
AWS
re:Invent

API303

Application migration & hybrid cloud with RabbitMQ

Sam Dengler

Principal Solutions Architect

AWS

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Message architectures drive critical business applications



Financial services
Trades and transactions



Retail
Order processing and fulfillment



Health care
Clinical data exchange



Media and entertainment
Image, or video processing



Publication
Document capture and publication search



Education
Learning and engagement



Manufacturing
Multi-step process



Government
Citizen engagement

Loosely coupled, larger scale, more fault-tolerant systems with less dependencies → faster innovation

Messaging architectures

Born in cloud



AWS Cloud



Amazon Simple Queue Services (Amazon SQS)



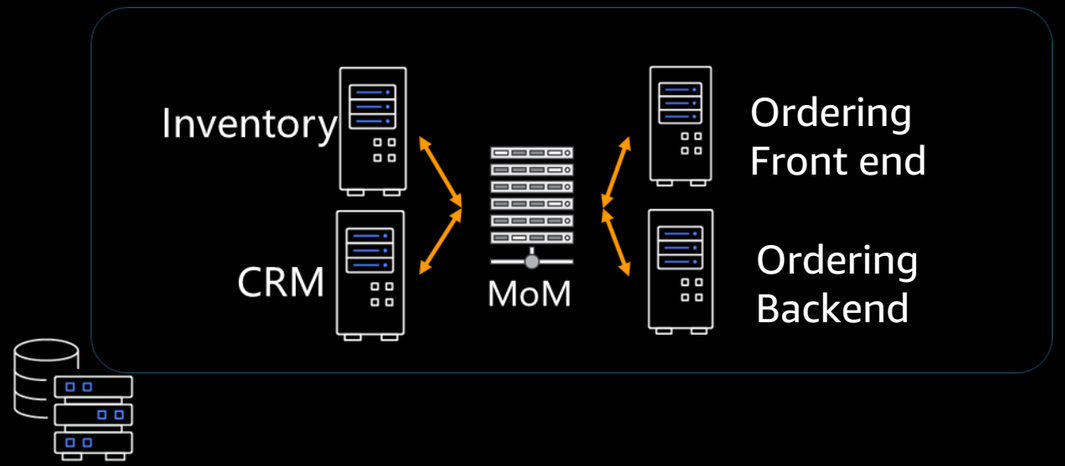
Amazon Simple Notification Service (Amazon SNS)



Amazon EventBridge

Message queuing service
Publish / subscribe messaging and notification service
Event-based Architecture

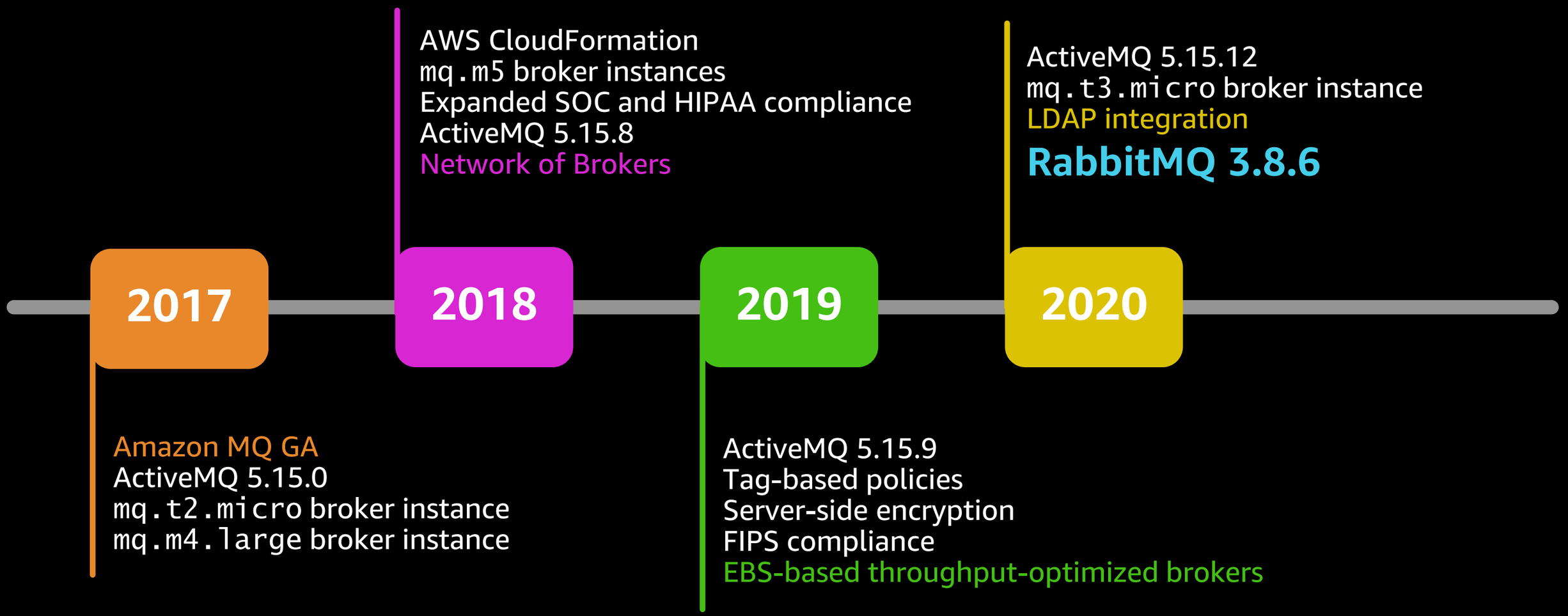
Traditional Applications



IBM MQ TIBCO ActiveMQ RabbitMQ

Message-oriented middleware (MoM) or Message Broker

Amazon MQ history



Amazon MQ for RabbitMQ



AWS re:Invent

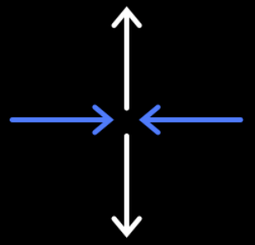
© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Why use Amazon MQ for RabbitMQ?



Setup

Automatic provisioning of single-node and clustered brokers with best practices by default



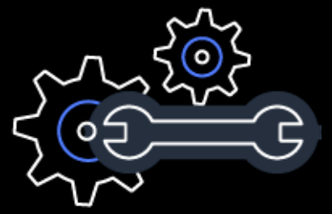
Scaling

Vertically scale between Amazon MQ broker instance types



Security

Encryption in transit over TLS and encryption at rest using KMS keys



Upgrades

Managed administrative tasks, such as software upgrades



Maintenance

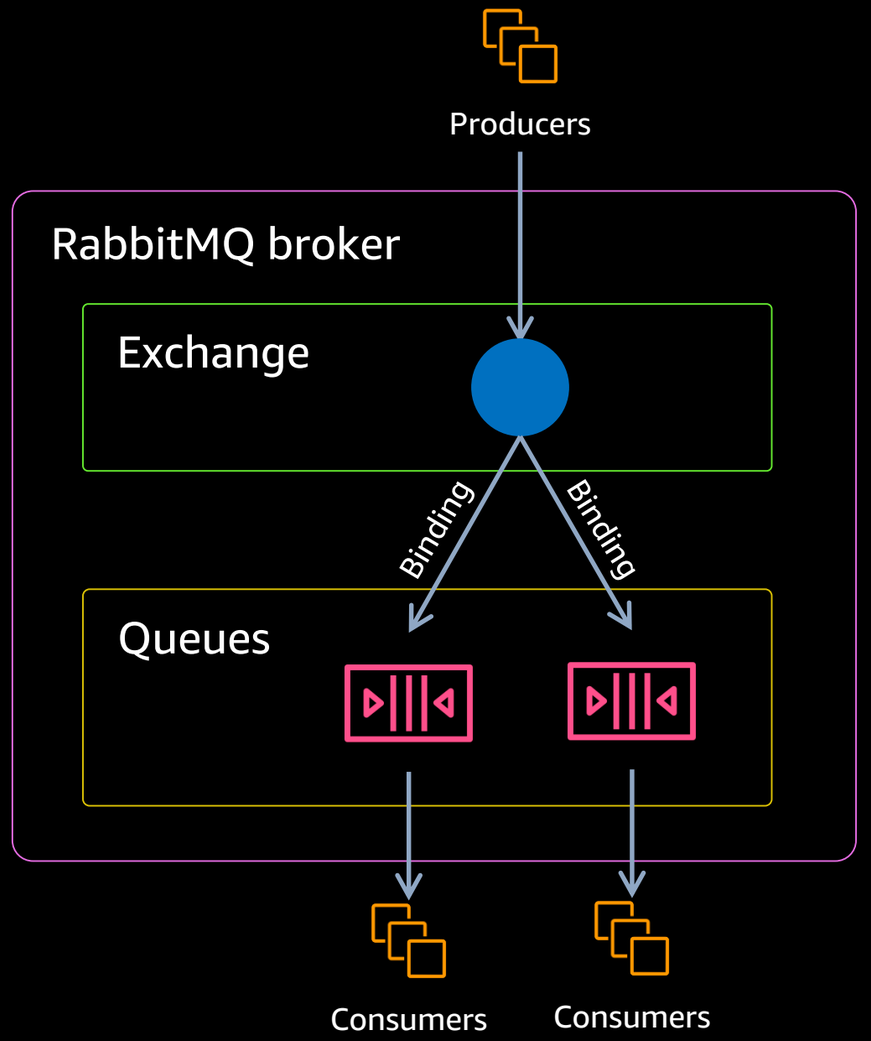
Customized maintenance window schedule

Getting Started with AmazonMQ for RabbitMQ

- Same open source as RabbitMQ v3.8.6
- AMQP 0-9-1 support
- Management, Monitoring, Shovel, and Federation plugins installed
- Choice of single-instance or three-node, clustered broker
- AWS CloudWatch metrics for broker monitoring
- AWS CloudWatch Logs for broker logging

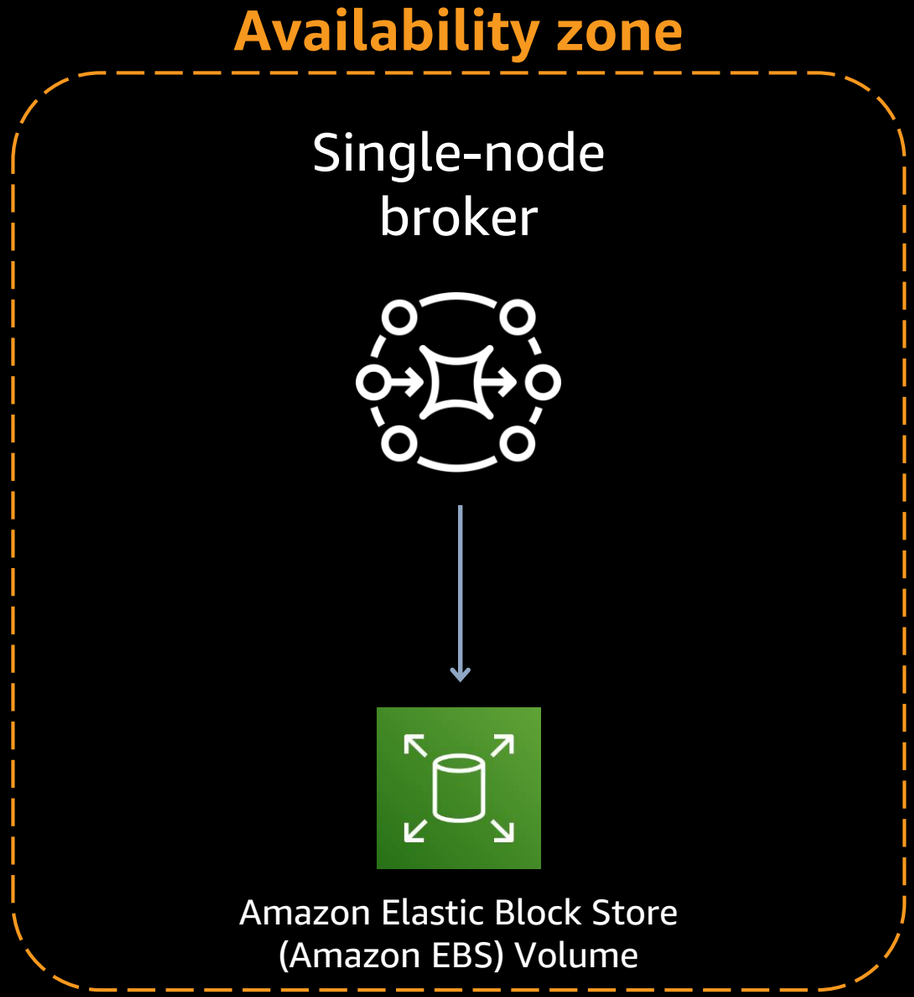
Exchanges, bindings, and queues

- Exchanges take a message and route it into zero or more queues; exchange types – Direct, Topic, Fanout, Headers
- Bindings are rules that exchanges use to route messages to queues
- Queues store messages that are consumed by applications



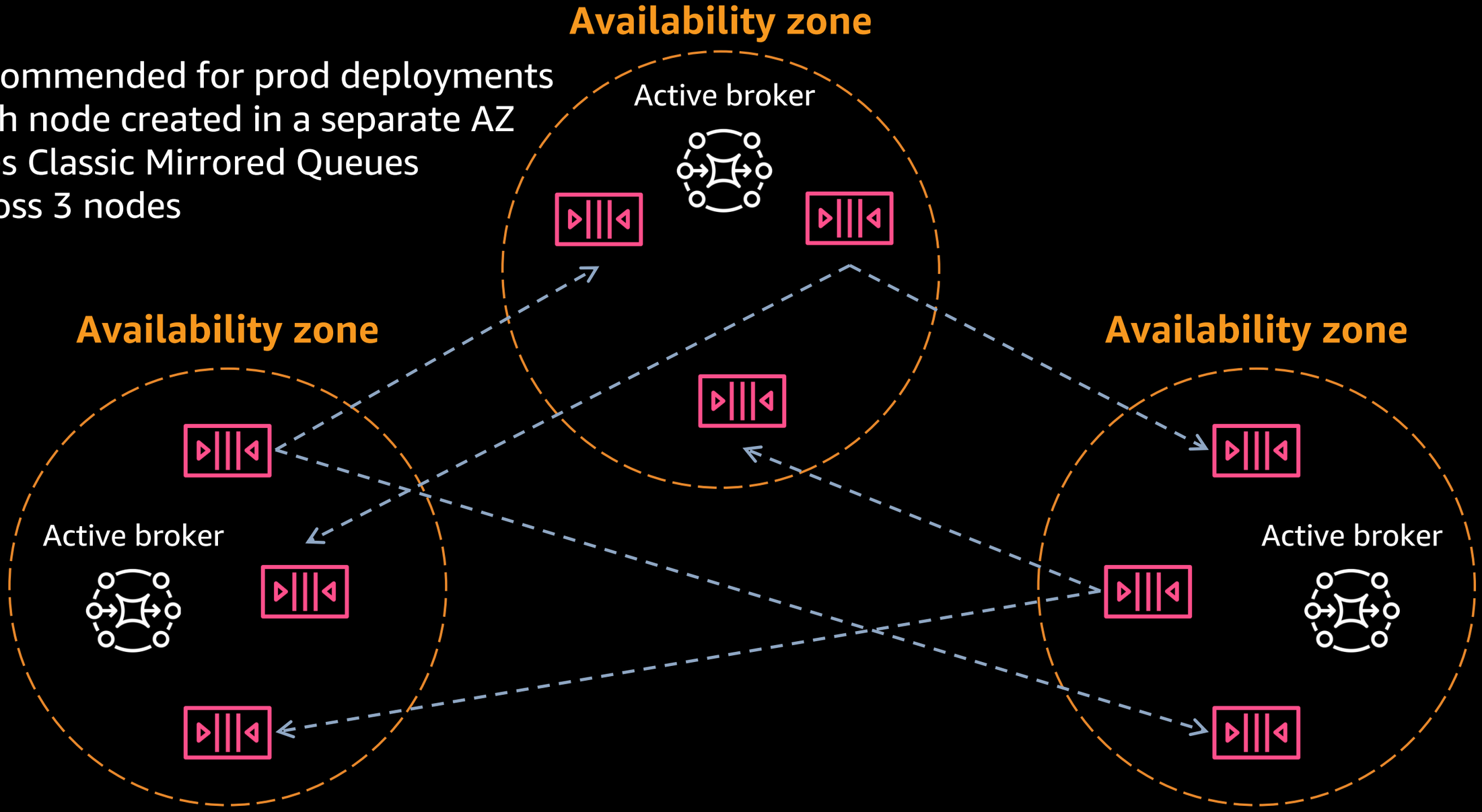
RabbitMQ single-node broker deployment

Recommended for non-production deployments and low-latency use cases with non-persistent messages



RabbitMQ three-node cluster deployment

- Recommended for prod deployments
- Each node created in a separate AZ
- Uses Classic Mirrored Queues across 3 nodes



Three-node cluster resilience

- Single endpoint for producers and consumers to connect to the cluster
- Amazon MQ enforces mirrored queue configurations on all RabbitMQ policies using `ha-mode=all` and `ha-sync-mode=automatic`

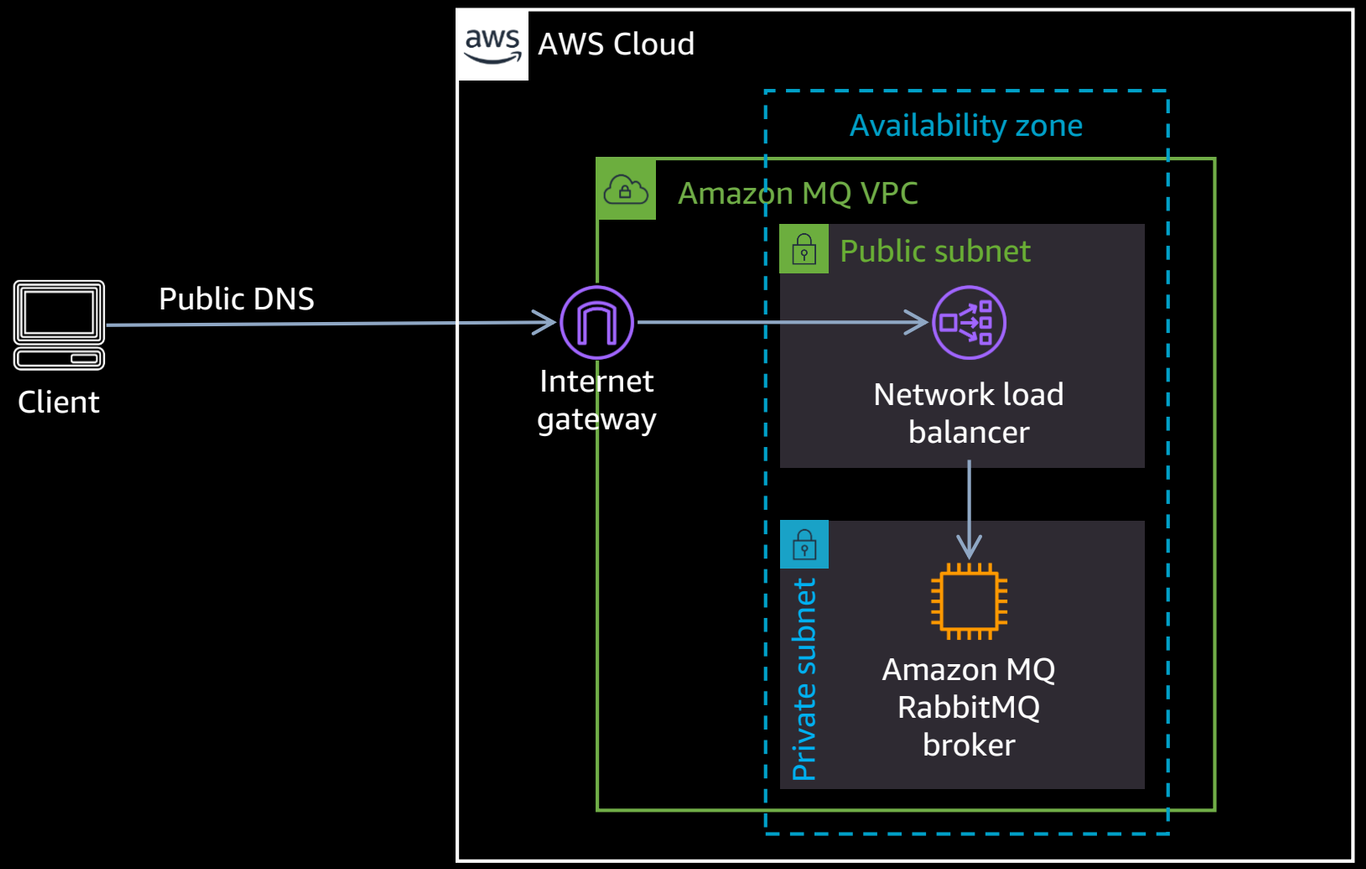
In the event of cluster node replacement –

1. Amazon MQ replaces the node automatically
2. Consumers are disconnected from the cluster and must reconnect
3. Queues will automatically synchronize when a new mirror joins

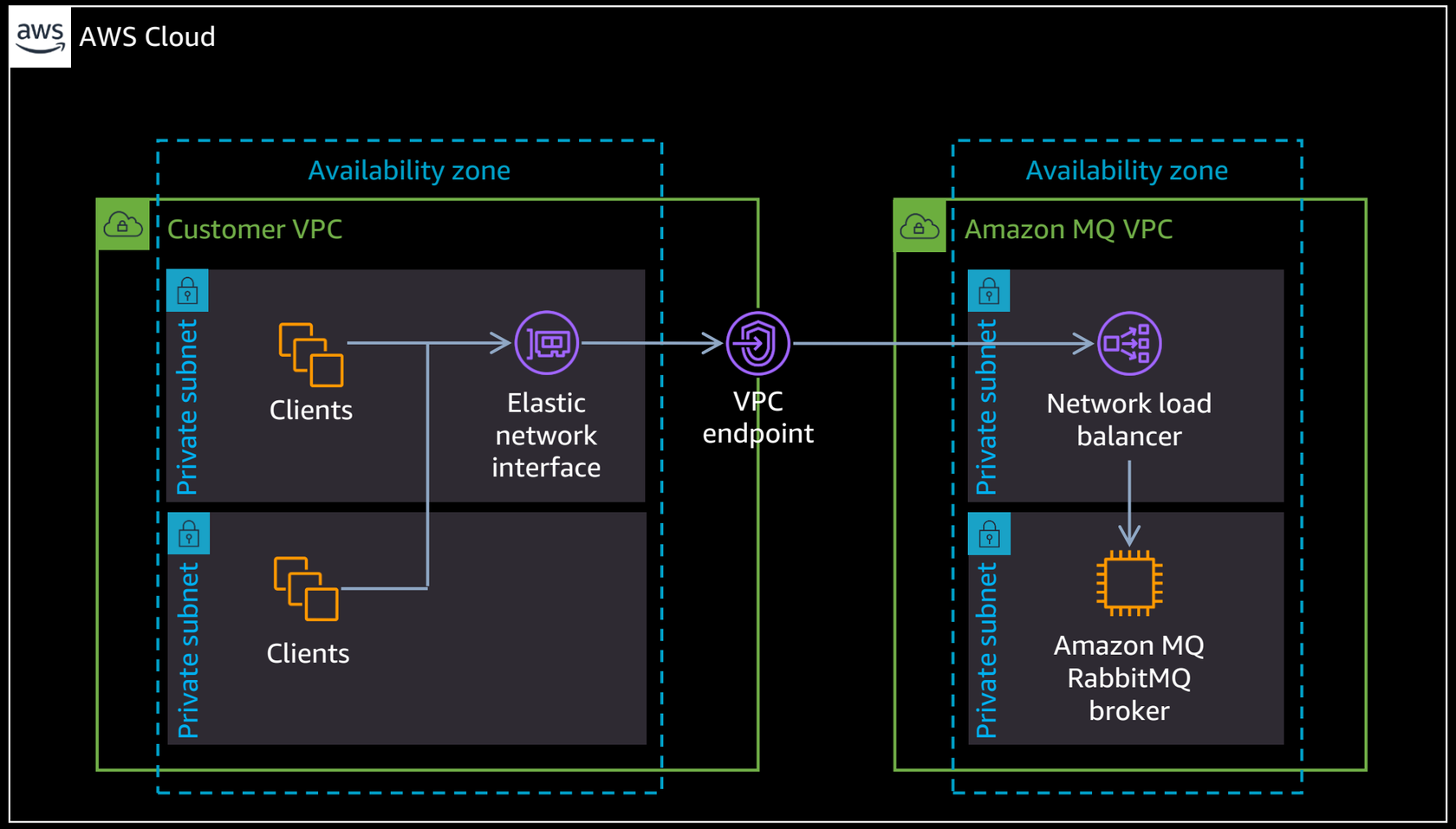
Queue synchronization is a blocking operation per queue

Keep queues small by consuming messages quickly

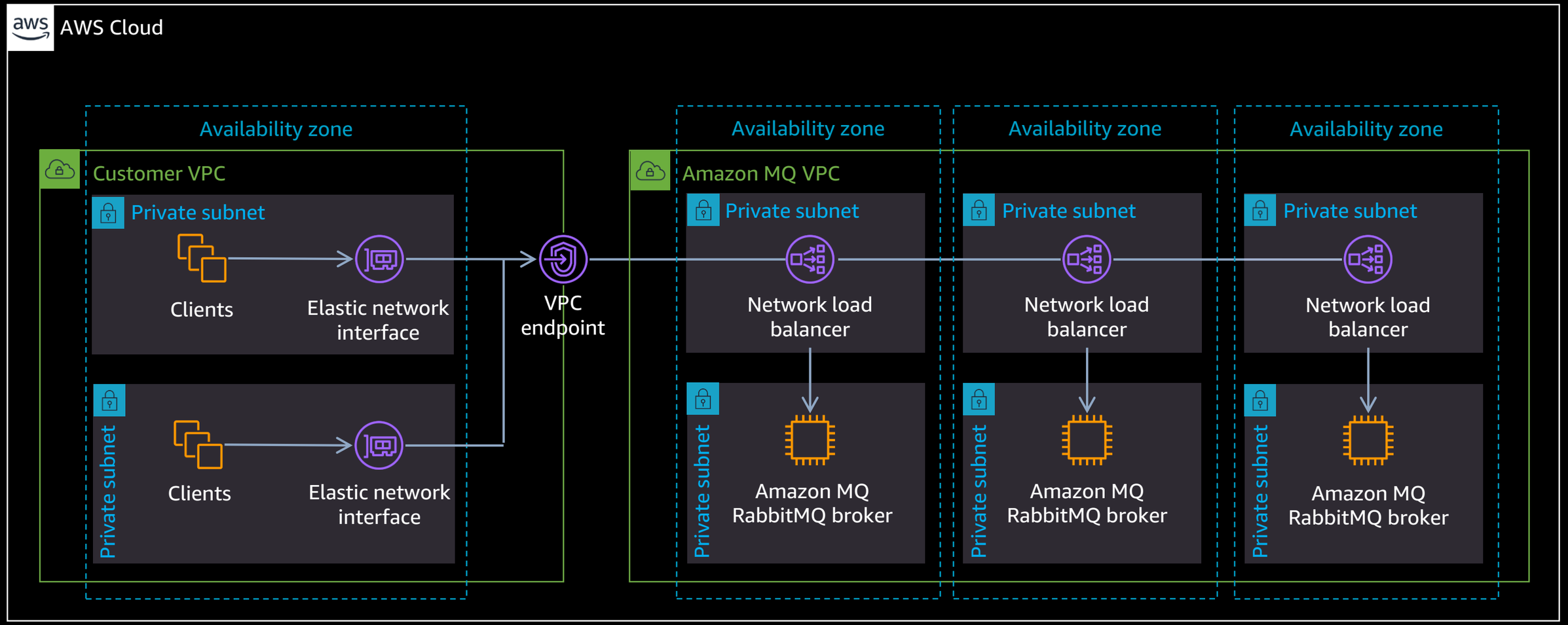
Single-node broker with public accessibility



Single-node broker without public accessibility



Clustered broker without public accessibility



Broker administration



AWS re:Invent

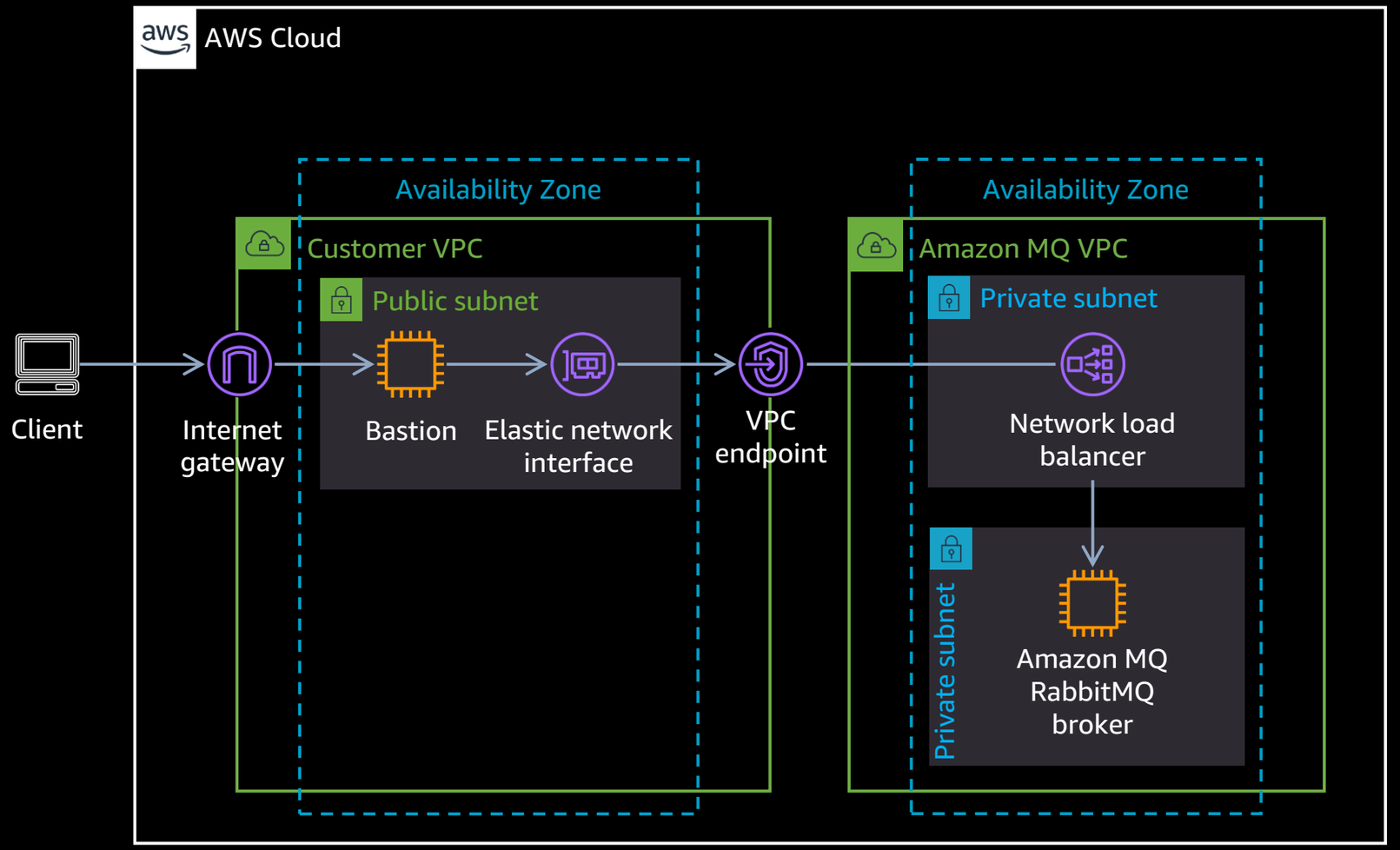
© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Remote administration using rabbitmqadmin

- RabbitMQ REST API client
- Download from https://*BROKER_HOST*/cli/rabbitmqadmin
- Specific to a RabbitMQ version
- `-s` used to connect with ssl
- `-P 443` specifies the port for Amazon MQ RabbitMQ web console

```
% rabbitmqadmin -s -P 443 -H BROKER_HOST -u USERNAME -p PASSWORD list queues
+----------+----------+
| name     | messages |
+----------+----------+
| my-queue | 0        |
+----------+----------+
```

Remote access to the private broker web console



1. Create publically-accessible Amazon Elastic Compute Cloud (Amazon EC2) instance (bastion host)
2. The security group used for the bastion host must allow inbound connections from the client on port 22
3. Install SSH private key on the client
4. Create an SSH tunnel to the bastion host *
5. Install a proxy client (e.g. FoxProxy) on the client
6. Open browser on client to http://localhost:8080

* `% ssh -i ~/.ssh/id_rsa -N -C -q -f -D 8080 ec2-user@BASTION_IP`

Application migration

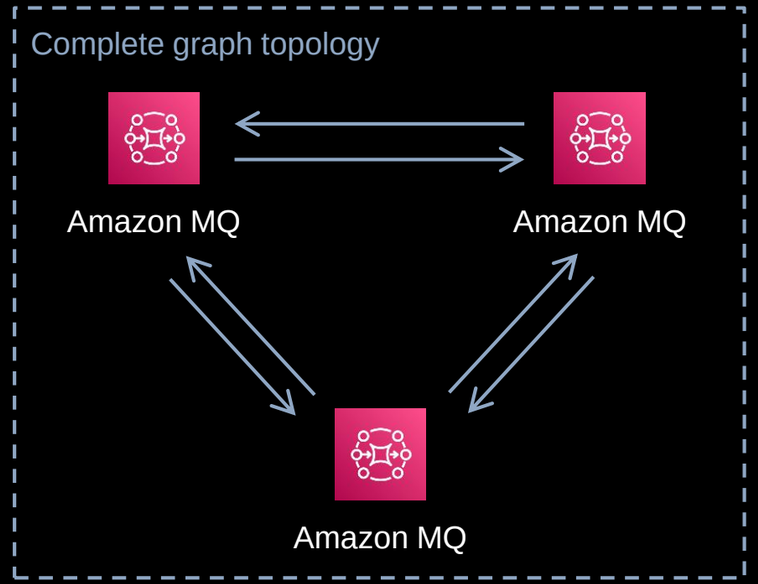


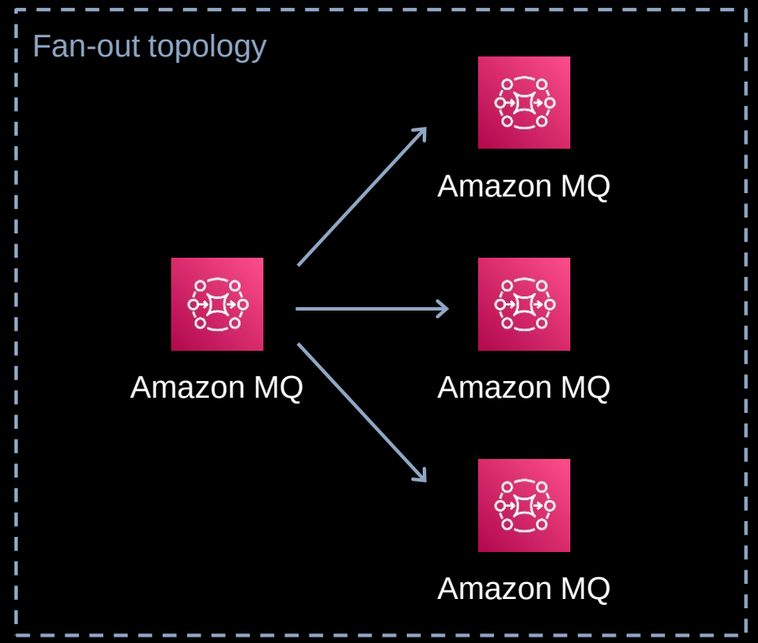
AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Federation plugin

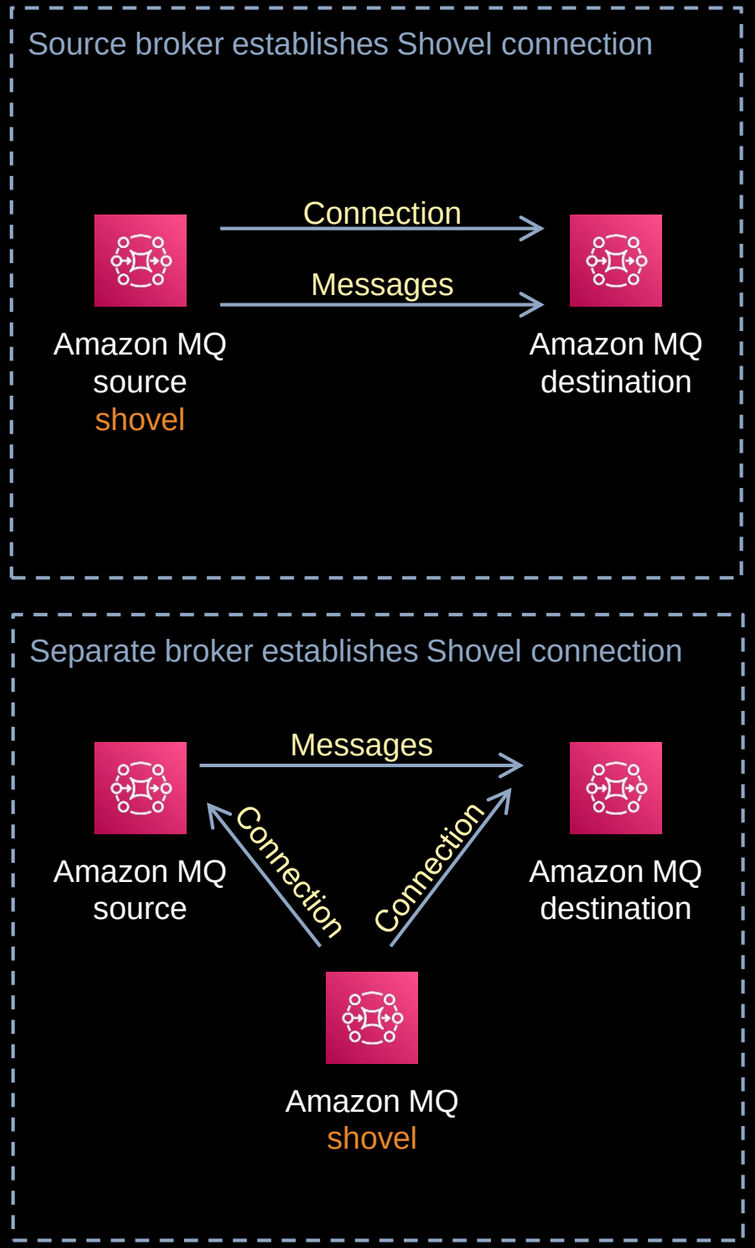
- Downstream brokers initiate one-way, point-to-point connections to upstream brokers
- Federated queues **move** messages from upstream queues to perform load balancing across downstream consumers
- Federated exchanges **copy** messages from upstream exchanges to downstream exchanges, using connected topologies such as pairing, complete graphs, and fan-out; loop protection is provided using a max-hops value
- Because the downstream broker initiates the connection to the upstream broker, the upstream Amazon MQ broker must be public



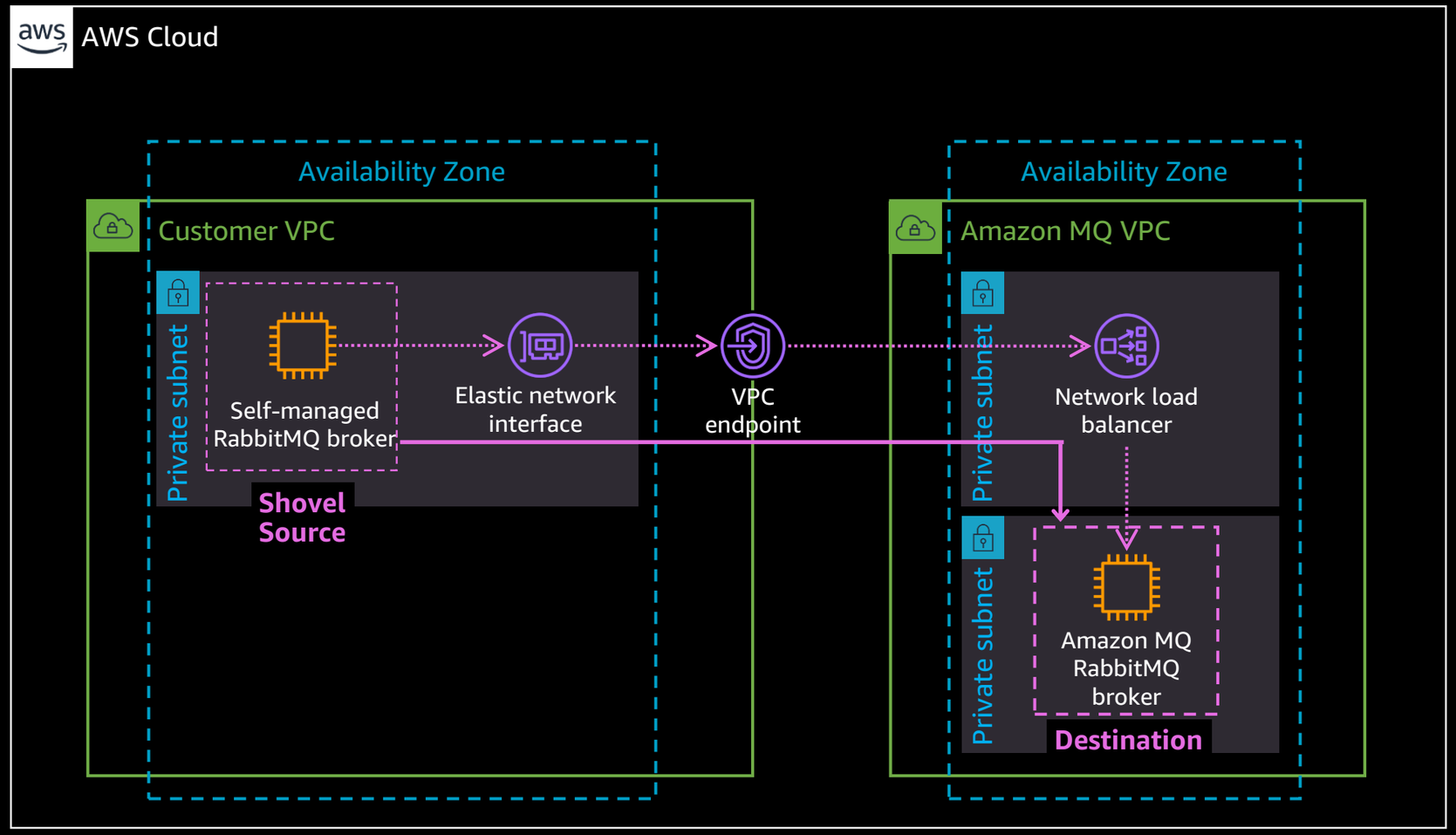


Shovel plugin

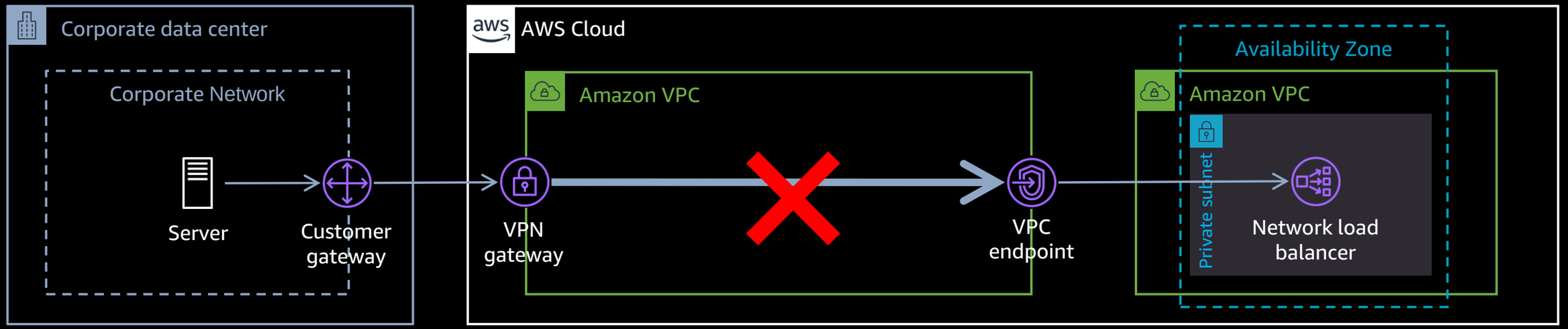
- A shovel acts like a well-written client application, consuming messages from a queue on a source broker and moving them to a queue or exchange on a destination broker
- Shovel connections can be initiated from the source broker, destination broker, or a separate node or cluster
- After a connection is established, queues, exchanges, and bindings can be declared automatically, according to the shovel definition
- Flexible topology support, but no loop detection

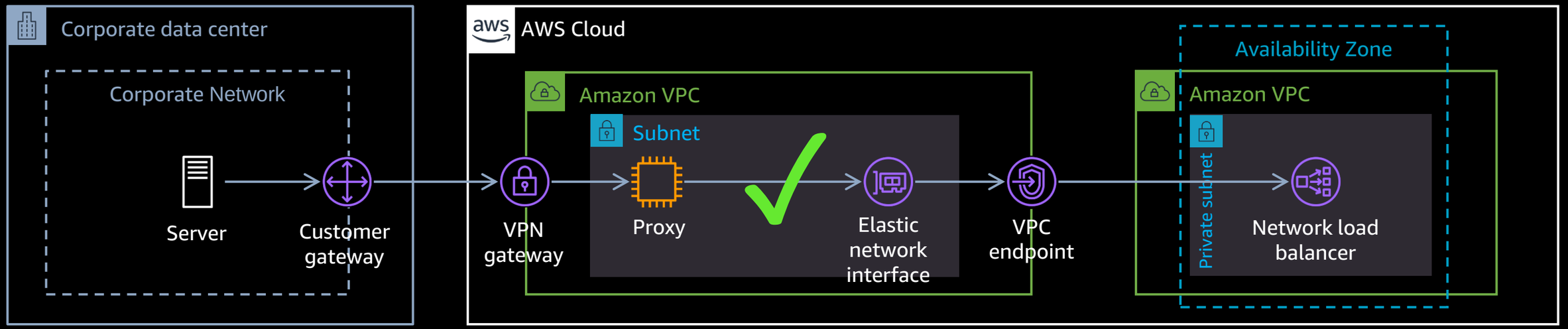


AWS private self-managed broker to private Amazon MQ broker

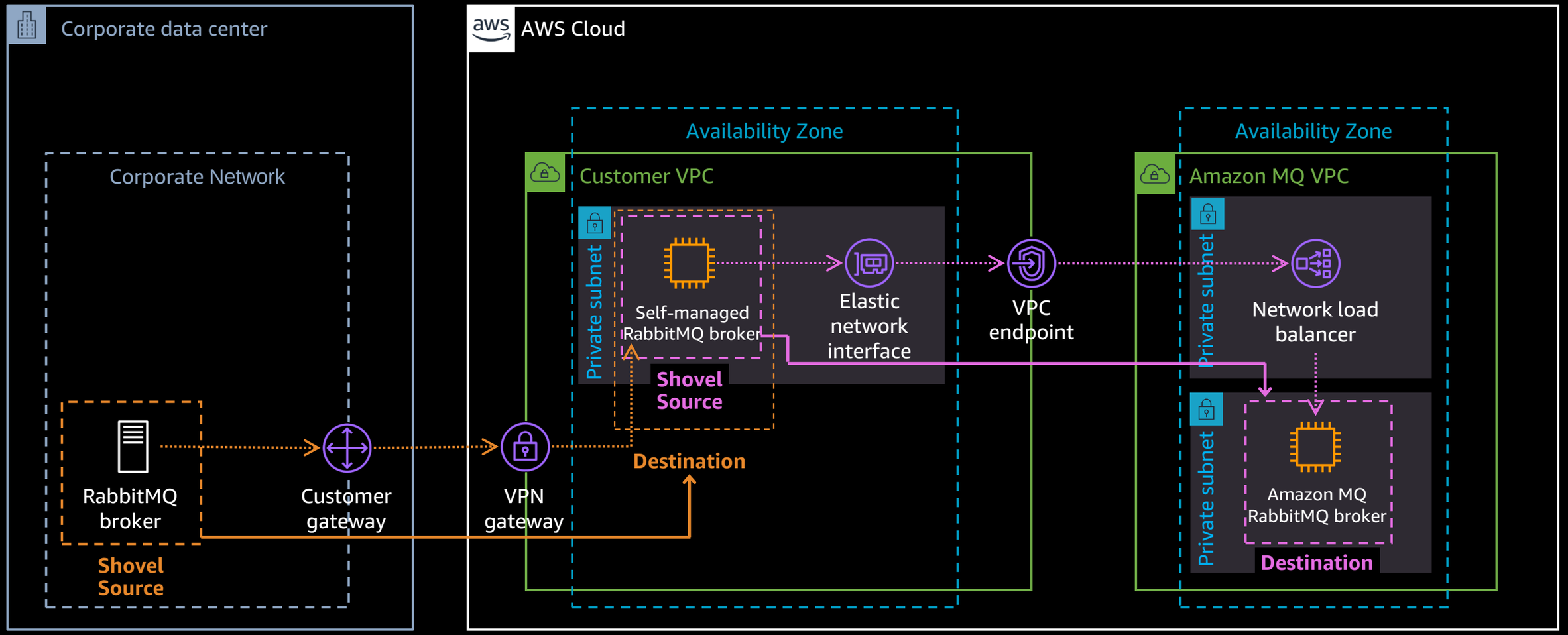


Reminder: VPC transit routing is prohibited

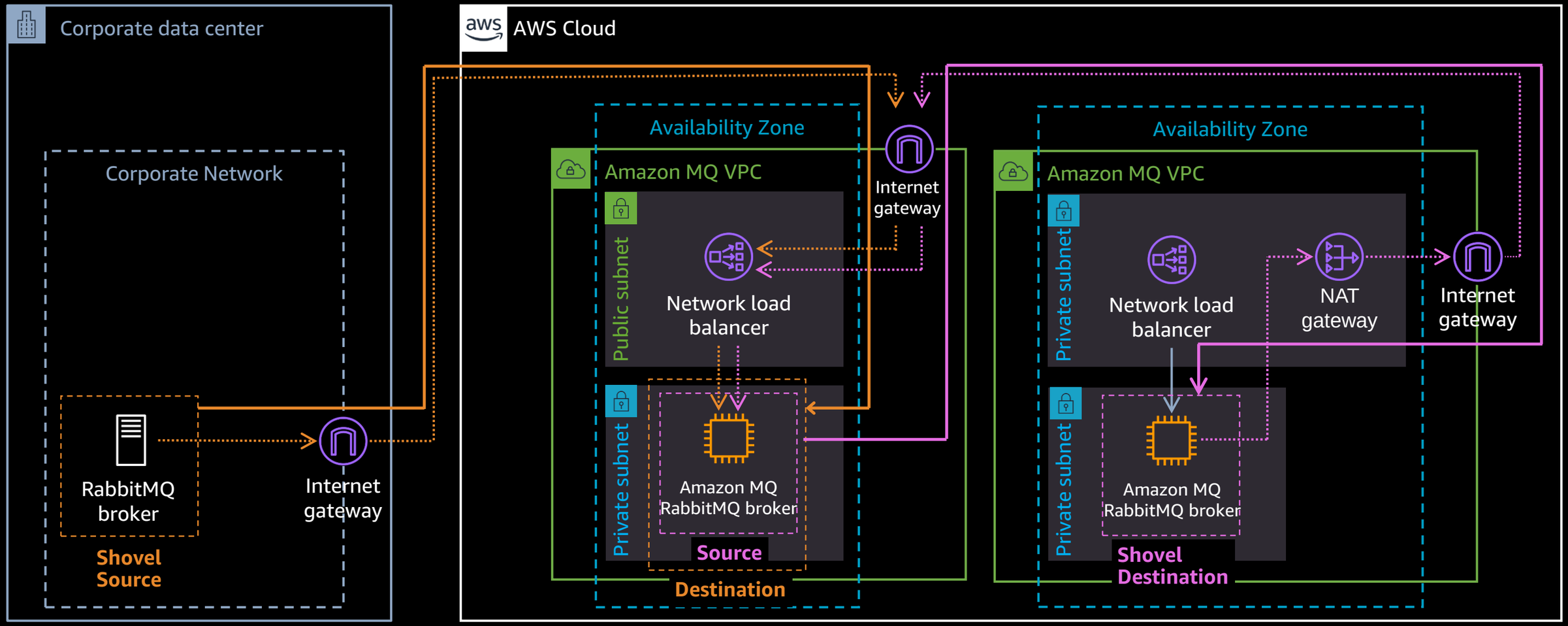




On-prem broker without Internet access to private Amazon MQ broker



On-prem broker with Internet access to private Amazon MQ broker



Thank you!

Sam Dengler

Principal Solutions Architect
@samdengler

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.