Architecting and Preparing Applications for ECS

Justin Menga
FULL STACK TECHNOLOGIST
@jmenga pseudo.co.de

Introduction

Microtrader Architecture

- Microtrader AWS Architecture

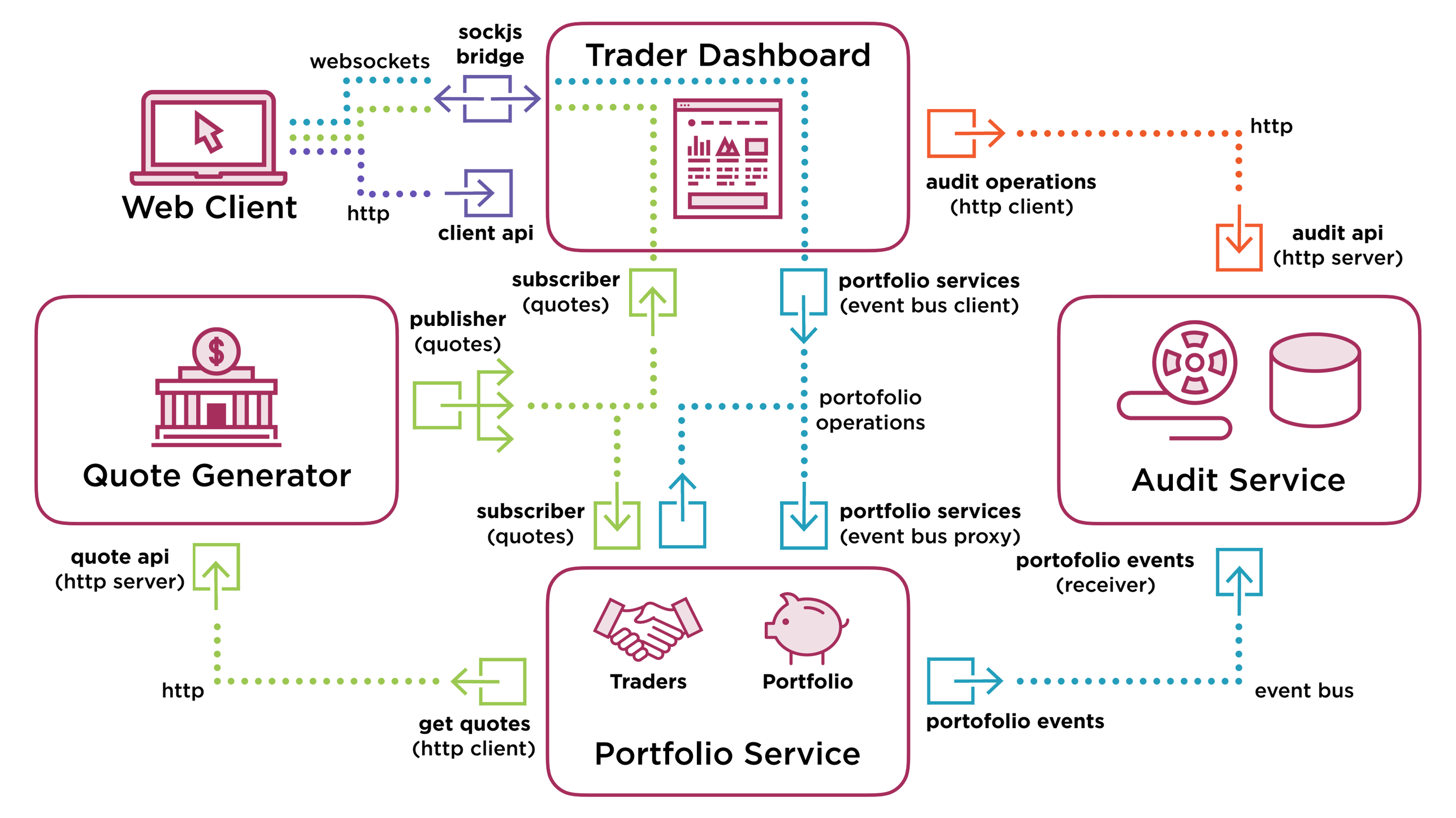
Docker and AWS Challenges

- Cluster Discovery

Dynamic Configuration

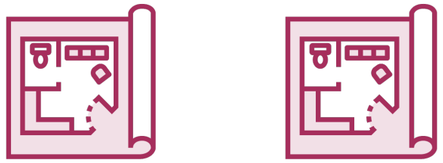
- Configuration files on the fly
- Local Docker vs AWS ECS environment
- Confd

Microtrader AWS Architecture


Trader Dashboard
websockets
sockjs
bridge
Web Client
http
client api
audit operations
(http client)
http
audit api
(http server)
subscriber
(quotes)
portfolio services
(event bus client)
publisher
(quotes)
portofolio
operations
Quote Generator
Audit Service
subscriber
(quotes)
portfolio services
(event bus proxy)
quote api
(http server)
portofolio events
(receiver)
Traders
Portfolio
http
get quotes
(http client)
portofolio events
event bus
Portfolio Service

Microtrader Application Stack

ECS Task Definitions



Route 53 Private DNS

dev-microtrader.dockerproductionaws.org

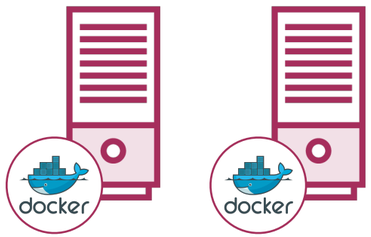
Public Load Balancer
(Internet Facing)

Dashboard Endpoint

Portfolio Service

Audit Service

Dashboard Service



Autoscaling Group

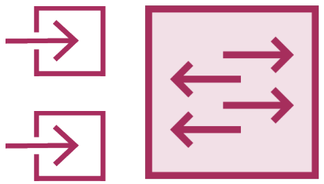
Quote Service

ECS Cluster

Application Load Balancer
(Internal)

Audit Endpoint

Quote Endpoint

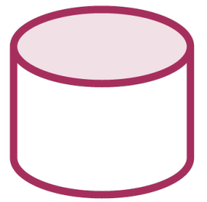


CloudWatch Log Groups

System Logs

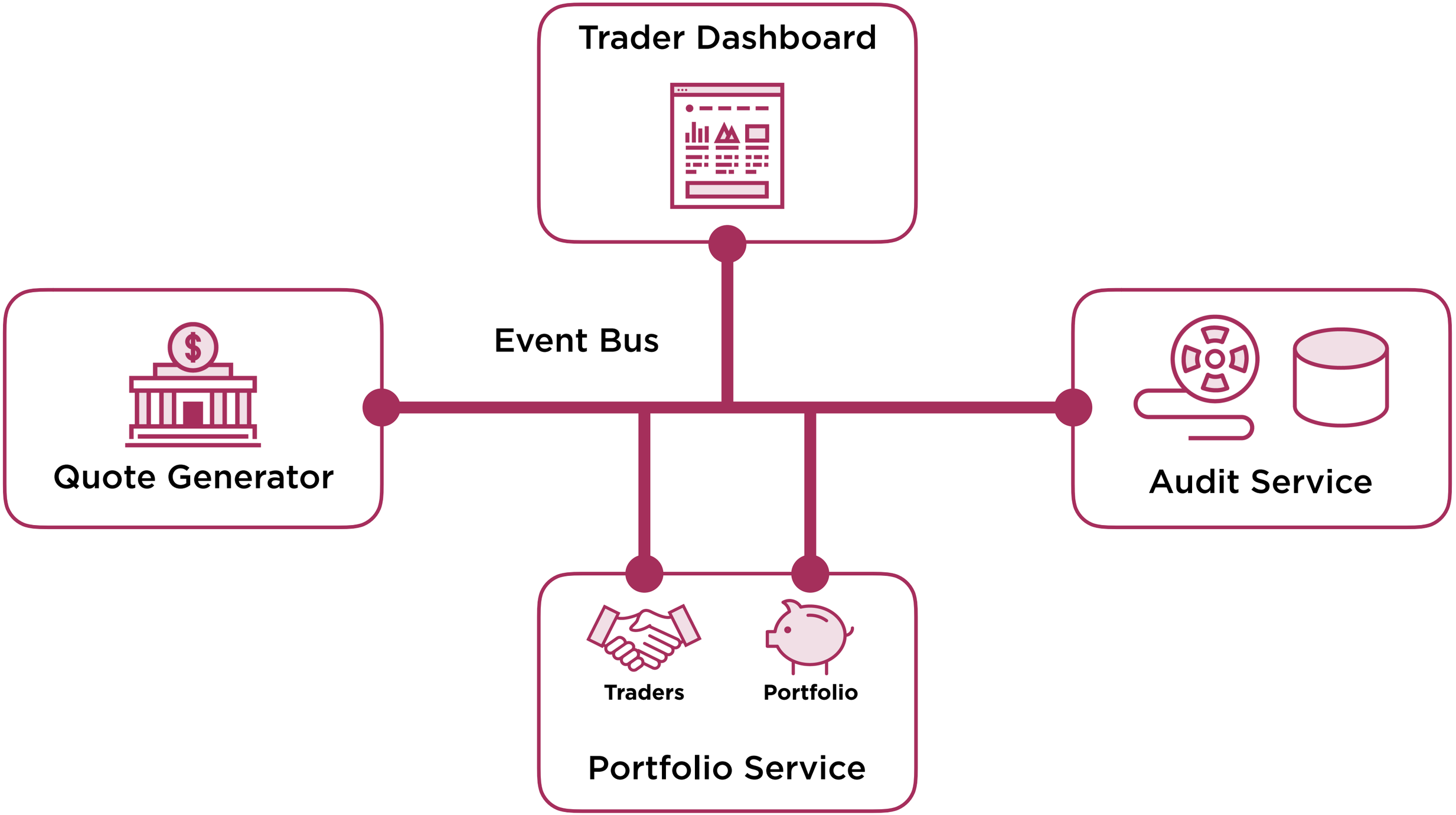
Container Logs

RDS Instance

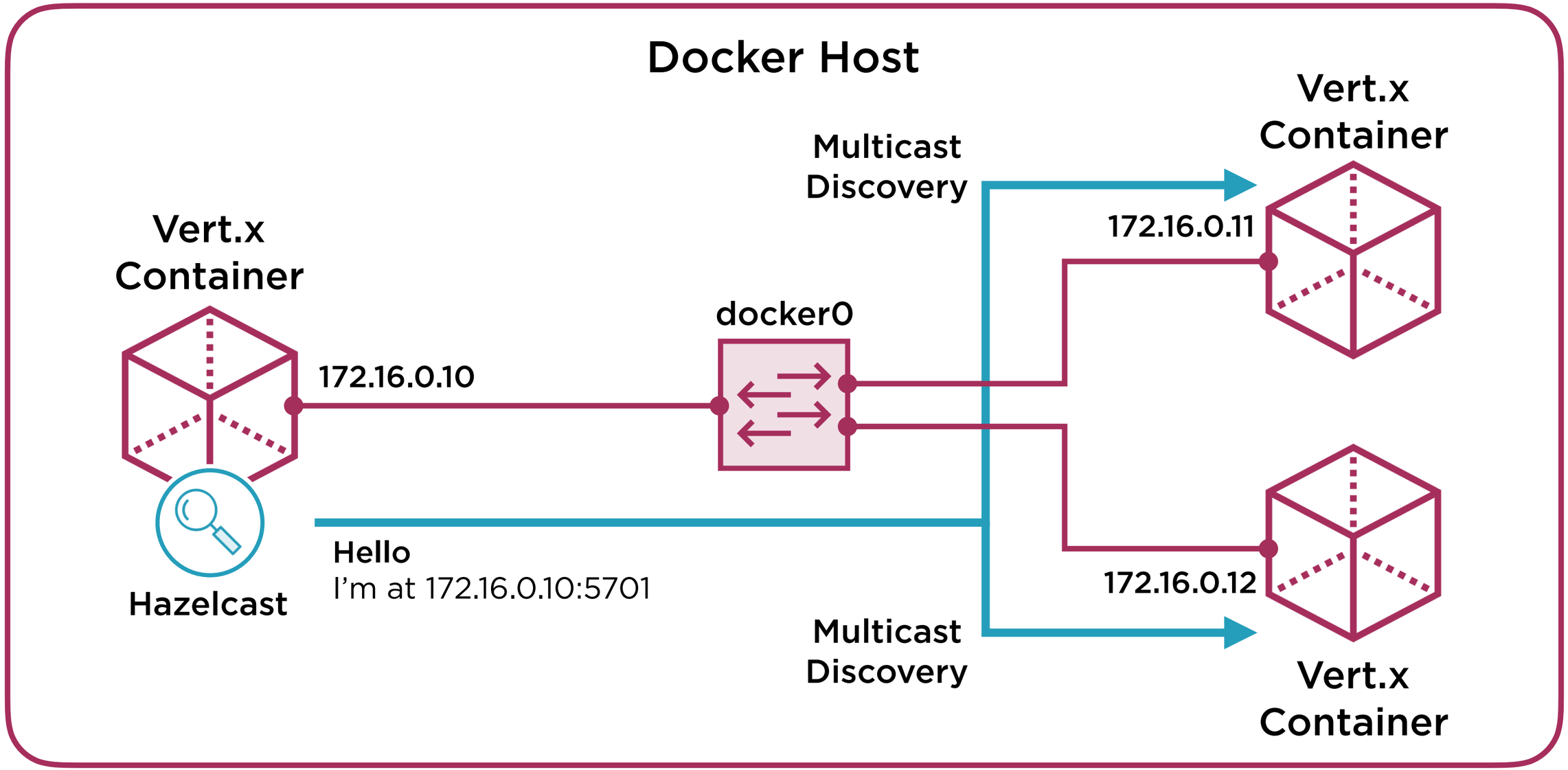


Audit Database

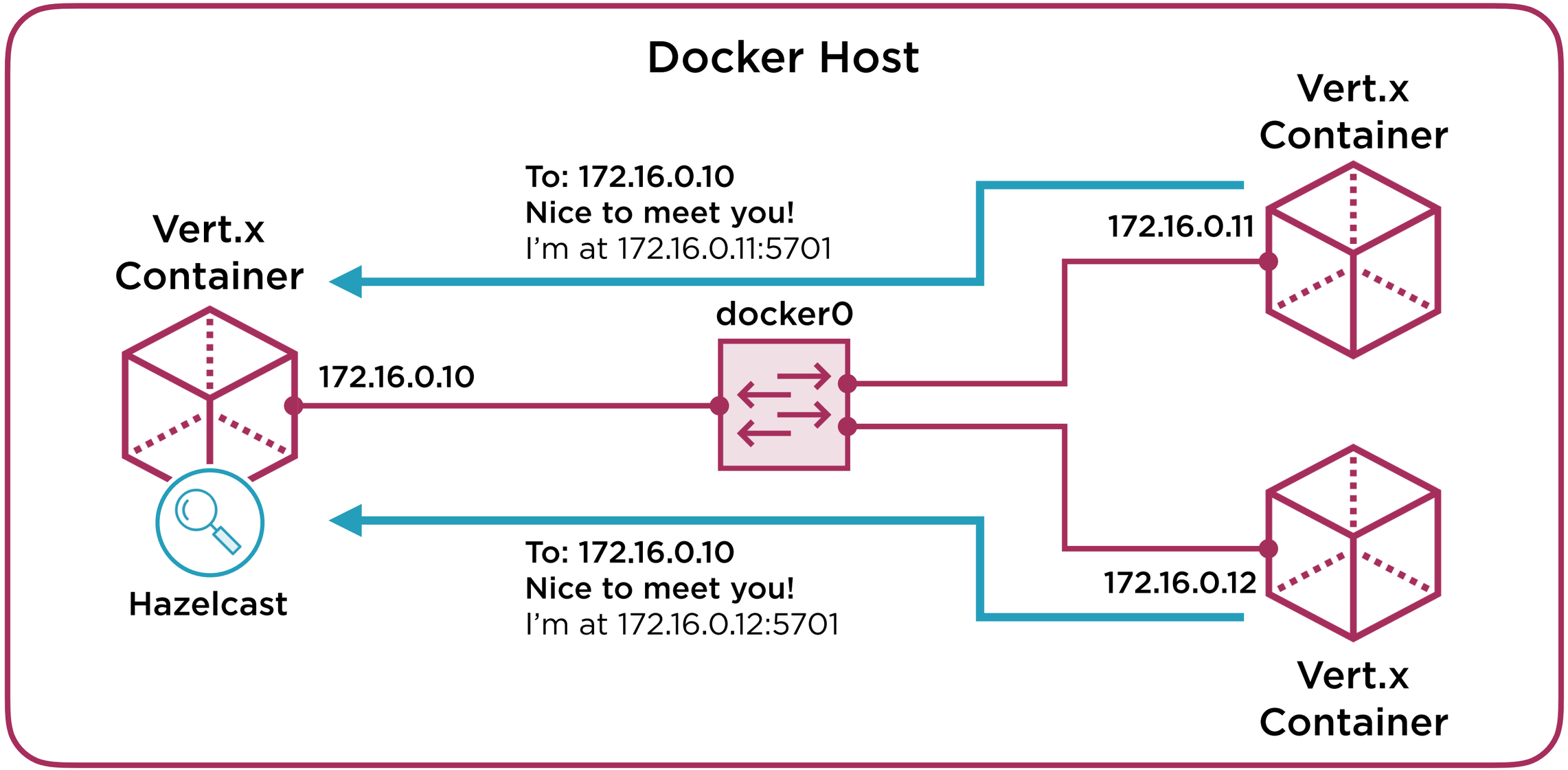
Docker and AWS Challenges

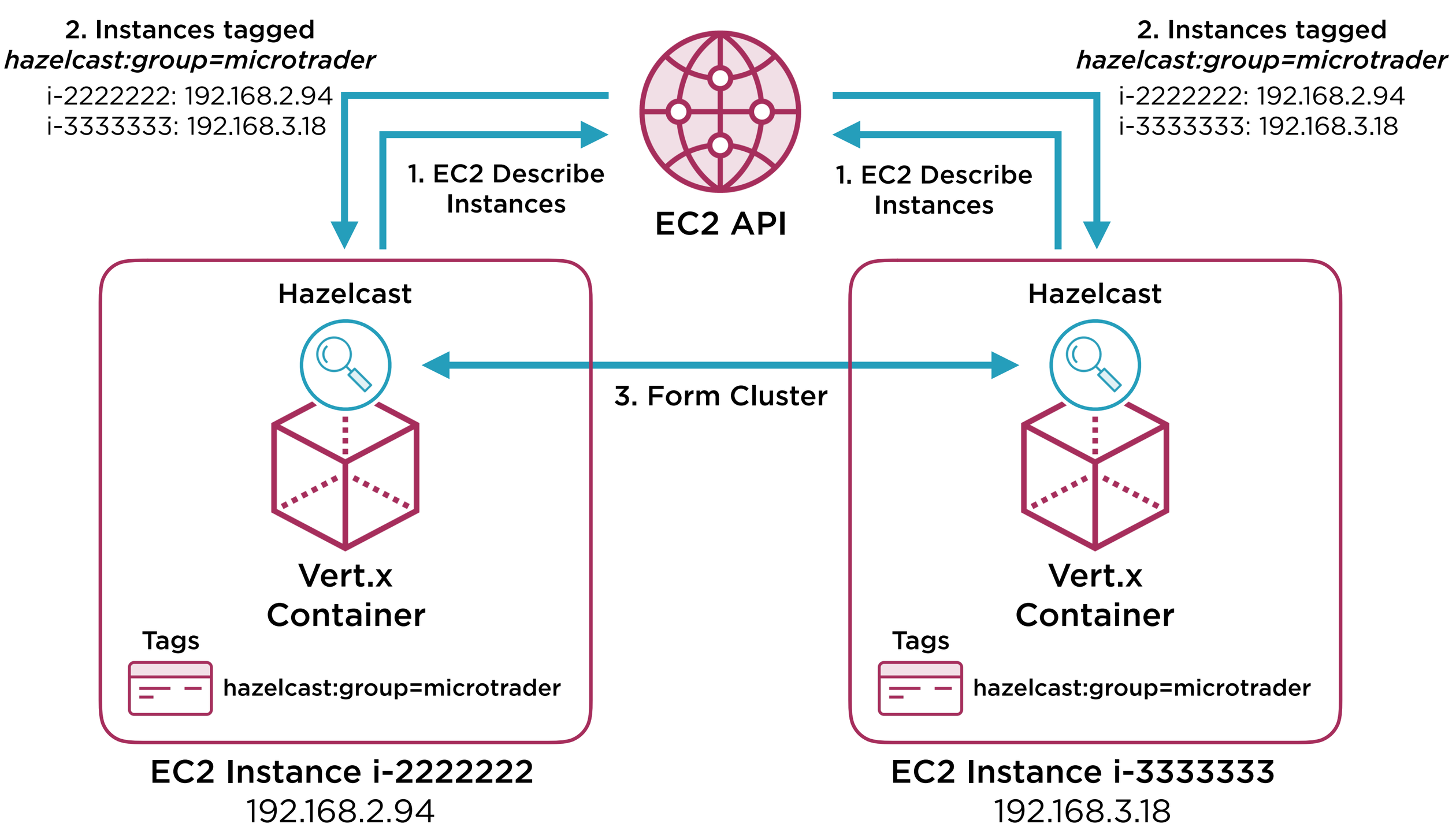

Trader Dashboard
Event Bus
Quote Generator
Audit Service
Traders
Portfolio
Portfolio Service

Vert.x Cluster Discovery

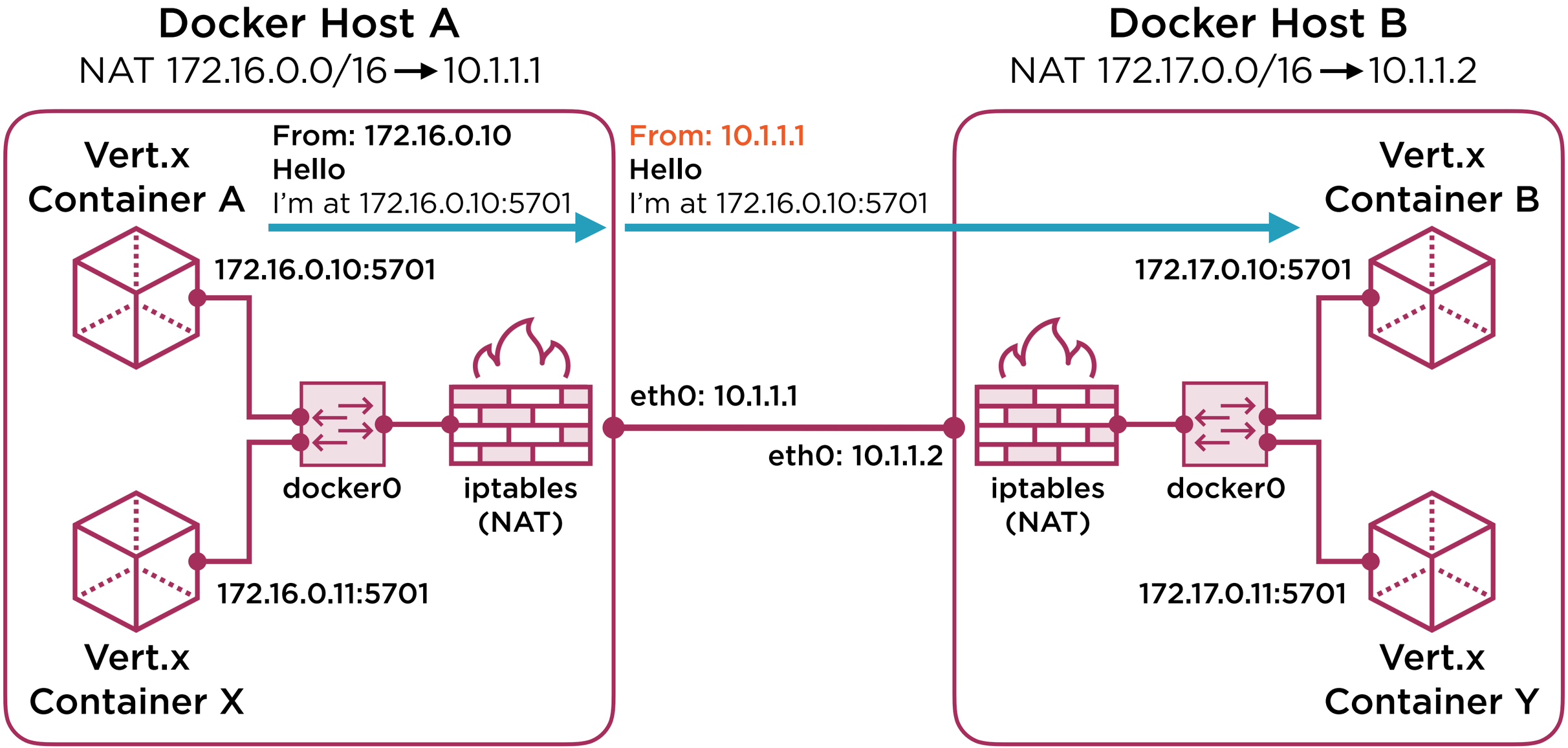


Vert.x Cluster Discovery

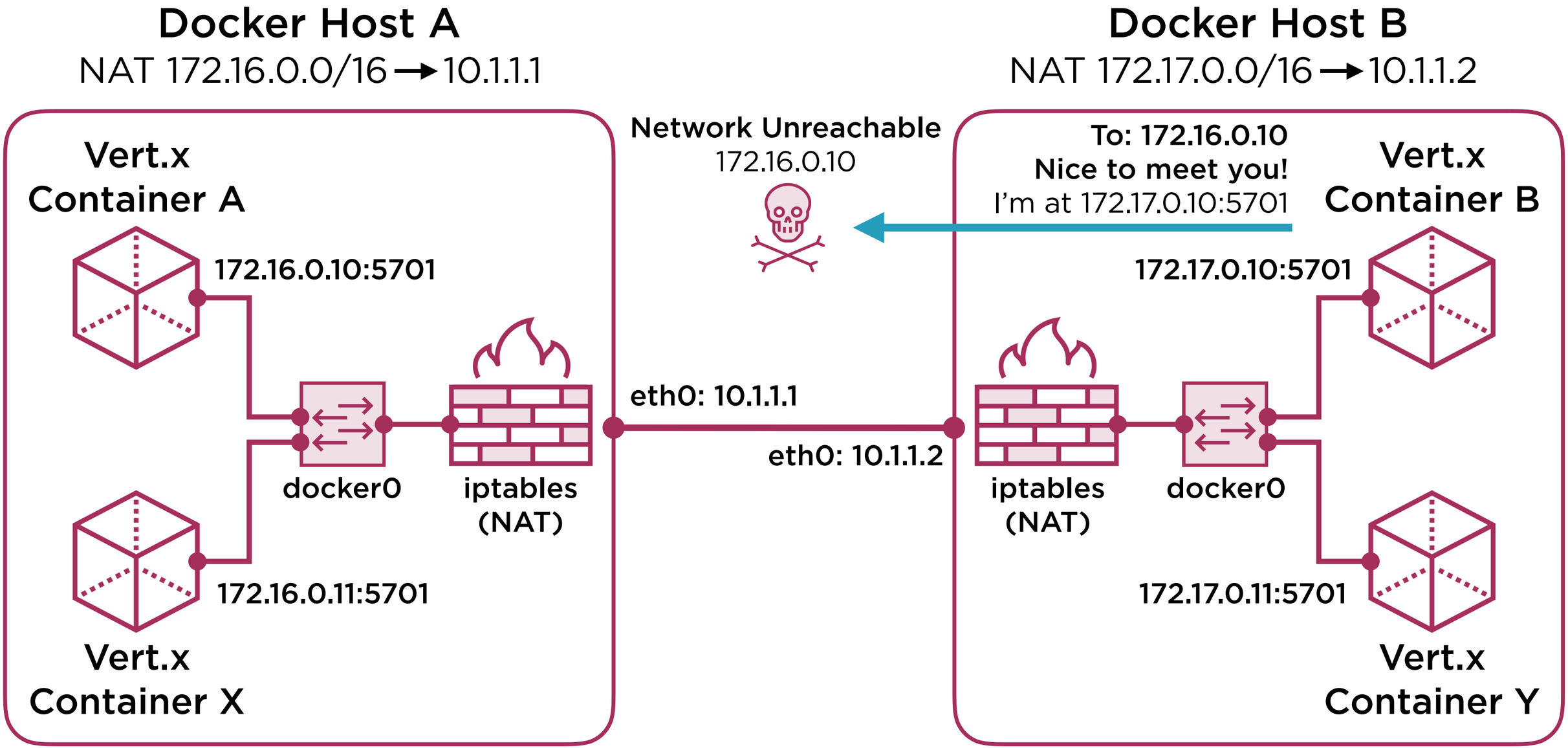



2. Instances tagged
hazelcast:group=microtrader
i-2222222: 192.168.2.94
i-3333333: 192.168.3.18
1. EC2 Describe Instances
EC2 API
2. Instances tagged
hazelcast:group=microtrader
i-2222222: 192.168.2.94
i-3333333: 192.168.3.18
1. EC2 Describe Instances
Hazelcast
Vert.x Container
Tags
hazelcast:group=microtrader
3. Form Cluster
Hazelcast
Vert.x Container
Tags
hazelcast:group=microtrader
EC2 Instance i-2222222
192.168.2.94
EC2 Instance i-3333333
192.168.3.18

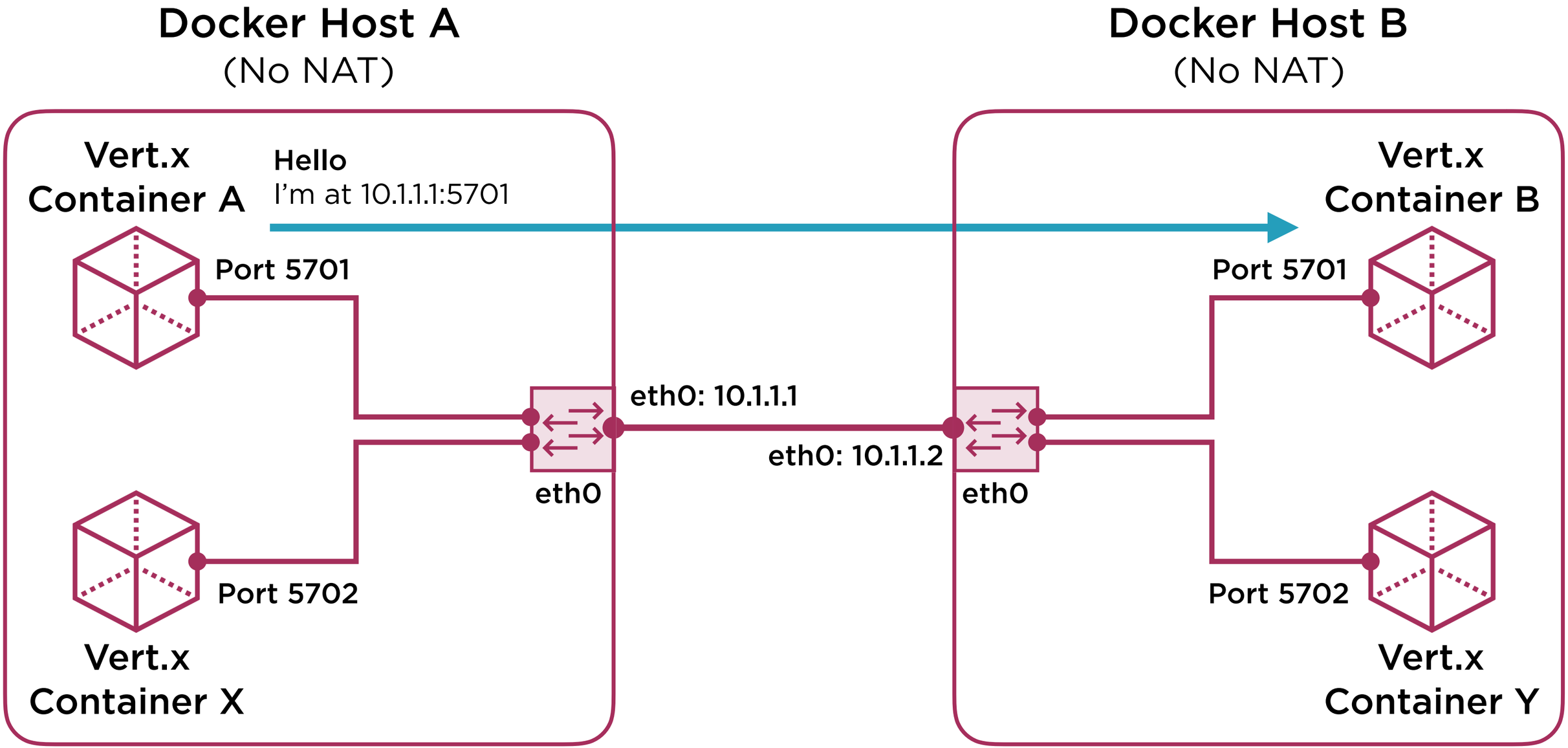
Network Address Translation Challenges



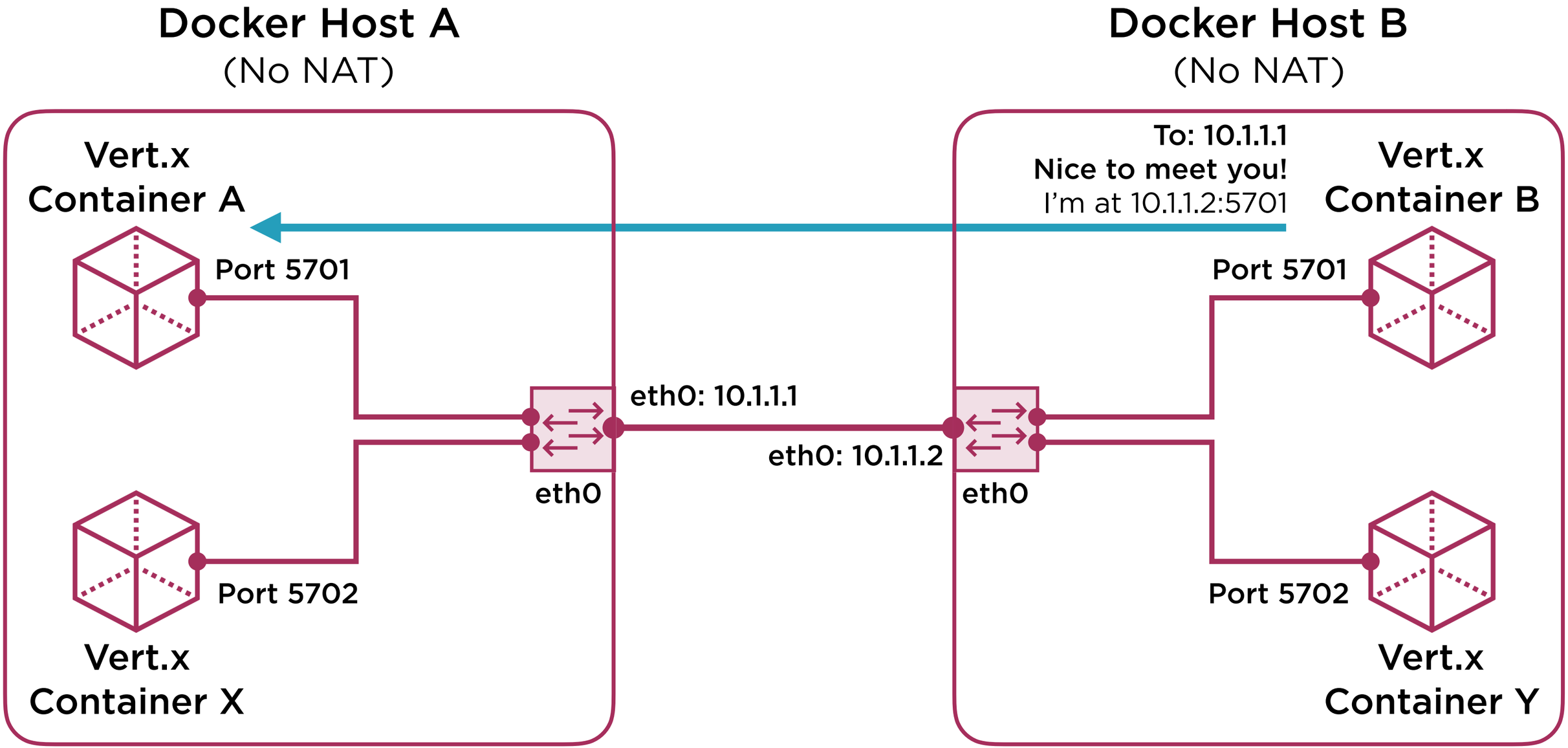
Network Address Translation Challenges

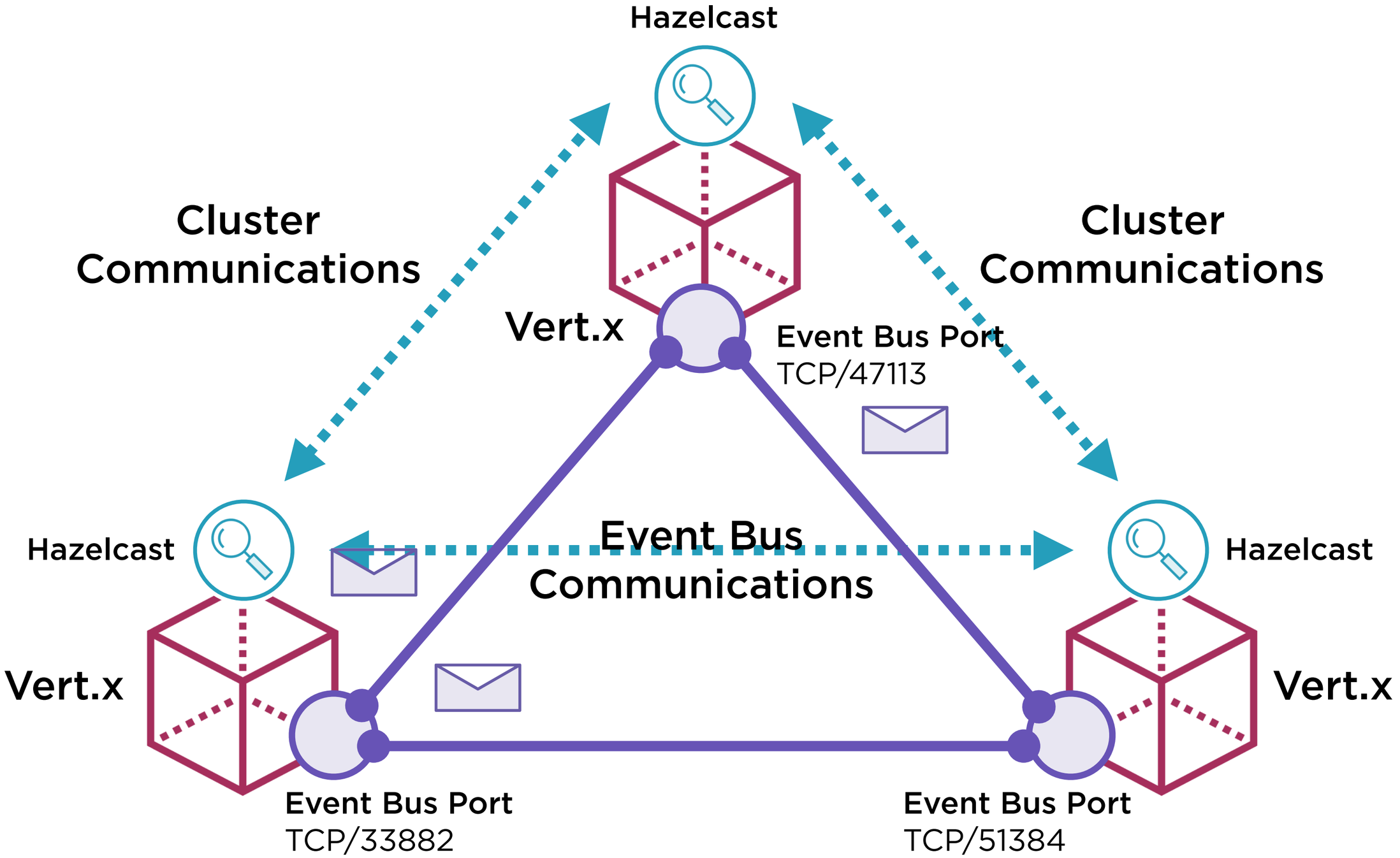


Docker Host Networking

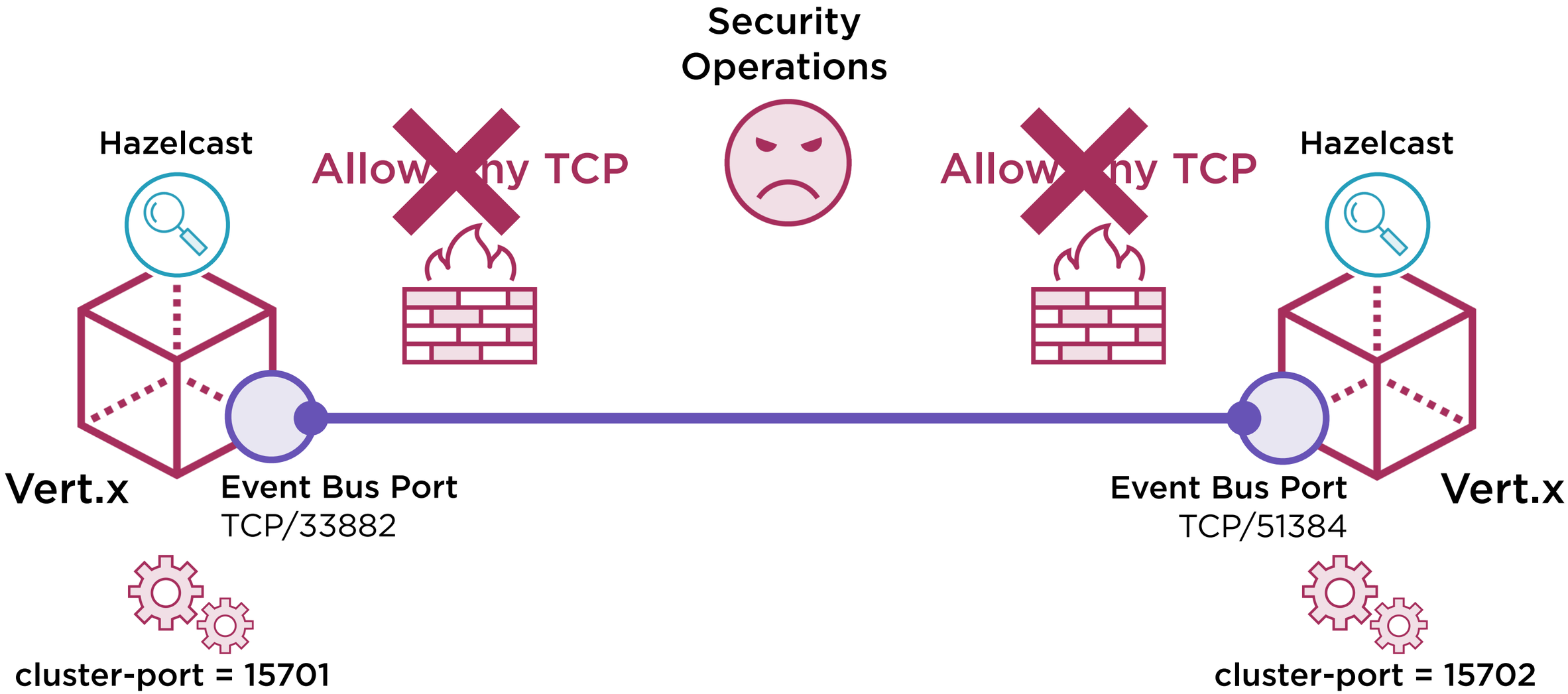


Docker Host Networking

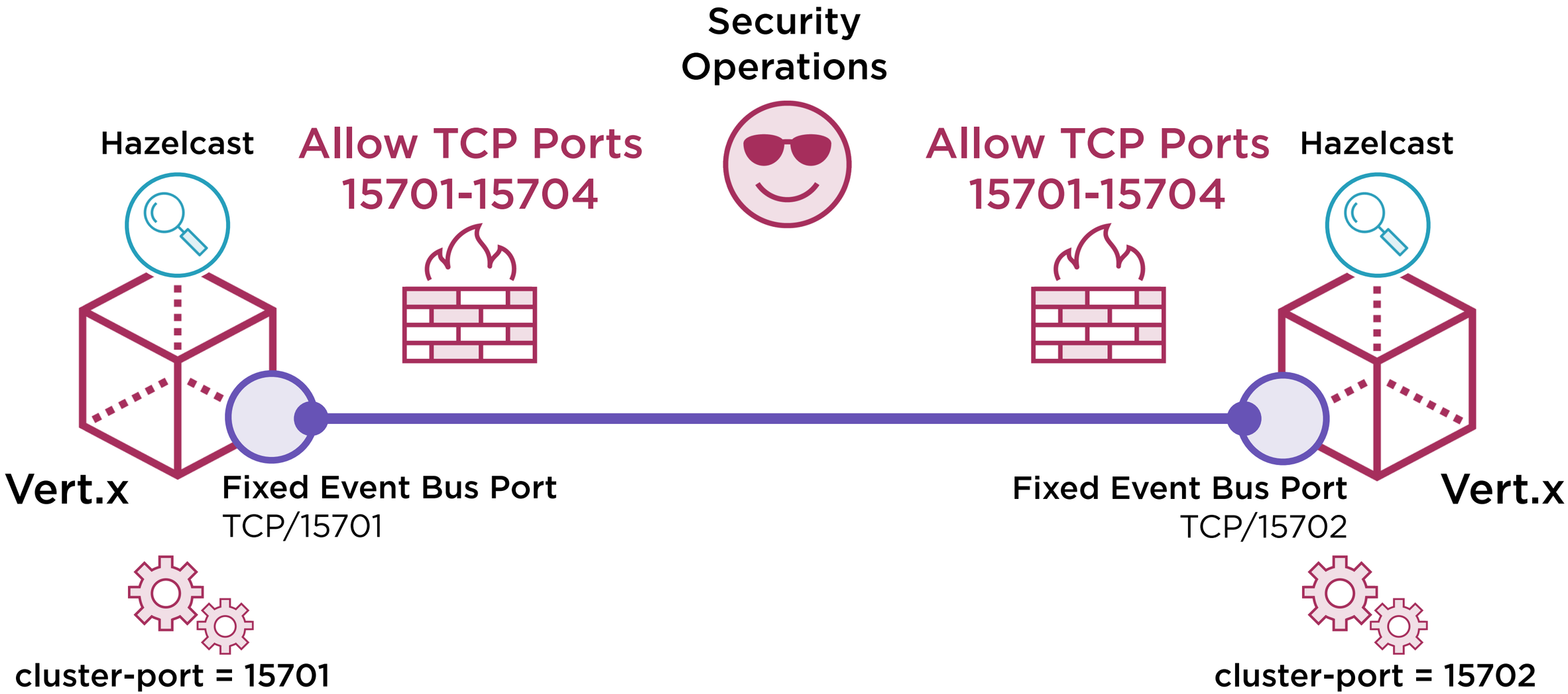



Hazelcast
Cluster
Communications
Cluster
Communications
Vert.x
Event Bus Port
TCP/47113
Hazelcast
Event Bus
Communications
Hazelcast
Vert.x
Vert.x
Event Bus Port
TCP/33882
Event Bus Port
TCP/51384

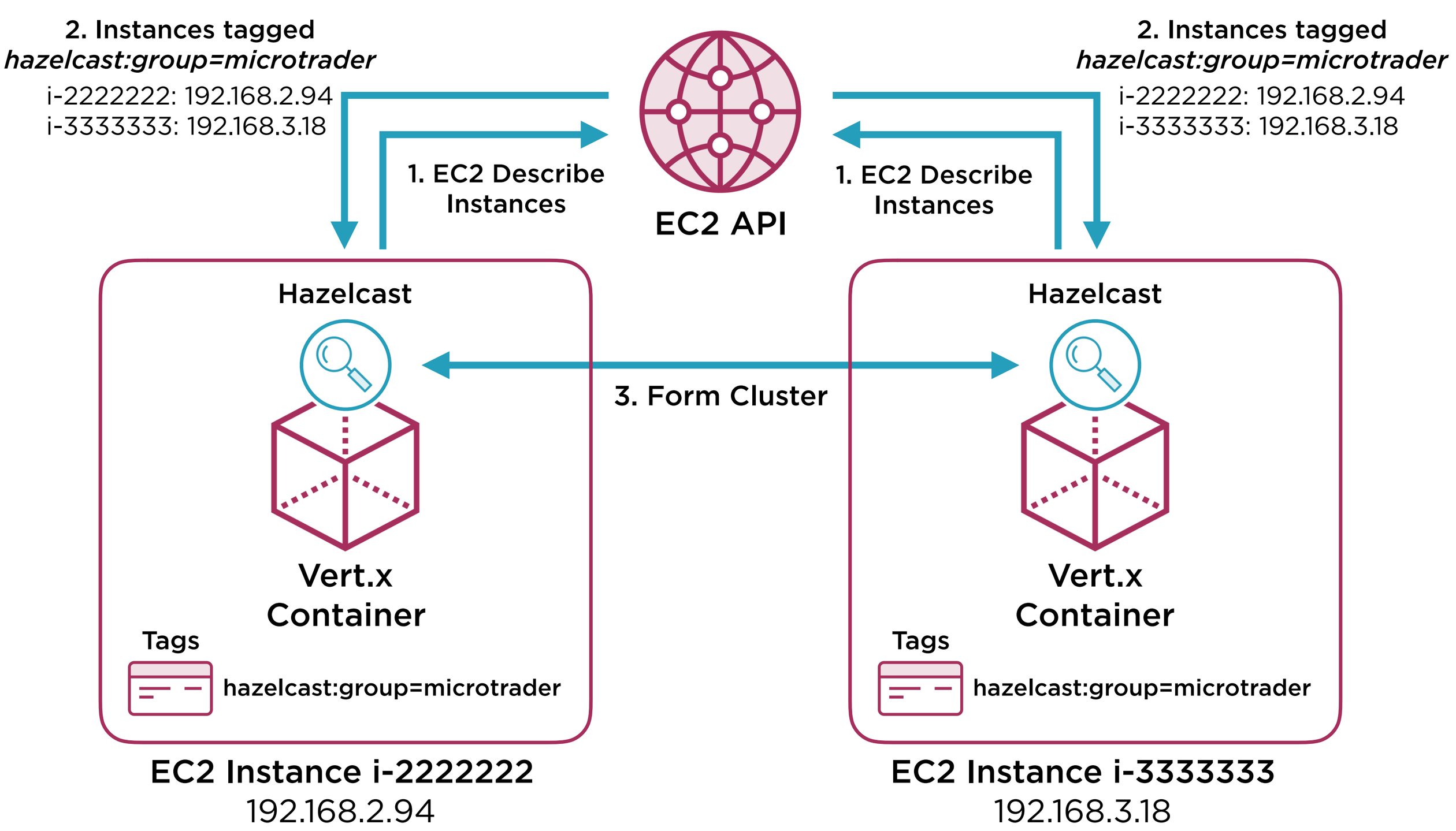
Securing Event Bus Communications



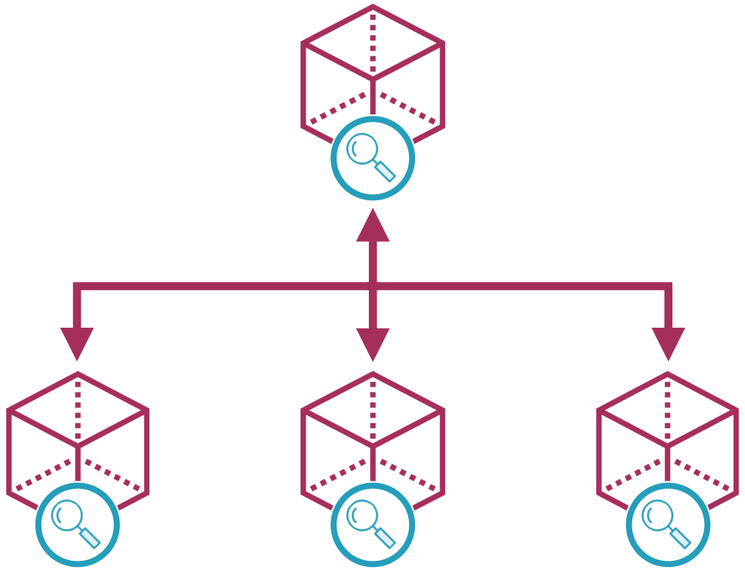
Securing Event Bus Communications



Controlling Cluster Auto Discovery


2. Instances tagged
hazelcast:group=microtrader
i-2222222: 192.168.2.94
i-3333333: 192.168.3.18
1. EC2 Describe Instances
EC2 API
2. Instances tagged
hazelcast:group=microtrader
i-2222222: 192.168.2.94
i-3333333: 192.168.3.18
1. EC2 Describe Instances
Hazelcast
Vert.x Container
Tags
hazelcast:group=microtrader
3. Form Cluster
Hazelcast
Vert.x Container
Tags
hazelcast:group=microtrader
EC2 Instance i-2222222
192.168.2.94
EC2 Instance i-3333333
192.168.3.18

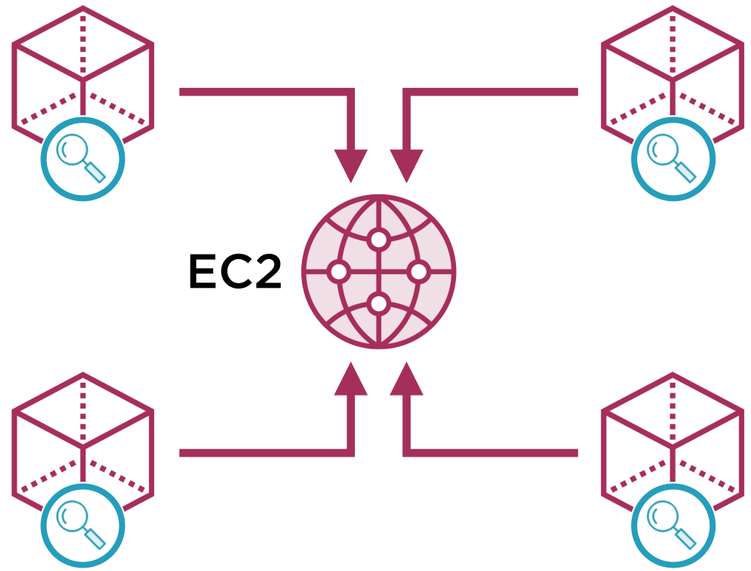
Local Docker Environment
Multicast Discovery



cluster.xml

```
…
<join>
  <multicast enabled="true"></multicast>
  …
</join>
…
```

AWS Environment
EC2 Discovery

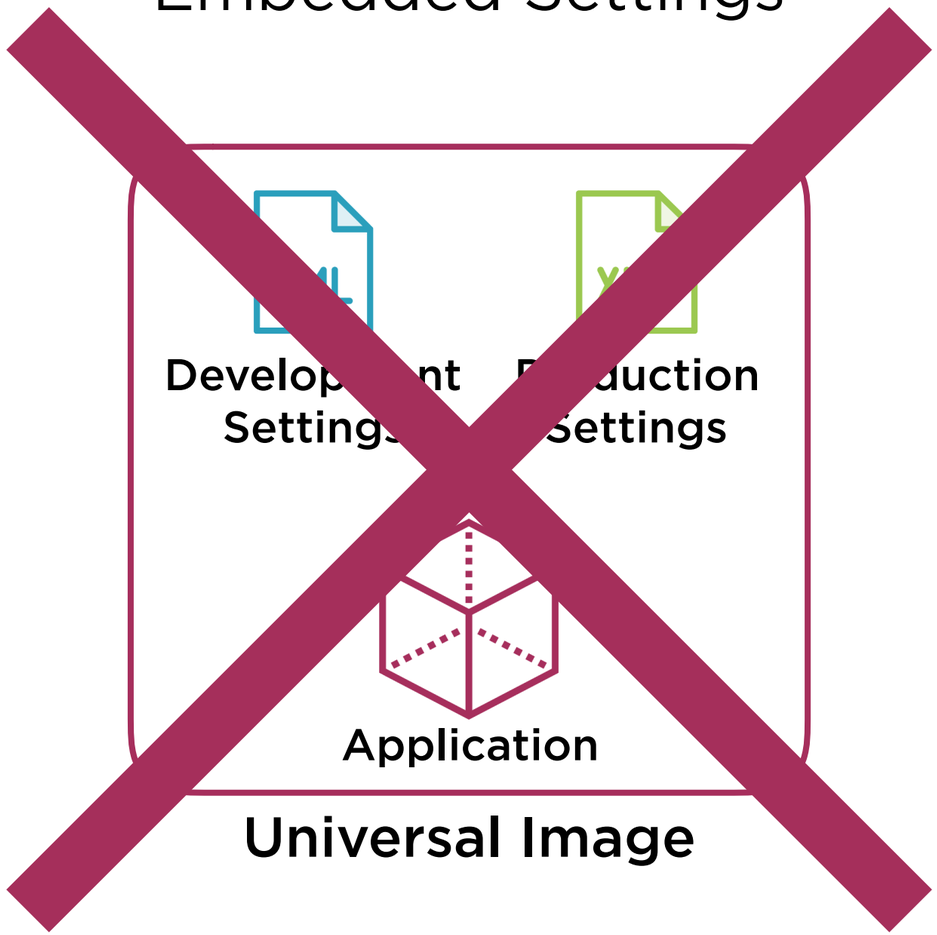


cluster.xml

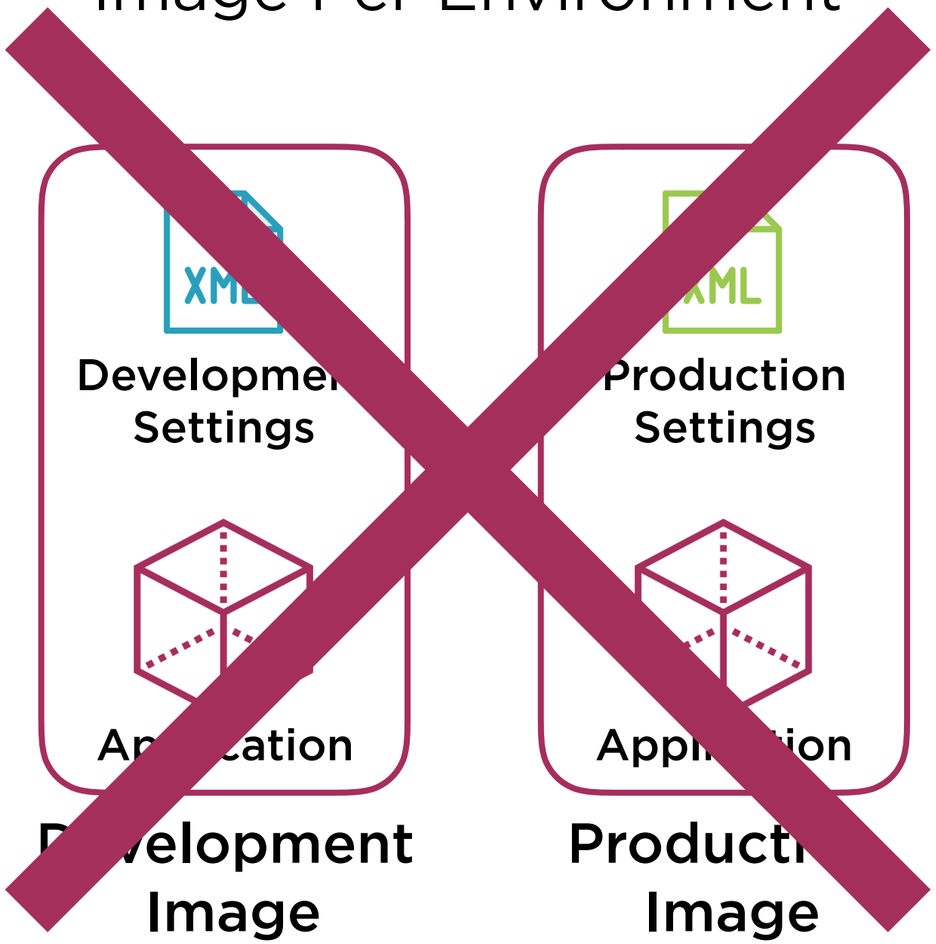
```
…
<join>
  <multicast enabled="false"></multicast>
  <aws enabled="true">…</aws>
</join>
…
```

File-based Environment Configurations

Option 1
Embedded Settings

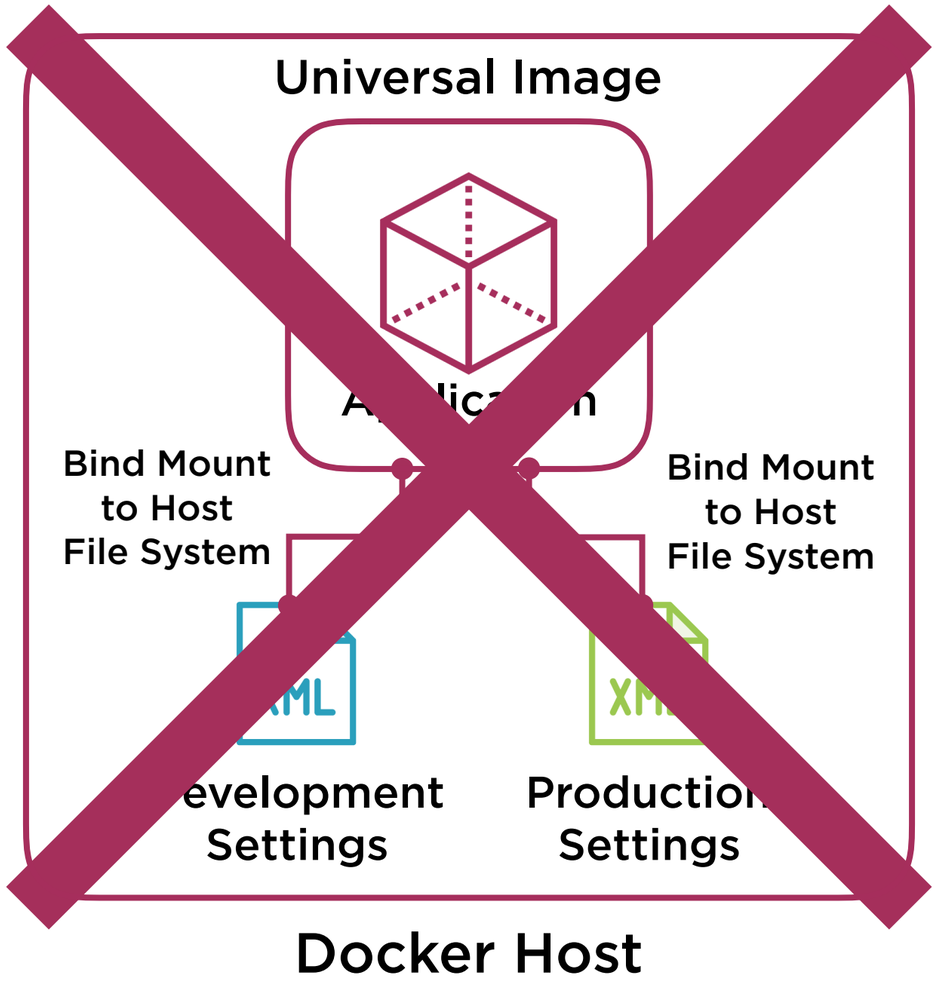


Option 2
Image Per Environment

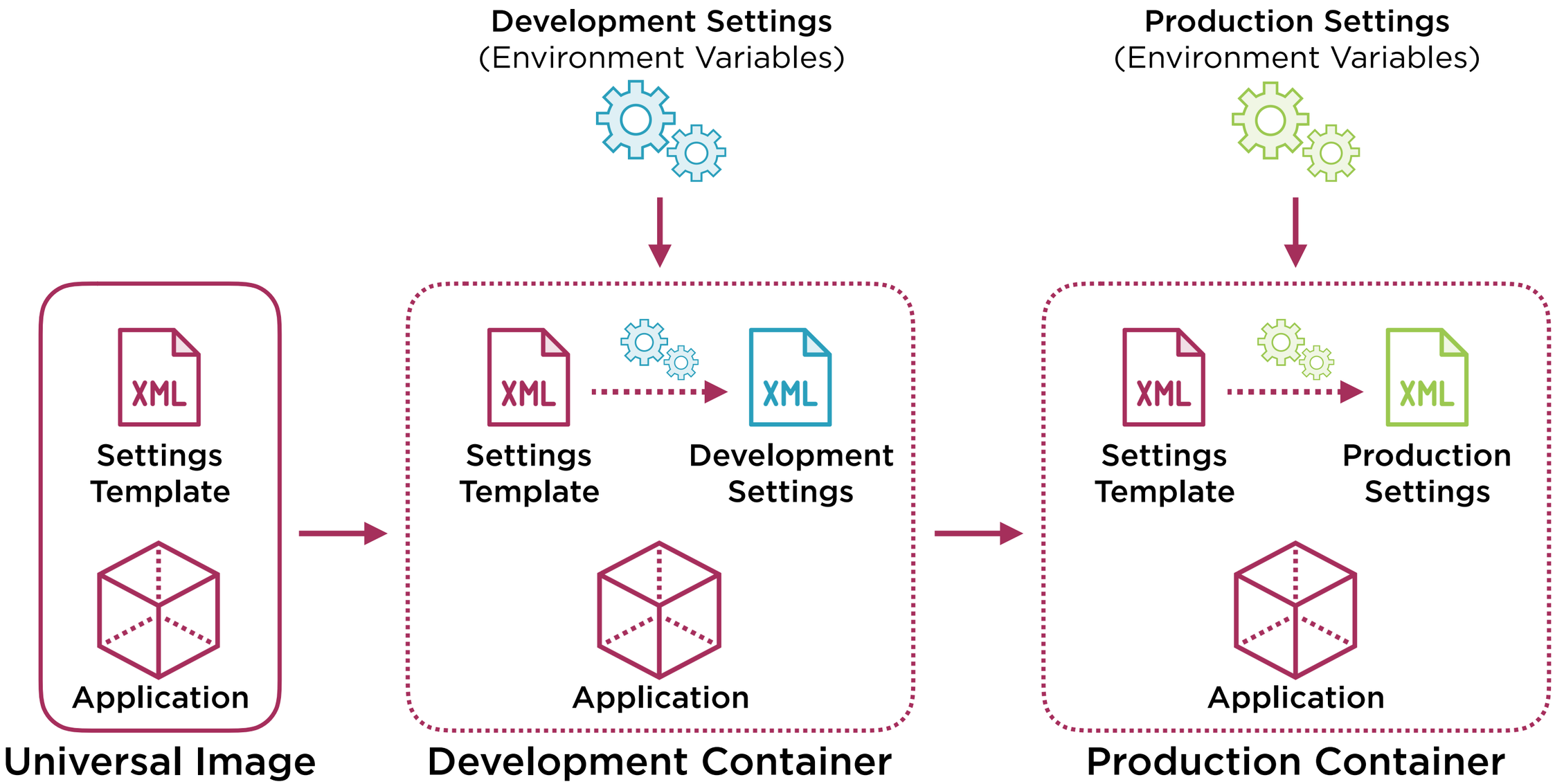


File-based Environment Configurations

Option 3
Bind Mounts



Configuration Files on the Fly



```
#!/bin/bash

tee cluster.xml << EOF > /dev/null
…
…
<join>
<aws>
   <region>${AWS_REGION}</region>
   <tag-key>${TAG_KEY}</tag-key>
   …
   …
</aws>
</join>
EOF
```

Configuration on the Fly using Bash Scripts

Easy for simple use cases

Can grow complex very quickly

- Conditional logic is hard

Customized for each use case

- Difficult to reuse

Confd Template Generation

cluster.xml.tmpl
(Go Template)

```
…
<join>
  {{ if eq (getv "/aws/enabled" "false")) "true" }}
  <aws enabled="true">
    <region>{{getv "/aws/region"}}</region>
    <iam-role>{{getv "/aws/iam/role"}}</iam-role>
    <tag-key>{{getv "/tag/key"}}</tag-key>
    <tag-value>{{getv "/tag/value"}}</tag-value>
  </aws>
  {{ end }}
</join>
…
```

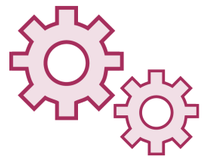
Template Variables

```
/aws/enabled
/aws/region
/aws/iam/role
/tag/key
/tag/value
```

Environment Settings
(Environment Variables)

```
AWS_ENABLED=true
AWS_REGION=us-west-2
AWS_IAM_ROLE=DEFAULT
TAG_KEY=hazelcast:group
TAG_VALUE=microtrader
```

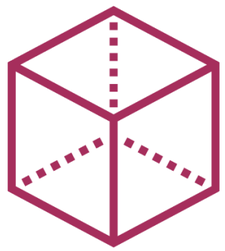
confd



Generate Template

/app/conf/cluster.xml

```
…
<join>
  <aws enabled="true">
    <region>us-west-2</region>
    <iam-role>DEFAULT</iam-role>
    <tag-key>hazelcast:group</tag-key>
    <tag-value>microtrader</tag-value>
  </aws>
</join>
…
```



Application

Confd Template Generation

cluster.xml.tmpl
(Go Template)

```
…
<join>
  {{ if eq (getv "/aws/enabled" "false")) "true" }}
  <aws enabled="true">
    <region>{{getv "/aws/region"}}</region>
    <iam-role>{{getv "/aws/iam/role"}}</iam-role>
    <tag-key>{{getv "/tag/key"}}</tag-key>
    <tag-value>{{getv "/tag/value"}}</tag-value>
  </aws>
  {{ end }}
</join>
…
```

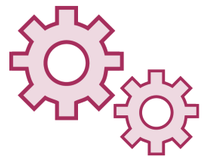
Template Variables

```
/aws/enabled
/aws/region
/aws/iam/role
/tag/key
/tag/value
```

Environment Settings
(consul, etcd and more)

```
/aws/enabled: true
/aws/region: us-west-2
/aws/iam/role: DEFAULT
/tag/key: hazelcast:group
/tag/value: microtrader
```

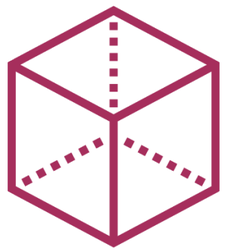
confd



Generate Template

/app/conf/cluster.xml

```
…
<join>
  <aws enabled="true">
    <region>us-west-2</region>
    <iam-role>DEFAULT</iam-role>
    <tag-key>hazelcast:group</tag-key>
    <tag-value>microtrader</tag-value>
  </aws>
</join>
…
```



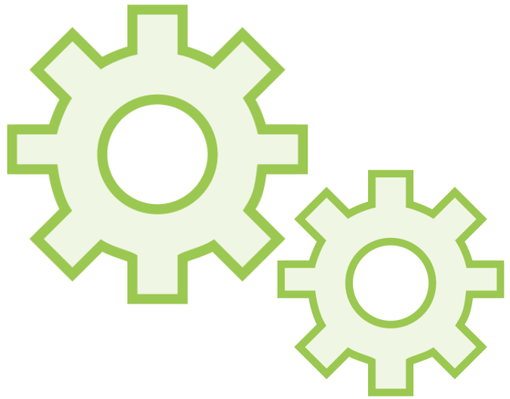
Application

Configuring Cluster Auto Discovery using Confd

Configuring Cluster Auto Discovery using Confd

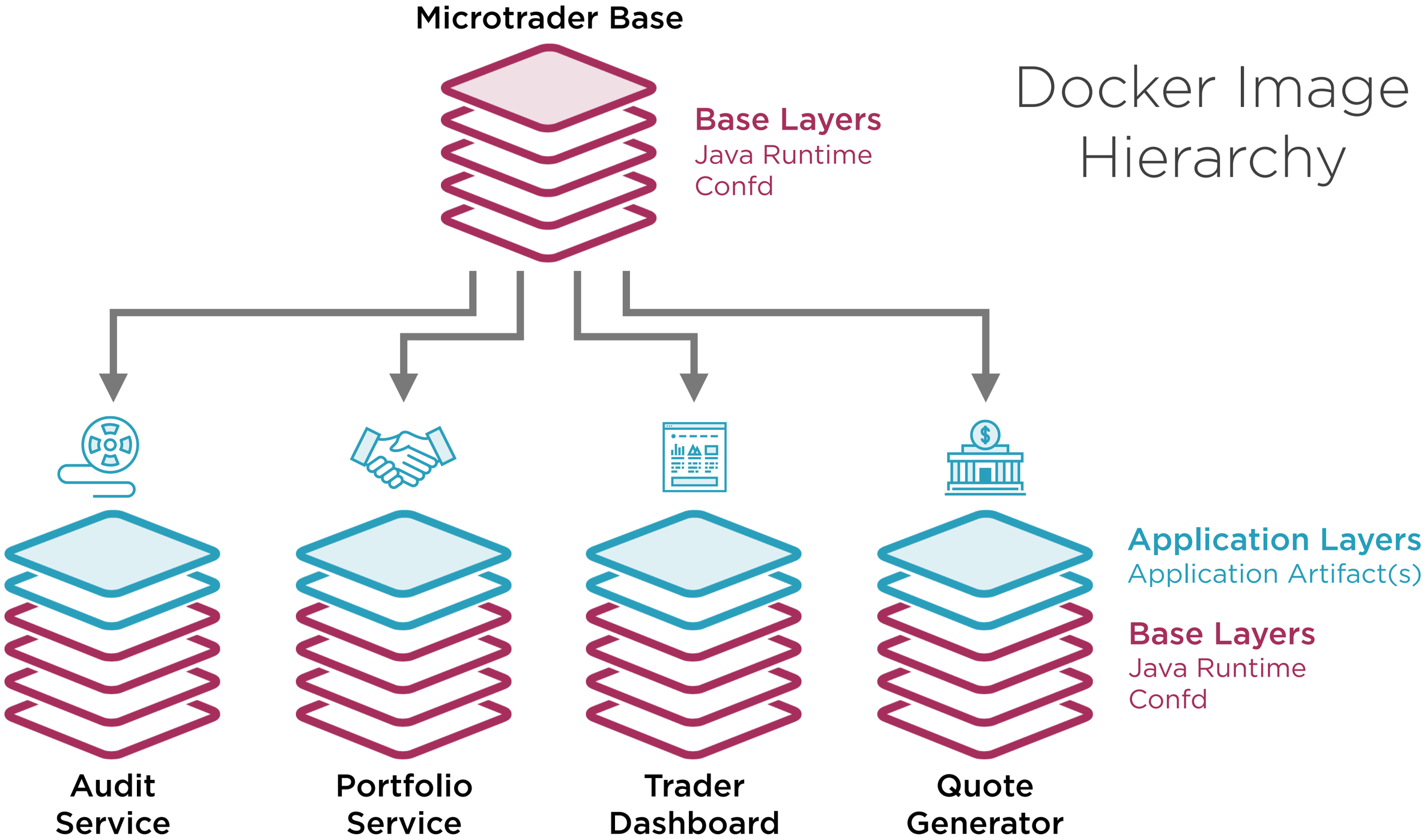
Install Confd Application

Create Templates



Render Templates at Container Startup

Docker Image Hierarchy



Summary

AWS Architecture

- Autoscaling Groups and Load Balancers
- RDS instance
- ECS resources

AWS Challenges

- Hazelcast EC2 auto discovery
- Event bus communications

Dynamic Configuration using Confd

- Generate Go templates from environment
- Generate cluster.xml for local Docker
- Generate cluster.xml for AWS ECS