Protect your network from DNS exfiltration attacks

Edge Modernization | 09/30/2021

Vadim Omeltchenko
Sr. Solution Architect
Amazon Web Services

Vishal Lakhotia
Solution Architect
Amazon Web Services

Agenda

- Role of Amazon Route53 in AWS Edge services
- What is DNS data exfiltration
- Outbound Network Traffic inspection
- DNS Traffic Inspection
- Amazon Route 53 Resolver DNS Firewall Deployment patterns
- Deployment Steps

The role Route53 plays in AWS Edge services

Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service. It is designed to give developers and businesses an extremely reliable and cost effective way to route end users to Internet applications

- Route 53 resolver
- Traffic flow rules
- DNSSEC
- Load balancer integrations
- Application recover functions
- Geo DNS
- Integrated Route53 Resolver DNS firewall

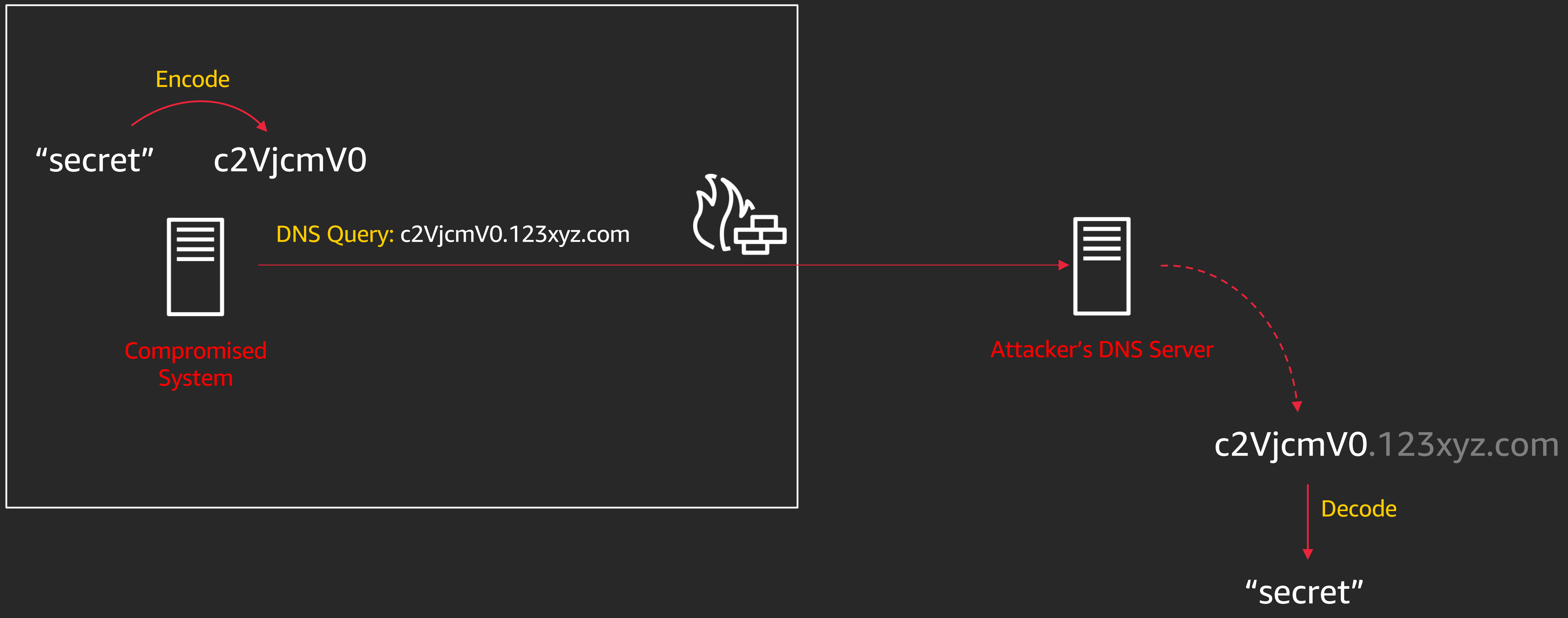
What is DNS data exfiltration?

What is DNS data exfiltration?

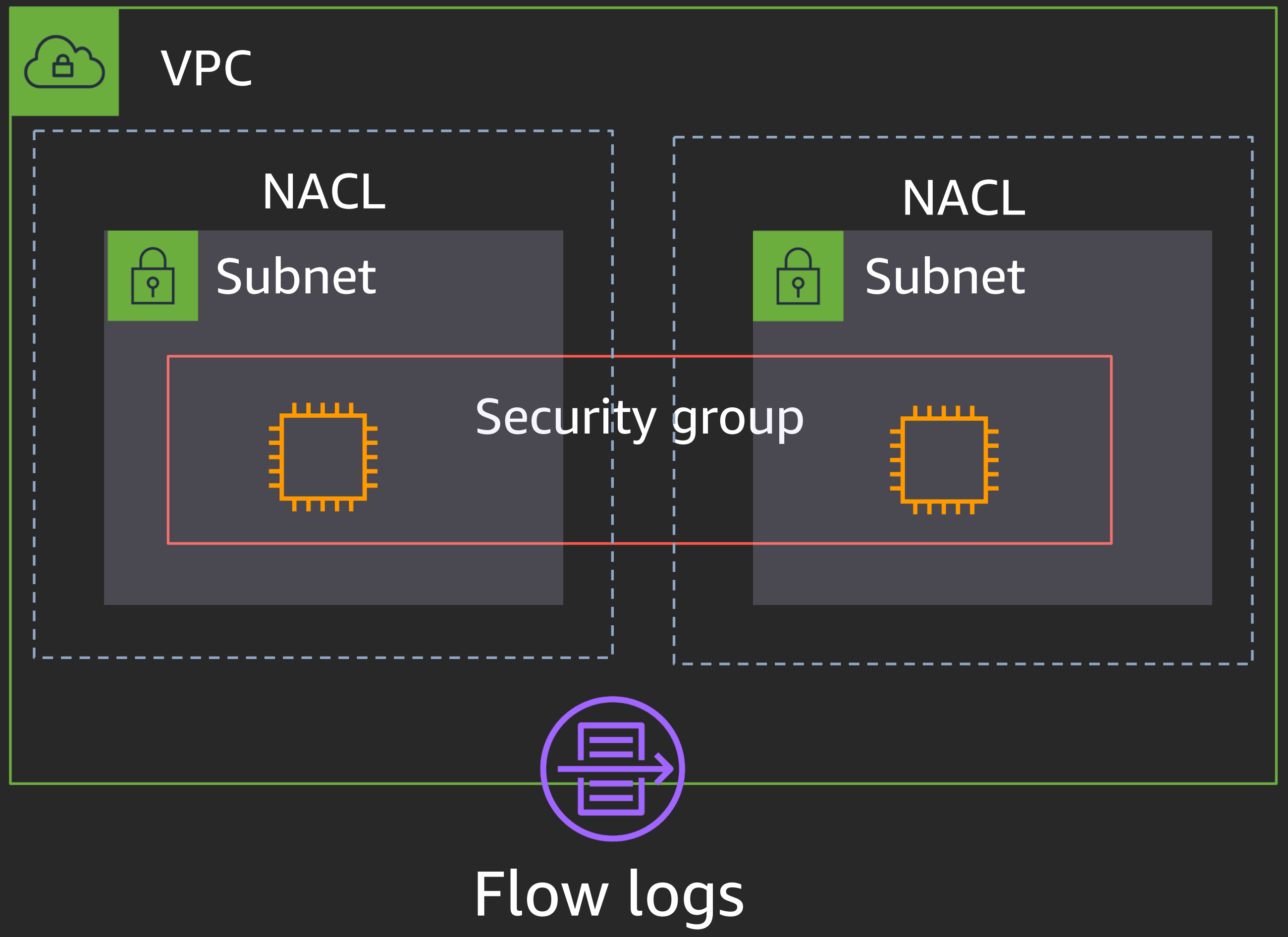
Unauthorized transfer of data from a compromised system to a remote host over DNS protocol.

- Target system is compromised
- Sensitive data is moved out of the environment
- Data transfer takes place over DNS
- Custom/exploited DNS server on the receiving end
- Can be prevented with Firewalls, IDS/IPS

How DNS data exfiltration works?



VPC Security Options



DNS firewall

Traffic mirroring

AWS Shield

Amazon GuardDuty

AWS WAF

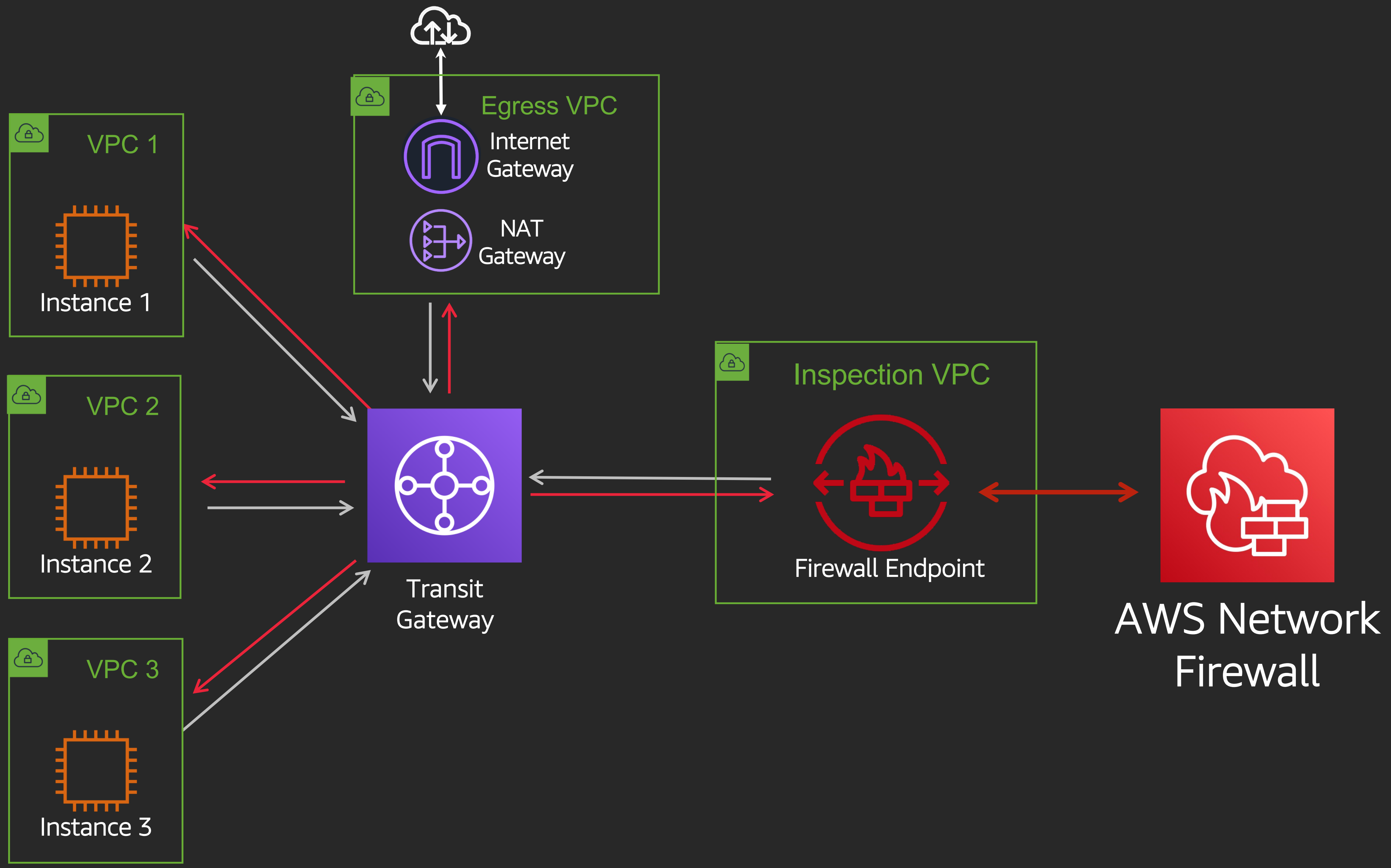
AWS Network Firewall

Gateway Load Balancer + 3rd party appliances

Outbound Network Inspection

With AWS Network Firewall

Centralized Security Inspection



Leveraging threat intelligence feeds

With AWS GuardDuty

GuardDuty Findings

Amazon GuardDuty identifies threats by continuously monitoring the network activity, data access patterns, and account behavior within the AWS environment. It comes integrated with up-to-date threat intelligence feeds from AWS, CrowdStrike, and Proofpoint.

Examples of GuardDuty DNS related findings

`Backdoor:EC2/C&CActivity.B!DNS`

`CryptoCurrency:EC2/BitcoinTool.B!DNS`

`Trojan:EC2/BlackholeTraffic!DNS`

`Trojan:EC2/DGADomainRequest.C!DNS`

`Trojan:EC2/DNSDataExfiltration`

`Trojan:EC2/DriveBySourceTraffic!DNS`

`Trojan:EC2/DropPoint!DNS`

`Trojan:EC2/PhishingDomainRequest!DNS`

`UnauthorizedAccess:EC2/MetadataDNSRebind`

DNS Traffic Inspection

With AWS Route 53 Resolver DNS Firewall

DNS Firewall Features

DNS Filtering

- Domain name based filtering
- Create: Denylists, allow lists
- Custom Deny Actions
- Filtering on Resolver and Resolver Endpoints

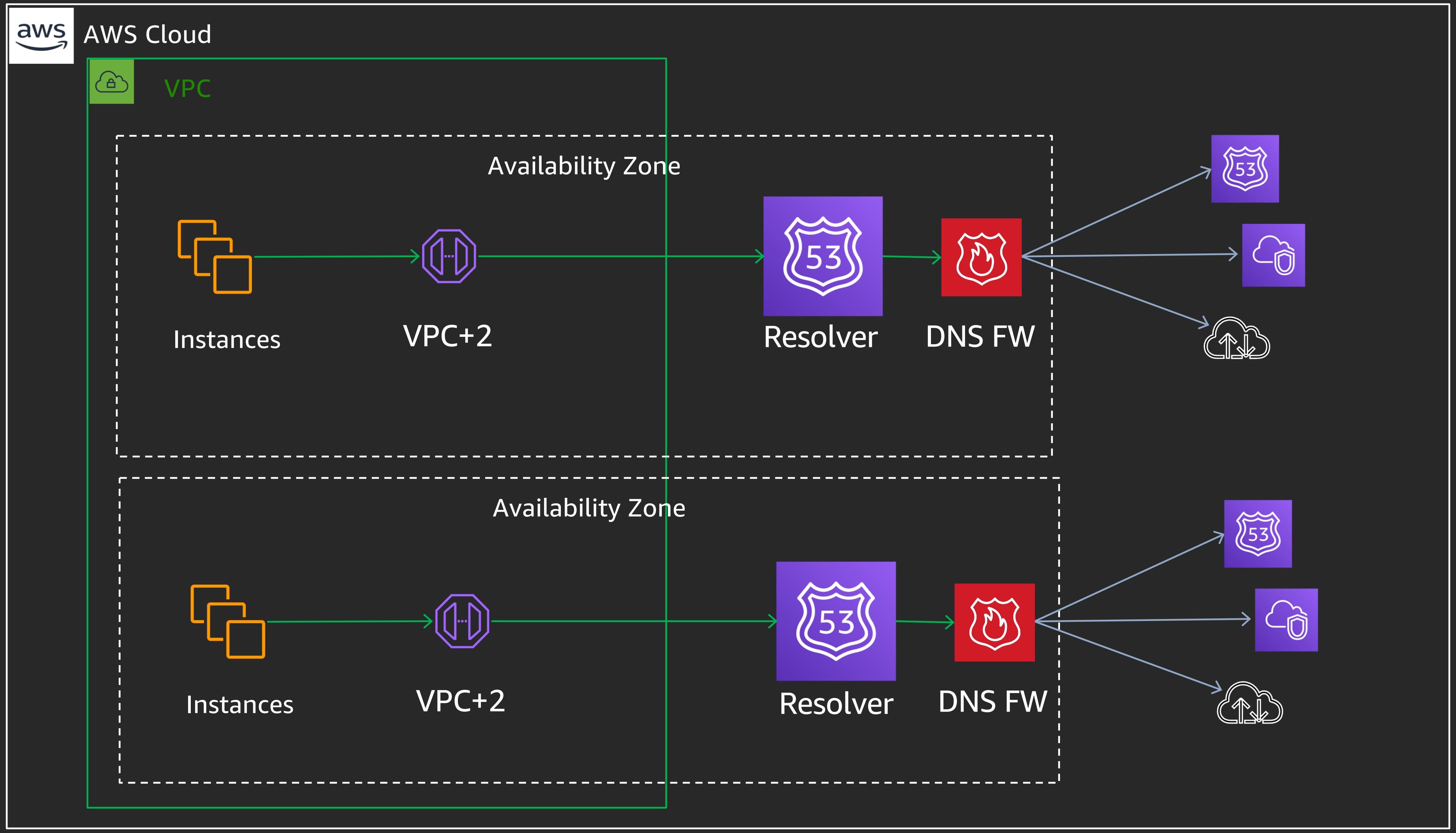
Managed Domain Lists

- Domain name based lists managed by AWS
- Provide protection against:
  - Malware
  - Botnet (C & C)

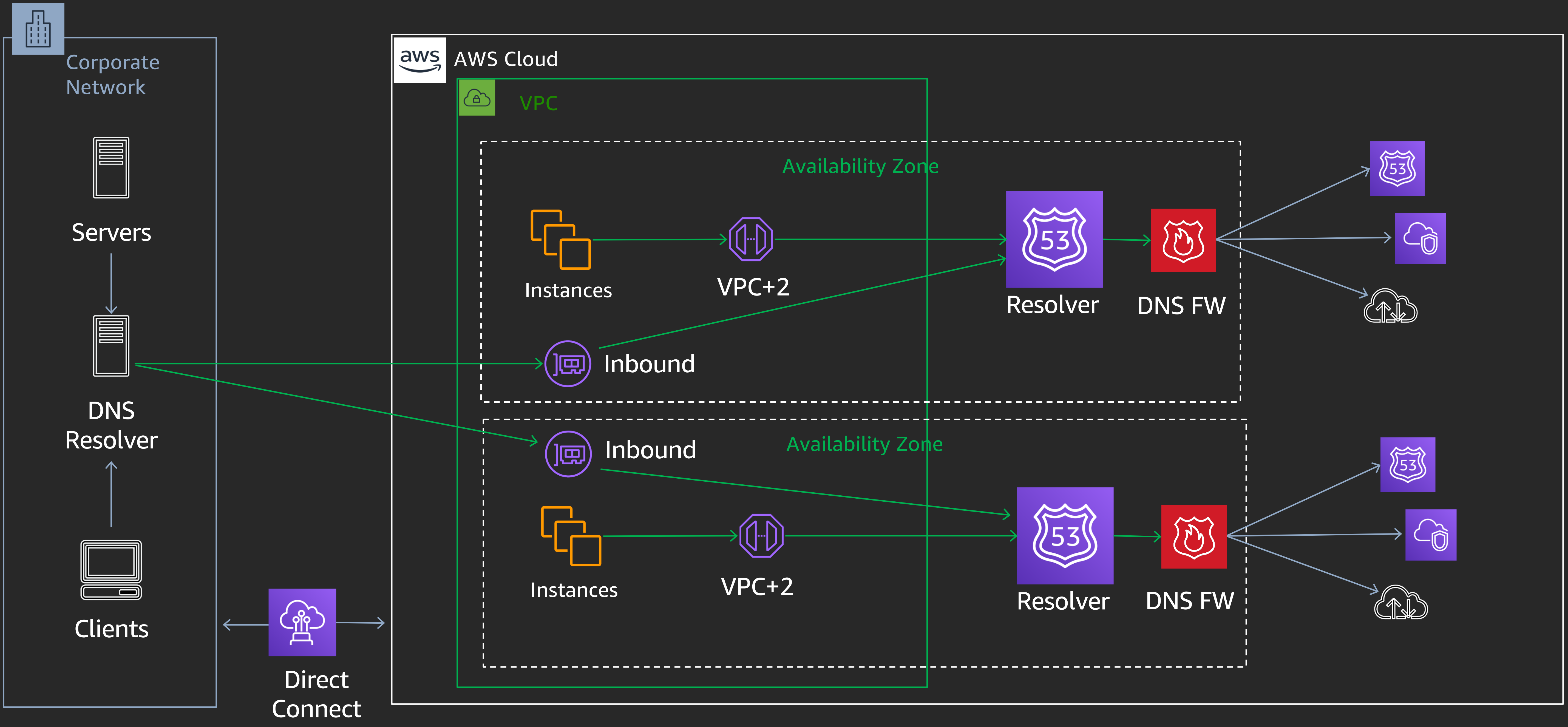
Visibility & Reporting

- Per Rule CloudWatch metrics
- Configurable logs sent to S3, CloudWatch, Kinesis

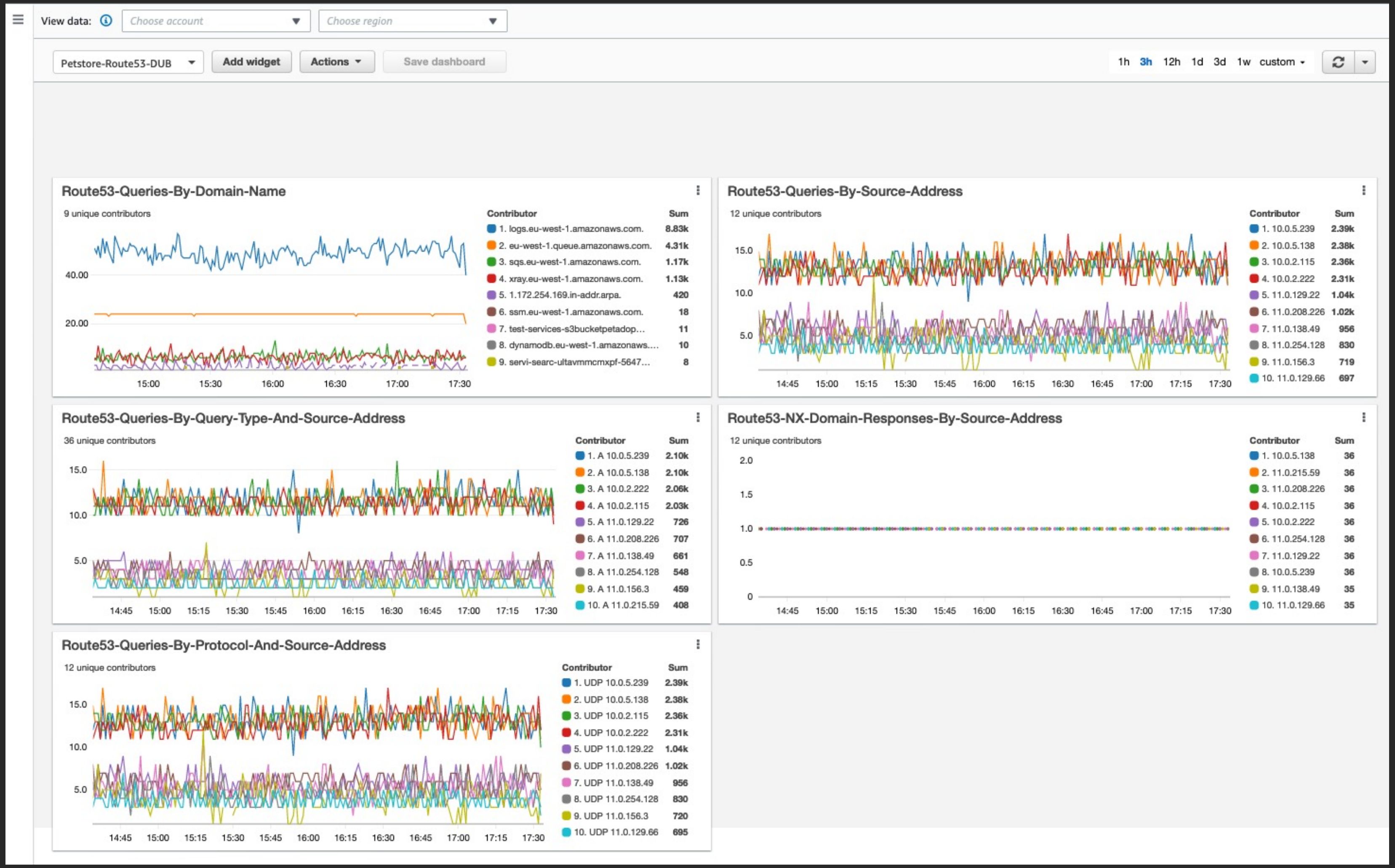
Deployment Model: Cloud-Only



Deployment Model: Hybrid



CloudWatch Contributor Insights



Route53-Queries-By-Domain-Name

9 unique contributors

Contributor	Sum
1. logs.eu-west-1.amazonaws.com.	8.83k
2. eu-west-1.queue.amazonaws.com.	4.31k
3. sqs.eu-west-1.amazonaws.com.	1.17k
4. xray.eu-west-1.amazonaws.com.	1.13k
5. 1.172.254.169.in-addr.arpa.	420
6. ssm.eu-west-1.amazonaws.com.	18
7. test-services-s3bucketpetadop...	11
8. dynamodb.eu-west-1.amazonaws....	10
9. servi-searc-ultavmmcmxpf-5647...	8

Route53-Queries-By-Source-Address

12 unique contributors

Contributor	Sum
1. 10.0.5.239	2.39k
2. 10.0.5.138	2.38k
3. 10.0.2.115	2.36k
4. 10.0.2.222	2.31k
5. 11.0.129.22	1.04k
6. 11.0.208.226	1.02k
7. 11.0.138.49	956
8. 11.0.254.128	830
9. 11.0.156.3	719
10. 11.0.129.66	697

Route53-Queries-By-Query-Type-And-Source-Address

36 unique contributors

Contributor	Sum
1. A 10.0.5.239	2.10k
2. A 10.0.5.138	2.10k
3. A 10.0.2.222	2.06k
4. A 10.0.2.115	2.03k
5. A 11.0.129.22	726
6. A 11.0.208.226	707
7. A 11.0.138.49	661
8. A 11.0.254.128	548
9. A 11.0.156.3	459
10. A 11.0.215.59	408

Route53-NX-Domain-Responses-By-Source-Address

12 unique contributors

Contributor	Sum
1. 10.0.5.138	36
2. 11.0.215.59	36
3. 11.0.208.226	36
4. 10.0.2.115	36
5. 10.0.2.222	36
6. 11.0.254.128	36
7. 11.0.129.22	36
8. 10.0.5.239	36
9. 11.0.138.49	35
10. 11.0.129.66	35

Route53-Queries-By-Protocol-And-Source-Address

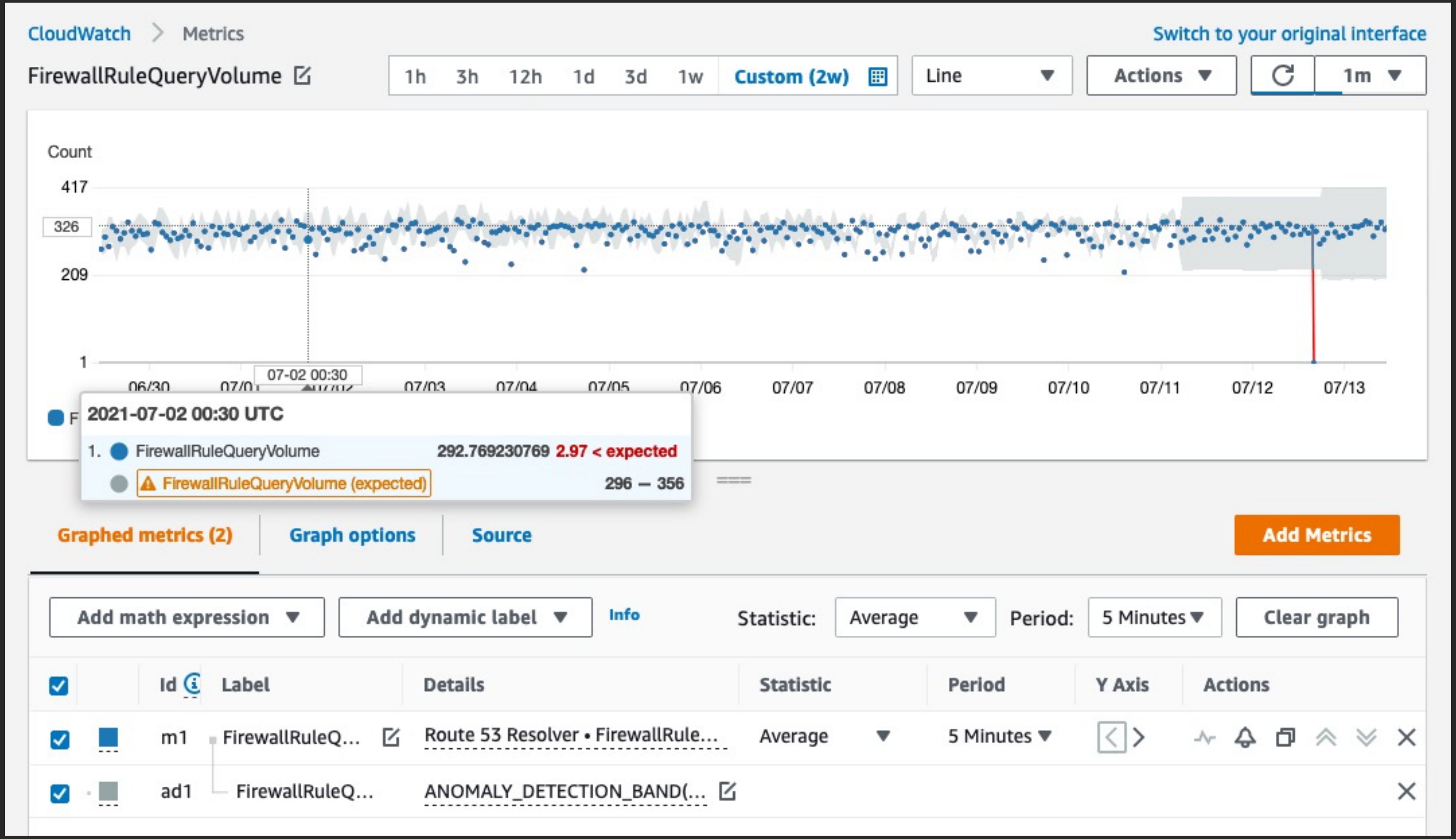
12 unique contributors

Contributor	Sum
1. UDP 10.0.5.239	2.39k
2. UDP 10.0.5.138	2.38k
3. UDP 10.0.2.115	2.36k
4. UDP 10.0.2.222	2.31k
5. UDP 11.0.129.22	1.04k
6. UDP 11.0.208.226	1.02k
7. UDP 11.0.138.49	956
8. UDP 11.0.254.128	830
9. UDP 11.0.156.3	720
10. UDP 11.0.129.66	695

- Surface outliers and top talkers
- Identify impacted users and resources
- Get actionable alerts & take remedial actions

CloudWatch Anomaly Detection

Use CloudWatch Anomaly Detection to help avoid manual configuration of static thresholds, and to more clearly differentiate between normal and problematic behavior



2021-07-02 00:30 UTC

1. FirewallRuleQueryVolume 292.769230769 2.97 < expected

FirewallRuleQueryVolume (expected) 296 — 356

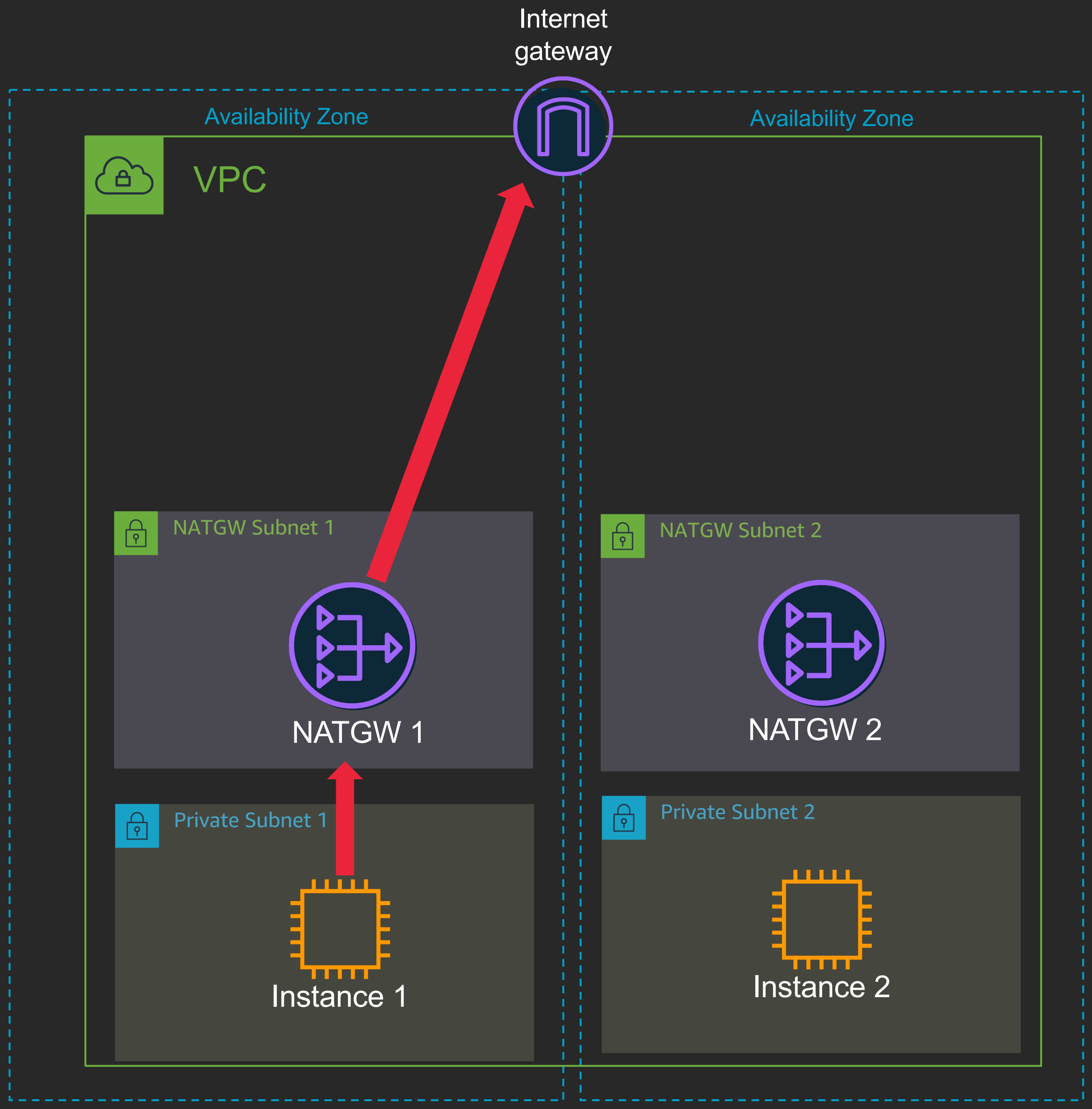
Statistic: Average

Period: 5 Minutes

☑		Id	Label	Details	Statistic	Period	Y Axis	Actions
☑	🟦	m1	FirewallRuleQ...	Route 53 Resolver • FirewallRule...	Average	5 Minutes		
☑	■	ad1	FirewallRuleQ...	ANOMALY_DETECTION_BAND(...				

Deployment patterns

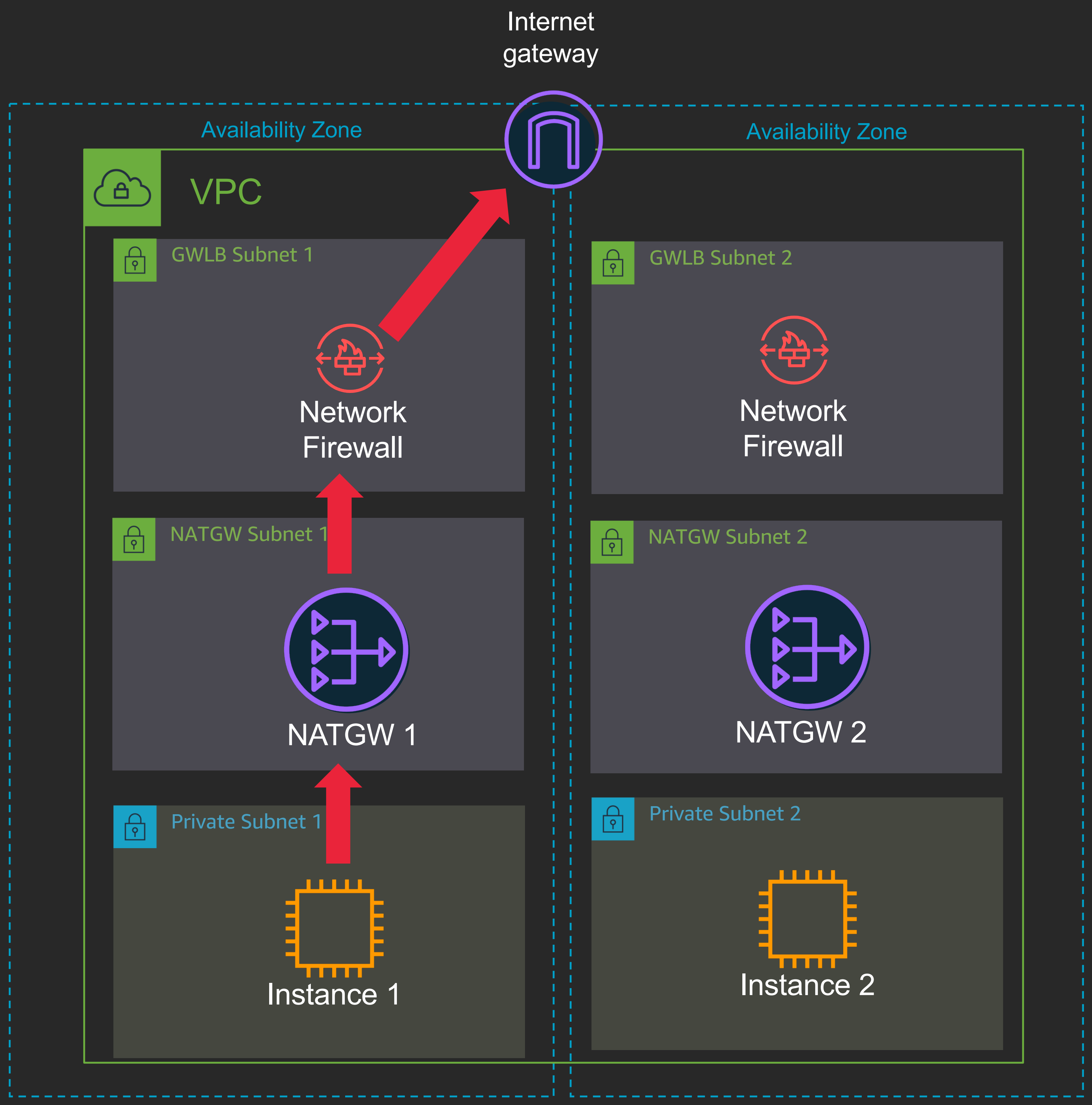
DNS Firewall, Network Firewall, Guard Duty



Instances using an external DNS server

Considerations

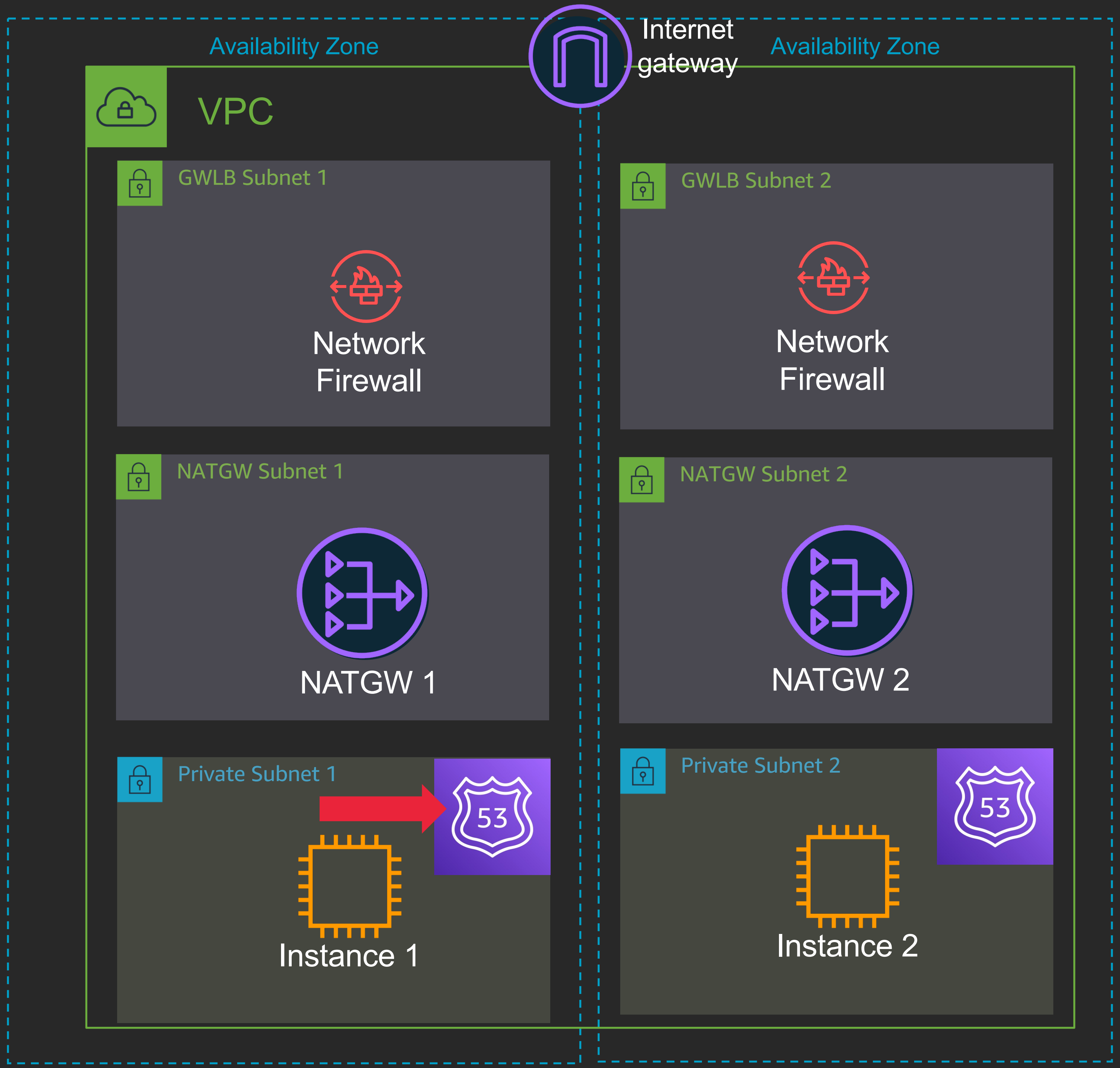
- No visibility into what FQDNs are being queried
- Bypasses GuardDuty DNS query detections
- No visibility into C&C traffic



Instances using an external DNS server with Network Firewall

Considerations

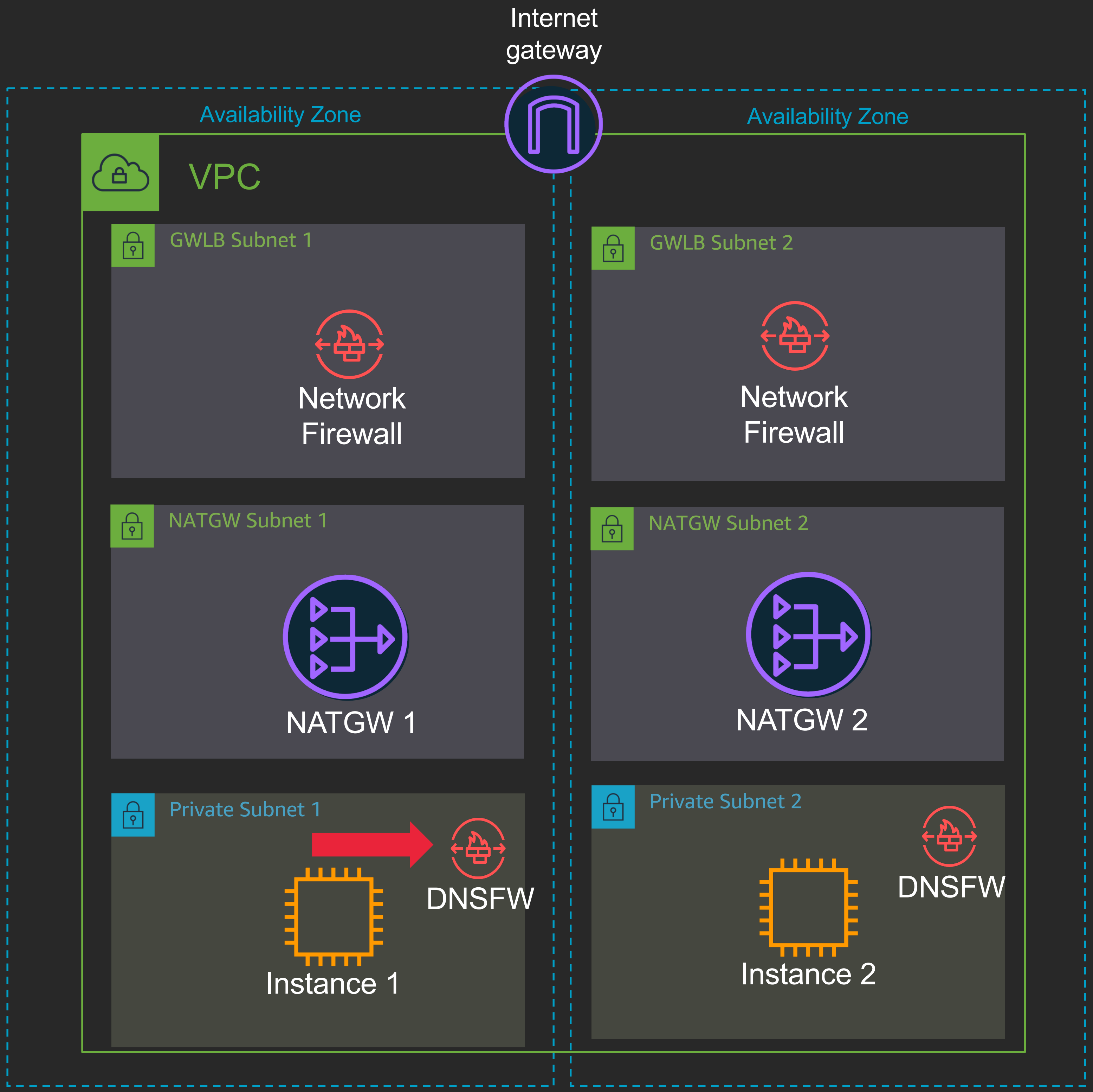
- Network Firewall gives visibility and control over DNS requests leveraging external DNS servers
- Bypasses GuardDuty’s DNS query detections



Instances using Route 53 Resolver

Considerations

- DNS requests bypass Network Firewall
- DNS Query Logging for FQDN visibility can be enabled
- No control over what queries are answered
- GuardDuty can provide visibility and alert to bad domains being queried, and DNS tunneling / exfiltration



Instances using Route 53 Resolver DNSFW

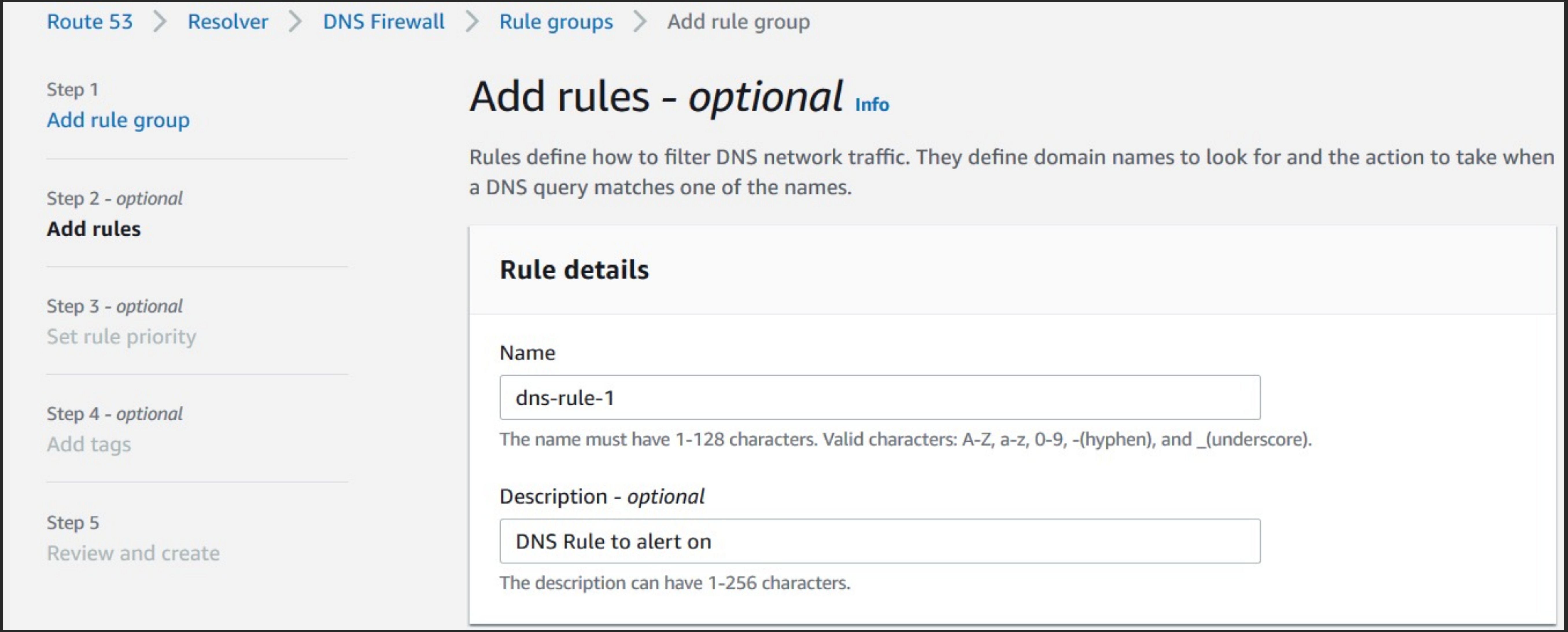
Considerations

- Defense in depth
- Visibility and control over requests to R53 endpoint and external requests
- Maximum visibility and control

Deployment steps

AWS Console

Creating DNS Firewall Rule



Route 53 > Resolver > DNS Firewall > Rule groups > Add rule group

Step 1
Add rule group

Step 2 - *optional*
Add rules

Step 3 - *optional*
Set rule priority

Step 4 - *optional*
Add tags

Step 5
Review and create

Add rules - *optional* Info

Rules define how to filter DNS network traffic. They define domain names to look for and the action to take when a DNS query matches one of the names.

Rule details

Name

dns-rule-1

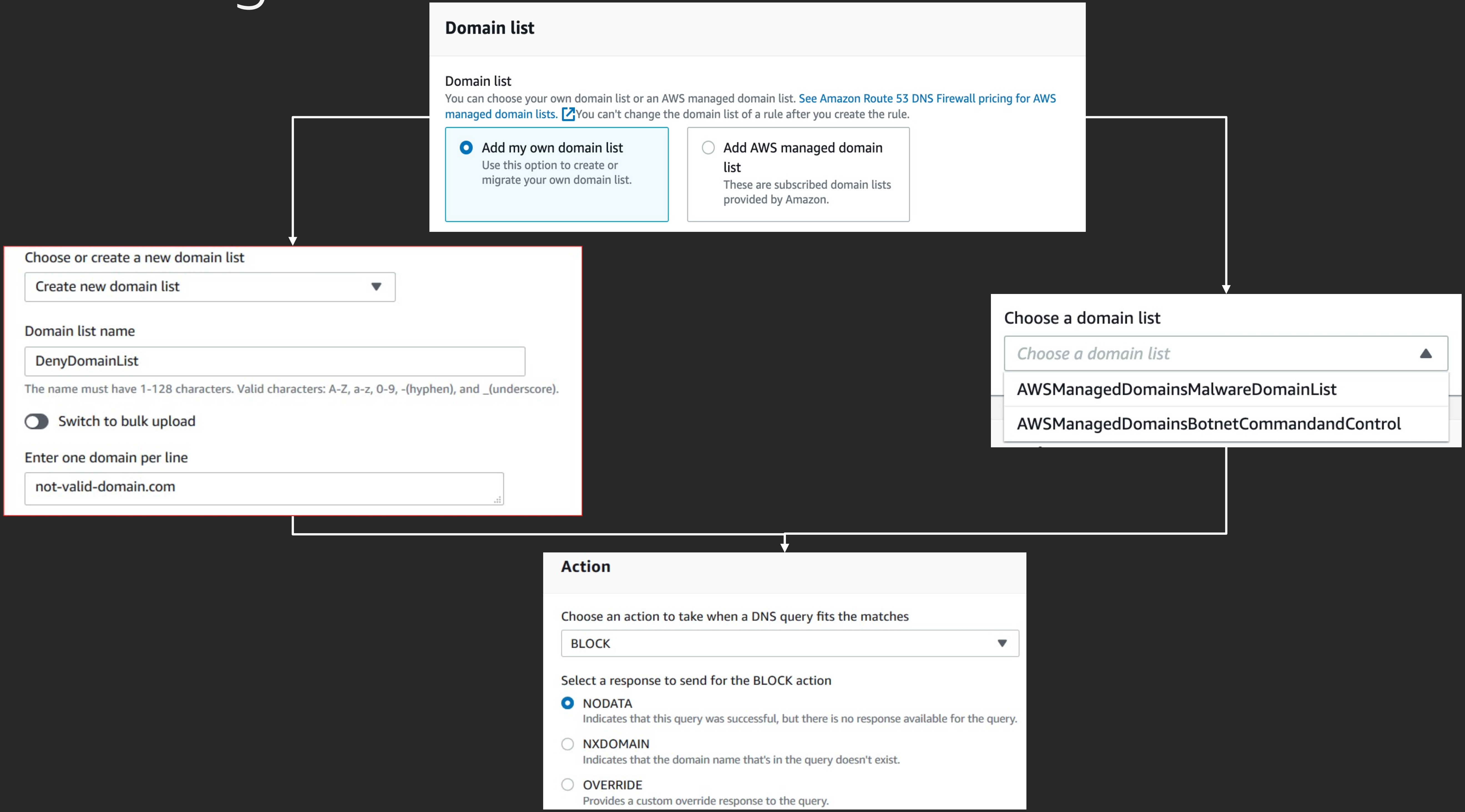
The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, -(hyphen), and _(underscore).

Description - *optional*

DNS Rule to alert on

The description can have 1-256 characters.

Creating DNS Firewall Rule



Domain list

Domain list

You can choose your own domain list or an AWS managed domain list. See Amazon Route 53 DNS Firewall pricing for AWS managed domain lists. You can't change the domain list of a rule after you create the rule.

● Add my own domain list
Use this option to create or migrate your own domain list.

○ Add AWS managed domain list
These are subscribed domain lists provided by Amazon.

Choose or create a new domain list

Create new domain list ▼

Domain list name

DenyDomainList

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, -(hyphen), and _(underscore).

Switch to bulk upload

Enter one domain per line

not-valid-domain.com

Choose a domain list

Choose a domain list ▲

AWSManagedDomainsMalwareDomainList

AWSManagedDomainsBotnetCommandandControl

Action

Choose an action to take when a DNS query fits the matches

BLOCK ▼

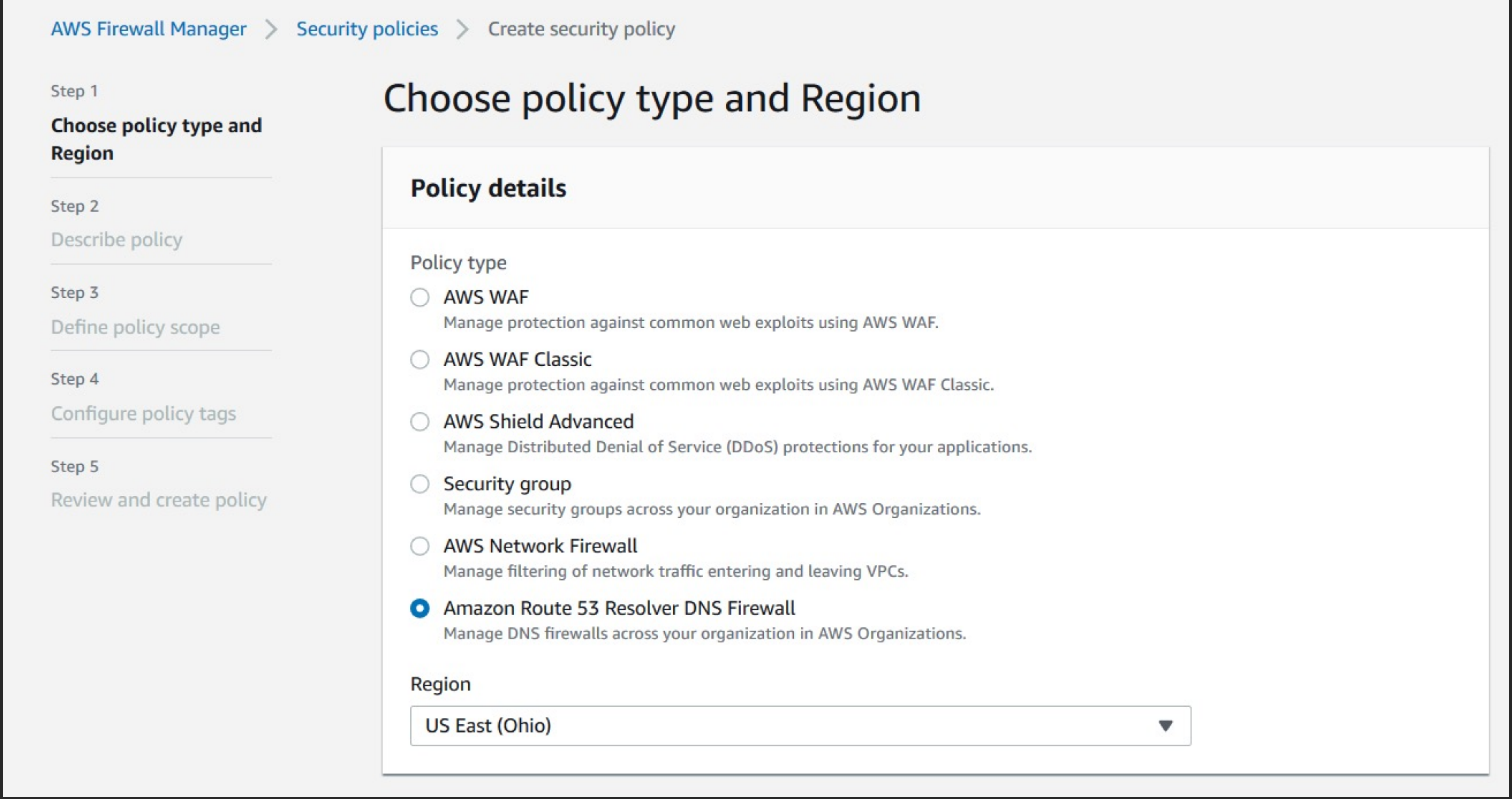
Select a response to send for the BLOCK action

● NODATA
Indicates that this query was successful, but there is no response available for the query.

○ NXDOMAIN
Indicates that the domain name that's in the query doesn't exist.

○ OVERRIDE
Provides a custom override response to the query.

Create DNS Firewall Policy



AWS Firewall Manager > Security policies > Create security policy

Step 1

Choose policy type and Region

Step 2

Describe policy

Step 3

Define policy scope

Step 4

Configure policy tags

Step 5

Review and create policy

Choose policy type and Region

Policy details

Policy type

○ AWS WAF
Manage protection against common web exploits using AWS WAF.

○ AWS WAF Classic
Manage protection against common web exploits using AWS WAF Classic.

○ AWS Shield Advanced
Manage Distributed Denial of Service (DDoS) protections for your applications.

○ Security group
Manage security groups across your organization in AWS Organizations.

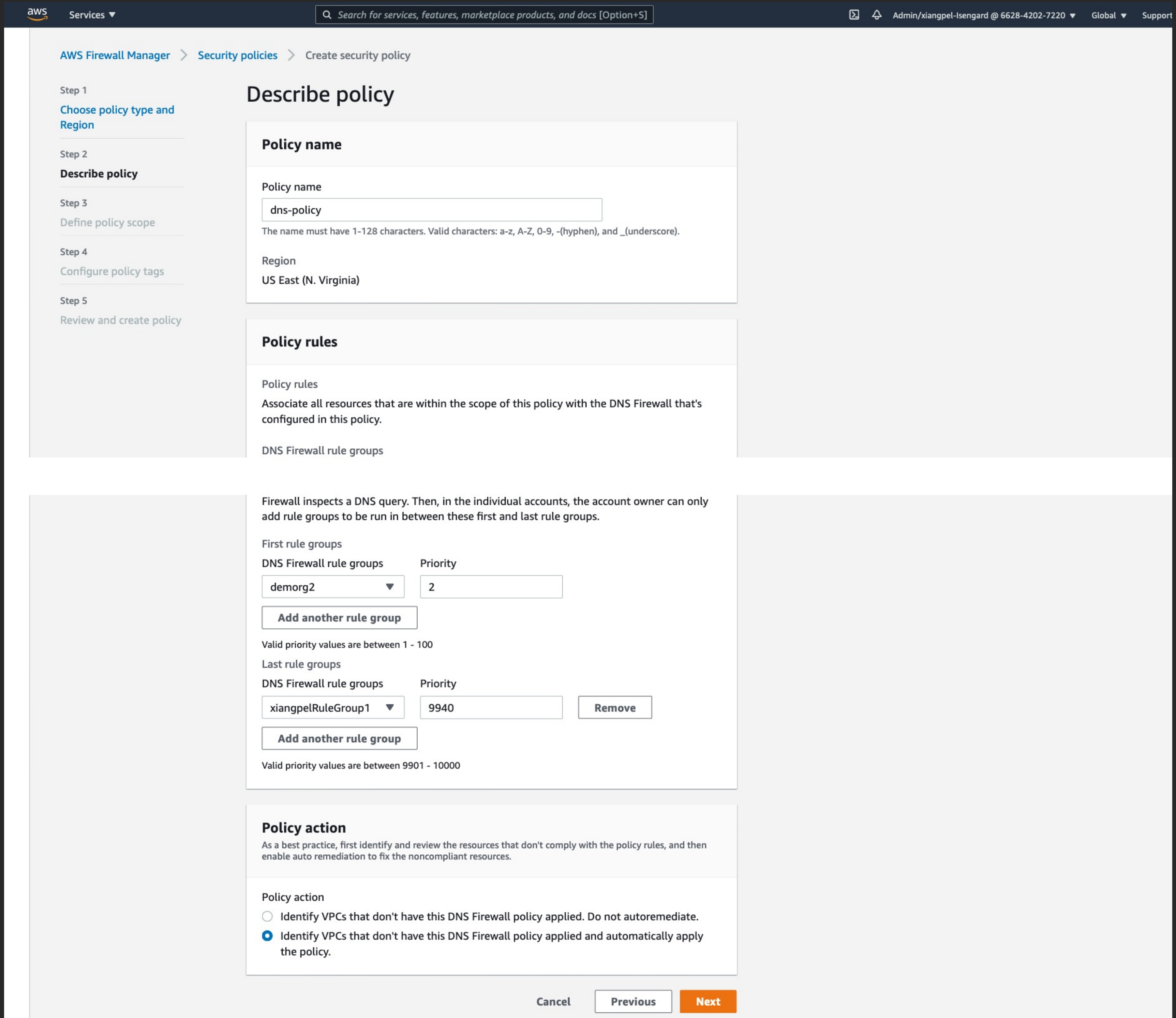
○ AWS Network Firewall
Manage filtering of network traffic entering and leaving VPCs.

● Amazon Route 53 Resolver DNS Firewall
Manage DNS firewalls across your organization in AWS Organizations.

Region

US East (Ohio) ▼

Describe Policy



aws
Services ▼
Search for services, features, marketplace products, and docs [Option+S]
Admin/xiangpel-Isengard @ 6628-4202-7220 ▼
Global ▼
Support

AWS Firewall Manager > Security policies > Create security policy

Step 1
Choose policy type and Region

Step 2
Describe policy

Step 3
Define policy scope

Step 4
Configure policy tags

Step 5
Review and create policy

Describe policy

Policy name

Policy name

dns-policy

The name must have 1-128 characters. Valid characters: a-z, A-Z, 0-9, -(hyphen), and _(underscore).

Region

US East (N. Virginia)

Policy rules

Policy rules

Associate all resources that are within the scope of this policy with the DNS Firewall that's configured in this policy.

DNS Firewall rule groups

Firewall inspects a DNS query. Then, in the individual accounts, the account owner can only add rule groups to be run in between these first and last rule groups.

First rule groups

DNS Firewall rule groups	Priority
demorg2 ▼	2

Add another rule group

Valid priority values are between 1 - 100

Last rule groups

DNS Firewall rule groups	Priority	
xiangpelRuleGroup1 ▼	9940	Remove

Add another rule group

Valid priority values are between 9901 - 10000

Policy action

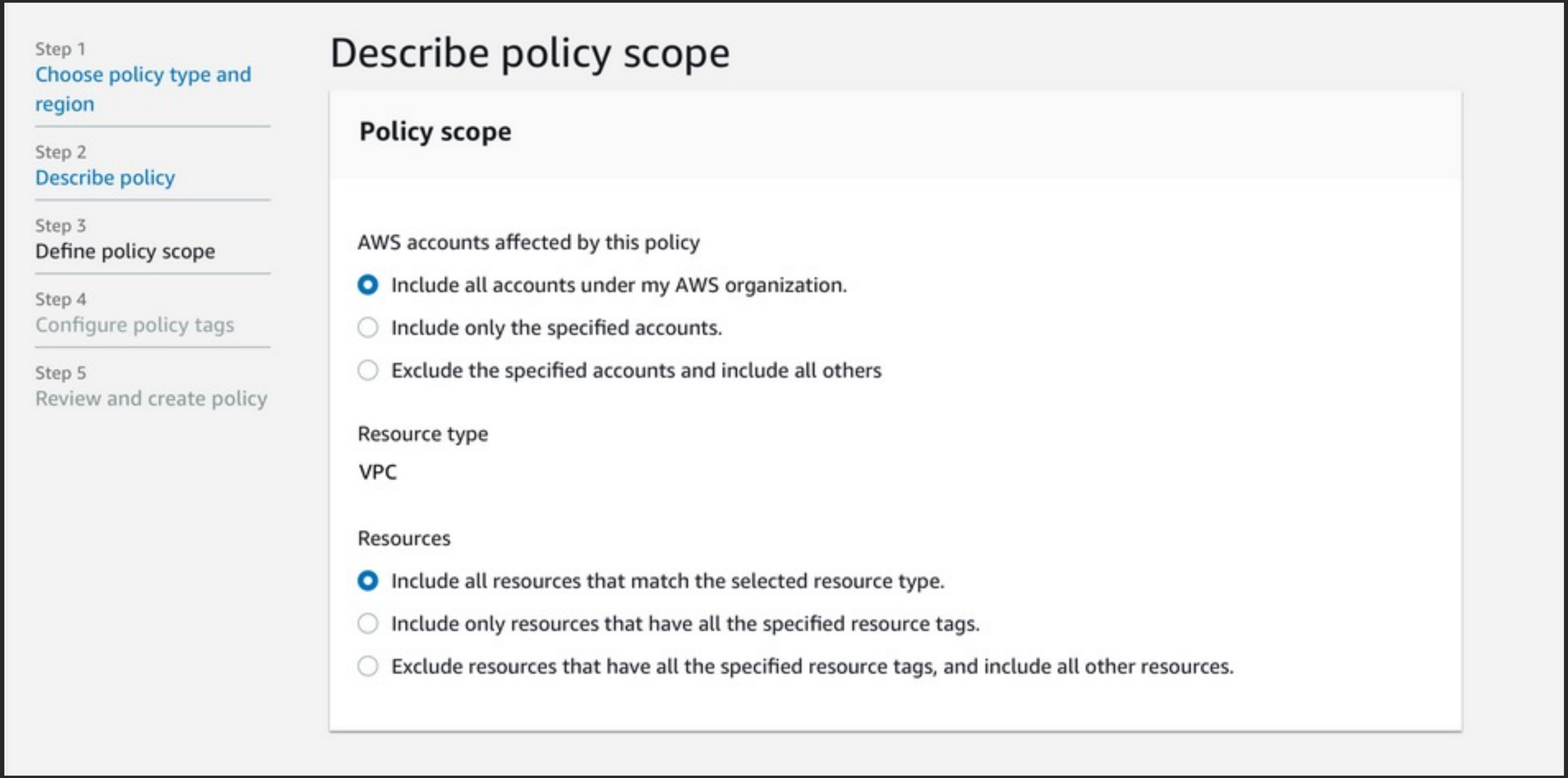
As a best practice, first identify and review the resources that don't comply with the policy rules, and then enable auto remediation to fix the noncompliant resources.

Policy action

○ Identify VPCs that don't have this DNS Firewall policy applied. Do not autoremediate.

● Identify VPCs that don't have this DNS Firewall policy applied and automatically apply the policy.

Define Policy Scope



Step 1
Choose policy type and region

Step 2
Describe policy

Step 3
Define policy scope

Step 4
Configure policy tags

Step 5
Review and create policy

Describe policy scope

Policy scope

AWS accounts affected by this policy

● Include all accounts under my AWS organization.

○ Include only the specified accounts.

○ Exclude the specified accounts and include all others

Resource type

VPC

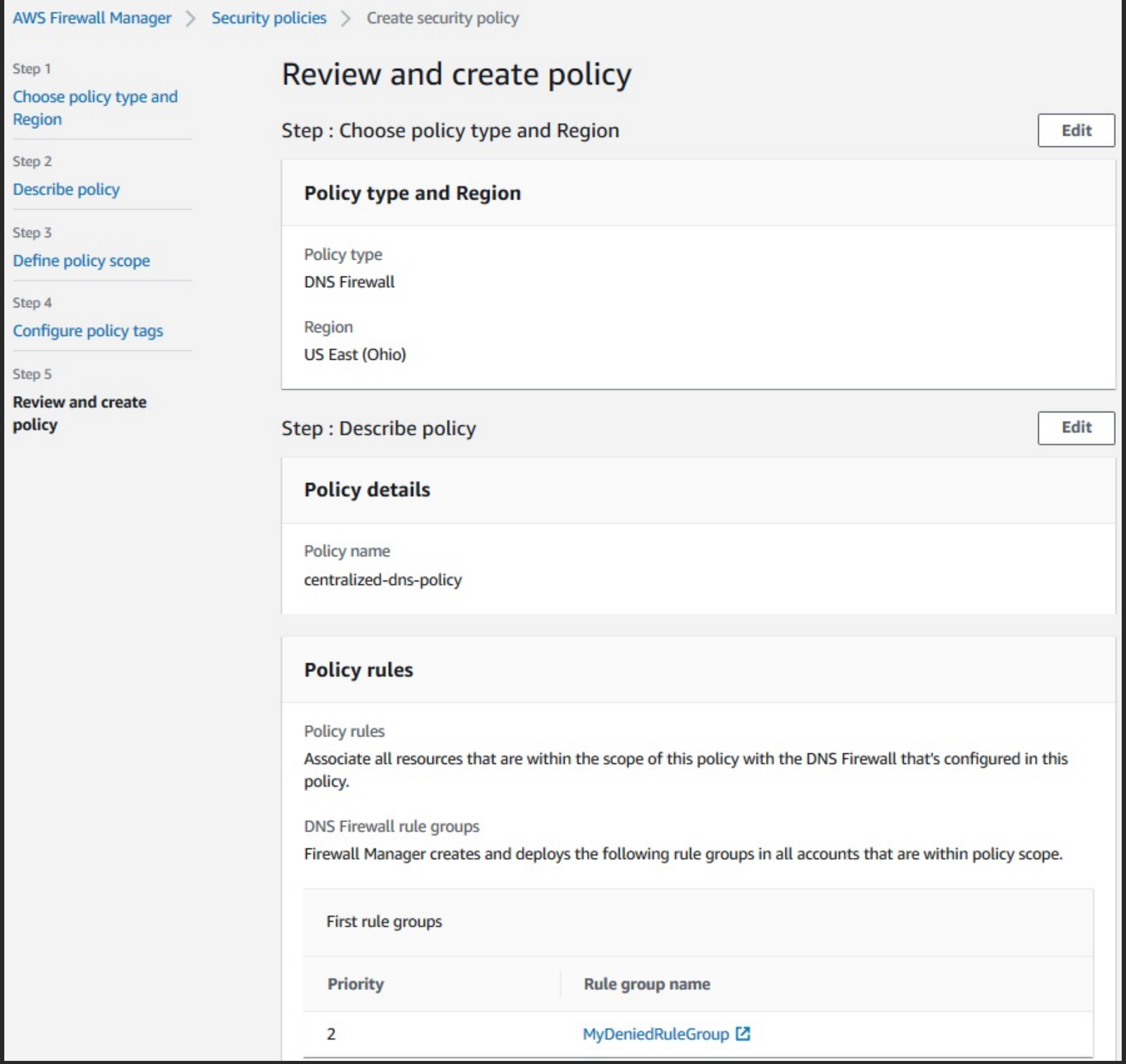
Resources

● Include all resources that match the selected resource type.

○ Include only resources that have all the specified resource tags.

○ Exclude resources that have all the specified resource tags, and include all other resources.

Review and complete



AWS Firewall Manager > Security policies > Create security policy

Step 1
Choose policy type and Region

Step 2
Describe policy

Step 3
Define policy scope

Step 4
Configure policy tags

Step 5
Review and create policy

Review and create policy

Step : Choose policy type and Region

Policy type and Region

Policy type
DNS Firewall

Region
US East (Ohio)

Step : Describe policy

Policy details

Policy name
centralized-dns-policy

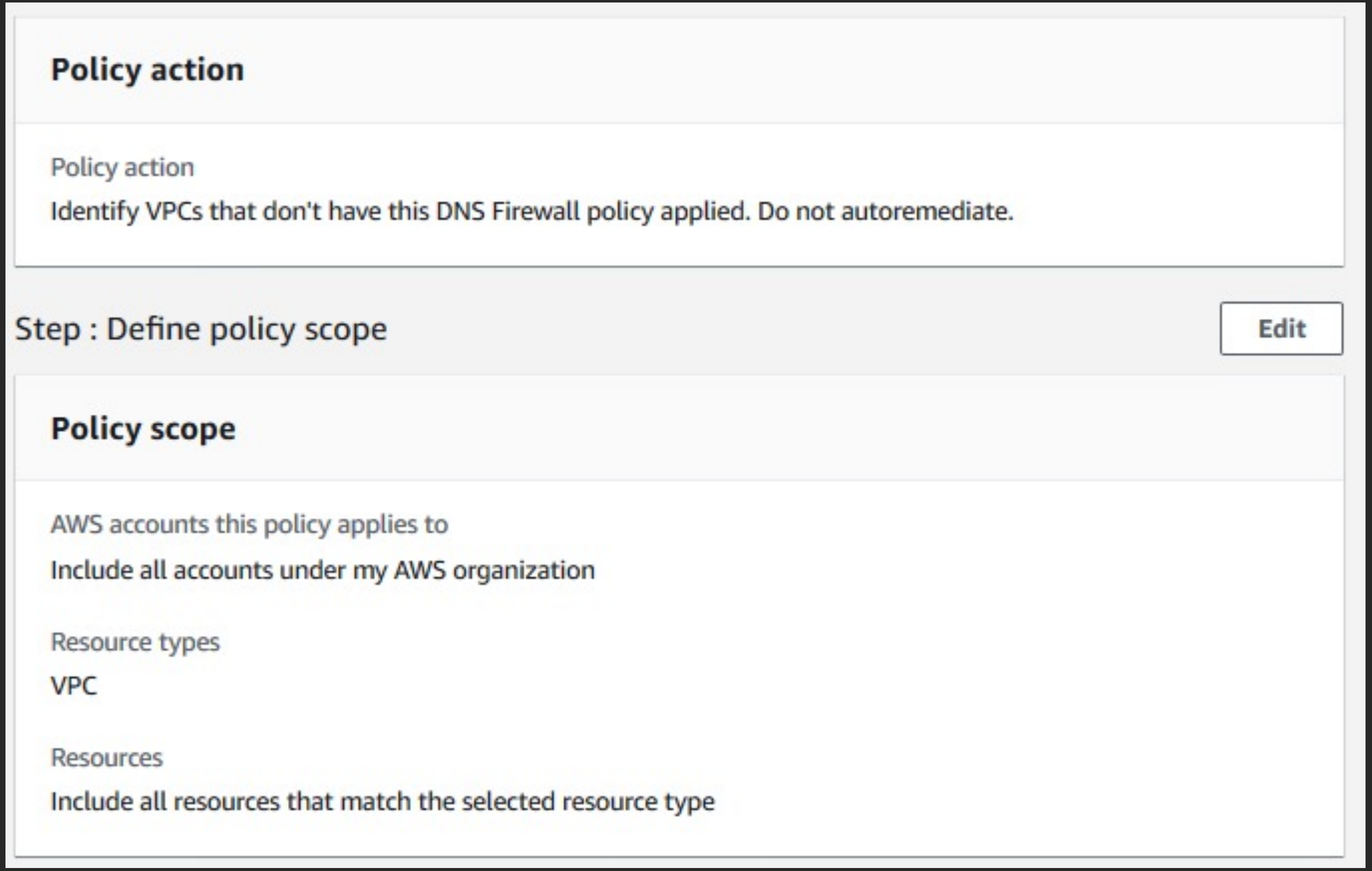
Policy rules

Policy rules
Associate all resources that are within the scope of this policy with the DNS Firewall that's configured in this policy.

DNS Firewall rule groups
Firewall Manager creates and deploys the following rule groups in all accounts that are within policy scope.

First rule groups

Priority	Rule group name
2	MyDeniedRuleGroup



Policy action

Policy action
Identify VPCs that don't have this DNS Firewall policy applied. Do not autoremediate.

Step : Define policy scope

Policy scope

AWS accounts this policy applies to
Include all accounts under my AWS organization

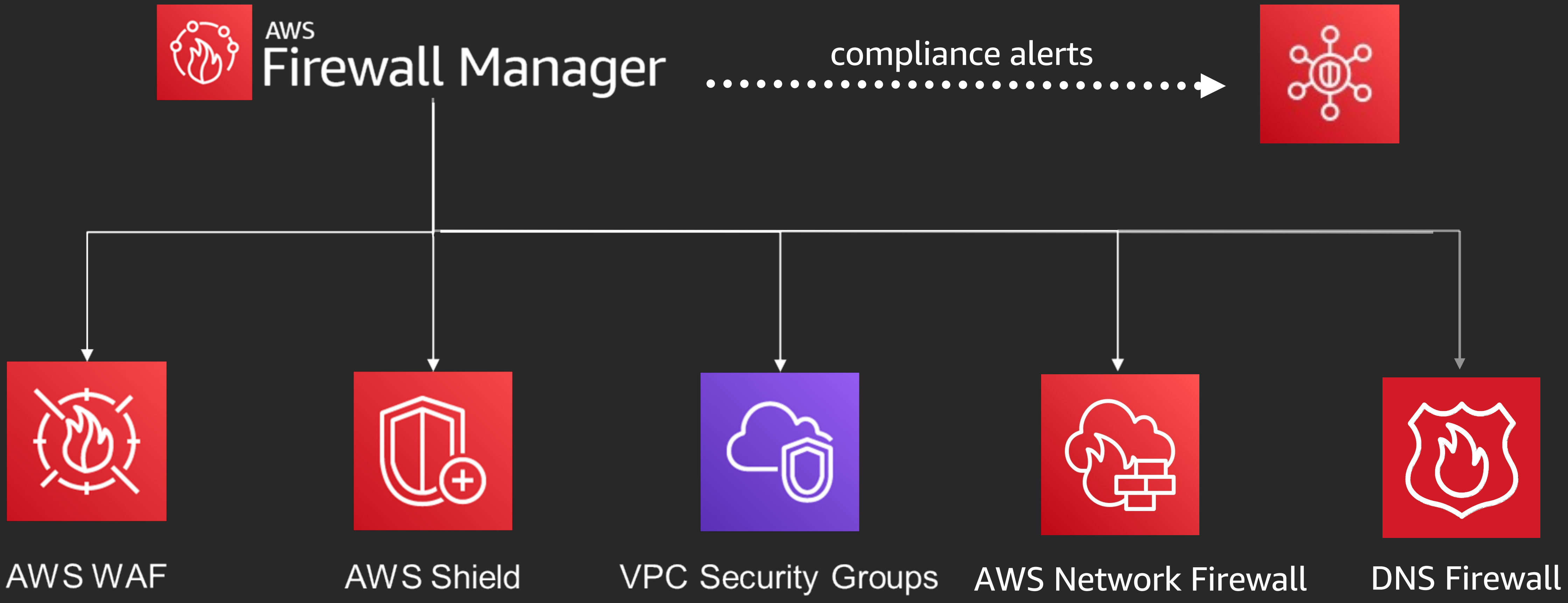
Resource types
VPC

Resources
Include all resources that match the selected resource type

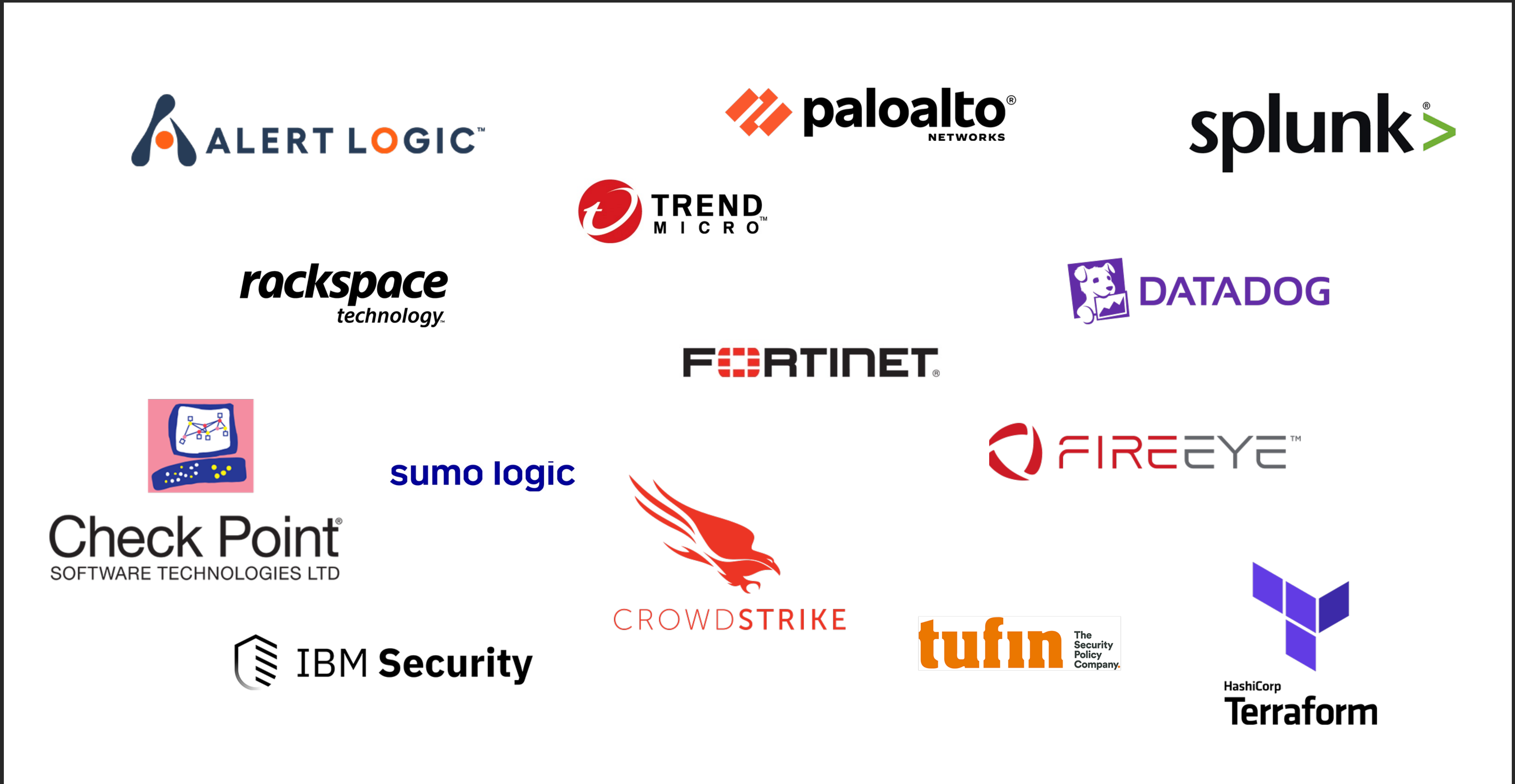
Manage it together

AWS Firewall Manager

Centralized AWS Firewall Manager



AWS Partners supporting AWS Network Firewall



AWS Partners supporting DNS Firewall

FIREEYE™

DATADOG

HashiCorp Terraform

splunk>

dynatrace

IBM QRadar

sumo logic

Thank you!

Vadim Omeltchenko
vadimo@amazon.cm

Vishal Lakhotia
lakhov@amazon.com