
# Extending Envoy with Go and Cilium

Thomas Graf, Cilium (@tgraf__)


# About the Speaker

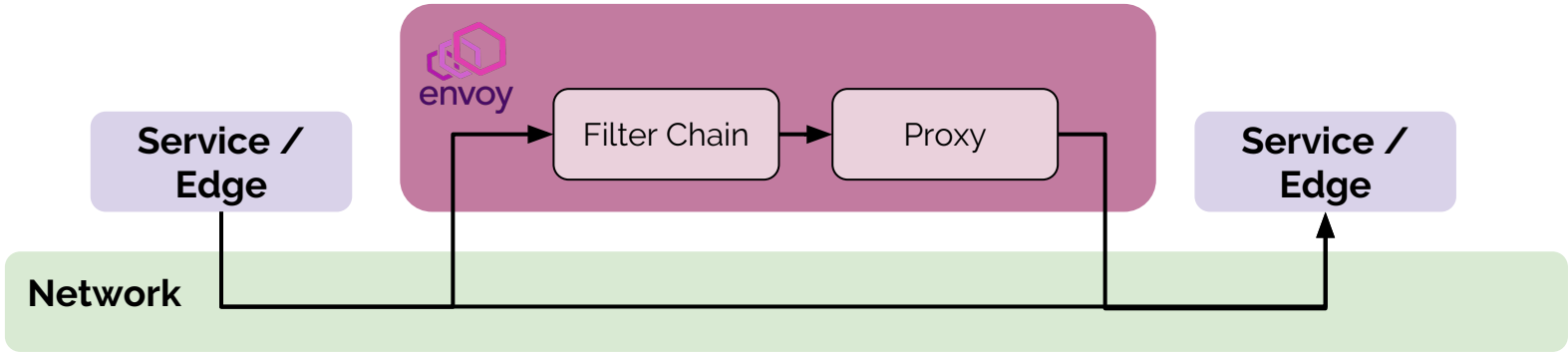

## Thomas Graf

- Linux kernel developer for many years
- Working on networking, security and BPF
- Founder of the Cilium project

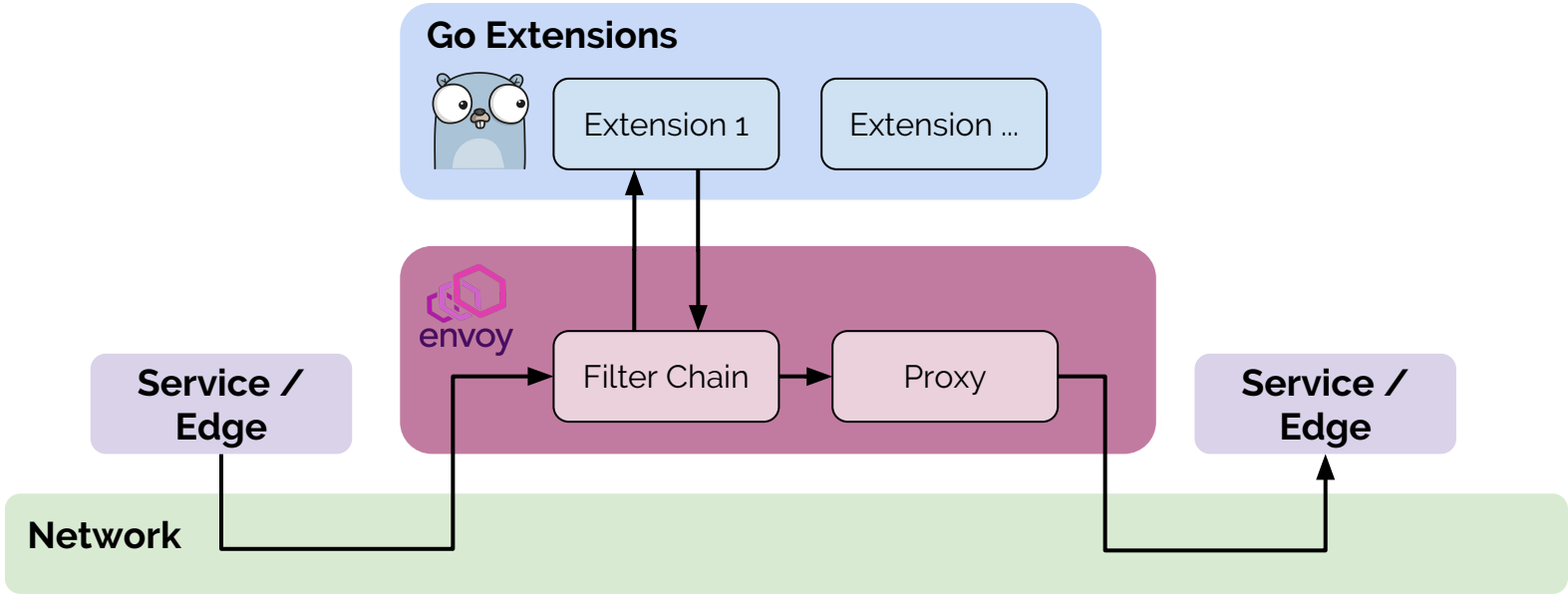

# Cilium


- Based on new BPF technology
- Networking (CNI)
- Envoy Integration
  - Accelerated proxy redirection
  - Transparent SSL visibility
  - BPF metadata filter, Network filter, HTTP filter
- Network security
  - Identity-based, DNS aware, API aware
- Kubernetes services implementation with multi-cluster routing
- Go Extensions

# Envoy Basics

# Go Extensions



# Motivation


## Secret Plot to get Matt to embrace Go

- Already features C++ style exceptions via `panic()` and `recover()`
- Willing to support Go++

Real Motivation:

# Making Envoy Data Aware


## Type of service communication

- **Edge to Service** ← Envoy is already good at this
  - HTTP, SSL termination
- **Service to Service** ← Envoy is already good at this
  - HTTP/REST, gRPC, mTLS
- **Service to Resources/Data** ← Motivation for Go extensions and kTLS (SSL visibility)
  - Cassandra, Kafka, SQL variants, Redis, Mongo, Memcached, …
  - {Storage|Database|Messaging} cloud providers services

# Design Principles


## Clean & stable API

- No need to know Envoy internals to extend Envoy
- Ability to leverage existing Go code

## Safe & flexible

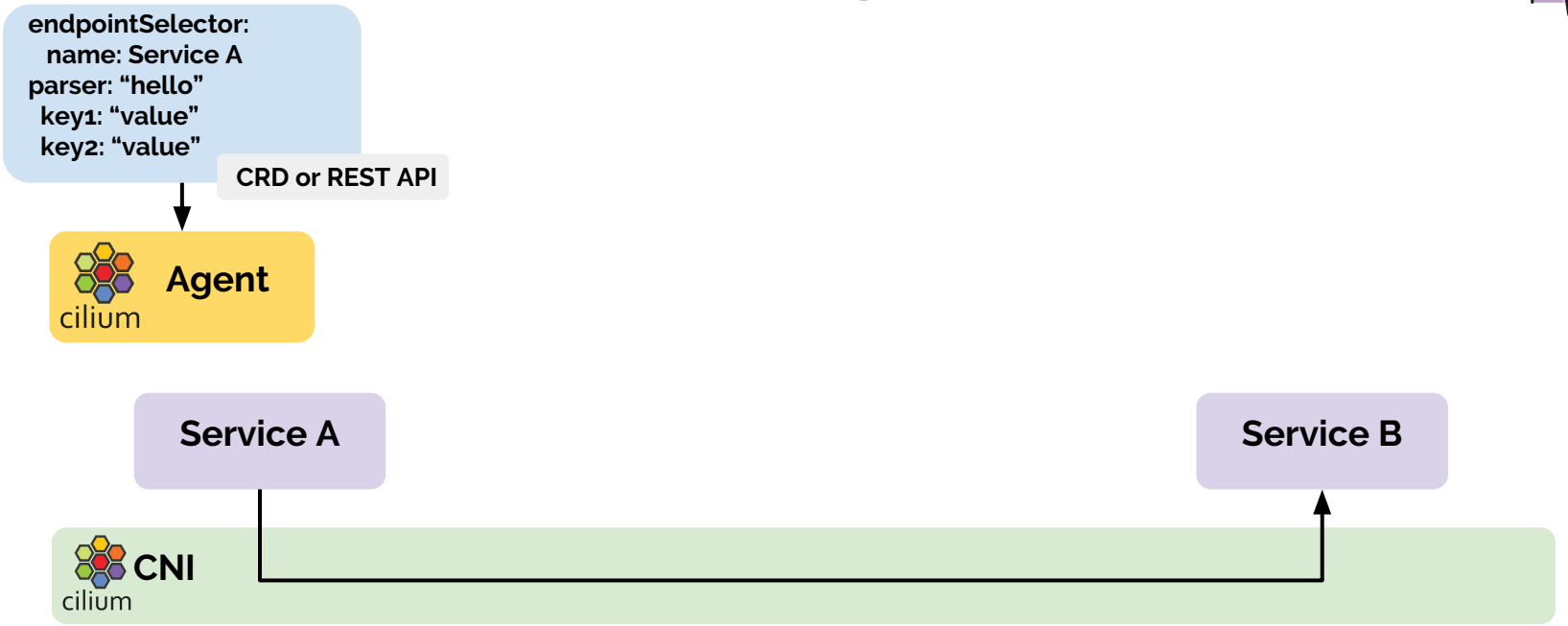
- Extensions are loaded at runtime
- Bug in Go extension cannot crash Envoy

## Preserve Envoy's performance & latency

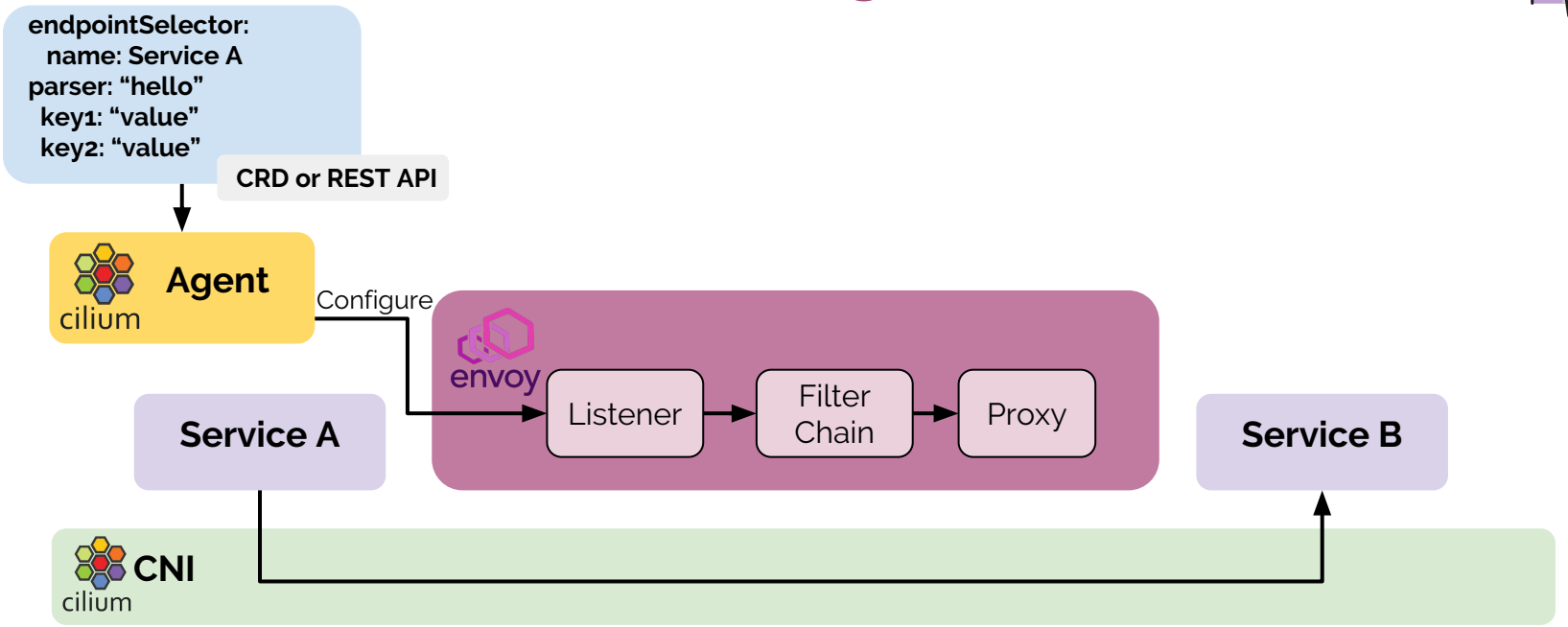
- Native execution environment
- Zerocopy data exchange

# Cilium + Envoy Stack

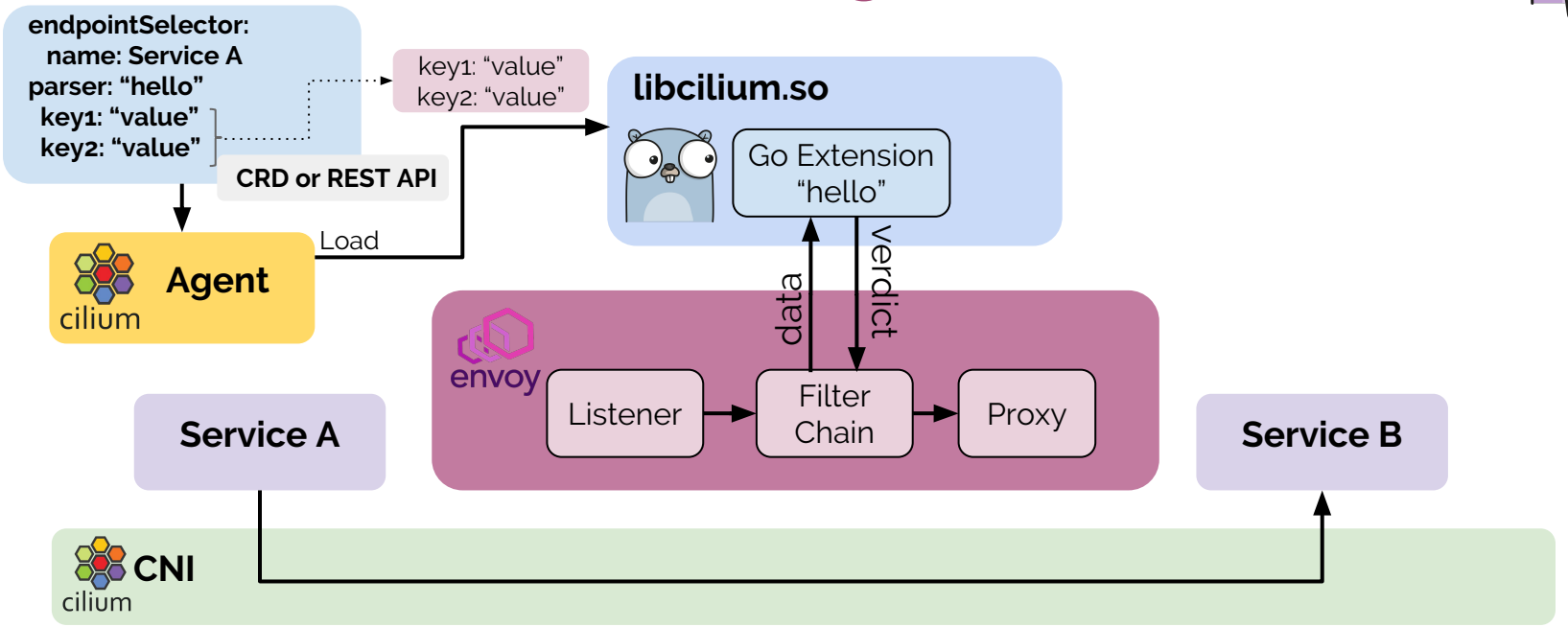

# Cilium + Envoy Stack

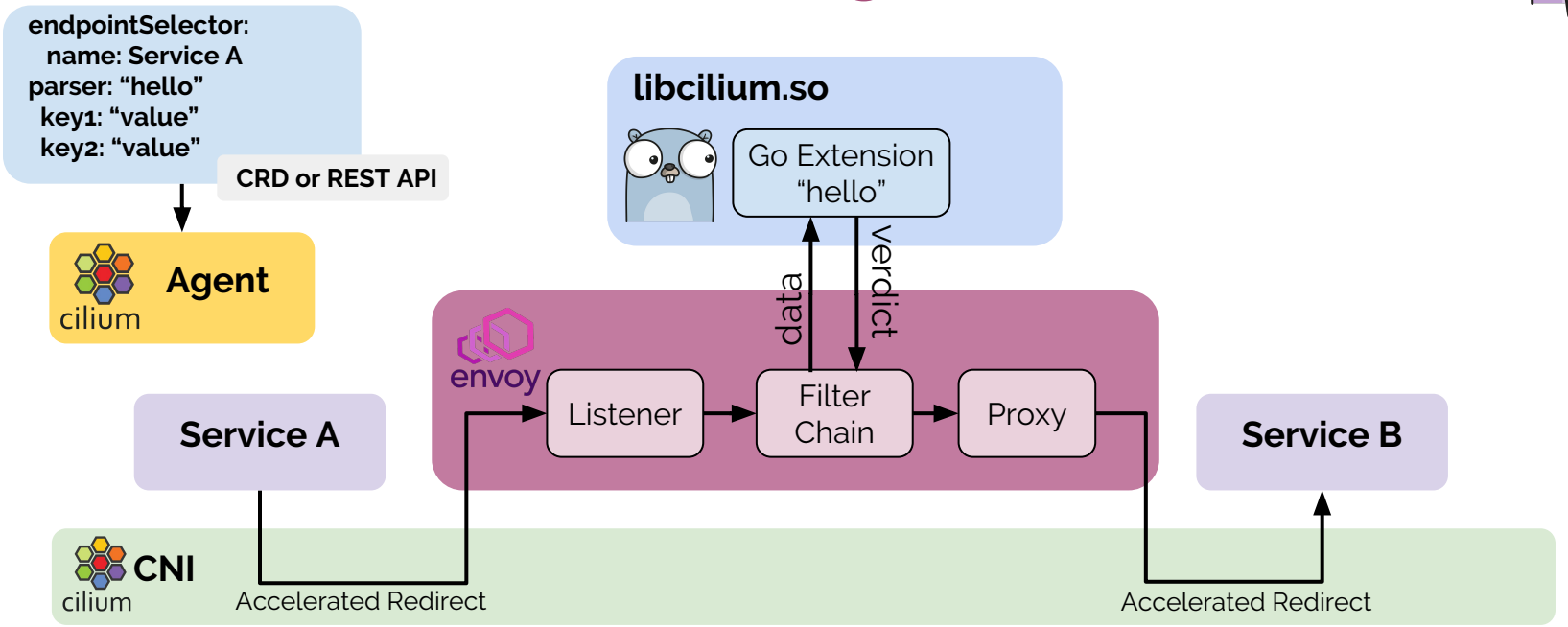
# Cilium + Envoy Stack

# Cilium + Envoy Stack

# Cilium + Envoy Stack

# Cassandra Example

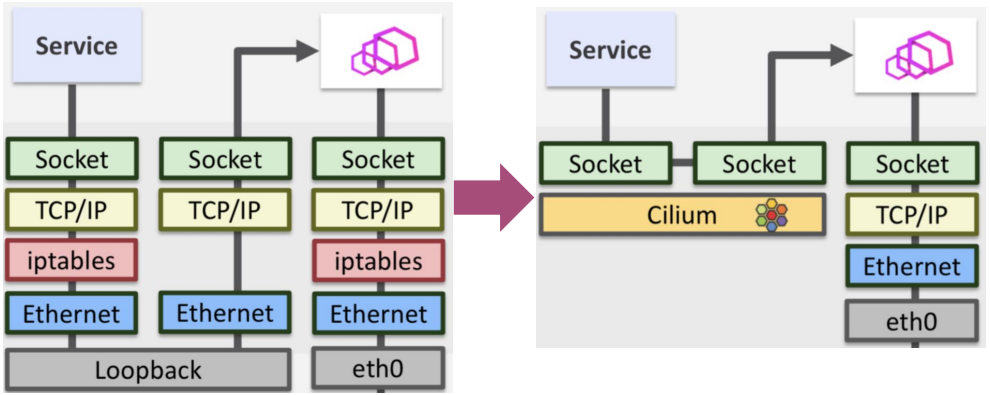
```
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
[...]
specs:
- endpointSelector:
  matchLabels:
    app: cassandra
ingress:
- toPorts:
  - ports:
    - port: "9042"
      protocol: TCP
      l7proto: cassandra
      l7:
      - query_action: "select"
        query_table: "myTable"
```

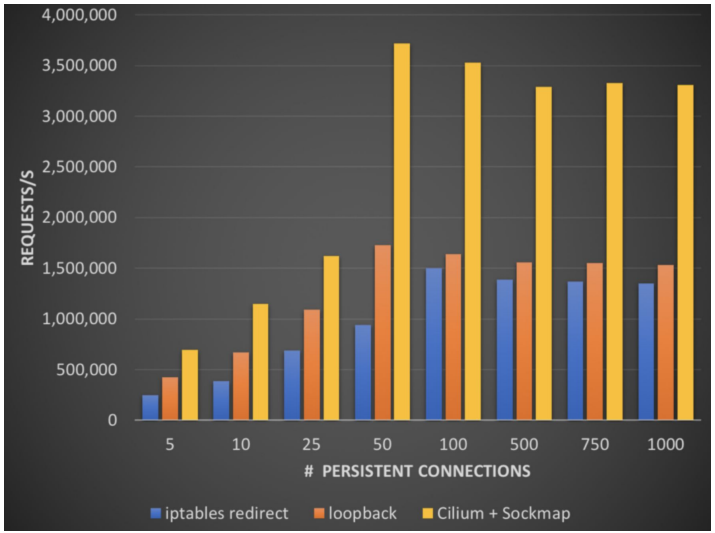
# API Basics: OnData()


- **MORE:** Parser needs *n* more bytes to continue parsing.
- **PASS:** Pass along *n* bytes of the data stream.
- **DROP:** Drop *n* bytes of the data stream.
- **INJECT:** Inject *n* bytes of data in the specified direction.
- **ERROR:** A parsing error has occurred, the connection must be closed.
- **NOP:** Do nothing.
- ...

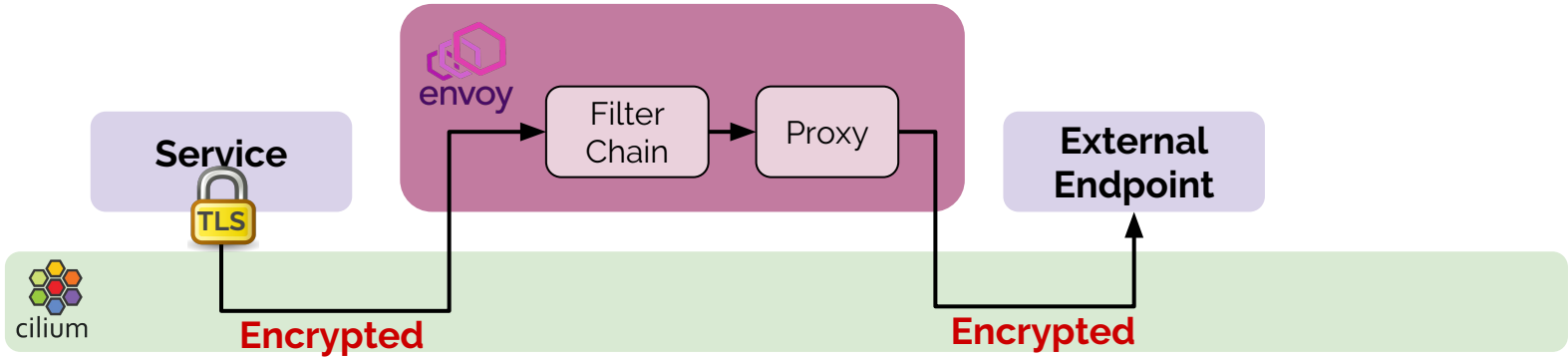
# Accelerated Redirect

## Performance of Unix Domain sockets using TCP sockets

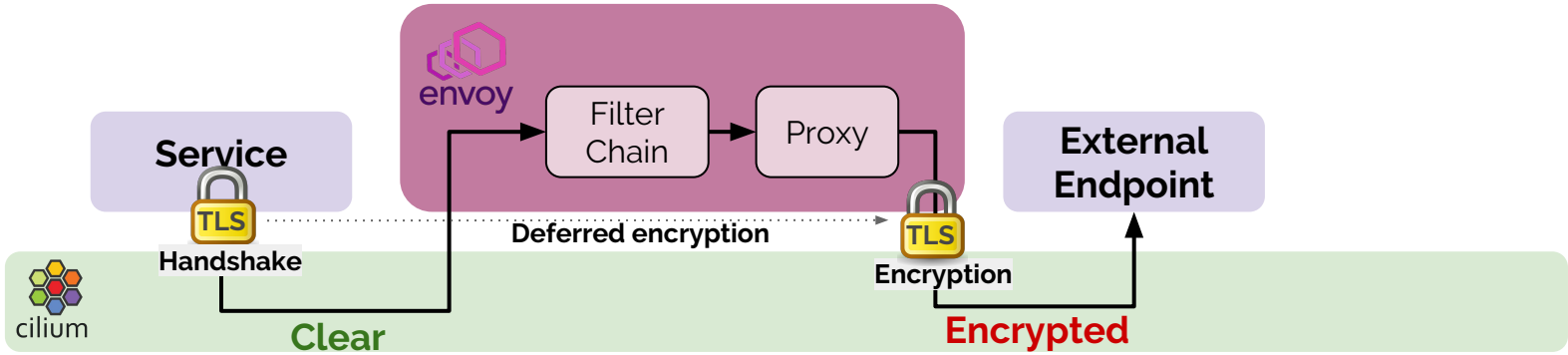


# Transparent SSL Visibility



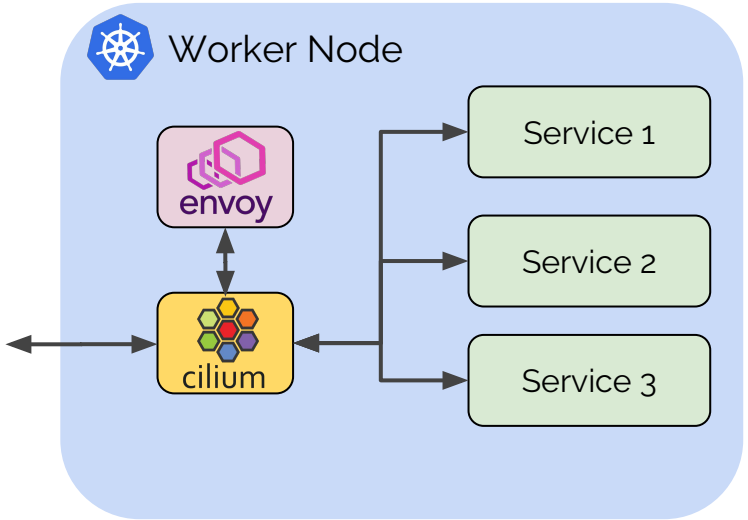
# Transparent SSL Visibility



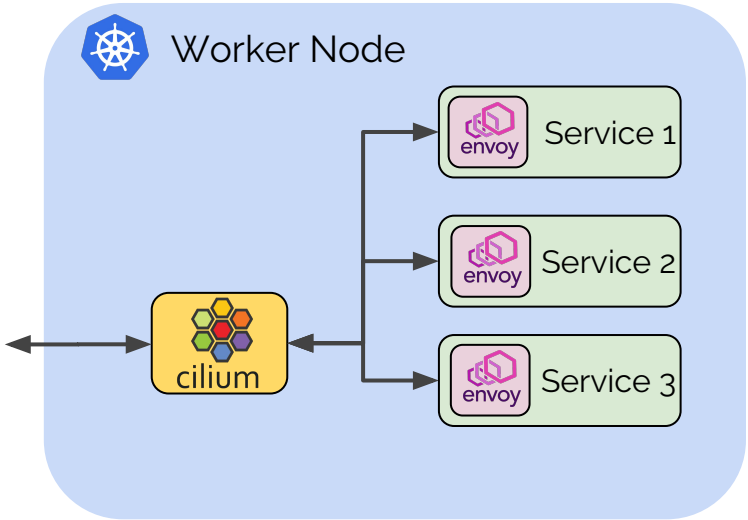

More info in KubeCon EU 2018 slides:
**Accelerating Envoy and Istio with Cilium and the Linux Kernel**
https://bit.ly/2G7DfIY

# Flexible Proxy Model


**One Envoy Per Node**




**Sidecar Proxy Mode**



# Summary


## Envoy

- Efficient L4-L7 proxy
- Written in C++

## Go Extensions

- Run on top of Envoy
- Loaded at runtime
- Configured via CRD
- Injected via Cilium

## Cilium

- Based on new BPF technology
- Networking (CNI)
- Envoy integration
  - Accelerated proxy redirection
  - Transparent SSL Visibility
  - BPF metadata filter, Network filter, HTTP filter
- Network security
  - Identity-based, DNS aware, API aware
  - Kernel accelerated service authentication (1.5)
- Efficient Kubernetes services implementation

# Thank You



**Join the community:**

GitHub: https://github.com/cilium/cilium

Slack: https://cilium.io/slack

Twitter: @ciliumproject

**Getting Started with Envoy Go Extensions:**

http://docs.cilium.io/en/stable/envoy/extensions/

https://cilium.io/blog/2018/10/23/cilium-13-envoy-go