# Learn about Bamboo Firewall solution

## Bamboo Firewall

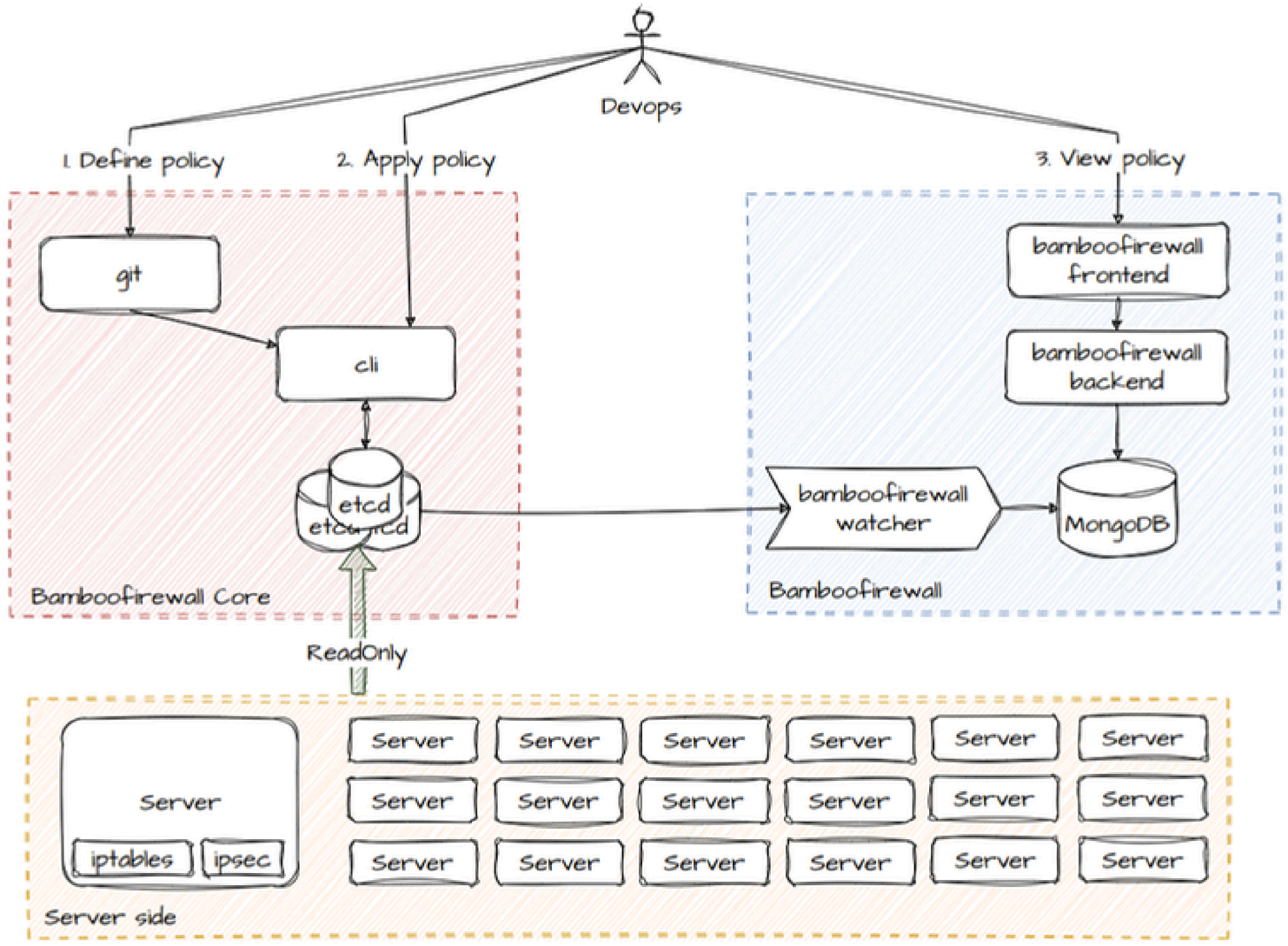
GHTK

# Giới thiệu

- software firewall
- central management
- rules/polices as code


Bamboo Firewall

# Mô hình triển khai




GHTK

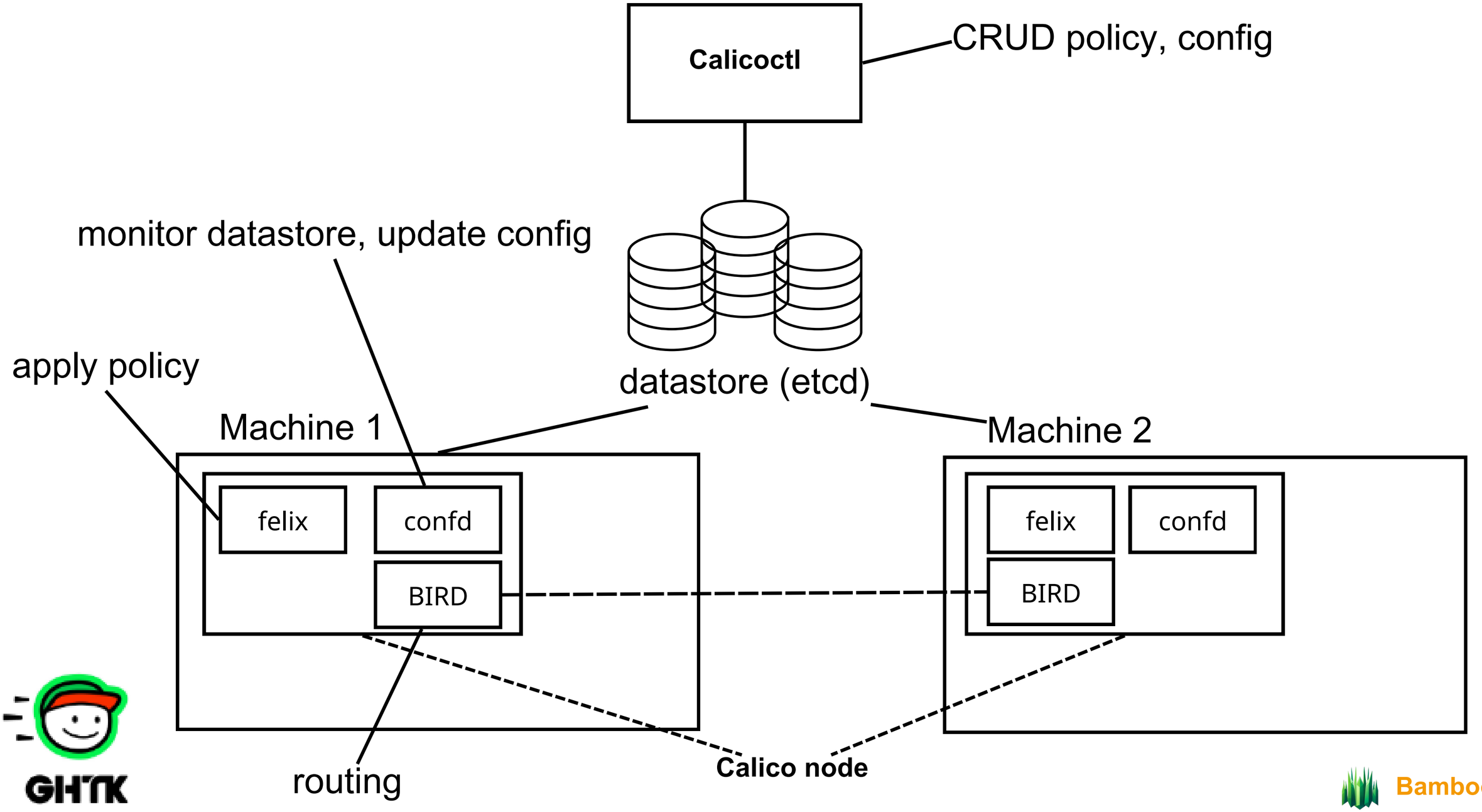
Bamboo Firewall

# Mô hình triển khai

- etcd cluster: Where stores metadata of bamboo firewall. ex: server endpoint, network zone, policies
- watcher: A job watch events from etcd to mongodb database (one way)
- backend (be): API backend server. It provides API for frontend
- frontend (fe): Frontend provides user interfaces via webview
- cli: Command line provides console interface for administrator
- Agent: Agent installed each server and connect to etcd cluster to apply policies

# Kiến trúc Calico

- calicoctl
- datastore
- calico node
  - felix
  - BIRD
  - confd

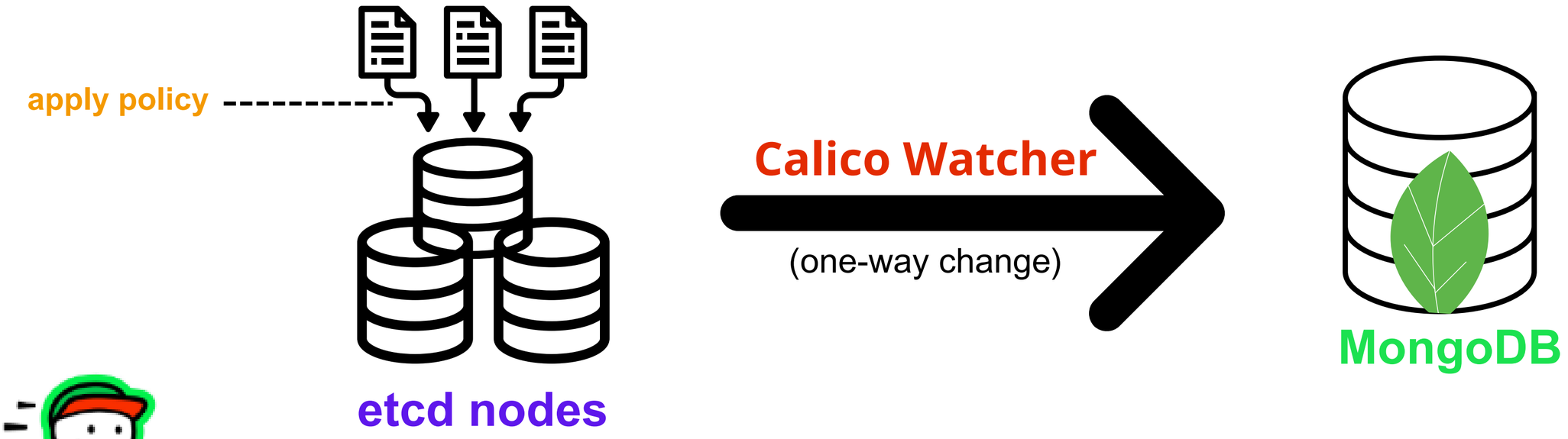




GHTK

Bamboo Firewall

# Calico Watcher

- Calico watcher will monitor changing on etcd cluster and update to mongodb database.



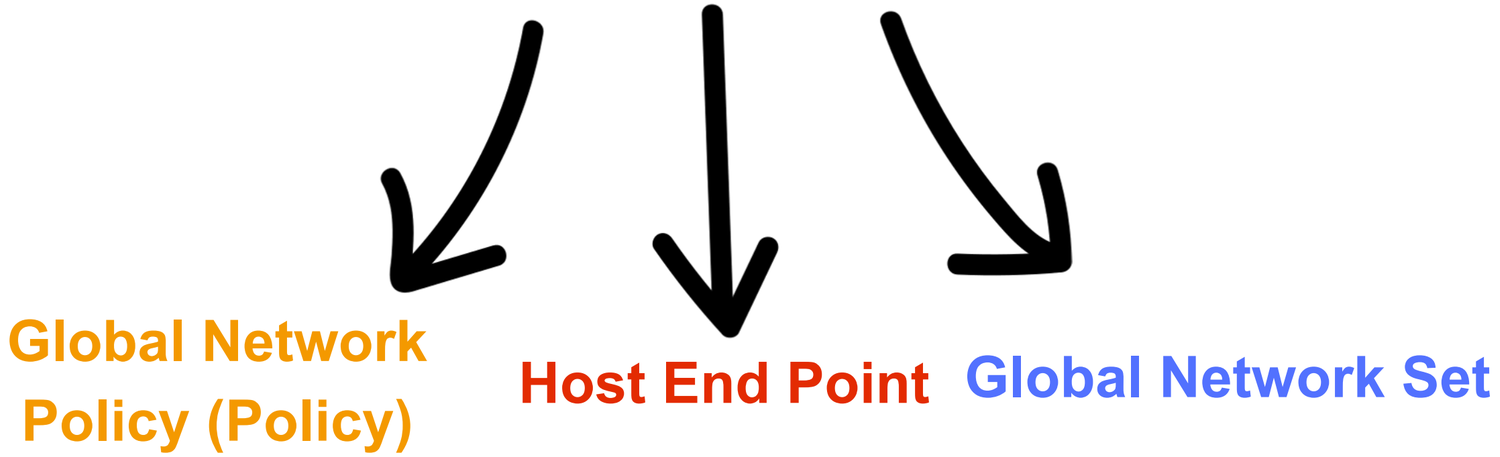

GHTK

Bamboo Firewall

# Bamboo API Server

## Overview

- Purpose: Gather and manage information related to Bamboo Firewall.




- Includes 3 types of APIs: **Public API**, **Protected API**, **Admin API - User management**

GHTK

Bamboo Firewall

# Public API

- Ping:

```
curl --location 'localhost:8080/api/ping
```

- Register:

```
curl -L 'localhost:8080/api/signup' -H 'Content-Type: application/json'
    --data-raw '{ "name": "", "email": "", "password": "" }'
```

- Login:

```
curl -L 'localhost:8080/api/login' -H 'Content-Type: application/json'
    --data-raw '{ "email": "", "password": "" }'
```


Bamboo Firewall

# Protected API

- Fetch API:

```
curl -L -X POST 'localhost:8080/api/v1/{type}/fetch'
    -H 'Authorization: Bearer <token>'
```

  - {type}: gnp, hep, gns, options
  - if type = options: add -d '{ "type": "", "label": "", "filter": [ ] }'

- Search API:

```
curl -L 'localhost:9091/api/v1/{type}/search'
    -H 'Content-Type: application/json'
    -H 'Authorization: Bearer <token>'
    -d '{ "options": [] }'
```

  - {type}: gnp, hep, gns

GHTK

Bamboo Firewall

# Protected API

- Statistic API:

  curl -L -X POST 'localhost:9091/api/v1/statistic/{type}'
  -H 'Authorization: Bearer <token>'

  - {type}: summary, project-summary

- Profile API:

  curl -L -X POST 'localhost:9091/api/v1/profile'
  -H 'Authorization: Bearer <token>'

  curl -L 'localhost:9091/api/v1/profile/update'
  -H 'Authorization: Bearer <token>'
  -H 'Content-Type: application/json' -d '{ "name": "" }'

# Admin API - User management

Required: Role = admin, -H 'Authorization: Bearer <token>'

- Fetch User:  curl -L -X POST 'localhost:9091/api/v1/admin/user/fetch'

- Create User: curl -L 'localhost:9091/api/v1/admin/user/create'
  -H 'Content-Type: application/json'
  --data-raw '{ "name": "","email": "", "password": "", "role": "admin" }'

- Delete User: curl -L 'localhost:9091/api/v1/admin/user/delete'
  -H 'Content-Type: application/json'  -d '{ ”id": "" }'

- Update User: curl -L 'localhost:9091/api/v1/admin/user/update'
  -H 'Content-Type: application/json'
  -d '{ "id": "", "name": "", "password": "", "role": "" }'


# Global Network Set

```yaml
apiVersion: projectcalico.org/v3
kind: GlobalNetworkSet
metadata:
  name: a-name-for-the-set
  labels:
    role: external-database
spec:
  nets:
    - 198.51.100.0/28
    - 203.0.113.0/24
```

- An arbitrary set of IP subnetworks/CIDRs
- When Calico is calculating the set of IPs that should match a source/destination selector, it includes the CIDRs from any network sets that match the selector.

# Calico host endpoint

```yaml
apiVersion: projectcalico.org/v3
kind: HostEndpoint
metadata:
  name: some.name
  labels:
    type: production
spec:
  interfaceName: eth0
  node: myhost
  expectedIPs:
    - 192.168.0.1
    - 192.168.0.2
  profiles:
    - profile1
    - profile2
  ports:
    - name: some-port
      port: 1234
      protocol: TCP
    - name: another-port
      port: 5432
      protocol: UDP
```

- Represents one or more real or virtual interfaces attached to a host that is running Calico.
- Enforces Calico policy on the traffic that is entering or leaving the host through those interfaces.


Bamboo Firewall

# HEP - interfaceName

```yaml
interfaceName: eth0
expectedIPs:
  - 192.168.0.1
  - 192.168.0.2
```

- interfaceName:
  - *: all host interfaces
  - eth0: name of the interface
  - empty and include expectedIPs: apply to interfaces have that IPs.

# HEP - profile

Group multiple endpoints so that they inherit a shared set of labels

```
profiles:
  - profile1
  - profile2

profiles:
  - projectcalico-default-allow
  - some-profile
```


Bamboo Firewall

# HEP - ports

```yaml
ports:
  - name: some-port
    port: 1234
    protocol: TCP
  - name: another-port
    port: 5432
    protocol: UDP
```

- Associates a name with a particular TCP/UDP/SCTP port of the endpoint
- Do not result in any change to the connectivity of the port

GHTK

Bamboo Firewall

# Global network policy

```yaml
apiVersion: projectcalico.org/v3
kind: GlobalNetworkPolicy
metadata:
  name: allow-tcp-6379
spec:
  selector: role == 'database'
  types:
    - Ingress
    - Egress
  ingress:
    - action: Allow
      metadata:
        annotations:
          from: frontend
          to: database
      protocol: TCP
      source:
        selector: role == 'frontend'
      destination:
        ports:
          - 6379
  egress:
    - action: Allow
```

- Ordered set of rules which are applied to a collection of endpoints that match a **label selector**


Bamboo Firewall

# GNP - Policy

Action:

- Allow | deny: allow, deny packets match selectors.
- Pass: skips to the next tier that contains a policy that applies to the endpoint, and processes the packet.
- Log - creates a log, and evaluation continues processing to the next rule

Protocol: TCP, UDP, ICMP, ICMPv6, SCTP, UDPLite, 1-255

Source: all(), global(), labels.

Destination: nets, selectors, ports