

Egress Woes: Debugging external service traffic in Istio

Greg Hanson
Twitter: @gihanson



#IstioCon

What is Istio?

?

?

?

?

#IstioCon



Debugging techniques

#IstioCon



“istioctl proxy-status” and “istioctl analyze”

#IstioCon



Access logs

- Telemetry API
 - Enable access logs on a per-workload basis
- Access logs start at the Listener
 - If the port and protocol don't match there may not be a log!
- PassthroughCluster beware!

#IstioCon



Engarde

github.com/nitishm/engarde

github.com/GregHanson/engarde-viewer

```
{
  "authority": "httpbin:8000",
  "bytes_received": "0",
  "bytes_sent": "551",
  "duration": "22",
  "forwarded_for": "-",
  "method": "GET",
  "protocol": "HTTP/1.1",
  "request_id": "492f0cf3-e6e4-9182-b7c4-ee9d2ec06cfc",
  "response_flags": "-",
  "status_code": "200",
  "timestamp": "2021-02-11T21:57:47.658Z",
  "upstream_service": "172.30.213.81:80",
  "upstream_service_time": "22",
  "upstream_cluster": "outbound|8000|httpbin.default.svc.cluster.local",
  "upstream_local": "172.30.213.80:43640",
  "downstream_local": "172.21.92.90:8000",
  "downstream_remote": "172.30.213.80:46244",
  "uri_path": "/headers",
  "user_agent": "curl/7.69.1",
  "mixer_status": "-",
  "original_message": "[2021-02-11T21:57:47.658Z] \"GET /headers HTTP/1.1\" 200
  \"httpbin:8000\" \"172.30.213.81:80\" outbound|8000|httpbin.default.svc.cluster"
}
```

Engarde-Viewer

Engarde Viewer

Enter an access log line:

```
[2022-04-01T13:19:38.282Z] "GET /productpage HTTP/1.1" 200 - via_upstream - "-" 0 5183 68 68 "10.112.0.1" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.84 Safari/537.36" "ee419ced-7a91-4840-afcc-1bbe8eb4d593" "35.231.243.205" "127.0.0.1:9080" inbound|9080|127.0.0.1:36240 10.112.0.66:9080 10.112.0.1:0 outbound_9080_ productpage.default.svc.cluster.local default
```

Use Istio

[Process](#)

Engarde Output

```
}
  authority: "35.231.243.205",
  bytes_received: 0,
  bytes_sent: 5183,
  connection_termination_details: -,
  duration: 68,
  forwarded_for: 10.112.0.1,
  method: GET,
  route_name: default,
  protocol: HTTP/1.1,
  request_id: ee419ced-7a91-4840-afcc-1bbe8eb4d593,
  response_flags: -,
  response_code_details: via_upstream,
  status_code: 200,
  timestamp: 2022-04-01T13:19:38.282Z,
  upstream_service: 127.0.0.1:9080,
```

Documentation

%ROUTE_NAME%
HTTP/TCP
Name of the route.

UDP
Not implemented ("-").

#IstioCon



istioctl proxy-config

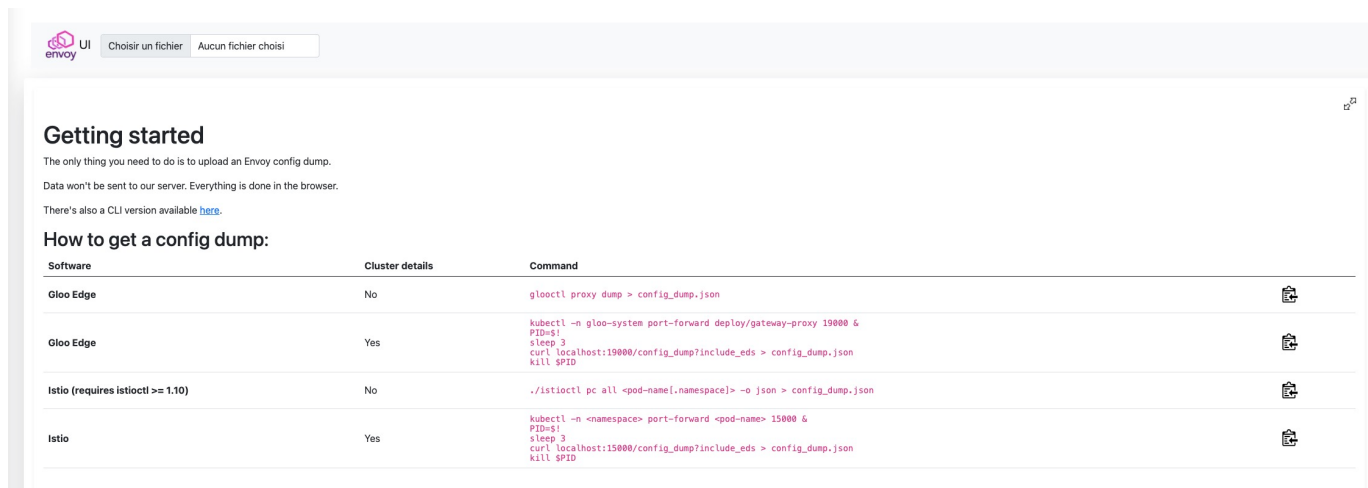
- **istioctl pc listeners**
 - Port
 - Protocol
 - SNI values
- **istioctl pc routes**
 - Hostnames
 - Path based routing
- **istioctl pc clusters**
 - TLS settings
- **istioctl pc secrets**
 - Custom certs
- **Embrace “-o json”**

#IstioCon



envoyctl and envoyui

- <https://github.com/djannot/envoyctl>
 - CLI tool for debugging Envoy
- <https://envoyui.solo.io>
 - View your config dumps with ease



The screenshot shows the Envoy UI interface. At the top, there's a header with the Envoy logo and the text 'UI Choisir un fichier Aucun fichier choisi'. Below the header, the main content area is titled 'Getting started' and contains the following text: 'The only thing you need to do is to upload an Envoy config dump. Data won't be sent to our server. Everything is done in the browser. There's also a CLI version available [here](#).' Below this, there's a section titled 'How to get a config dump:' followed by a table with three columns: 'Software', 'Cluster details', and 'Command'. The table lists commands for Gloo Edge and Istio, with 'Cluster details' indicating whether the command is for a specific cluster or not.

Software	Cluster details	Command
Gloo Edge	No	<code>glooctl proxy dump > config_dump.json</code>
Gloo Edge	Yes	<code>kubectl -n gloo-system port-forward deploy/gateway-proxy 19000 & PID=\$! sleep 3 curl localhost:19000/config_dump?include_eds > config_dump.json kill \$PID</code>
Istio (requires istioctl >= 1.10)	No	<code>./istioctl pc all <pod-name[.namespace]> -o json > config_dump.json</code>
Istio	Yes	<code>kubectl -n <namespace> port-forward <pod-name> 15000 & PID=\$! sleep 3 curl localhost:15000/config_dump?include_eds > config_dump.json kill \$PID</code>

#IstioCon



Common mistakes

#IstioCon



protocol: TCP

```
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: my-se
spec:
  hosts:
  - this.hostname.does.not.matter
  addresses: # these actually matter
  - 1.2.3.4
  ports:
  - number: 443
    name: tcp
    protocol: TCP
```

#IstioCon



Where did my hosts go?

```
apiVersion: networking.istio.io/v1alpha3
kind: Sidecar
metadata:
  name: default
  namespace: istio-config
spec:
  egress:
  - hosts:
    - ".*/*"
    - "istio-system/*"
```

#IstioCon



Double encryption

```
kind: ServiceEntry
metadata:
  name: my-se
spec:
  hosts:
  - my.secure.hostname
  ports:
  - number: 443
    name: tls
    protocol: TLS
  resolution: DNS
```

```
kind: Gateway
metadata:
  name: istio-egressgateway
spec:
  selector:
    istio: egressgateway
  servers:
  - port:
      number: 443
      name: tls
      protocol: TLS
    hosts:
    - my.secure.hostname
    tls:
      mode: ISTIO_MUTUAL
```

```
kind: DestinationRule
metadata:
  name: egressgateway-for-cnn
spec:
  host: egress.gateway.fqdn
  trafficPolicy:
    tls:
      mode: ISTIO_MUTUAL
      sni: edition.cnn.com
```

#IstioCon



Thank you!

- <https://istio.io/latest/docs/reference/config/telemetry/>
- <https://github.com/nitishm/engarde>
- <https://github.com/GregHanson/engarde-viewer>
- <https://github.com/djannot/envoyctl>
- <https://envoyui.solo.io>
- @gihanson on Istio Slack

#IstioCon

