Application-Layer Egress Control in Kubernetes

Current Solutions, Future Standards

Joshua Fox

SRECon EMEA 2023, Dublin
The challenge

Lock down egress
Threat model

Your application

Malware

exfiltration

api.stealyourmoney.com
But allow

Your application

Least viable permissions

api.stripe.com
graph.facebook.com
api.spotify.com
Our anonymized example

Your application

api.kadath.io

api.leng.com
The VPC

VPC: Egress blocked by default

Your application, configured to connect to api.kadath.com

Allow egress just to API

5.22.128.19 (dynamic)
api.kadath.io (static)

Third party API
Multiple layers

Each firewall works on a different one
The layers

- Application Layer (Layer 7)
  - FQDN
- Presentation Layer (Layer 6)
- Session Layer (Layer 5)
- Transport Layer (Layer 4)
- Network Layer (Layer 3)
- Data Link Layer (Layer 2)
- Physical Layer (Layer 1)

Protocols:
- HTTP, DNS
- TCP, UDP
- IP
Rapidly evolving
My articles and talks
TFiR article re Kubernetes


KubeCon | CloudNativeCon
Europe 2023
Re networking

bit.ly/egress-control
Talks

Level 7 Egress Control in Kubernetes

Joshua Fox
joshua@doit.com
Firewalls
Firewall

- VPC: Egress blocked by default
- Your application
- Allow egress
- 5.22.128.19
- Third party API
Google Firewall
Summary of AWS Firewall types

<table>
<thead>
<tr>
<th>Security Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network ACL</td>
</tr>
<tr>
<td>Network Firewall</td>
</tr>
<tr>
<td>Shield/WAF</td>
</tr>
<tr>
<td>Route 53 Firewall</td>
</tr>
</tbody>
</table>
Article on AWS Firewall types


AWS Firewalls 101: How and When to Use Each One

AWS offers a lot of firewall solutions, and now a new one. Here's how to make sense of all these firewalls.
AWS Security Groups
AWS Network ACL

VPC:
Egress blocked by default

Your application

Allow egress

5.22.128.19

Third party API
IP address changes

5.22.128.15

api.kadath.io

api.leng.com

Ἡράκλειτος
Implementations
Hosted
Implementation

5.22.128.19 → [..] → api.kadath.io
5.22.128.21 → [..] → api.leng.com
Don’t DIY!
Squid Proxy

api.kadath.io

5.22.128.19
5.22.128.21
Squid Proxy

```
  acl whitelist dstdomain api.kadath.io
  http_access allow whitelist

  http_access deny all
```
Multiple hosts on an IP address

5.22.128.19

API Gateway

api.kadath.io

api.leng.com
Server Name Indication
HTTPS Flow & SNI

TLS: SNI api.kadath.io

TLS: CERT api.kadath.io

HOST https://api.kadath.io

api.kadath.io

api.leng.com
Encrypted SNI

Encrypted DNS lookup

Public key

TLS: SNI api.kadath.io

api.kadath.io

api.leng.com

TLS: CERT api.kadath.io

HOST https://api.kadath.io
Squid Proxy: SslBump Peel and Splice

acl step1 at_step SslBump1
acl whitelist_ssl ssl::server_name "/etc/squid/whitelist"
acct whitelist dstdomain "/etc/squid/whitelist"
...
ssl_bump peek step1
ssl_bump splice whitelist_ssl
ssl_bump terminate all !whitelist_ssl
Commercial self-hosted

Chaser

DiscrimiNAT

Aviatrix

Aviatrix FQDN Egress Filtering on AWS
Don’t use a hosted solution
Services,
AWS services
AWS Network Firewall

With SNI!

api.leng.com
api.kadath.io
AWS Network Firewall

```json
{
    "RulesSource": {
        "RulesSourceList": {
            "Targets": [
                "api.kadath.io"
            ],
            "TargetTypes": [
                "HTTP_HOST",
                "TLS_SNI"
            ],
            "GeneratedRulesType": "ALLOWLIST"
        }
    }
}
```
Route 53 DNS Firewall

api.kadath.io

api.leng.com
Implementations

Google services
Google Secure Web Proxy

Supports SNI

See the first article by Chimbu Chinnadurai
Google FQDN Firewall Objects

No SNI

30 sec

See the second article by Chimbu Chinnadurai
Kubernetes
Another world
Cluster and Namespaces

- Frontend
- Shopping Cart
- API connector

api.kadath.io

api.leng.com
Plain: Kubernetes Network Policy

apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
...

spec:
podSelector:
  matchLabels:
    networking/access-kadath-api: "true"

egress:
- to:
  - ipBlock: 5.22.128.133/32

policyTypes:
- Egress

Nope!
Cilium

eBPF-based Networking layer
On each host
Kernel-adjacent
Cilium

eBPF-based Networking

Cilium Network Policy

Cilium Clusterwide Network Policy

With SNI!
Cilium Network Policy

```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: "fqdn"
spec:
  endpointSelector:
    matchLabels:
      function: api-conector
  egress:
    - toFQDNs:
      - matchName: "api.kadath.io"
```

Another clause blocks egress from all other pods.
Cilium

Why not?
Istio

Service A

Envoy proxy

Service B

Envoy proxy

Istio control plane

metrics

metrics
Istio ServiceEntry

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: kadath
spec:
  hosts:
    - ".kadath.io"
    - location: MESH_EXTERNAL
  ports:
    - number: 443
      name: https
      protocol: HTTPS
```

With SNI using `sniHosts`
Istio

Why not?
Kubernetes Network Policy

apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
...

spec:
  podSelector:
    matchLabels:
      networking/access-kadath-api: "true"
  egress:
    - to:
      - ipBlock: 5.22.128.133/32
  policyTypes:
    - Egress

Nope!
apiVersion: networking.gke.io/v1alpha1
group: FQDNNetworkPolicy
metadata:
  name: allow-out-fqdnnp
spec:
  podSelector:
    matchLabels:
      run: curl
  egress:
    - matches:
      - name: "api.kadath.io"
  ports:
    - protocol: "TCP" # to allow only HTTPS
      port: 443
Committee

- Kubernetes Project
- Special Interest Group Networking
- Extend K8s NetworkPolicy
<table>
<thead>
<tr>
<th>Technique</th>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Squid &amp; other self-hosted</td>
<td>Mature</td>
<td>You manage it, no K8s</td>
</tr>
<tr>
<td>Level 7 Firewall-as-a-service</td>
<td>As-a-Service</td>
<td>No K8s</td>
</tr>
<tr>
<td>Cilium Network Policies, Istio Service Mesh</td>
<td>Supports K8s, powerful, generally available</td>
<td>Overpowered</td>
</tr>
<tr>
<td>GKE K8s Network Policies</td>
<td>Built-in to K8s</td>
<td>Non-standard</td>
</tr>
<tr>
<td>Standard K8s Network Policies</td>
<td>Built-in to K8s, standard</td>
<td>Doesn’t exist</td>
</tr>
</tbody>
</table>
See the slides

We’re hiring!

joshua@doit.com

Questions?