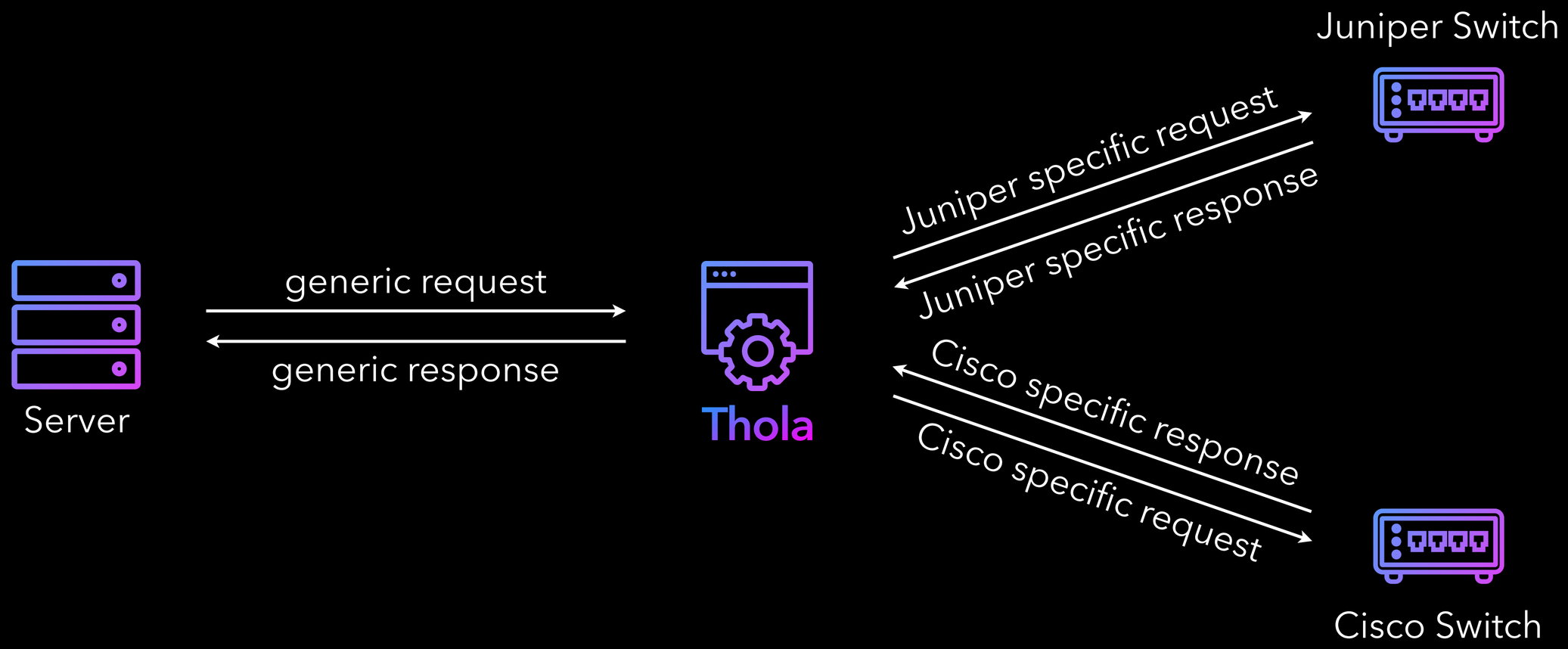
Introducing Thola

A unified interface for communication with network devices

Tobias Berdin · Mika Meyer · Niklas Schon

Overview


Juniper Switch
Juniper specific request
Juniper specific response
generic request
generic response
Server
Thola
Cisco specific response
Cisco specific request
Cisco Switch

Why is Thola that awesome?

- unified interface for communication with devices from Cisco, Juniper, Huawei, Nokia/ISAM and many more
- support for various devices like switches, routers, directional radio, UPS...
- easy way for adding additional device types

Why is Thola that awesome?

- usage of different protocols including SNMP, HTTP...
- compiled Go binary without dependencies
- low resource needs

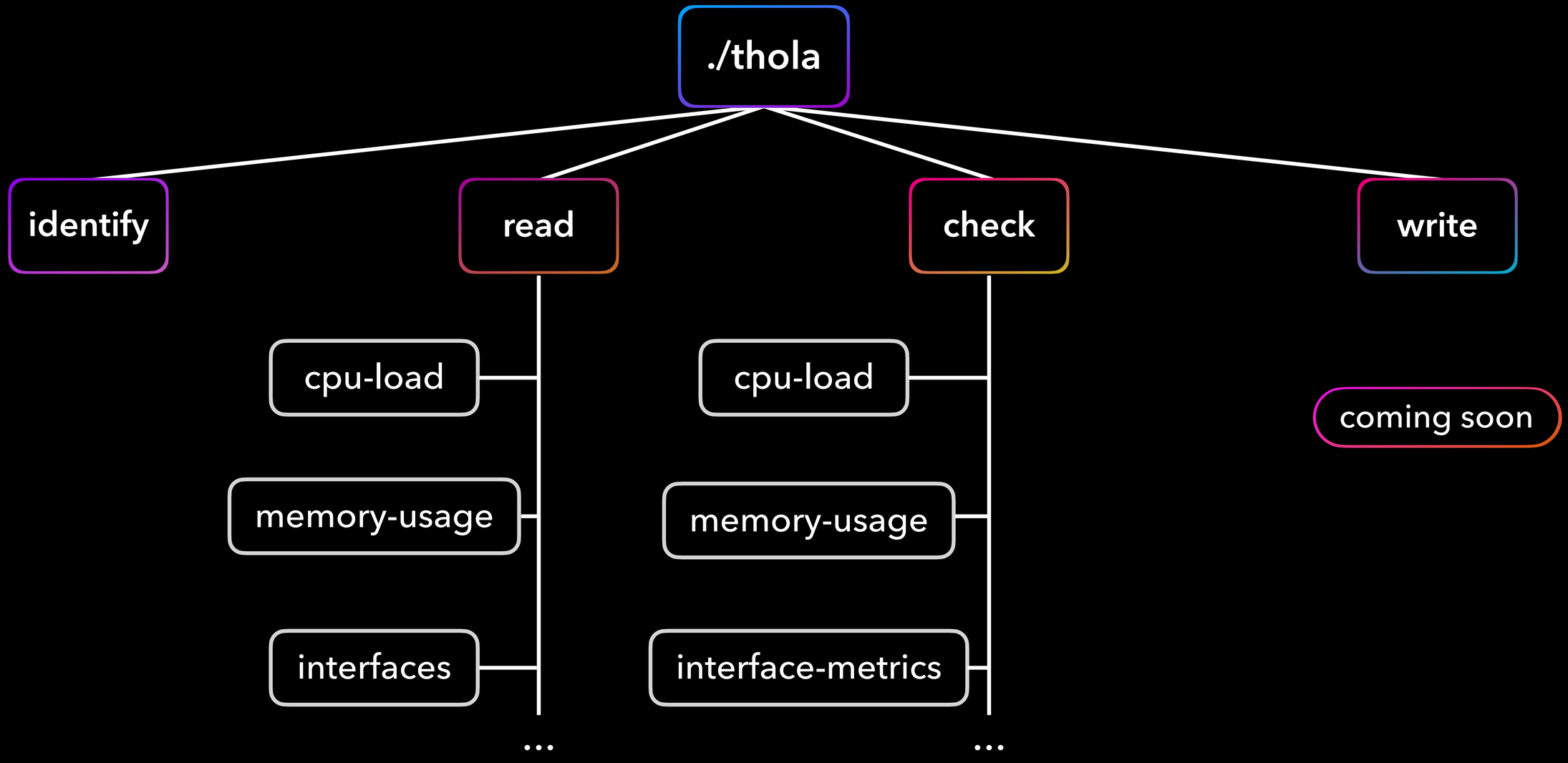
open source

How to use Thola

- Command Line Interface
- Rest API
- Check Plugin
- Integration in other monitoring systems coming soon

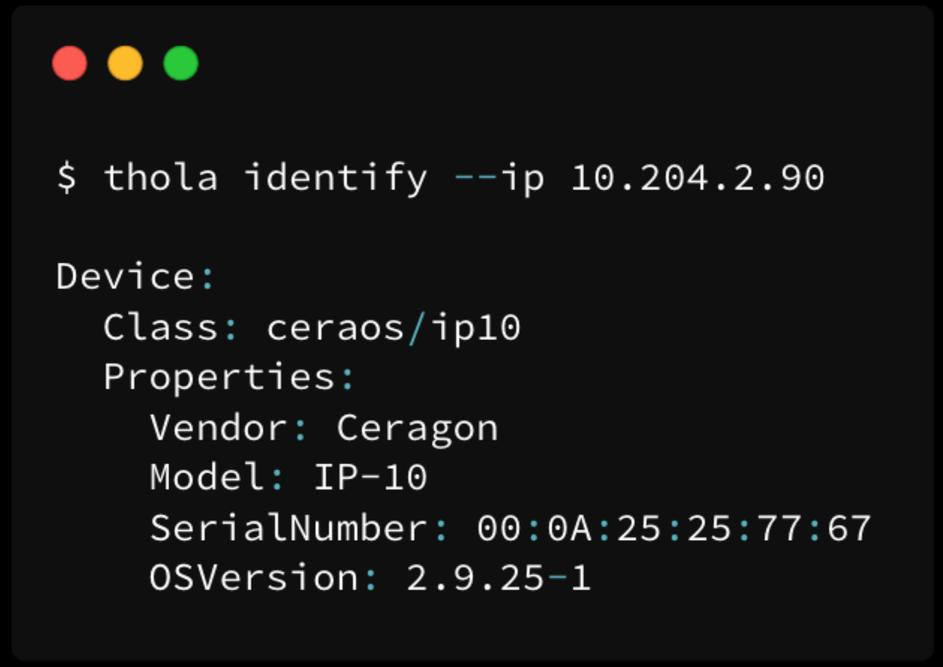
Operation modes

Operation modes



./thola identify

Automatically identify a device



```
$ thola identify --ip 10.204.2.90

Device:
  Class: ceraos/ip10
  Properties:
    Vendor: Ceragon
    Model: IP-10
    SerialNumber: 00:0A:25:25:77:67
    OSVersion: 2.9.25-1
```

./thola read interfaces

Read out special interface information

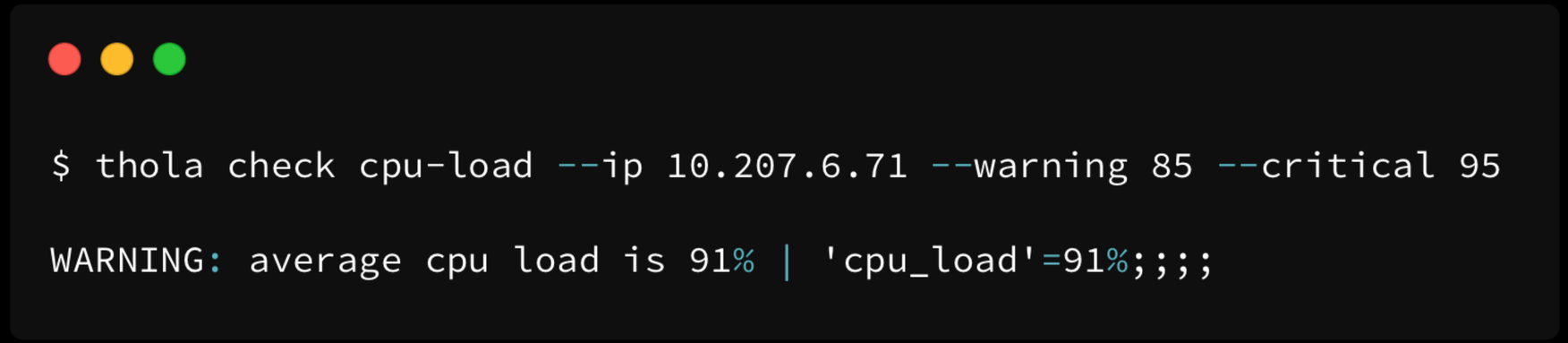
```
$ thola read interfaces --ip 10.204.2.90

Interfaces: [8]
  IfIndex: 1
  IfDescr: Radio Interface #0
  IfType: sonet
  IfMtu: 2430
  IfSpeed: 367000
  ...

  IfIndex: 5001
  IfDescr: Ethernet #7
  IfType: ethernetCsmacd
  IfMtu: 1548
  IfSpeed: 10000000
  IfPhysAddress: 00:0A:25:27:57:1E
  IfAdminStatus: up
  IfOperStatus: down
  ...
```

./thola check

- thola check output per default in check plugin format
- can be used in monitoring tools (e.g. Nagios, Icinga)

./thola check cpu-load

Check CPU usage



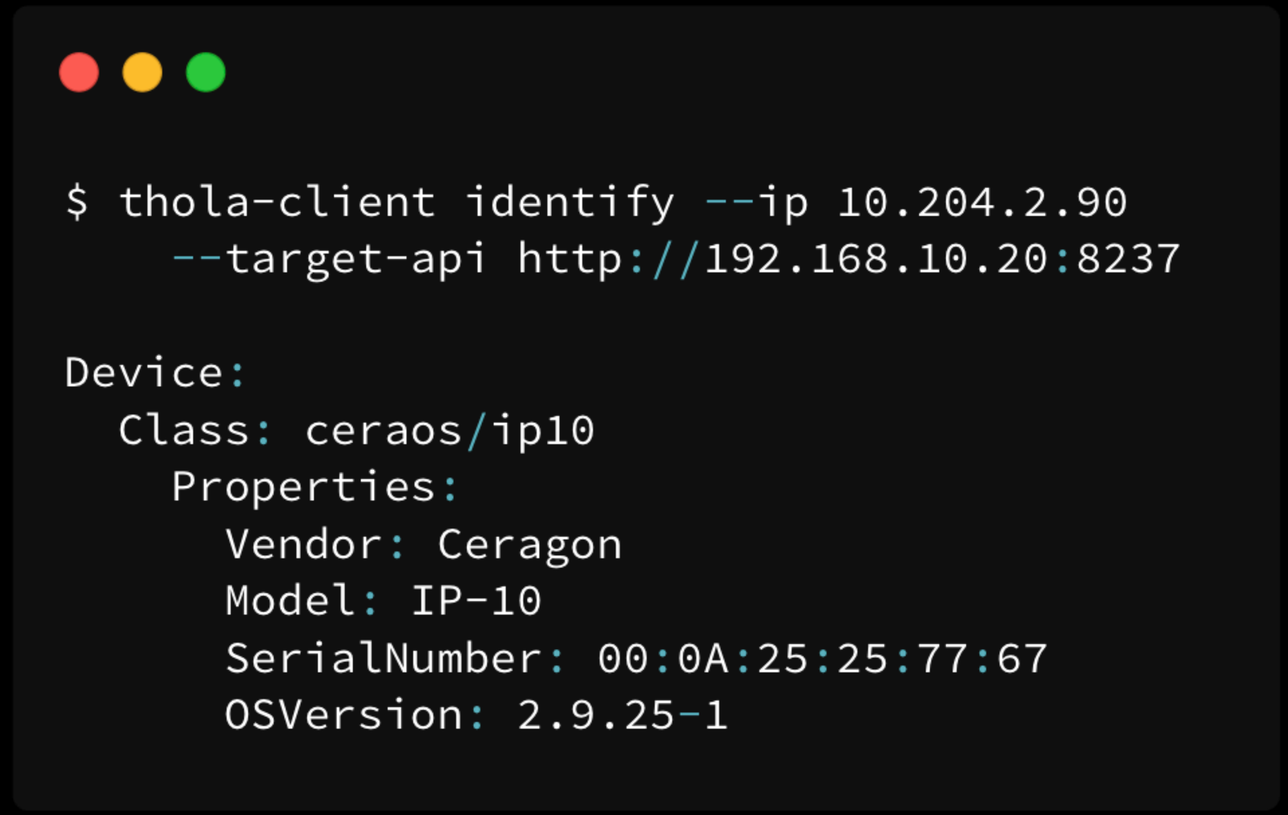
```
$ thola check cpu-load --ip 10.207.6.71 --warning 85 --critical 95

WARNING: average cpu load is 91% | 'cpu_load'=91%;;;;
```

./thola api

Start and configure the API

./thola-client

Let the API do the work

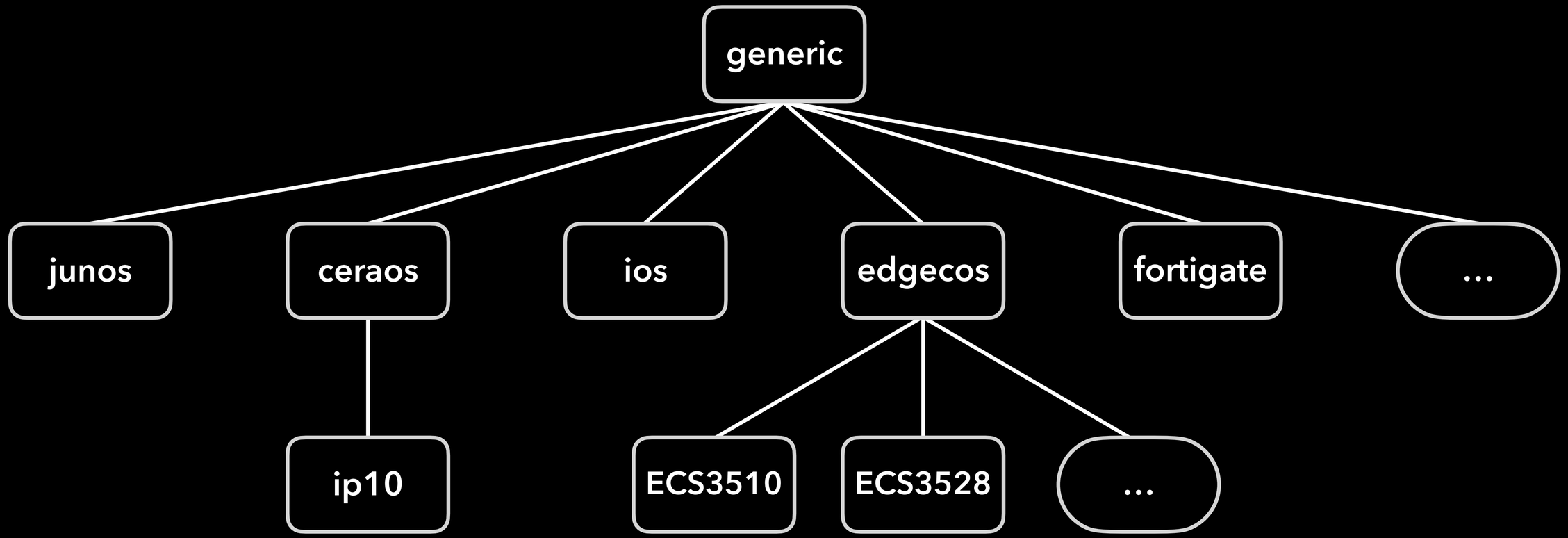


Device classes

Device class content

- conditions for assigning a device to this class
- available operations for the device
- specific network requests for generic operations

Device class hierarchy



Device classes in .yaml

- entire logic in .yaml configuration files
- easy to write
- not bound to a programming language
- embedded in binary
- can be extended with code

Example condition (ios)

```yaml
match:
  conditions:
    - type: SysObjectID
      match_mode: startsWith
      values:
        - .1.3.6.1.4.1.9.1.
    - type: SysDescription
      match_mode: regex
      values:
        - '(?i)^CISCO\s'
  logical_operator: OR
```

Example model identification (ios)

```yaml
model:
  - detection: snmpget
    oid: "1.3.6.1.2.1.47.1.1.1.1.13.1"
    operators:
      - type: modify
        modify_method: regexReplace
        regex: "(?i)^C(ISCO)?"
        replace: ""
```

Live Demo

Stay open source!

thola.io

github.com/inexio/thola

reddit.com/r/thola