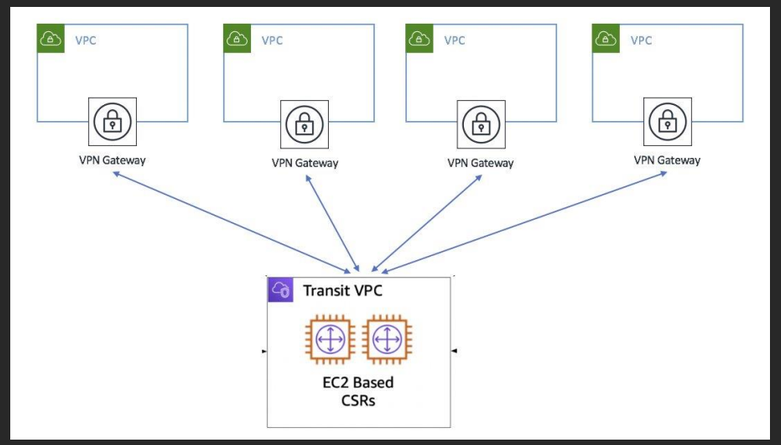

## VPC peering

- No transitive routing
- Full mesh design – Difficult to manage at scale
- 125 Max peering connections per VPC
- Full IP connectivity between VPC's - Use Security group and NACL's for security

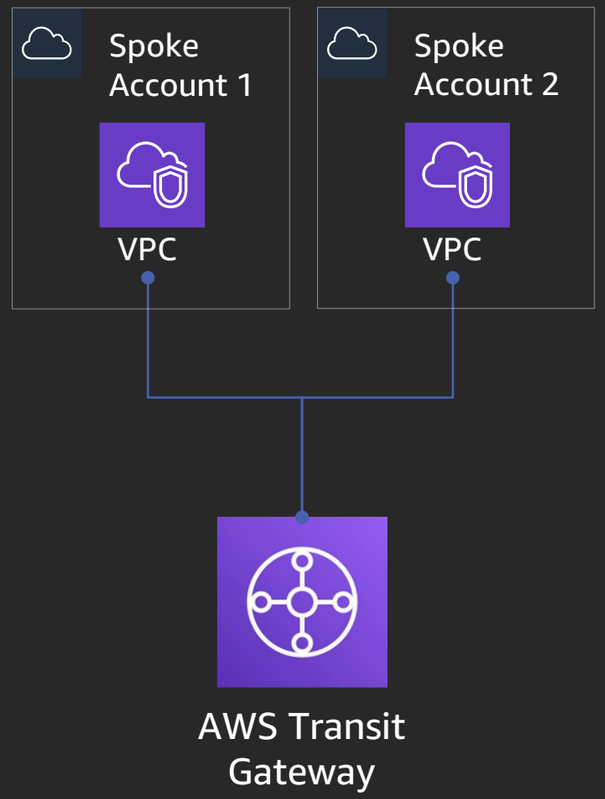
# Dissecting the reference architecture—VPC-to-VPC connectivity



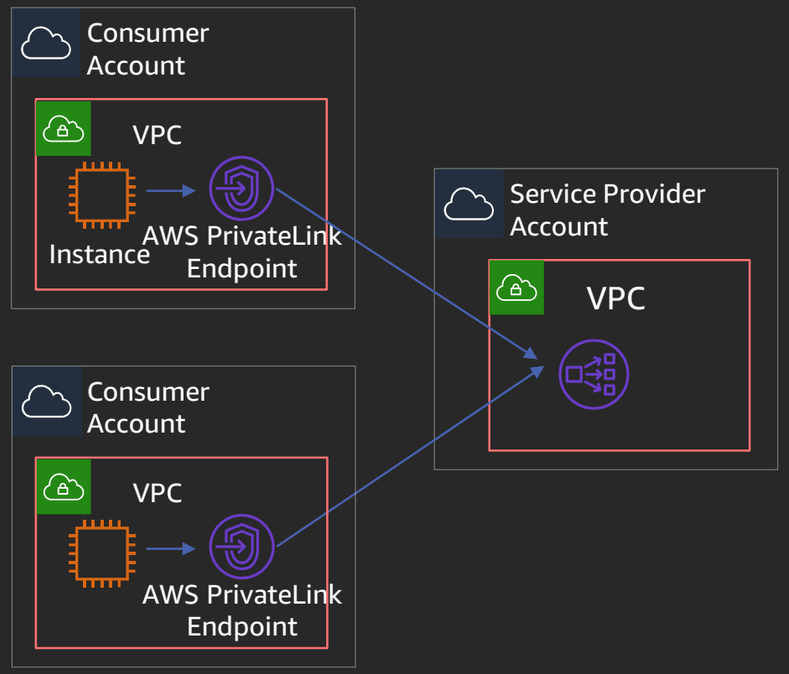

## VPC peering




## Transit VPC



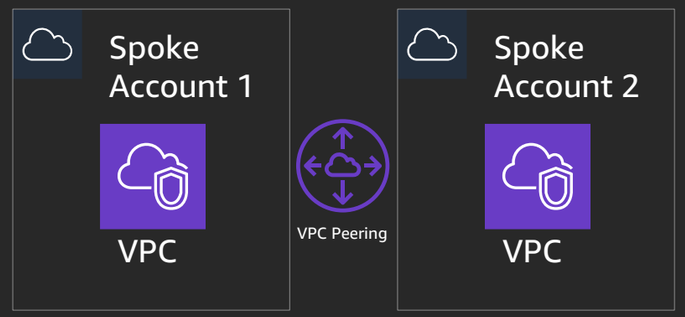

## Transit GW




## AWS PrivateLink

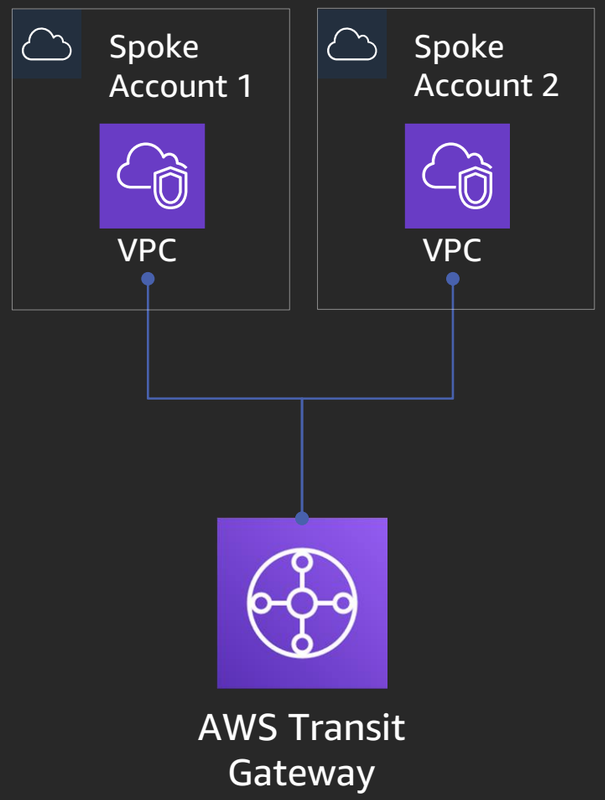
- Client/server design (many to one)
- Unidirectional connection initiation

# Dissecting the reference architecture—VPC-to-VPC connectivity

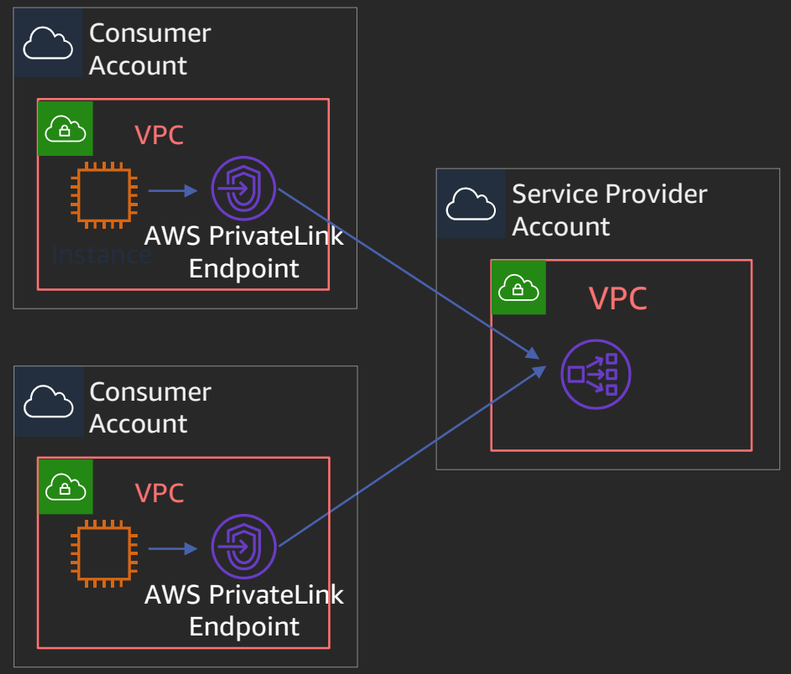



## VPC peering

- Low costs –
  - VPC peering - data transfer charges
  - AWS Transit Gateway - hourly charge per attachment + data processing charge
- No BW limits (flow limits apply) vs up to 50Gbps per attachment on TGW
- Lowest Latency
- Security Group referencing works with VPC peering



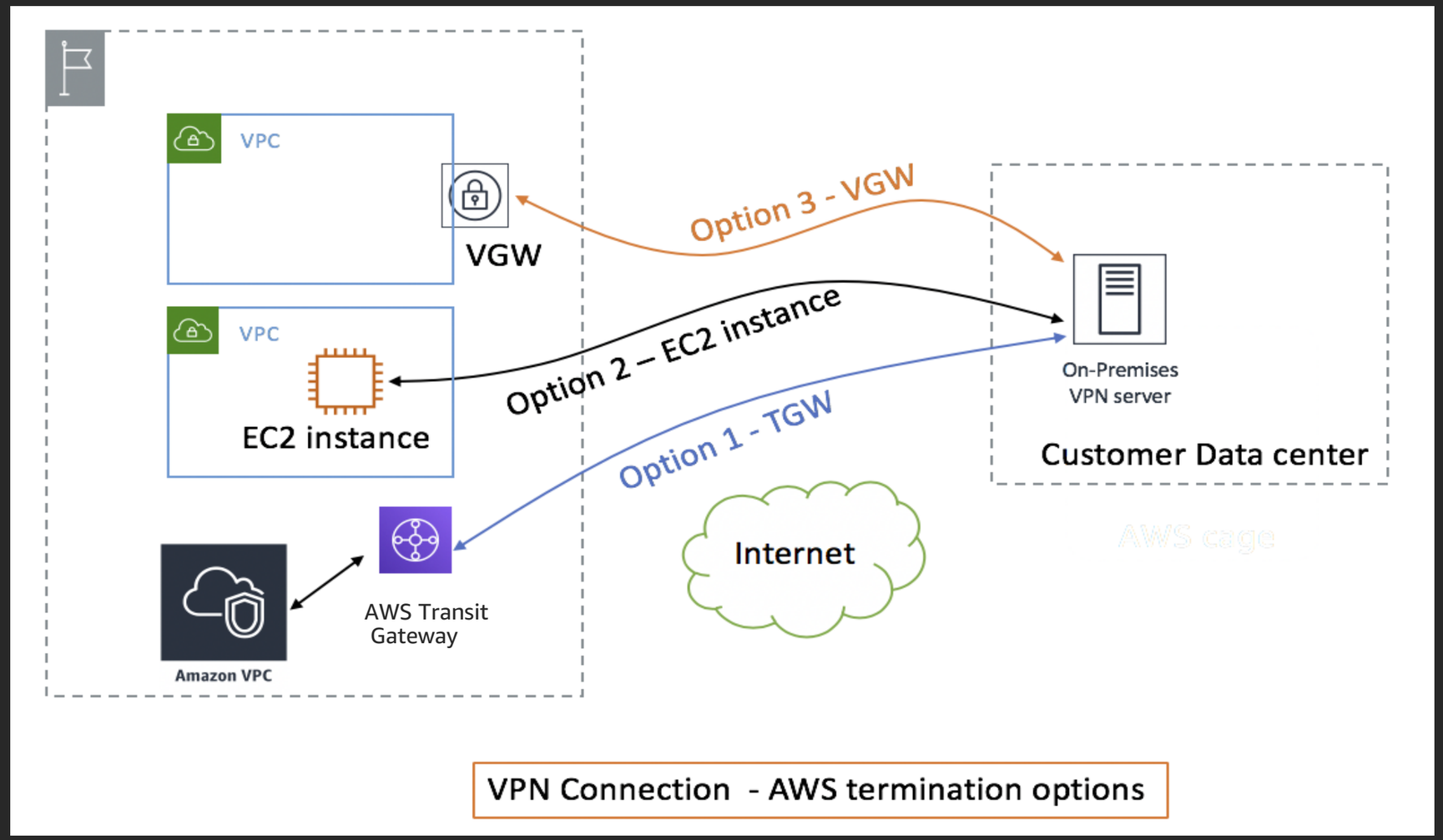

## Transit GW



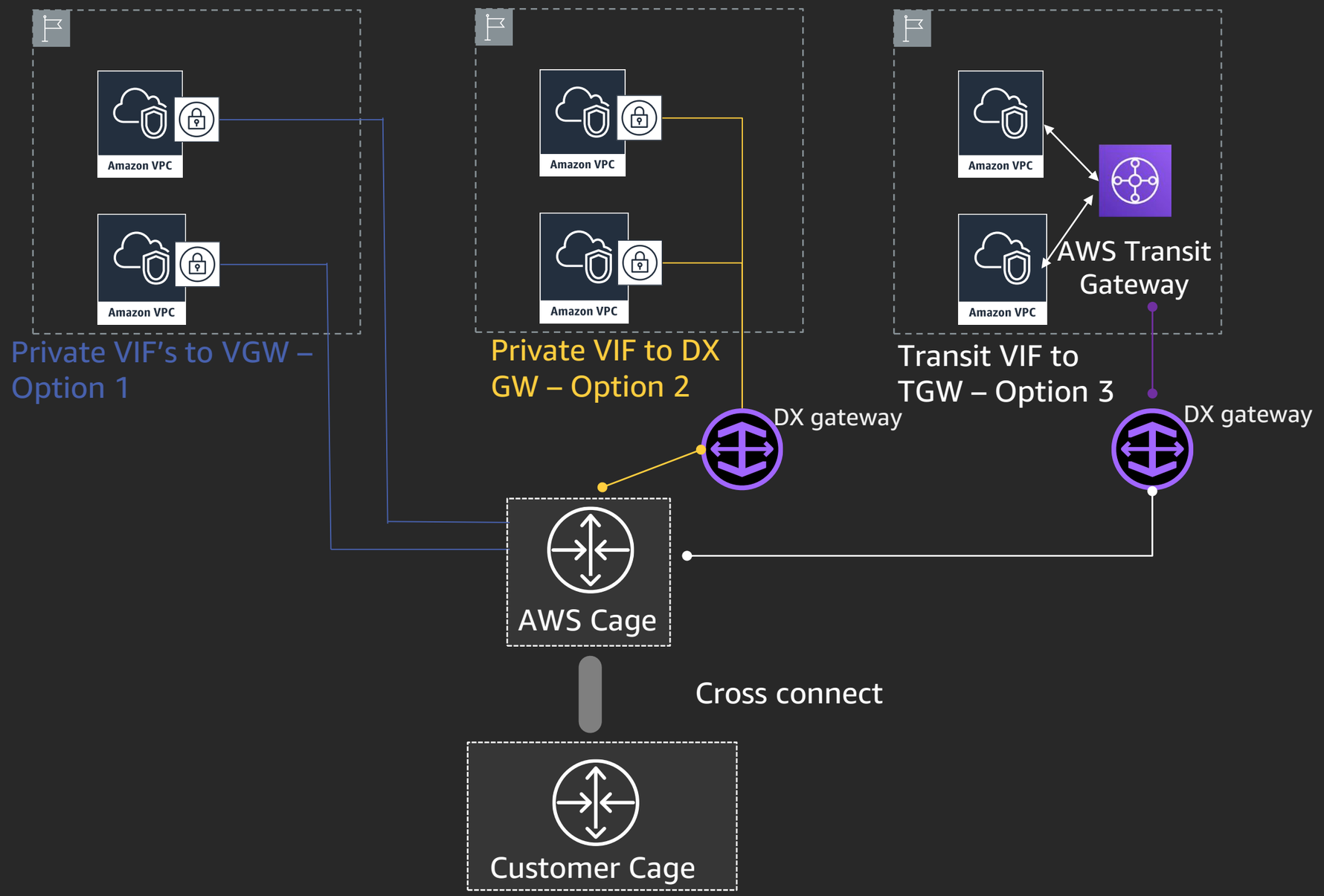

## AWS PrivateLink

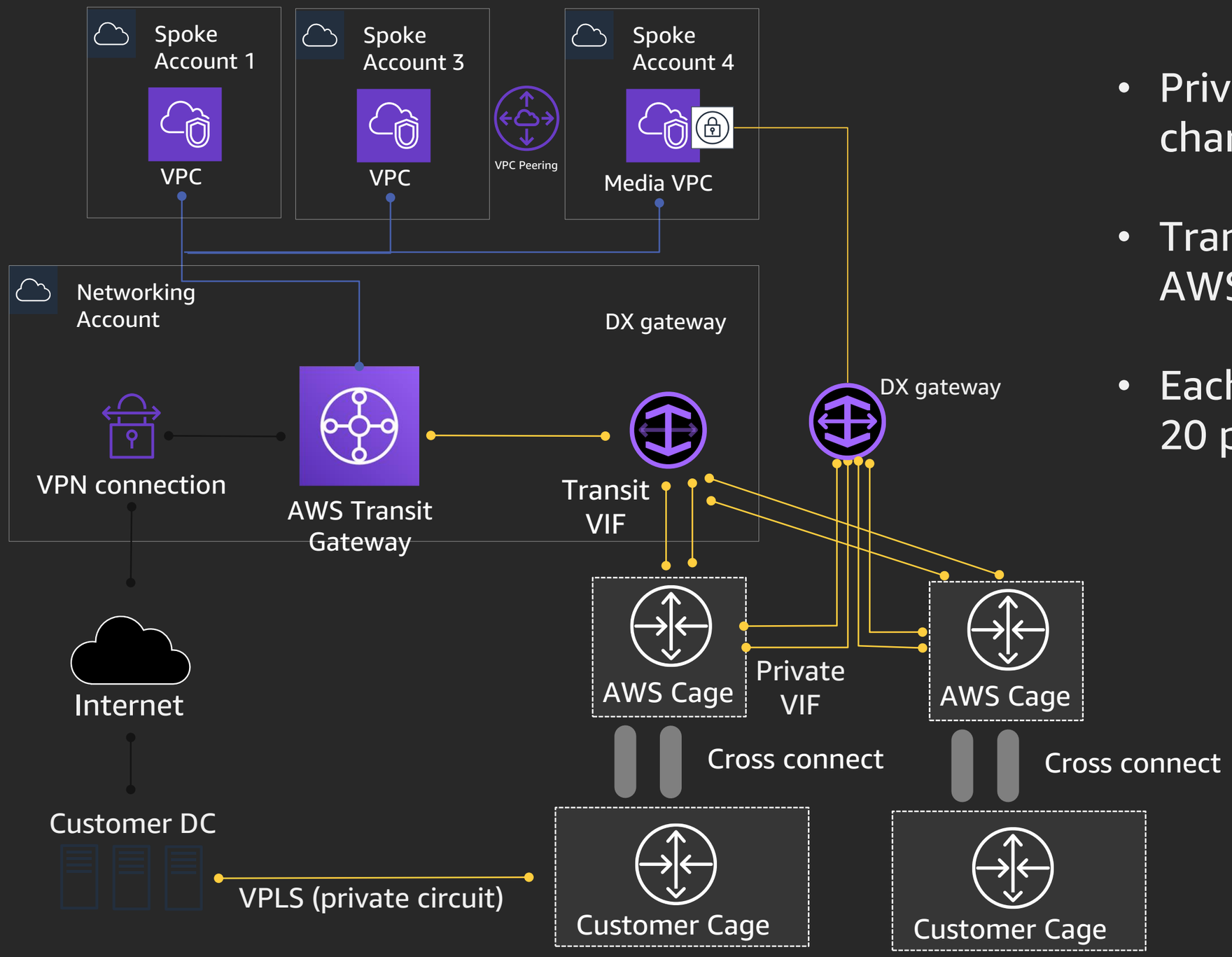
- Client/server design (many to one)
- Unidirectional connection initiation

# Hybrid connectivity options—VPN

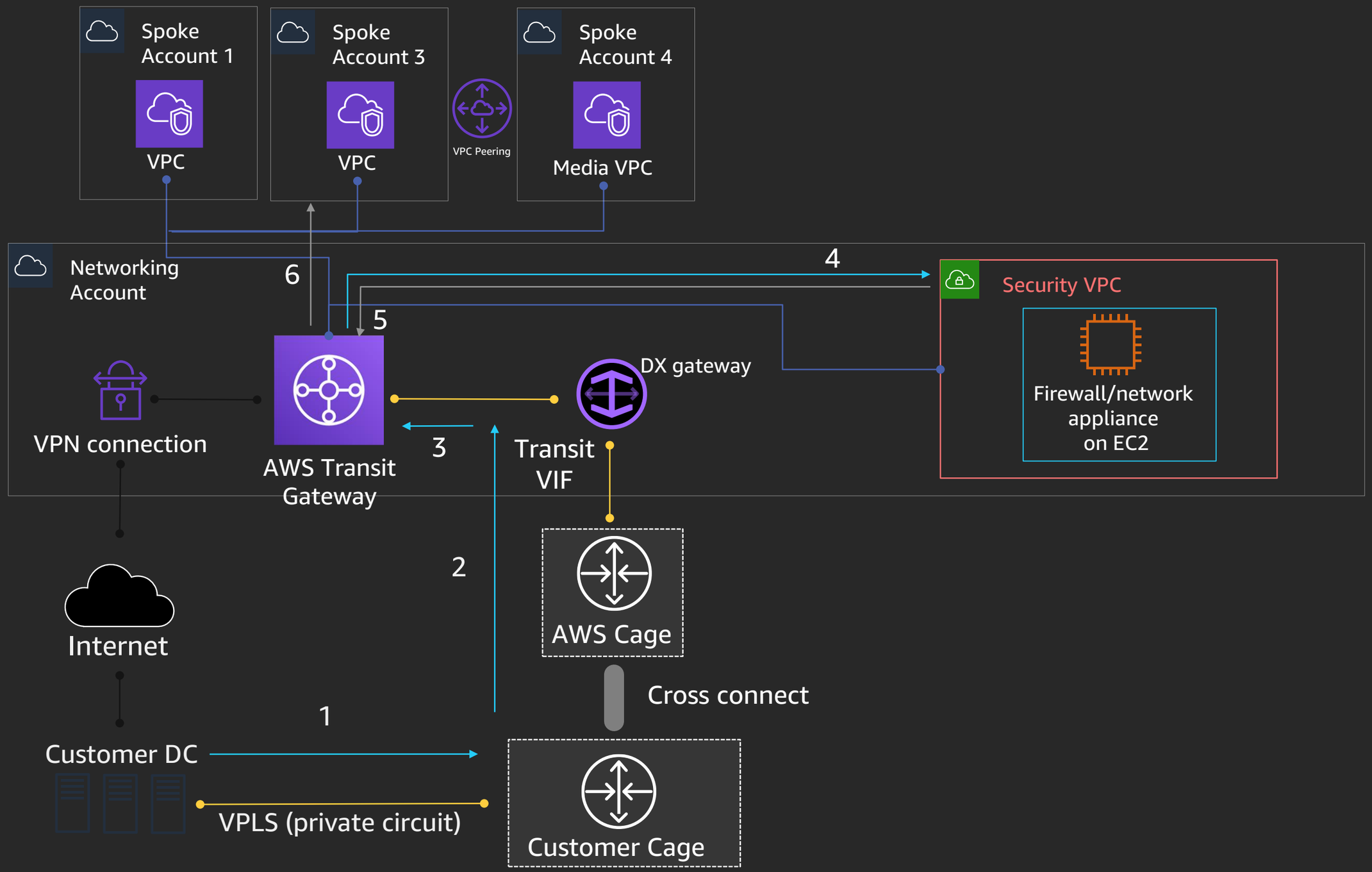


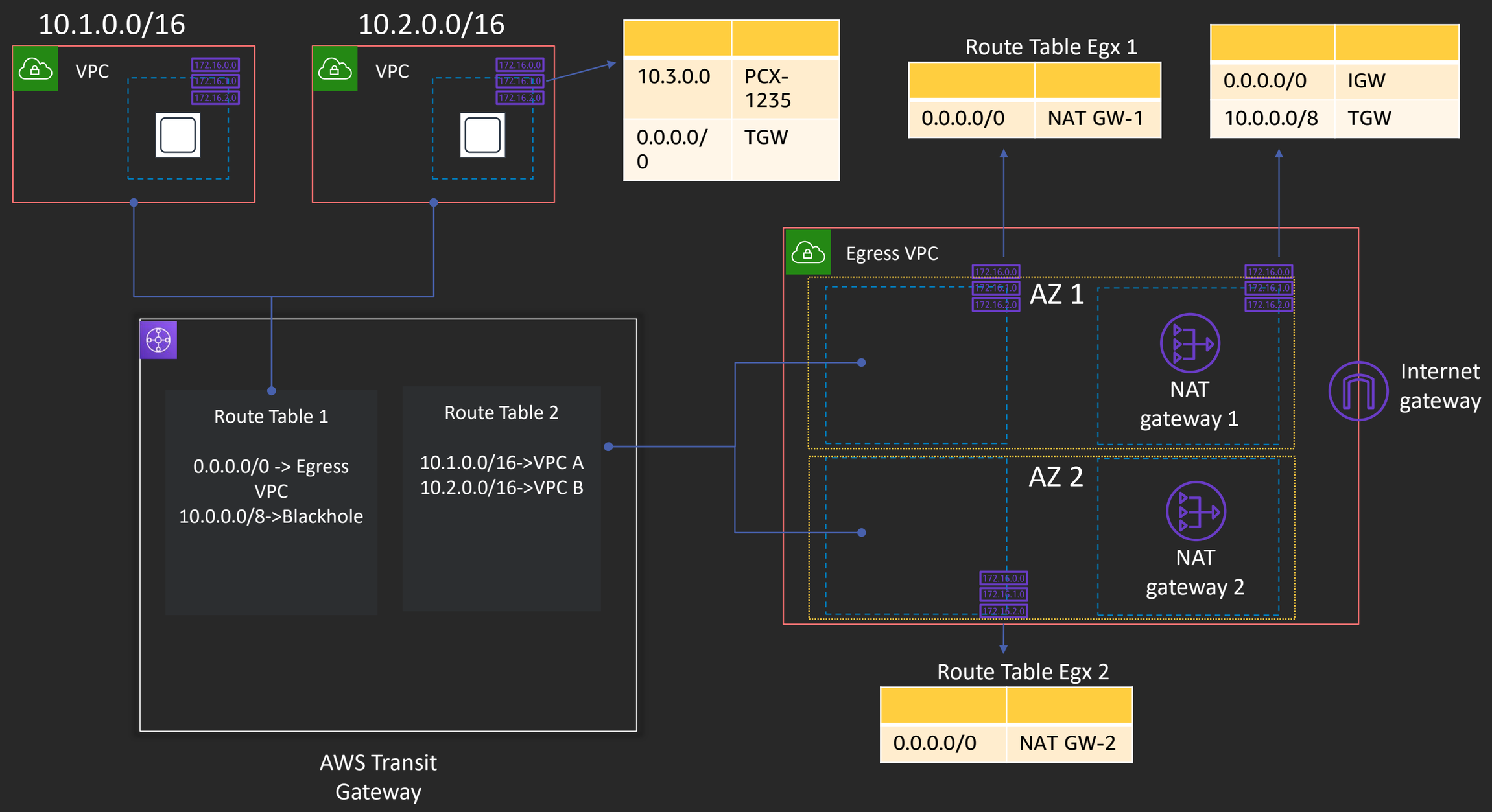
# Hybrid Connectivity options – DX



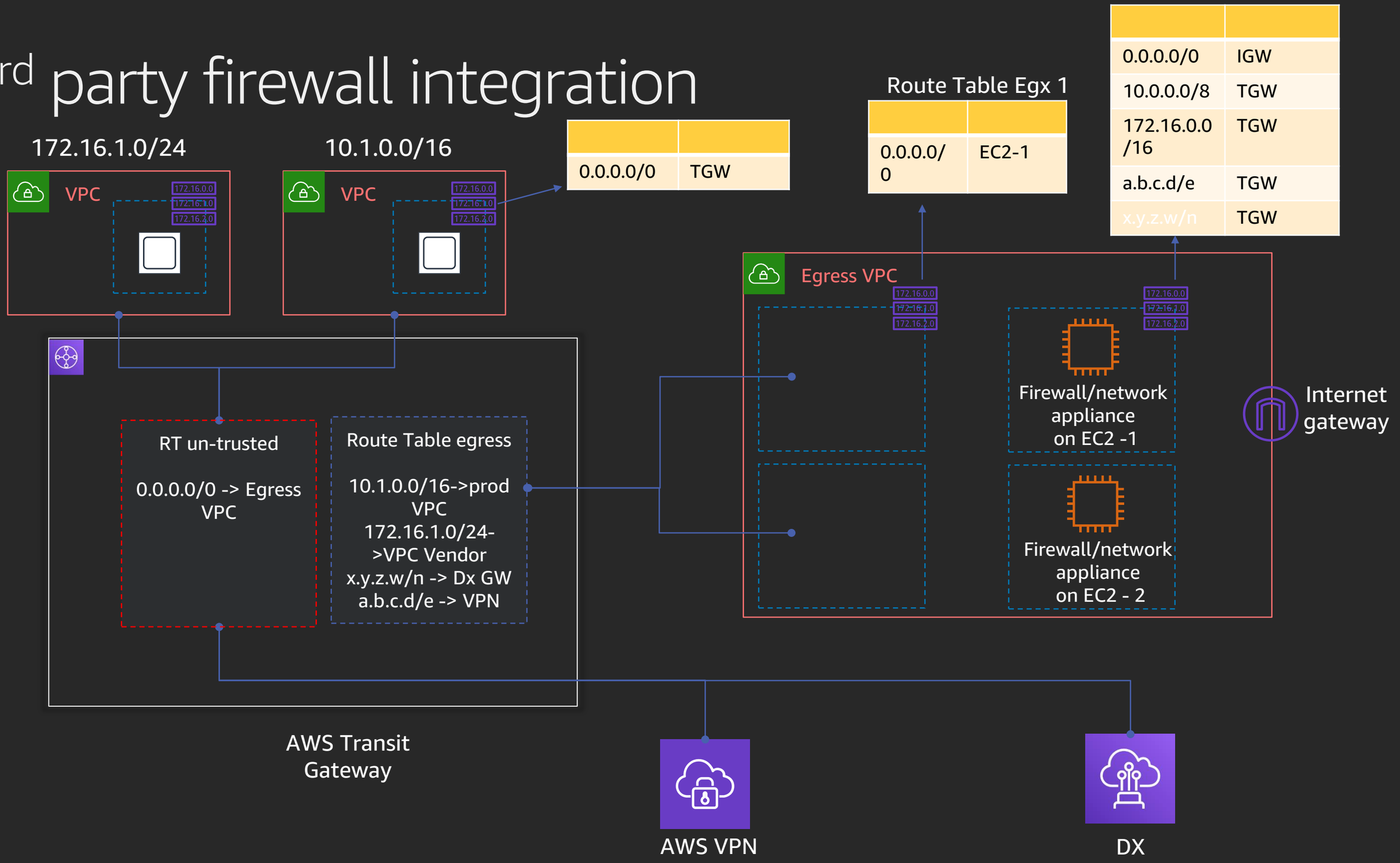



- Private VIF – No ingress charges
- Transit VIF – 3 TGW's (any AWS Region except China)
- Each TGW can advertise max 20 prefixes to DX GW

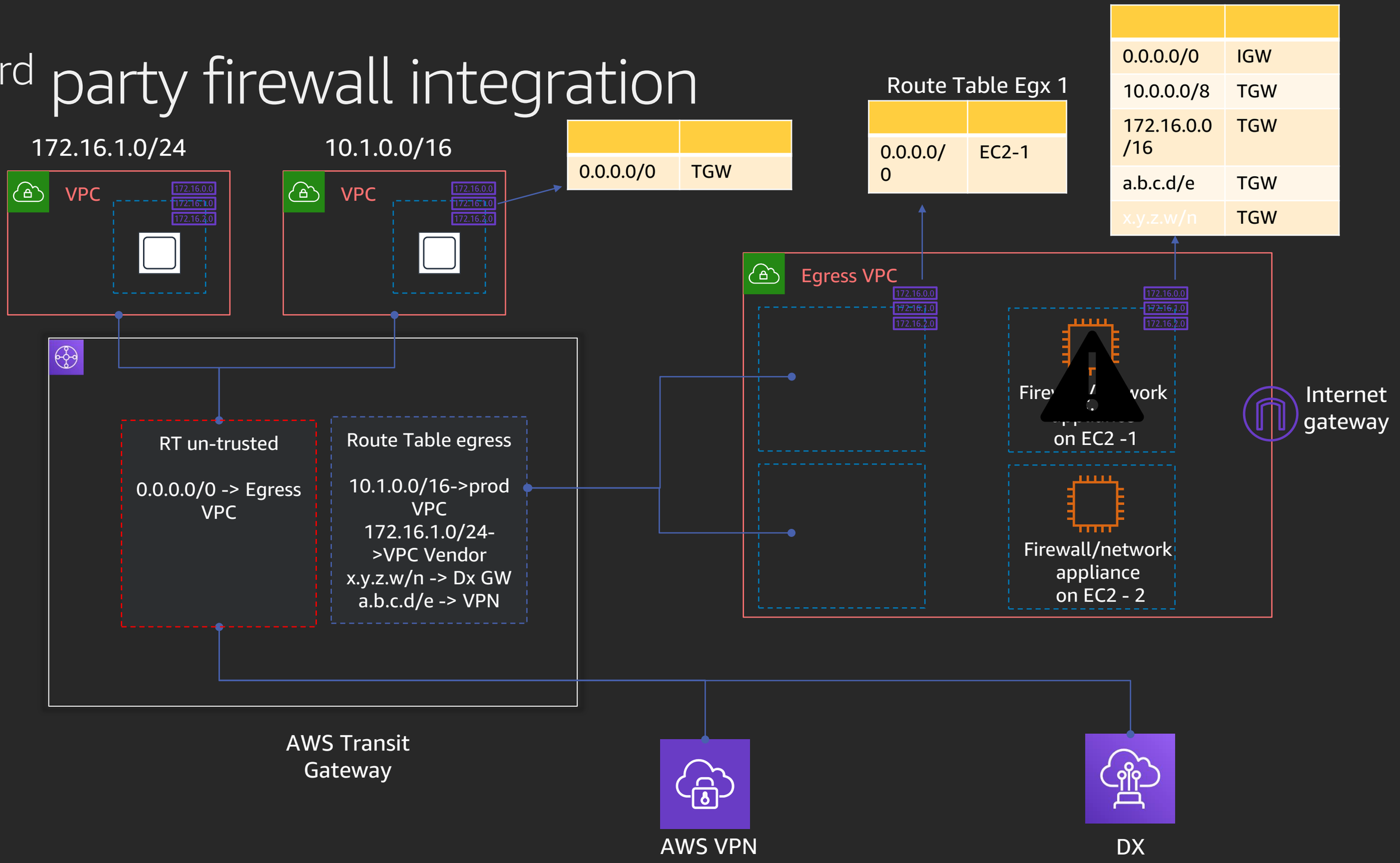

Spoke Account 1
VPC
Spoke Account 3
VPC
VPC Peering
Spoke Account 4
Media VPC
Networking Account
6
4
5
Security VPC
Firewall/network appliance on EC2
DX gateway
VPN connection
AWS Transit Gateway
3
Transit VIF
2
AWS Cage
Internet
Cross connect
1
Customer DC
VPLS (private circuit)
Customer Cage


10.1.0.0/16
VPC
172.16.0.0
172.16.1.0
172.16.2.0
10.2.0.0/16
VPC
172.16.0.0
172.16.1.0
172.16.2.0
10.3.0.0
PCX-1235
0.0.0.0/0
TGW
Route Table Egx 1
0.0.0.0/0
NAT GW-1
0.0.0.0/0
IGW
10.0.0.0/8
TGW
Egress VPC
AZ 1
AZ 2
NAT gateway 1
NAT gateway 2
Internet gateway
Route Table 1
0.0.0.0/0 -> Egress VPC
10.0.0.0/8->Blackhole
Route Table 2
10.1.0.0/16->VPC A
10.2.0.0/16->VPC B
AWS Transit Gateway
Route Table Egx 2
0.0.0.0/0
NAT GW-2

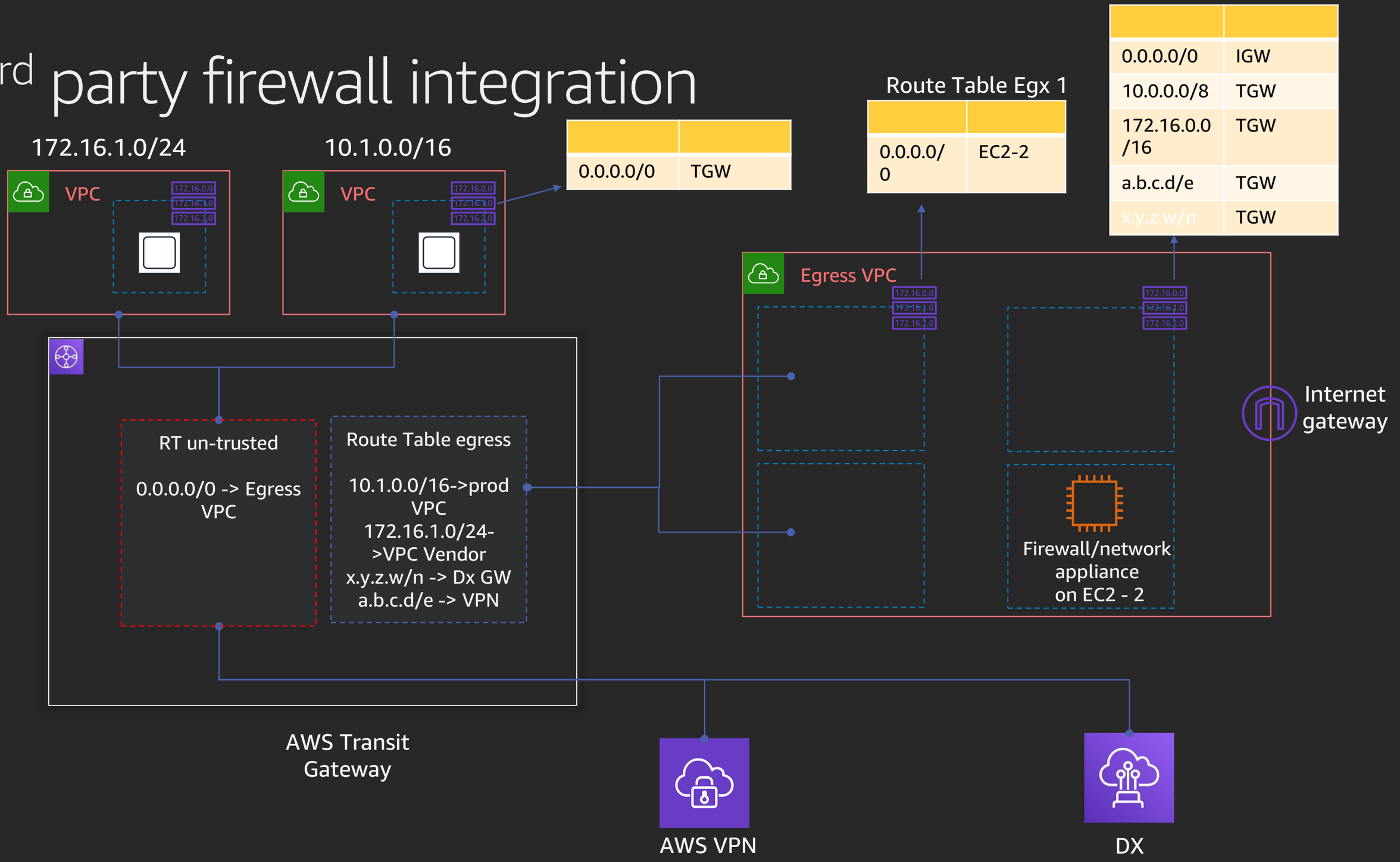
# 3rd party firewall integration



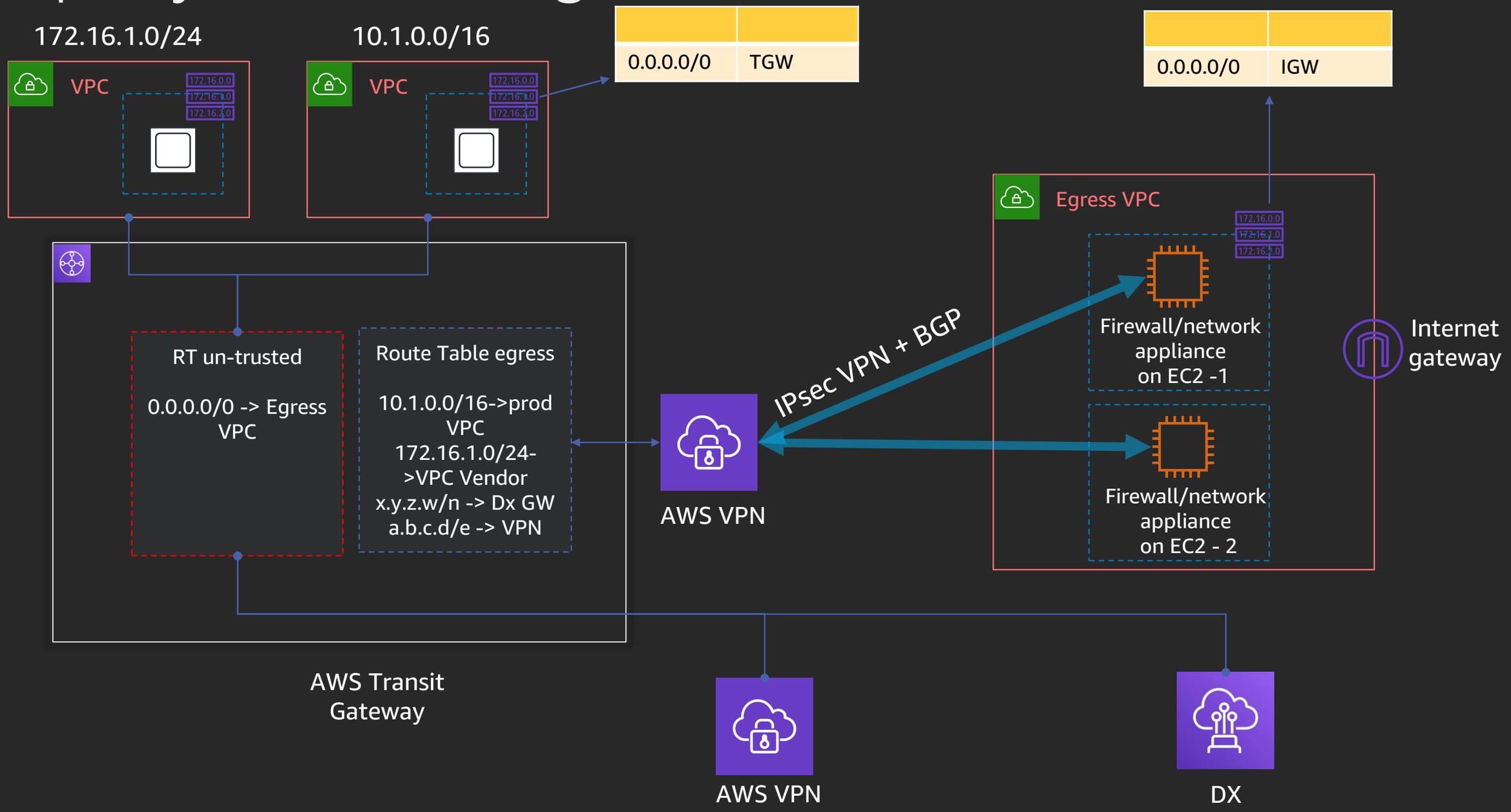
# 3rd party firewall integration



# 3rd party firewall integration



# 3rd party firewall integration



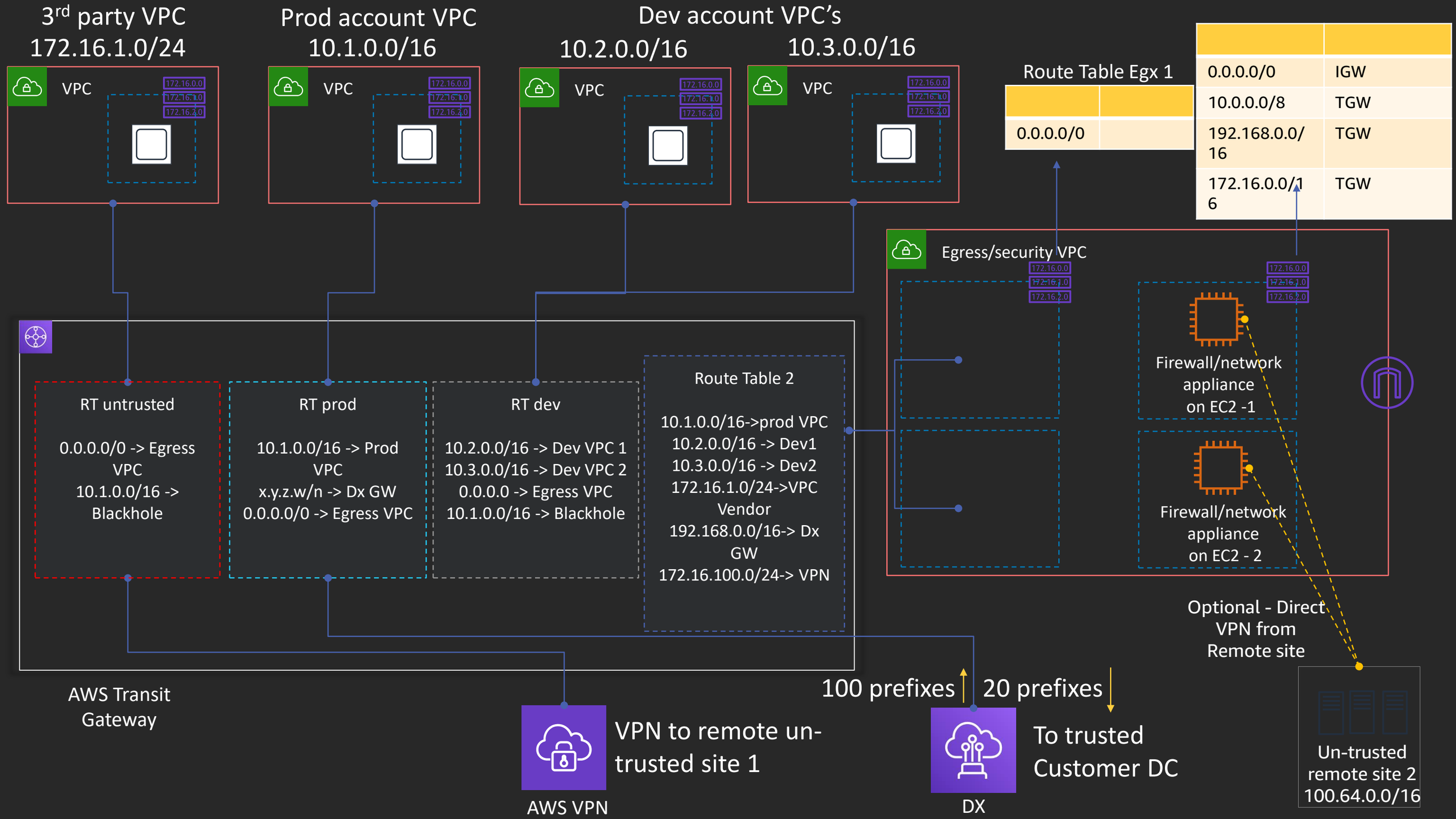
# 3rd party firewall integration

## Native Attachment

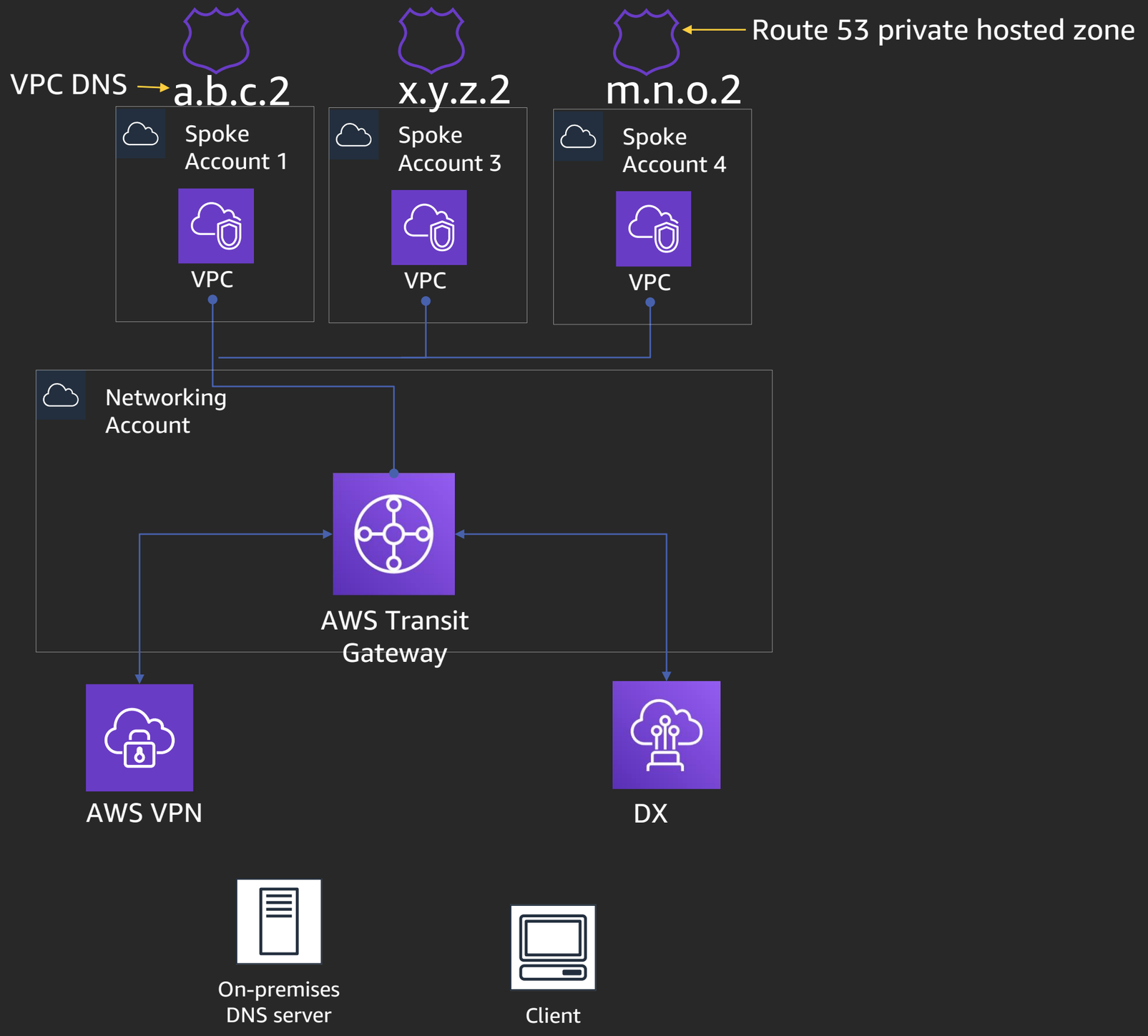
1. Automation required for failure detection and route table change
2. Active/Standby setup
3. Vertical scaling

## VPN attachment

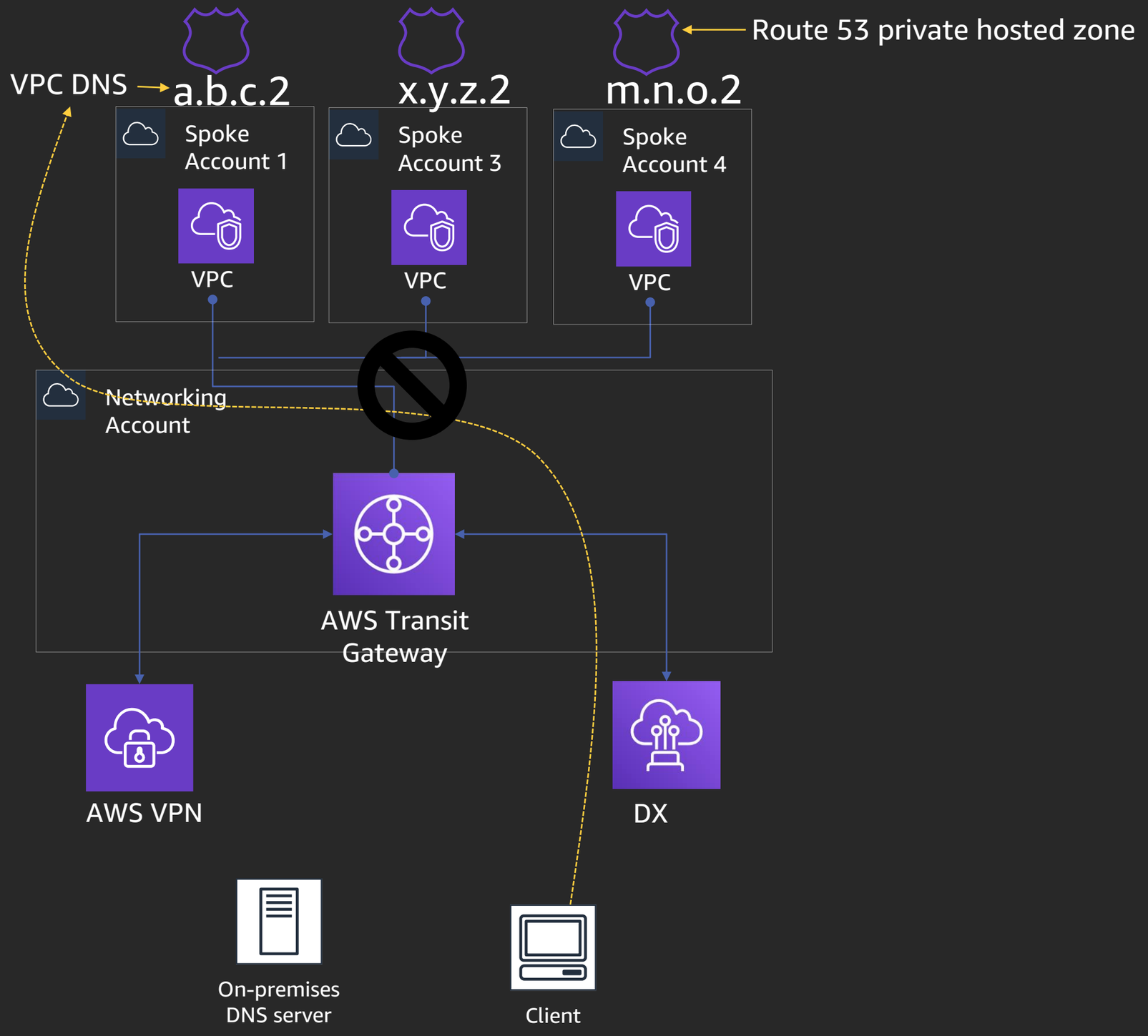
1. Failure and re-convergence taken care by BGP
2. Active/Active setup possible but SNAT required to guarantee same return path
3. Horizontal scaling with (BGP ECMP). SNAT required as above


3rd party VPC
172.16.1.0/24
Prod account VPC
10.1.0.0/16
Dev account VPC's
10.2.0.0/16
10.3.0.0/16
VPC
172.16.0.0
172.16.1.0
172.16.2.0
Route Table Egx 1
0.0.0.0/0
0.0.0.0/0 IGW
10.0.0.0/8 TGW
192.168.0.0/16 TGW
172.16.0.0/16 TGW
Egress/security VPC
Firewall/network appliance on EC2 -1
Firewall/network appliance on EC2 - 2
RT untrusted
0.0.0.0/0 -> Egress VPC
10.1.0.0/16 -> Blackhole
RT prod
10.1.0.0/16 -> Prod VPC
x.y.z.w/n -> Dx GW
0.0.0.0/0 -> Egress VPC
RT dev
10.2.0.0/16 -> Dev VPC 1
10.3.0.0/16 -> Dev VPC 2
0.0.0.0 -> Egress VPC
10.1.0.0/16 -> Blackhole
Route Table 2
10.1.0.0/16->prod VPC
10.2.0.0/16 -> Dev1
10.3.0.0/16 -> Dev2
172.16.1.0/24->VPC Vendor
192.168.0.0/16-> Dx GW
172.16.100.0/24-> VPN
Optional - Direct VPN from Remote site
AWS Transit Gateway
100 prefixes
20 prefixes
VPN to remote un-trusted site 1
AWS VPN
To trusted Customer DC
DX
Un-trusted remote site 2
100.64.0.0/16

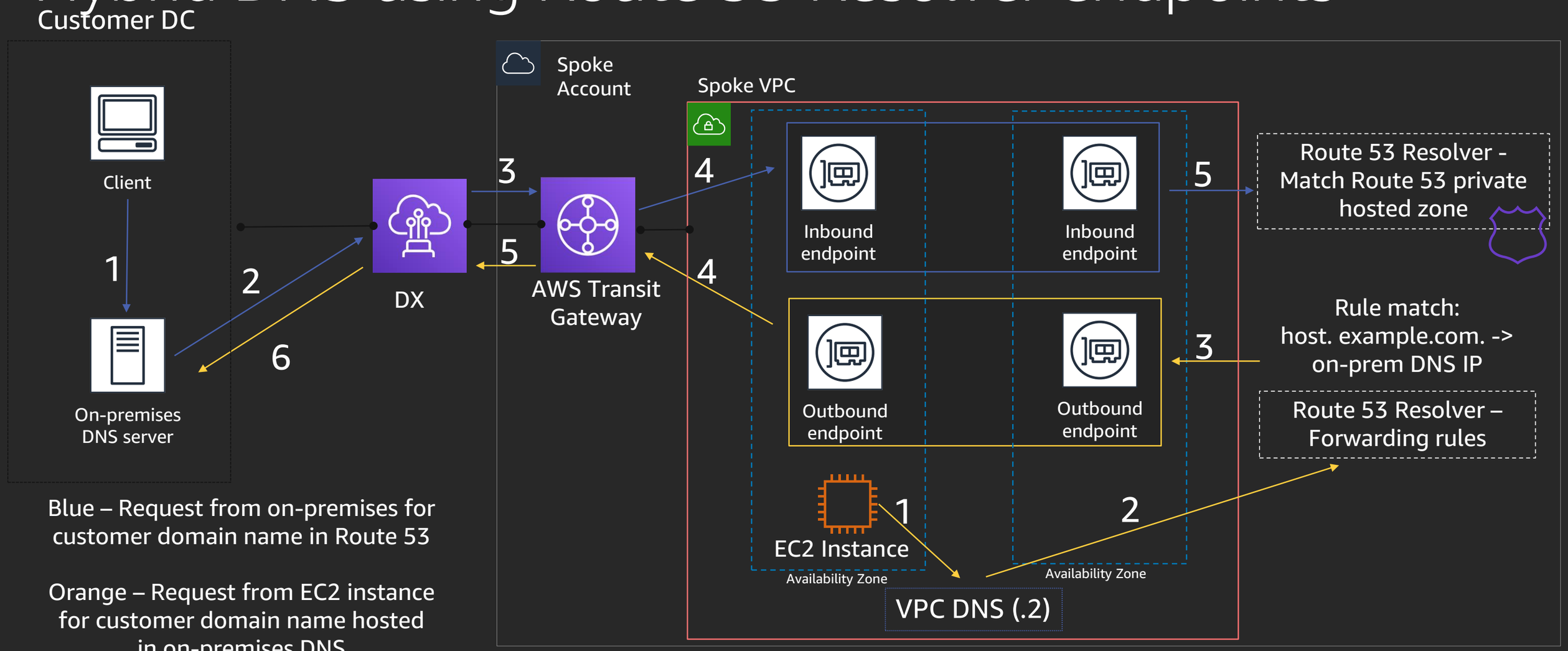
# DNS


Route 53 private hosted zone
VPC DNS
a.b.c.2
x.y.z.2
m.n.o.2
Spoke Account 1
VPC
Spoke Account 3
VPC
Spoke Account 4
VPC
Networking Account
AWS Transit Gateway
AWS VPN
DX
On-premises DNS server
Client

# DNS


Route 53 private hosted zone
VPC DNS
a.b.c.2
x.y.z.2
m.n.o.2
Spoke Account 1
VPC
Spoke Account 3
VPC
Spoke Account 4
VPC
Networking Account
AWS Transit Gateway
AWS VPN
DX
On-premises DNS server
Client

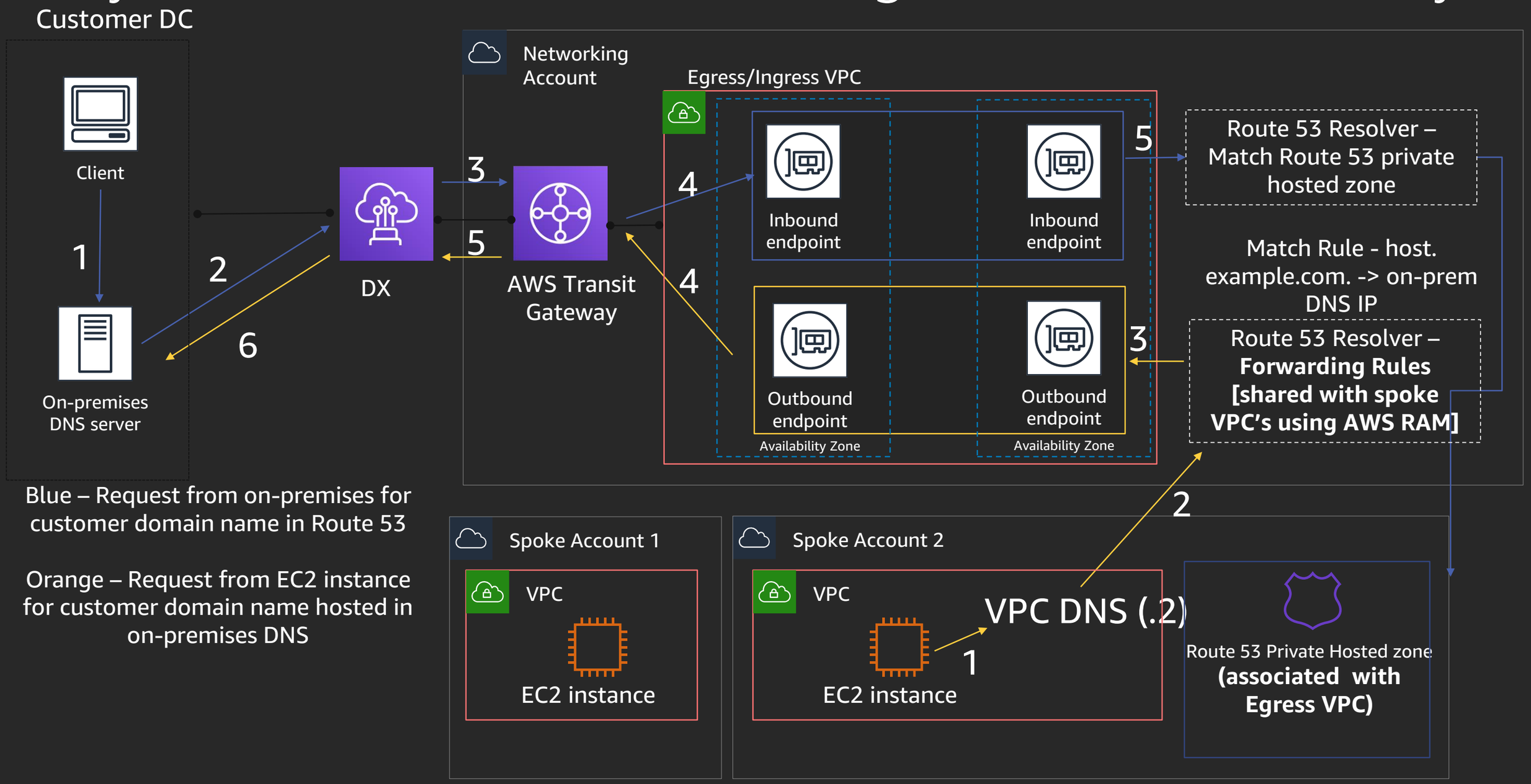
# Hybrid DNS using Route 53 Resolver endpoints




Blue – Request from on-premises for customer domain name in Route 53

Orange – Request from EC2 instance for customer domain name hosted in on-premises DNS

# Hybrid DNS—centralization using AWS Transit Gateway




Blue – Request from on-premises for customer domain name in Route 53

Orange – Request from EC2 instance for customer domain name hosted in on-premises DNS

# Hybrid DNS

**Amazon Route 53**
Developer Guide (API Version 2013-04-01)

Documentation - This Guide
Search

- What Is Amazon Route 53?
- Setting Up
- Getting Started

## Sharing Forwarding Rules with Other AWS Accounts and Using Shared Rules

You can share the forwarding rules that you created using one AWS account with other AWS accounts. To sha
Resolver console integrates with AWS Resource Access Manager. For more information about Resource Access
Resource Access Manager User Guide.

Note the following:

**Associating shared rules with VPCs**

**https://tinyurl.com/y6kabnjx**

**Amazon Route 53**
Developer Guide (API Version 2013-04-01)

Documentation - This Guide
Search

- What Is Amazon Route 53?
- Setting Up
- Getting Started
- Integration with Other Services

AWS Documentation » Amazon Route 53 » Developer Guide » Configuring Amazon Route 53 as Your DNS Service » Working with Hosted Zones » Working with Private Hosted Zones » Associating an Amazon VPC and a Private Hosted Zone That You Created with Different AWS Accounts

## Associating an Amazon VPC and a Private Hosted Zone That You Created with Different AWS Accounts

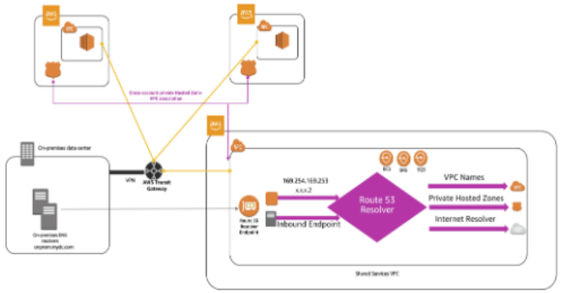
If you want to associate a VPC that you created with one AWS account with a private hosted zone that you created with a different account, perform the following procedure:

**To associate an Amazon VPC and a private hosted zone that you created with different AWS accounts**

1. Using the account that created the hosted zone, authorize the association of the VPC with the private hosted zone by using one of the following methods:

**https://tinyurl.com/yxgty6v6**

# Hybrid DNS




Centralized DNS management of hybrid cloud with Amazon Route 53 and AWS Transit Gateway

by Bhavin Desai | on 03 MAY 2019 | in Amazon Route 53, AWS Transit Gateway, Networking & Content Delivery | Permalink | Share

A successful hybrid networking strategy goes beyond private network connectivity. It often requires dealing with independent internal zones both in Amazon Virtual Private Cloud (Amazon VPC) and on-premises. Such a strategy needs Domain Name System (DNS) naming that spans the entire network. Typically, this is managed by providing name resolution services in the same place [...]

Read More

**https://tinyurl.com/y43u9o8b**

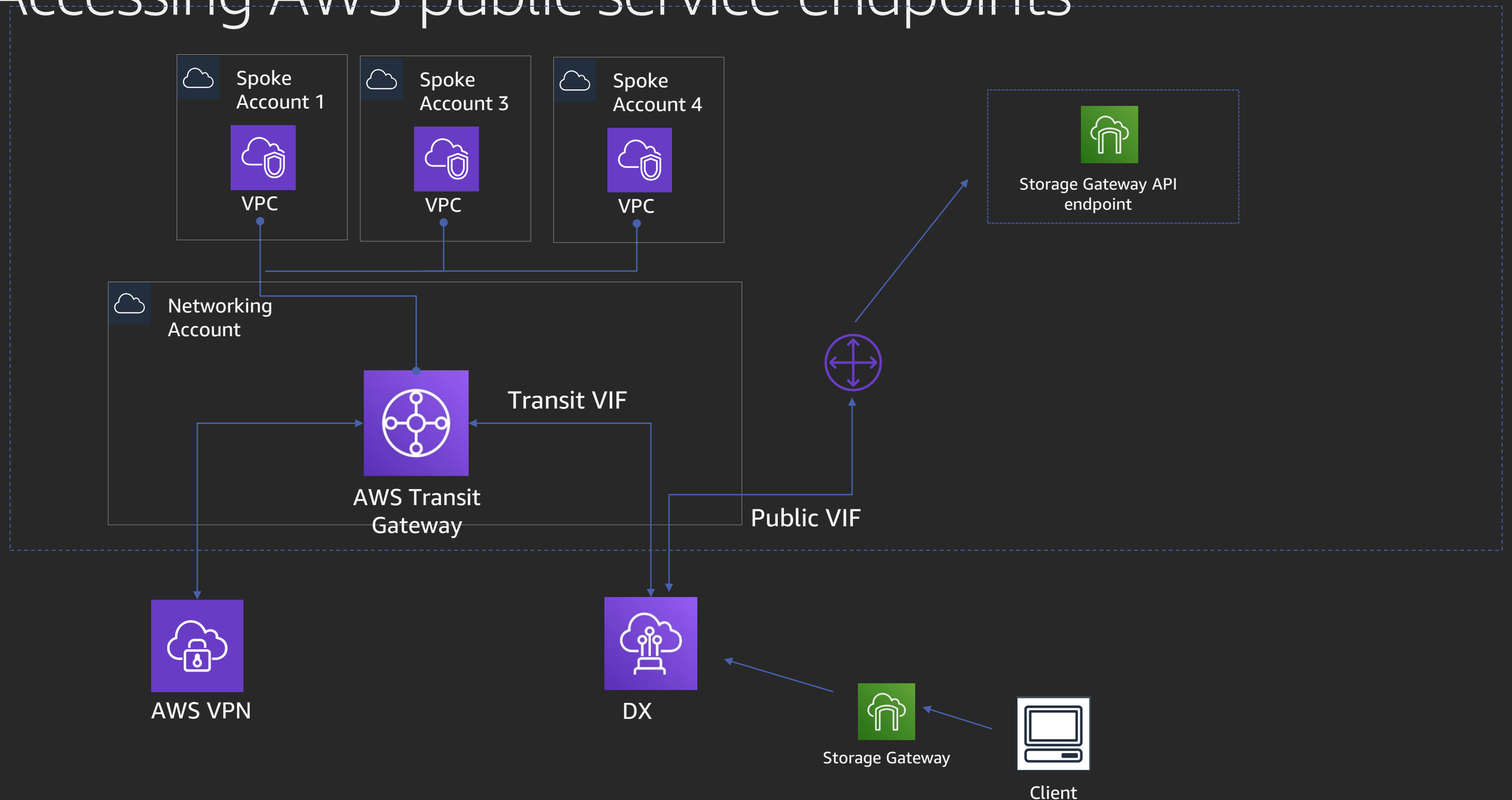
AWS re:Invent 2018: Introduction to Amazon Route 53 Resolver for Hybrid Cloud (NET215)

Amazon Web Services • 7.9K views • 8 months ago

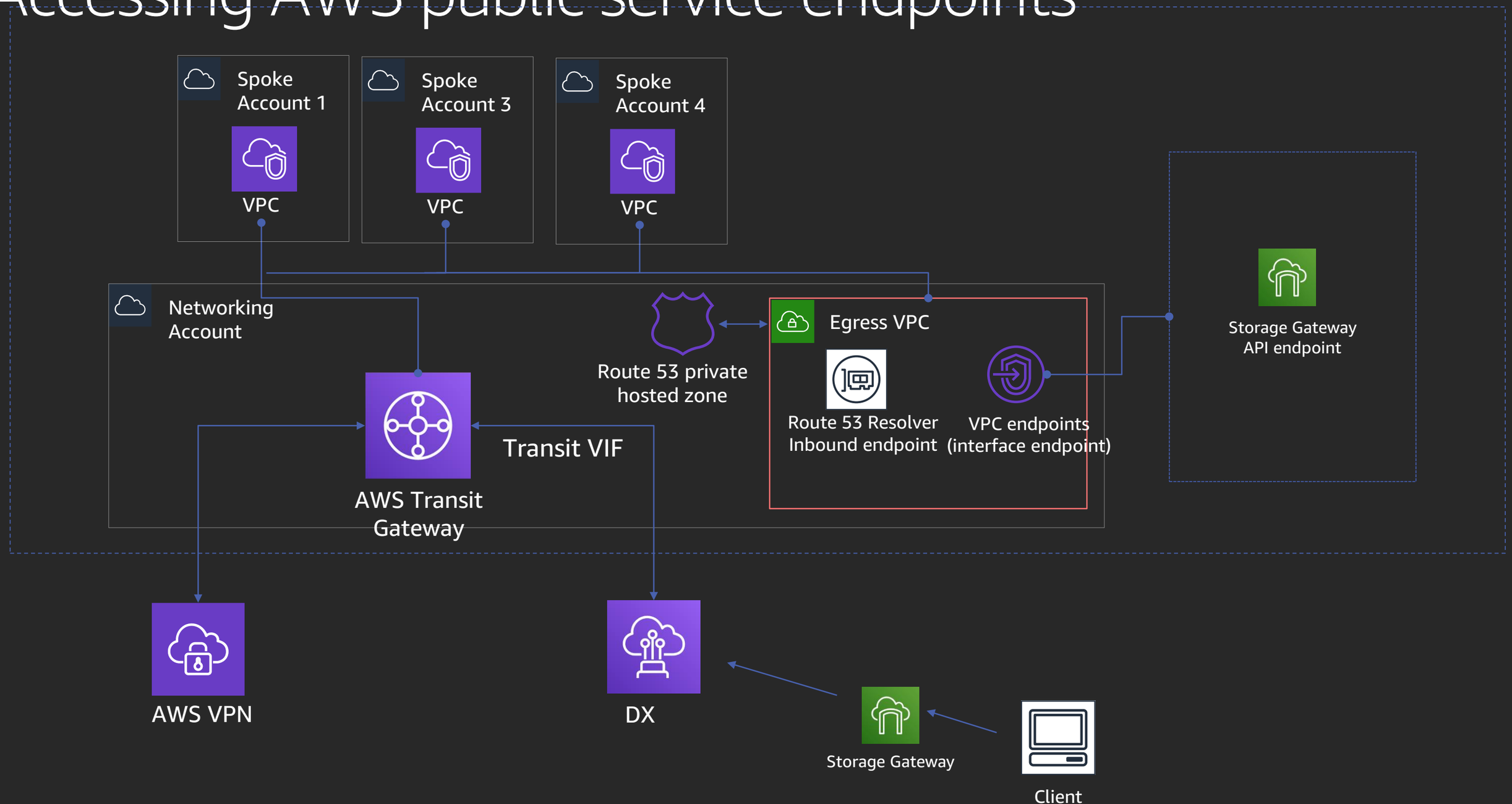
Amazon **Route 53 Resolver** provides recursive DNS for your Amazon VPC and on-premises networks over VPN or AWS Direct ...

**https://tinyurl.com/y6f4h7zm**

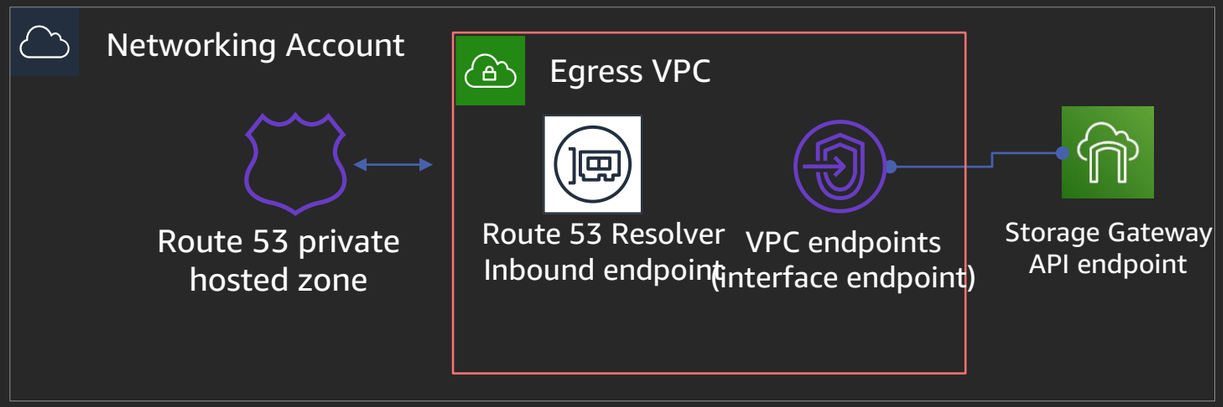
# Accessing AWS public service endpoints

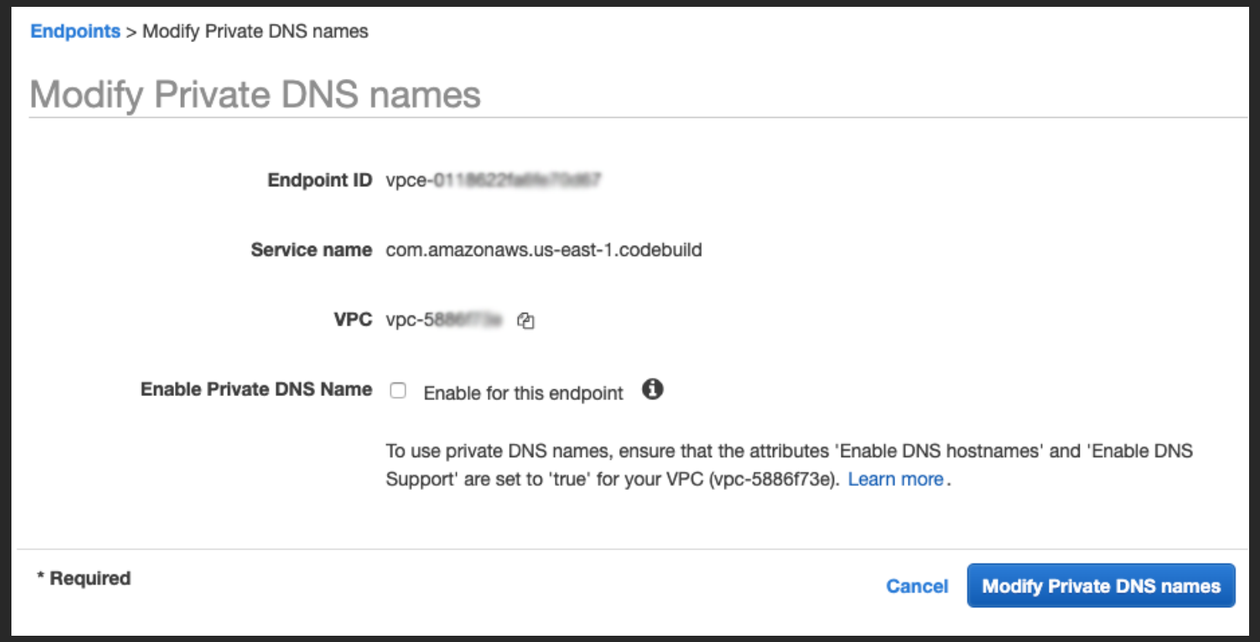


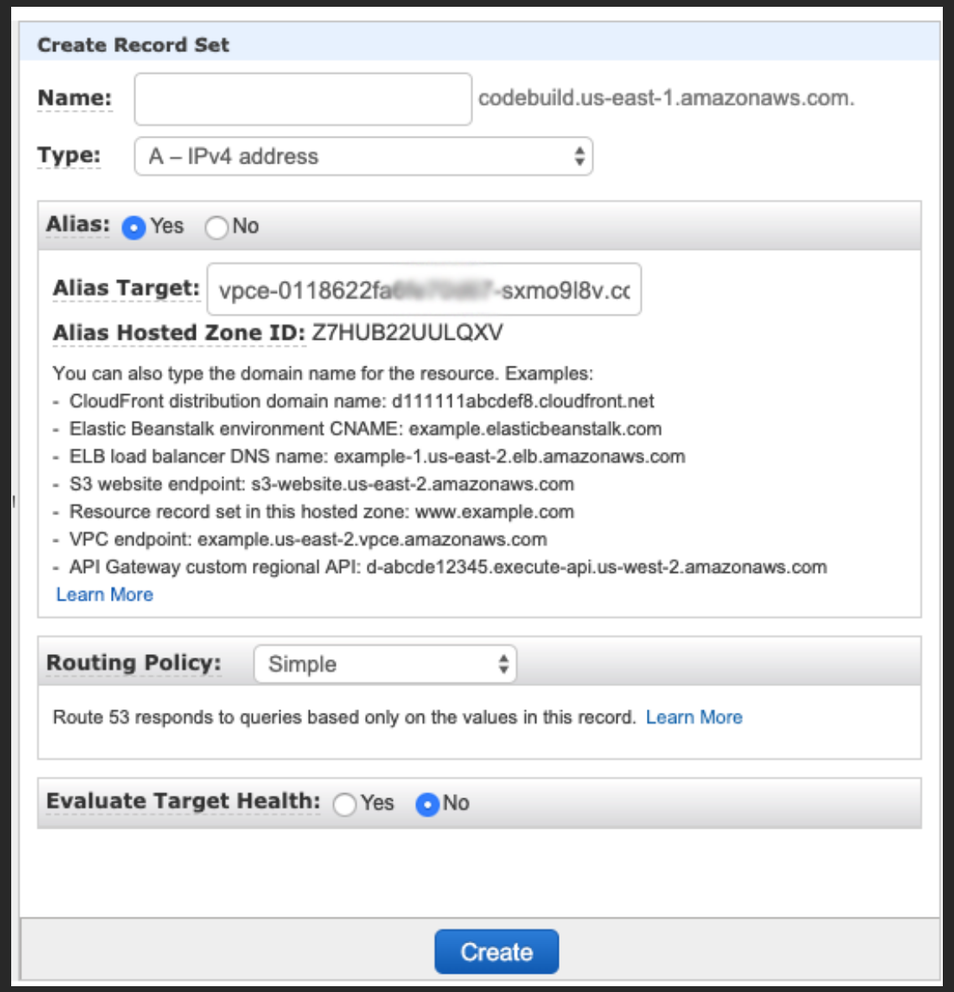
# Accessing AWS public service endpoints



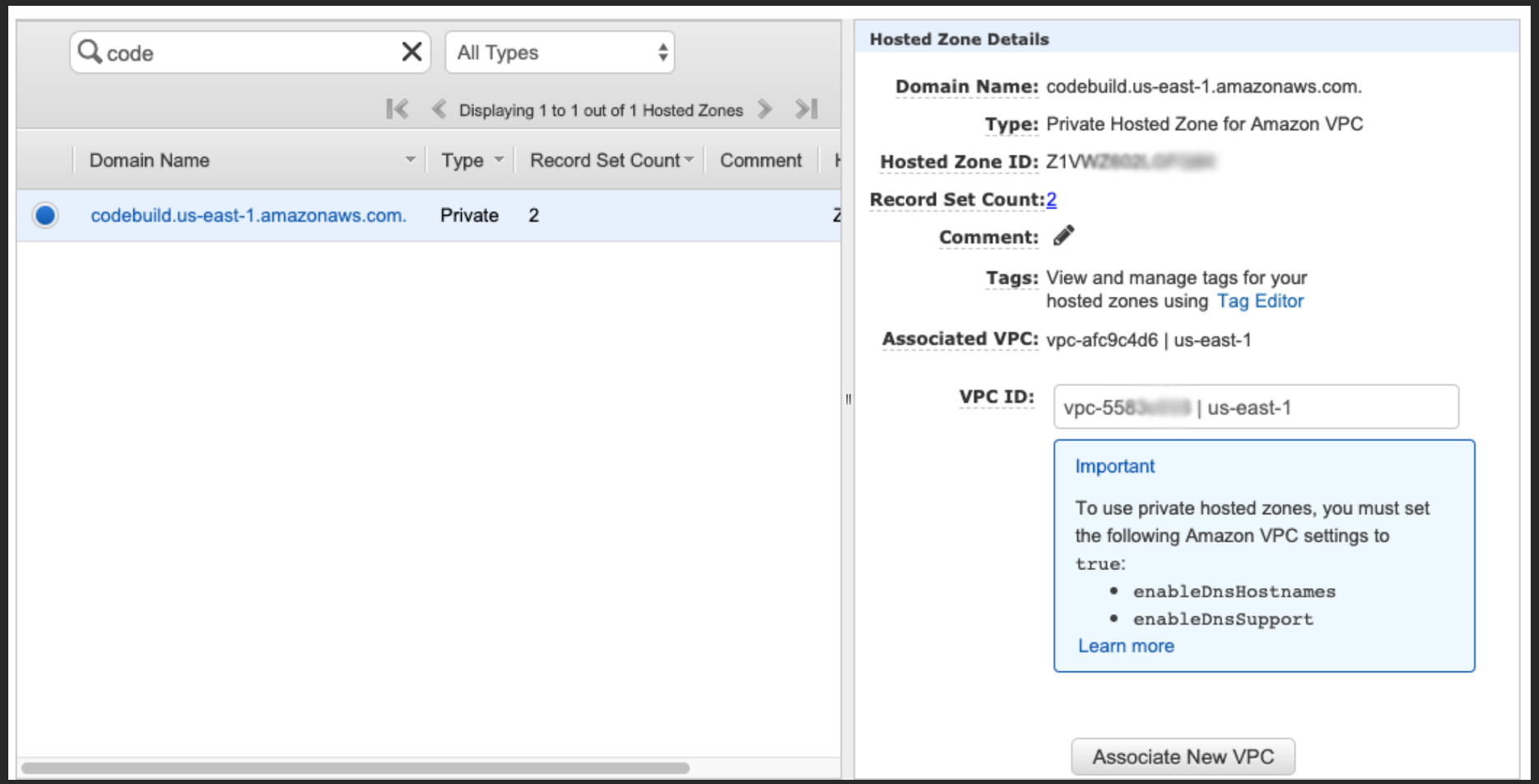
# Centralizing access to AWS public service endpoints



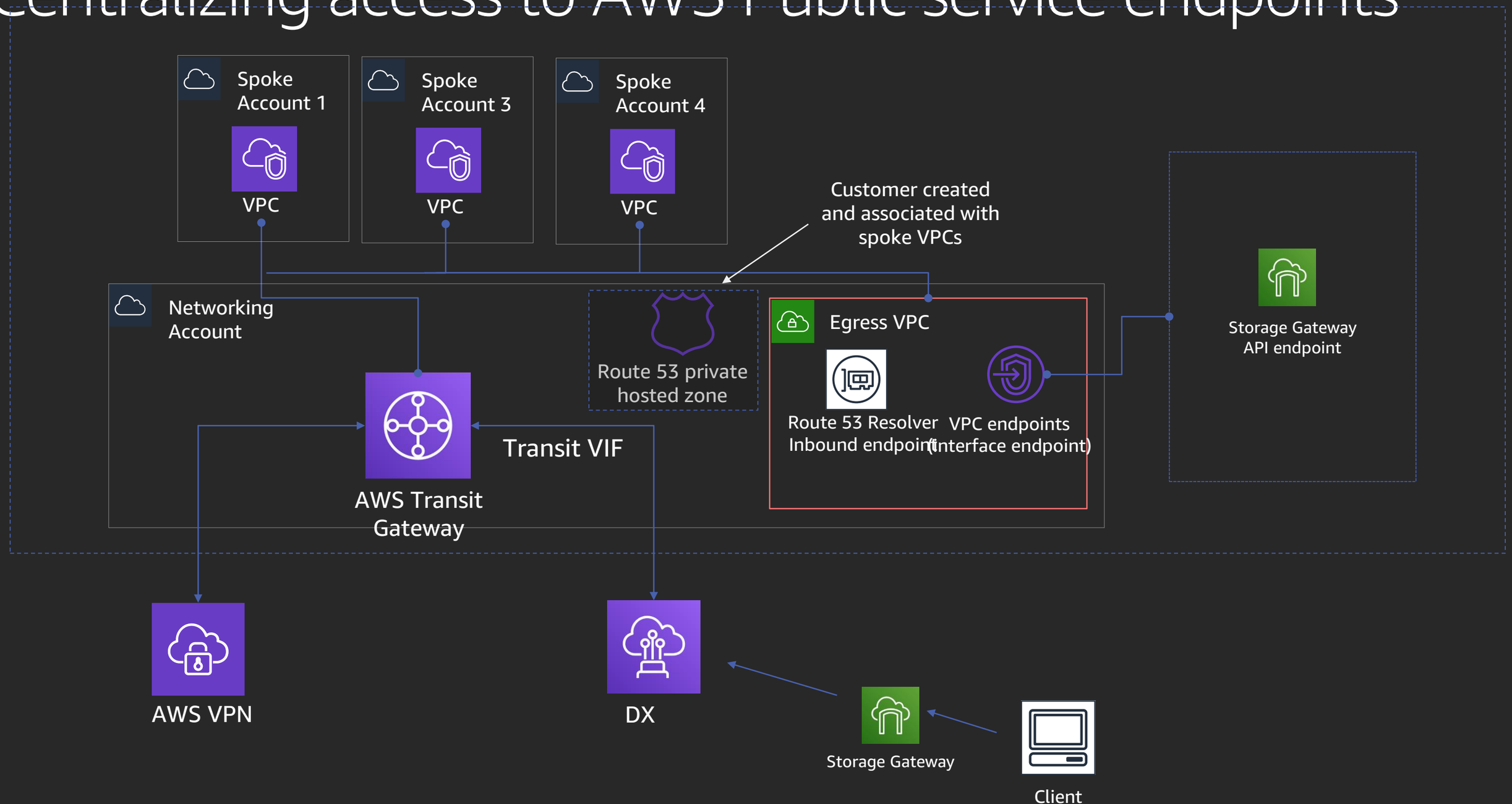



# Centralizing access to AWS Public service endpoints



# Centralizing access to AWS Public service endpoints



# Centralizing access to AWS Public service endpoints

Networking & Content Delivery

## Integrating AWS Transit Gateway with AWS PrivateLink and Amazon Route 53 Resolver

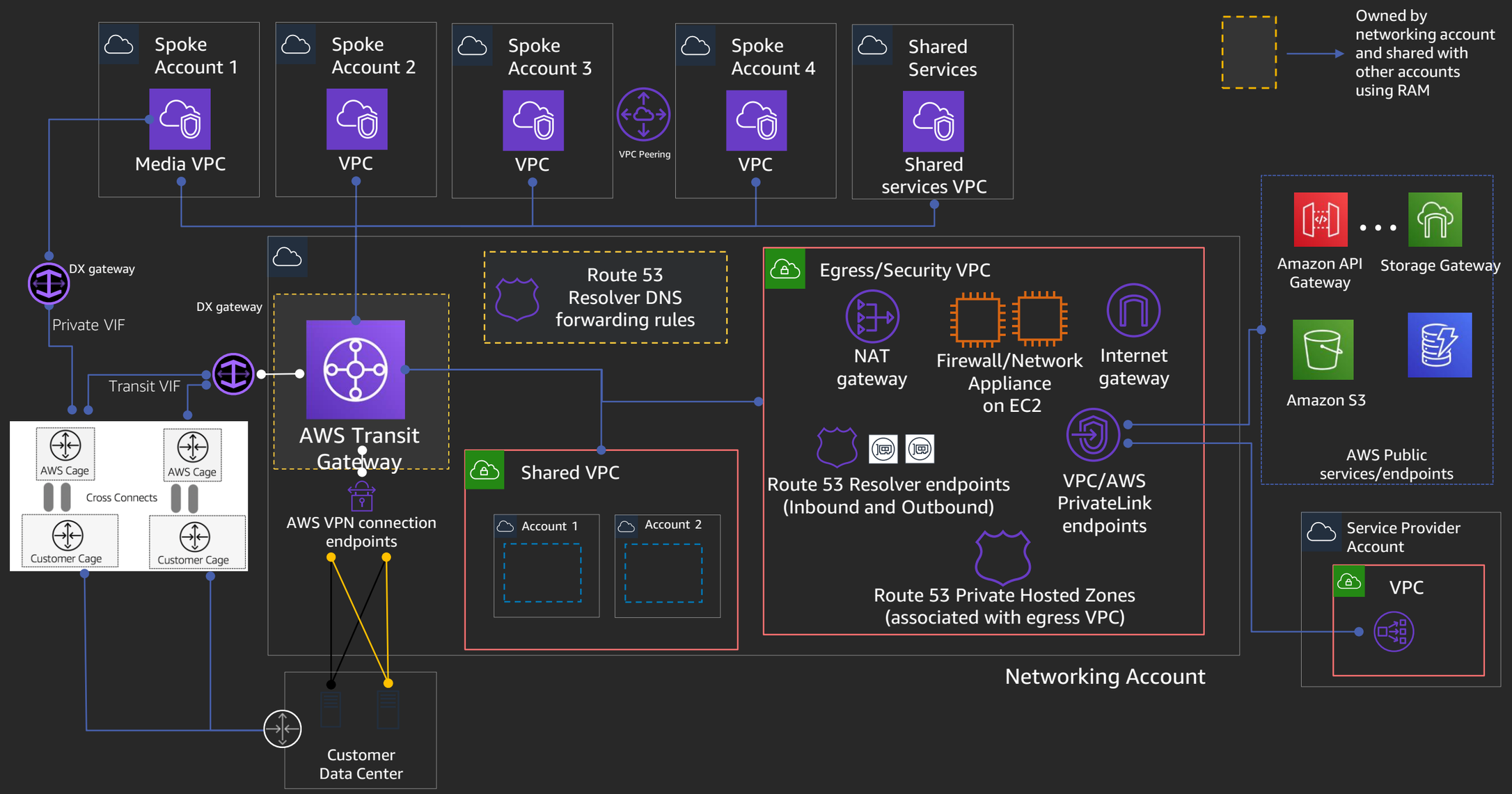
by James Devine | on 24 JUL 2019 | in *Learning Levels, Advanced (300), Amazon Route 53, AWS PrivateLink, AWS Transit Gateway, Expert (400), Networking & Content Delivery | Permalink | Share

I want to take some time to dive more deeply into a use case outlined in NET301 Best Practices for AWS PrivateLink. The use case involves using AWS Transit Gateway, along with Amazon Route 53 Resolver, to share AWS PrivateLink interface endpoints between multiple connected Amazon virtual private clouds (VPCs) and an on-premises environment. We've seen a lot of interest from customers in this architecture. It can greatly reduce the number of VPC endpoints, simplify VPC endpoint deployment, and help cost-optimize when deploying at scale.

### Architecture overview

For VPC endpoints that you use to connect to endpoint services (services you create using AWS PrivateLink behind a Network Load Balancer) the architecture is fairly straightforward. Since the DNS entries for the VPC endpoint are public, you just need layer-three connectivity between a VPC and its destination using either VPC peering, transit gateway, or a VPN. Where the architecture becomes more complex is when you want to share endpoints for AWS services and AWS PrivateLink SaaS.

https://tinyurl.com/yxs69pwl


Spoke Account 1
Media VPC
Spoke Account 2
VPC
Spoke Account 3
VPC
VPC Peering
Spoke Account 4
VPC
Shared Services
Shared services VPC
Owned by networking account and shared with other accounts using RAM
DX gateway
Private VIF
DX gateway
Transit VIF
AWS Cage
AWS Cage
Cross Connects
Customer Cage
Customer Cage
AWS Transit Gateway
AWS VPN connection endpoints
Route 53 Resolver DNS forwarding rules
Shared VPC
Account 1
Account 2
Egress/Security VPC
NAT gateway
Firewall/Network Appliance on EC2
Internet gateway
Route 53 Resolver endpoints (Inbound and Outbound)
VPC/AWS PrivateLink endpoints
Route 53 Private Hosted Zones (associated with egress VPC)
Amazon API Gateway
Storage Gateway
Amazon S3
AWS Public services/endpoints
Service Provider Account
VPC
Networking Account
Customer Data Center

# AWS CloudFormation resources

aws | Services | Resource Groups | S3 | EC2 | VPC | Lambda | Nick/nsci-Isengard @ nsciarrilli | VA us-east-1 | Support

CloudFormation > Stacks > NET335

**Stacks (26)**

Filter by stack name

Active

View nested

1

NET335
2019-10-20 18:05:20 UTC-0700
CREATE_COMPLETE

NESTED
testcat2-CodePipelineStack-1KLOPZLJ8QC26
2019-10-19 18:07:29 UTC-0700
CREATE_COMPLETE

NESTED
testcat2-SSMStack-1273WNLG3SL2G
2019-10-19 18:05:32 UTC-0700
CREATE_COMPLETE

NESTED

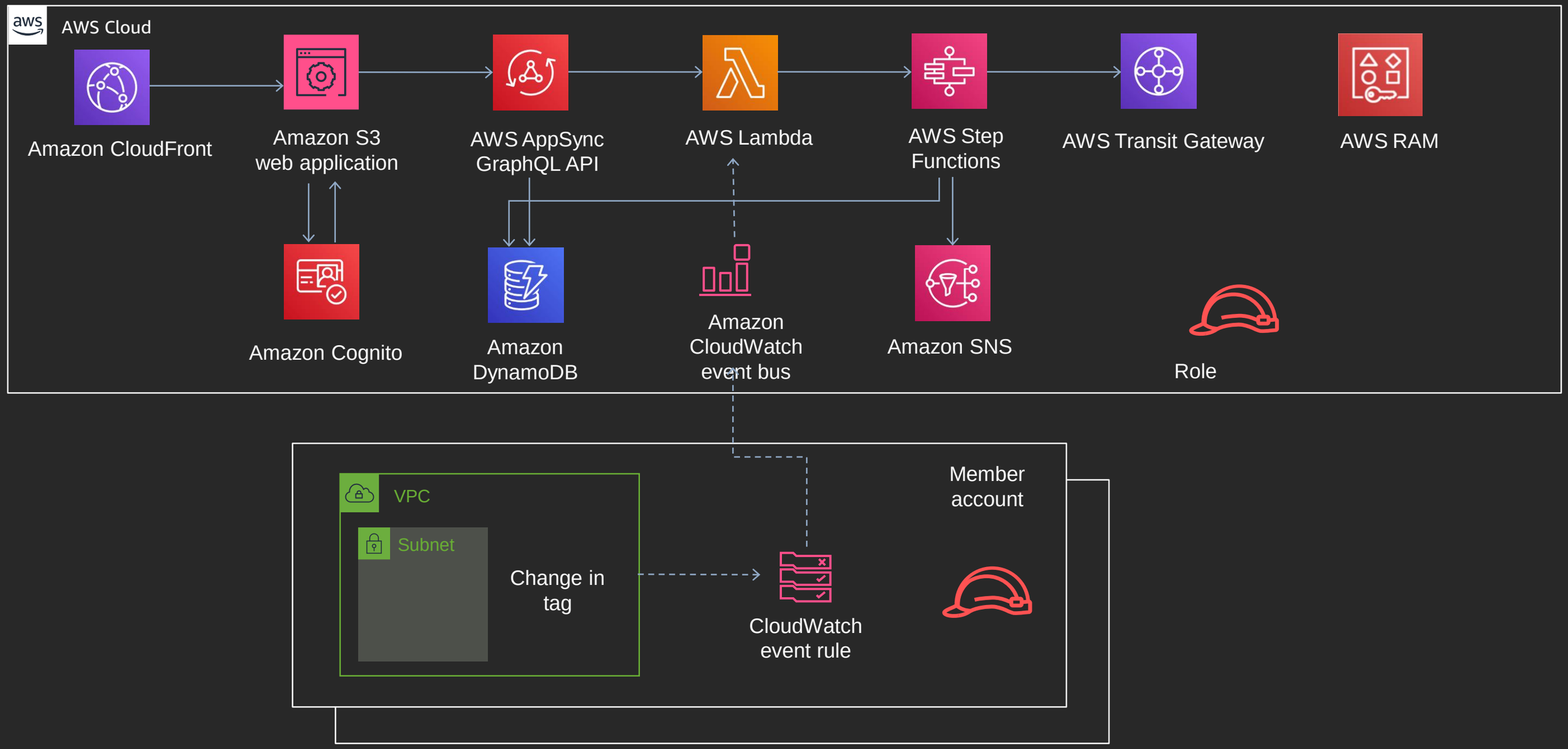
## NET335

Delete | Update | Stack actions | Create stack

Stack info | Events | Resources | Outputs | Parameters | Template | Change sets

**Resources (4)**

Search resources

| Logical ID | Physical ID | Type | Status |
|---|---|---|---|
| myATTACHMENT | tgw-attach-08013276791dd7d96 | AWS::EC2::TransitGatewayAttachment | CREATE_COMPLETE |
| mySUBNET | subnet-0738ee6bdcbce25ac | AWS::EC2::Subnet | CREATE_COMPLETE |
| myTGW | tgw-01d926f6cdd35c78f | AWS::EC2::TransitGateway | CREATE_COMPLETE |
| myVPC | vpc-0e127fc20d12c5b54 | AWS::EC2::VPC | CREATE_COMPLETE |

# Serverless Transit Network Orchestrator