Envoy

Christopher M Luciano
Advisory Software Engineer at IBM
cmluciano@us.ibm.com

IBM Developer

Agenda

- Envoy Basics
- Envoy Internals
- xDS APIs
- Contributing to Envoy

IBM **Developer**

Summary

The network should be transparent to applications. When network and application problems do occur it should be easy to determine the source of the problem.

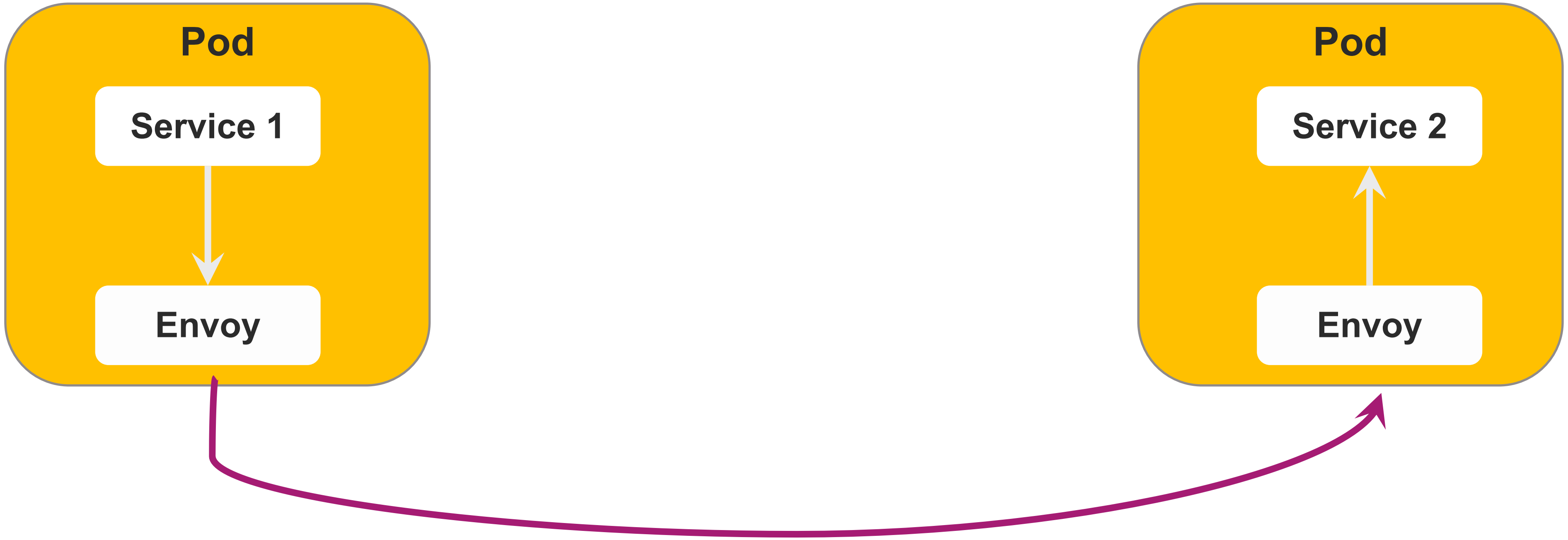
https://www.envoyproxy.io/docs/envoy/v1.7.0/intro/what_is_envoy

Envoy

Intelligent proxy deployed as a sidecar

- Intercept & manages network traffic
- Security/Identity
- TLS termination
- Low memory footprint
- Language Agnostic

Envoy’s Role in Istio



Listener

- One to many number of listeners per Envoy process
- Proxies event when connection is made to listeners
- Configured with network or listener filters
- Listener binding
  - Freebind
  - Transparent
- TCP only (UDP soon)

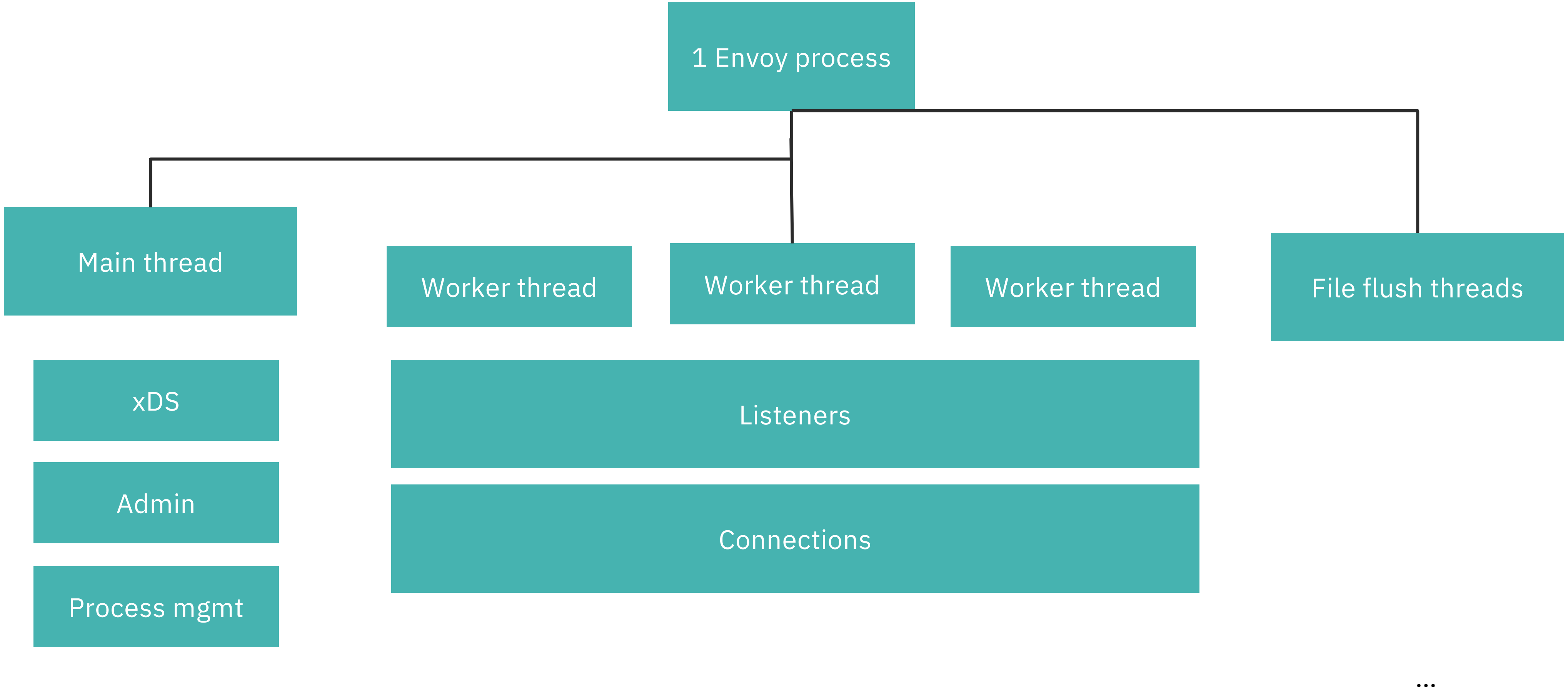
Clusters

- Collection of similar hosts for Envoy connections
- Determined through various service discovery systems
- Can be configured either from a static list or dynamic API
- Warmed on initial boot

Filters

- Listener
  - Original Destination
  - TLS Inspector
- Network (Layer 3 & 4)
  - Rate limiting
  - Mongo
  - TCP Proxy
- HTTP
  - Fault injection
  - Router

Envoy Threading Model



1 Envoy process

Main thread

Worker thread

Worker thread

Worker thread

File flush threads

xDS

Admin

Process mgmt

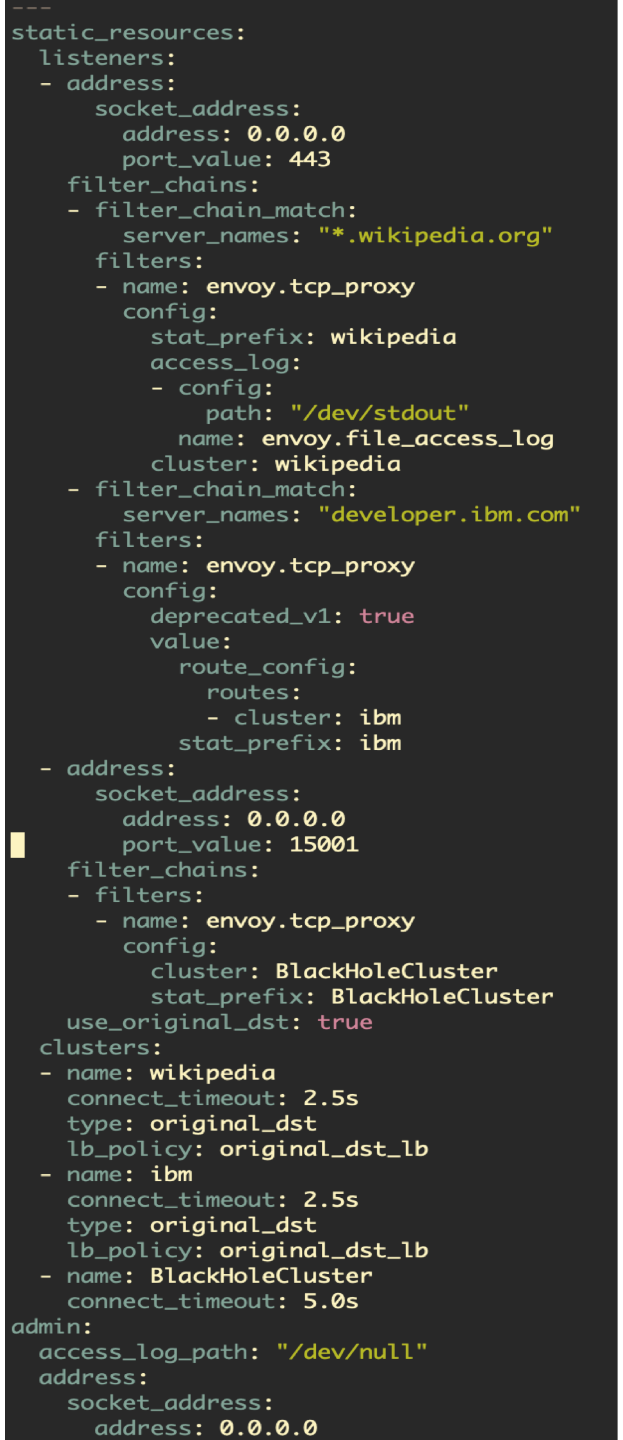
Listeners

Connections

...

SNI Proxy Example

https://github.com/IBM/envoy101/tree/master/assets/sni-proxy-https

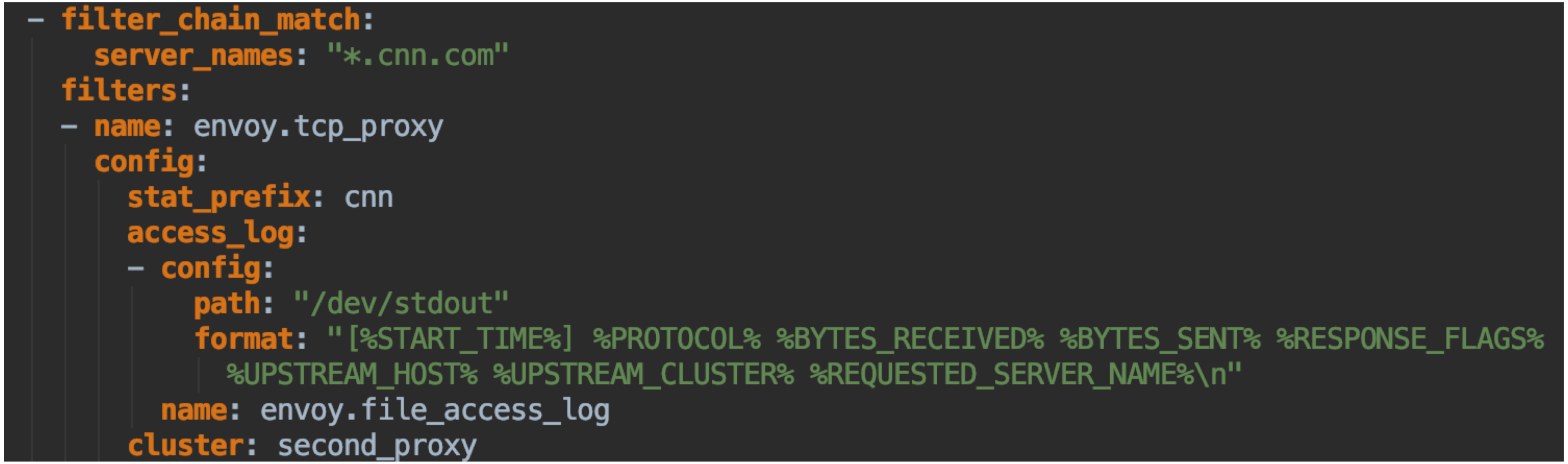


```
---
static_resources:
  listeners:
  - address:
      socket_address:
        address: 0.0.0.0
        port_value: 443
    filter_chains:
    - filter_chain_match:
        server_names: "*.wikipedia.org"
      filters:
      - name: envoy.tcp_proxy
        config:
          stat_prefix: wikipedia
          access_log:
          - config:
              path: "/dev/stdout"
            name: envoy.file_access_log
          cluster: wikipedia
    - filter_chain_match:
        server_names: "developer.ibm.com"
      filters:
      - name: envoy.tcp_proxy
        config:
          deprecated_v1: true
          value:
            route_config:
              routes:
              - cluster: ibm
            stat_prefix: ibm
  - address:
      socket_address:
        address: 0.0.0.0
        port_value: 15001
    filter_chains:
    - filters:
      - name: envoy.tcp_proxy
        config:
          cluster: BlackHoleCluster
          stat_prefix: BlackHoleCluster
    use_original_dst: true
  clusters:
  - name: wikipedia
    connect_timeout: 2.5s
    type: original_dst
    lb_policy: original_dst_lb
  - name: ibm
    connect_timeout: 2.5s
    type: original_dst
    lb_policy: original_dst_lb
  - name: BlackHoleCluster
    connect_timeout: 5.0s
admin:
  access_log_path: "/dev/null"
  address:
    socket_address:
      address: 0.0.0.0
```

DEMO SNI PROXY

https://github.com/IBM/envoy101/blob/master/assets/sni-proxy-https/docker-compose.yaml

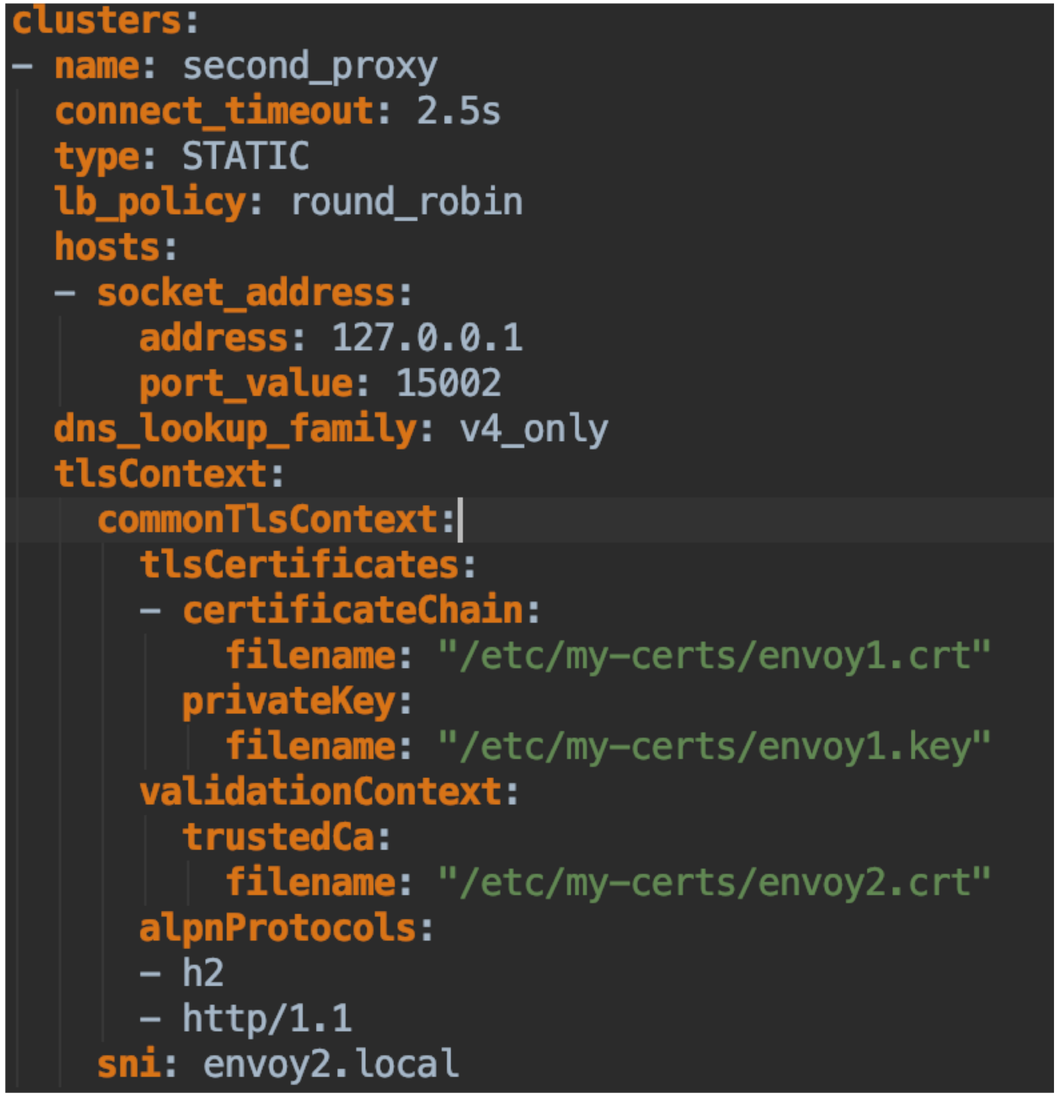
Double Proxy with MTLS filter_chain_match



```
- filter_chain_match:
    server_names: "*.cnn.com"
  filters:
  - name: envoy.tcp_proxy
    config:
      stat_prefix: cnn
      access_log:
      - config:
          path: "/dev/stdout"
          format: "[%START_TIME%] %PROTOCOL% %BYTES_RECEIVED% %BYTES_SENT% %RESPONSE_FLAGS%
            %UPSTREAM_HOST% %UPSTREAM_CLUSTER% %REQUESTED_SERVER_NAME%\n"
        name: envoy.file_access_log
      cluster: second_proxy
```

https://github.com/IBM/envoy101/blob/master/assets/double-proxy-mtls/envoy_config1.yaml

Double Proxy with MTLS cluster



```yaml
clusters:
- name: second_proxy
  connect_timeout: 2.5s
  type: STATIC
  lb_policy: round_robin
  hosts:
  - socket_address:
      address: 127.0.0.1
      port_value: 15002
  dns_lookup_family: v4_only
  tlsContext:
    commonTlsContext:
      tlsCertificates:
      - certificateChain:
          filename: "/etc/my-certs/envoy1.crt"
        privateKey:
          filename: "/etc/my-certs/envoy1.key"
      validationContext:
        trustedCa:
          filename: "/etc/my-certs/envoy2.crt"
      alpnProtocols:
      - h2
      - http/1.1
    sni: envoy2.local
```

https://github.com/IBM/envoy101/blob/master/assets/double-proxy-mtls/envoy_config2.yaml

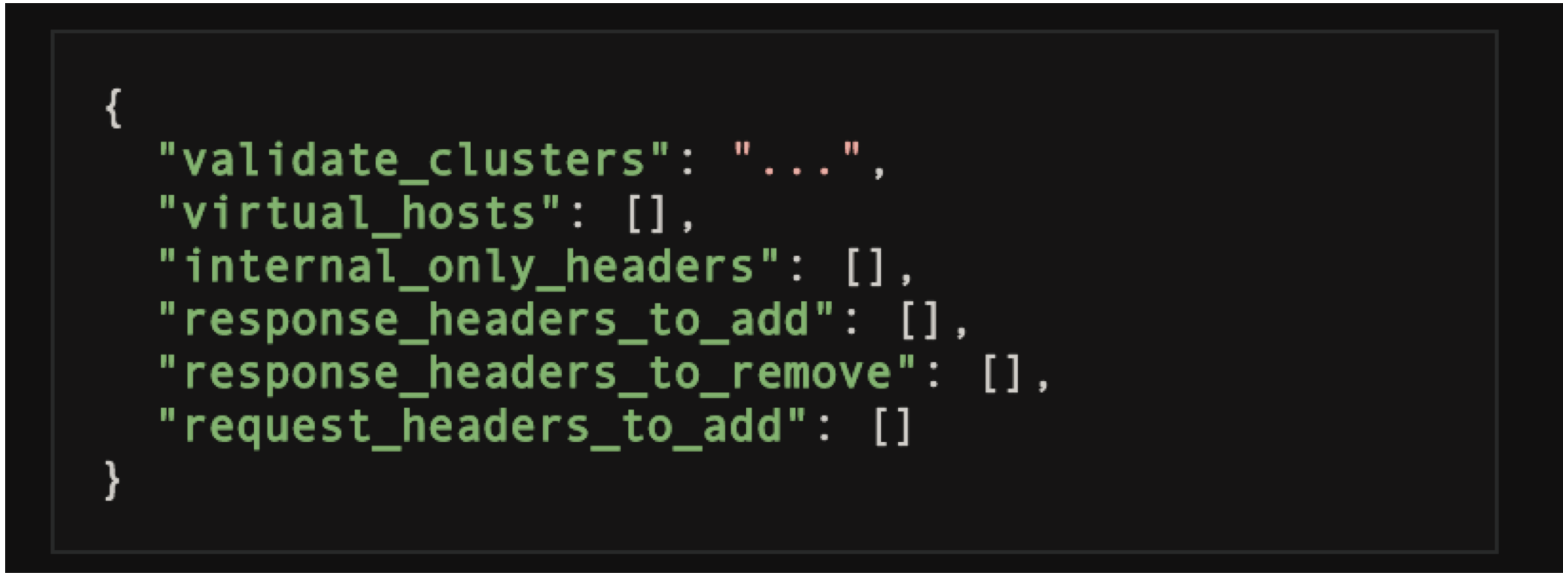
DEMO DOUBLE PROXY MTLS

https://github.com/IBM/envoy101/blob/master/assets/double-proxy-mtls/docker-compose.yaml

xDS APIs

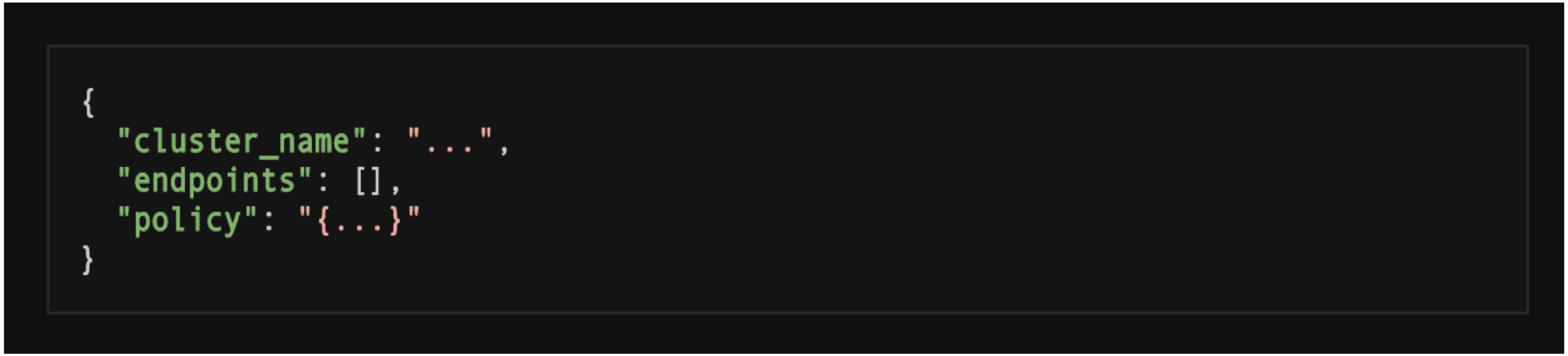
- Route Discovery Service (RDS)
- Endpoint Discovery Service (EDS)
- Cluster Discovery Service (CDS)
- Listener Discovery Service (LDS)
- Aggregated Discovery Service (ADS)

Route Discovery Service



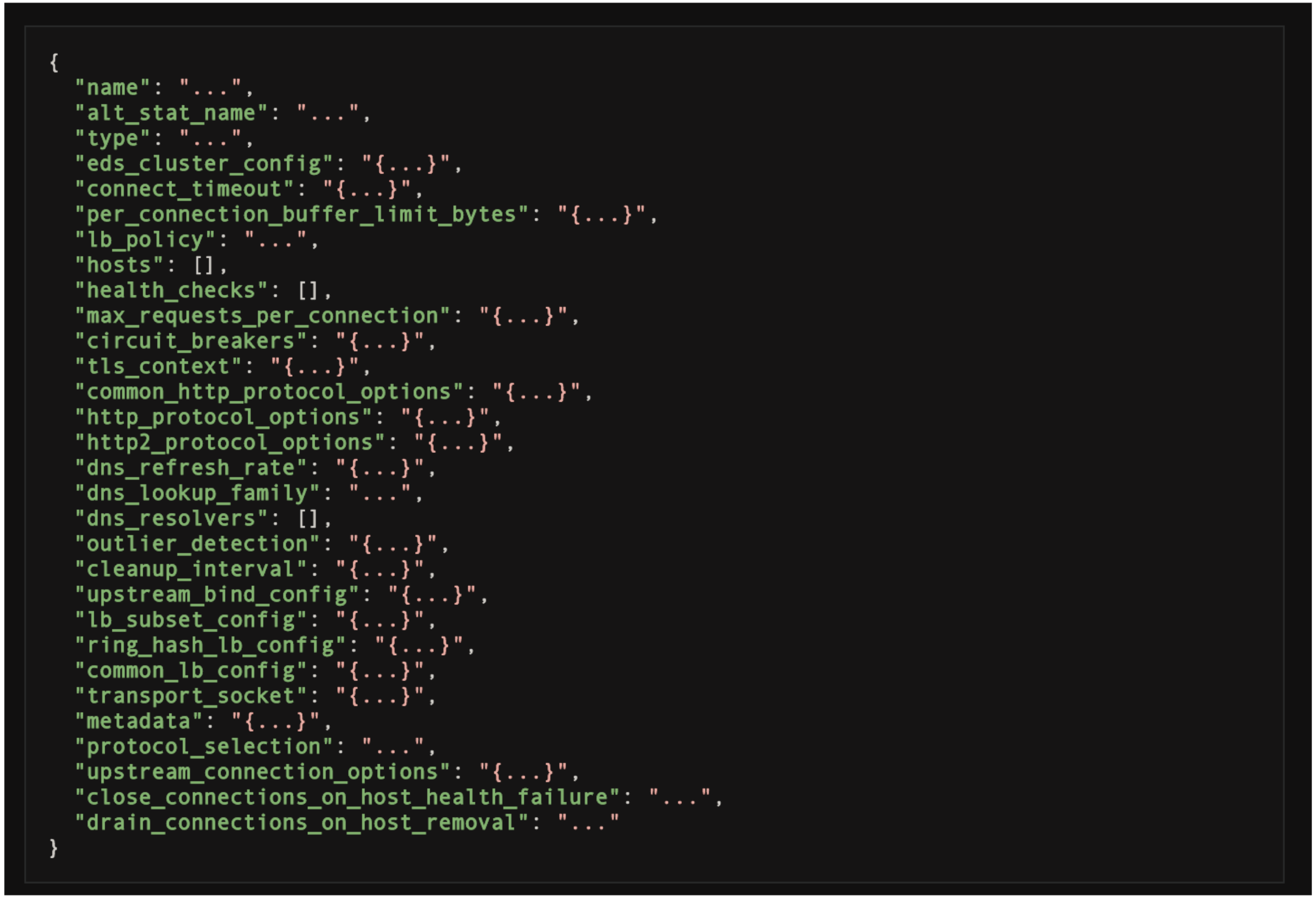
```
{
  "validate_clusters": "...",
  "virtual_hosts": [],
  "internal_only_headers": [],
  "response_headers_to_add": [],
  "response_headers_to_remove": [],
  "request_headers_to_add": []
}
```

Endpoint Discovery Service



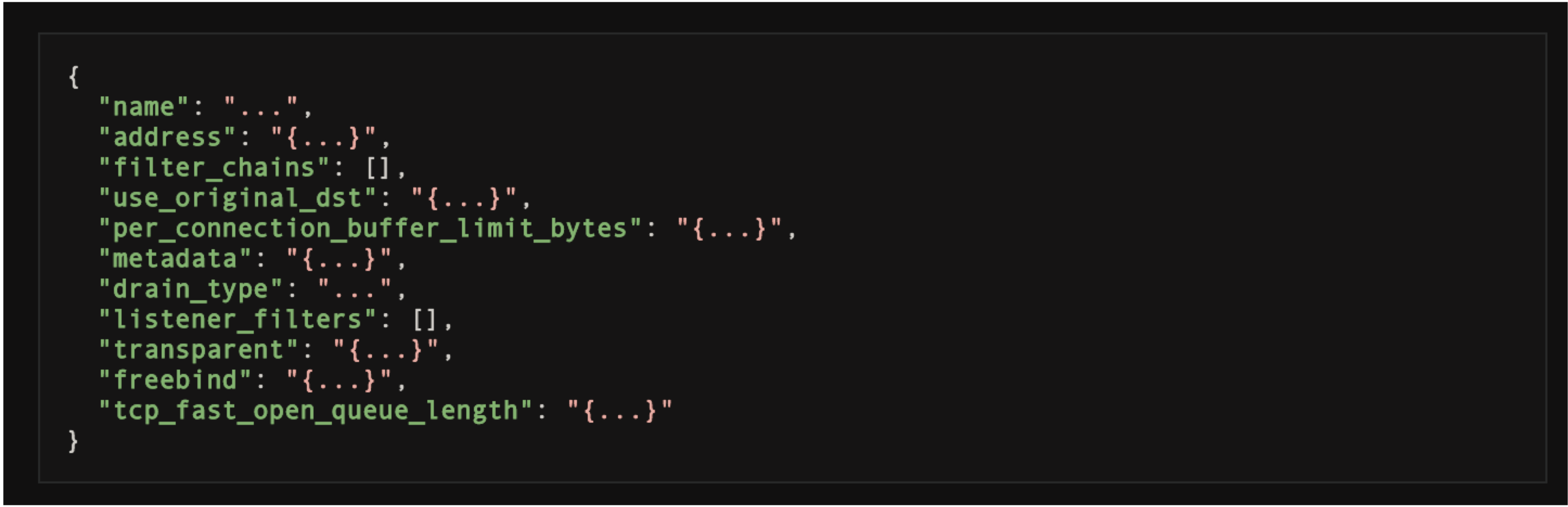
```
{
  "cluster_name": "...",
  "endpoints": [],
  "policy": "{...}"
}
```

Cluster Discover Service



```
{
  "name": "...",
  "alt_stat_name": "...",
  "type": "...",
  "eds_cluster_config": "{...}",
  "connect_timeout": "{...}",
  "per_connection_buffer_limit_bytes": "{...}",
  "lb_policy": "...",
  "hosts": [],
  "health_checks": [],
  "max_requests_per_connection": "{...}",
  "circuit_breakers": "{...}",
  "tls_context": "{...}",
  "common_http_protocol_options": "{...}",
  "http_protocol_options": "{...}",
  "http2_protocol_options": "{...}",
  "dns_refresh_rate": "{...}",
  "dns_lookup_family": "...",
  "dns_resolvers": [],
  "outlier_detection": "{...}",
  "cleanup_interval": "{...}",
  "upstream_bind_config": "{...}",
  "lb_subset_config": "{...}",
  "ring_hash_lb_config": "{...}",
  "common_lb_config": "{...}",
  "transport_socket": "{...}",
  "metadata": "{...}",
  "protocol_selection": "...",
  "upstream_connection_options": "{...}",
  "close_connections_on_host_health_failure": "...",
  "drain_connections_on_host_removal": "..."
}
```

Listener Discover Service (LDS)



```
{
  "name": "...",
  "address": "{...}",
  "filter_chains": [],
  "use_original_dst": "{...}",
  "per_connection_buffer_limit_bytes": "{...}",
  "metadata": "{...}",
  "drain_type": "...",
  "listener_filters": [],
  "transparent": "{...}",
  "freebind": "{...}",
  "tcp_fast_open_queue_length": "{...}"
}
```

Aggregated Discovery Service (ADS)

- Single sequenced delivery of configuration
  - Istio 1.0 style
- Combines CDS/RDS/EDS
- Bidirectional stream
- gRPC style only

Contributing to Envoy

- github.com/envoyproxy/envoy
- Bazel based build system
- Open issues designated with “help/wanted”
  - Beginner tags for “good first issues”
- cmluciano.blog for contribution guide

Thank you

cmluciano@us.ibm.com

twitter.com/cmluciano_

github.com/cmluciano

developer.ibm.com