# prometheus/blackbox\_exporter:

Blackbox prober exporter

https://github.com/prometheus/blackbox\_exporter

The blackbox exporter allows blackbox probing of endpoints over HTTP, HTTPS, DNS, TCP and ICMP.

## Building and running

### Local Build

make

./blackbox\_exporter <flags>

Visiting <http://localhost:9115/probe?target=google.com&module=http_2xx> will return metrics for a HTTP probe against google.com. The probe\_success metric indicates if the probe succeeded.

#编译blackbox\_exporter

go build .

### Building with Docker

docker build -t blackbox\_exporter .

docker run -d -p 9115:9115 --name blackbox\_exporter -v `pwd`:/config blackbox\_exporter -config.file=/config/blackbox.yml

## Configuration

A configuration showing all options is below:

modules:

http\_2xx\_example:

prober: http

timeout: 5s

http:

valid\_status\_codes: [] # Defaults to 2xx

method: GET

headers:

Host: vhost.example.com

Accept-Language: en-US

no\_follow\_redirects: false

fail\_if\_ssl: false

fail\_if\_not\_ssl: false

fail\_if\_matches\_regexp:

- "Could not connect to database"

fail\_if\_not\_matches\_regexp:

- "Download the latest version here"

tls\_config:

insecure\_skip\_verify: false

protocol: "tcp" # accepts "tcp/tcp4/tcp6", defaults to "tcp"

preferred\_ip\_protocol: "ip4" # used for "tcp", defaults to "ip6"

http\_post\_2xx:

prober: http

timeout: 5s

http:

method: POST

headers:

Content-Type: application/json

body: '{}'

tcp\_connect\_v4\_example:

prober: tcp

timeout: 5s

tcp:

protocol: "tcp4"

irc\_banner\_example:

prober: tcp

timeout: 5s

tcp:

query\_response:

- send: "NICK prober"

- send: "USER prober prober prober :prober"

- expect: "PING :([^ ]+)"

send: "PONG ${1}"

- expect: "^:[^ ]+ 001"

icmp\_example:

prober: icmp

timeout: 5s

icmp:

protocol: "icmp"

preferred\_ip\_protocol: "ip4"

dns\_udp\_example:

prober: dns

timeout: 5s

dns:

query\_name: "www.prometheus.io"

query\_type: "A"

valid\_rcodes:

- NOERROR

validate\_answer\_rrs:

fail\_if\_matches\_regexp:

- ".\*127.0.0.1"

fail\_if\_not\_matches\_regexp:

- "www.prometheus.io.\t300\tIN\tA\t127.0.0.1"

validate\_authority\_rrs:

fail\_if\_matches\_regexp:

- ".\*127.0.0.1"

validate\_additional\_rrs:

fail\_if\_matches\_regexp:

- ".\*127.0.0.1"

dns\_tcp\_example:

prober: dns

dns:

protocol: "tcp" # accepts "tcp/tcp4/tcp6/udp/udp4/udp6", defaults to "udp"

preferred\_ip\_protocol: "ip4" # used for "udp/tcp", defaults to "ip6"

query\_name: "www.prometheus.io"

HTTP, HTTPS (via the http prober), DNS, TCP socket and ICMP (v4 only, see permissions section) are currently supported. Additional modules can be defined to meet your needs.

## Prometheus Configuration

The blackbox exporter needs to be passed the target as a parameter, this can be done with relabelling.

Example config:

scrape\_configs:

- job\_name: 'blackbox'

metrics\_path: /probe

params:

module: [http\_2xx] # Look for a HTTP 200 response.

static\_configs:

- targets:

- prometheus.io # Target to probe

relabel\_configs:

- source\_labels: [\_\_address\_\_]

regex: (.\*)(:80)?

target\_label: \_\_param\_target

replacement: ${1}

- source\_labels: [\_\_param\_target]

regex: (.\*)

target\_label: instance

replacement: ${1}

- source\_labels: []

regex: .\*

target\_label: \_\_address\_\_

replacement: 127.0.0.1:9115 # Blackbox exporter.

## Permissions

The ICMP probe requires elevated privileges to function:

* Windows: Administrator privileges are required.
* Linux: root user or CAP\_NET\_RAW capability is required.
  + Can be set by executing setcap cap\_net\_raw+ep blackbox\_exporter
* BSD / OS X: root user is required.

# [question] Dynamic Prometheus configuration · Issue #2041 ·

prometheus/prometheus https://github.com/prometheus/prometheus/issues/2041

Great, file\_sd\_configs worked nicely with my blackbox job.

As far the refresh\_interval, the documentation says

# Refresh interval to re-read the files.

[ refresh\_interval: <duration> | default = 5m ]

... however the refresh rate for me was a couple of secs.

I tried to configure the refresh\_interval as:

scrape\_configs:

- job\_name: 'blackbox-http-2xx-secure'

metrics\_path: /probe

params:

module: [http\_2xx] # Look for a HTTP 200 response.

file\_sd\_configs:

- files:

- '/etc/prometheus/blackbox/\*.yml'

- refresh\_interval: 10s

relabel\_configs:

- source\_labels: [\_\_address\_\_]

regex: (.\*)(:80)?

target\_label: \_\_param\_target

replacement: ${1}

- source\_labels: [\_\_param\_target]

regex: (.\*)

target\_label: instance

replacement: ${1}

- source\_labels: []

regex: .\*

target\_label: \_\_address\_\_

replacement: 127.0.0.1:9115 # Blackbox exporter.

And it gives this error message:

time="2016-09-30T19:23:31Z" level=error msg="Error loading config: couldn't load configuration (-config.file=/etc/prometheus/prometheus.yml): file service discovery config must contain at least one path name" source="main.go:126"

it works fine without the refresh\_interval on the job configuration.

[**@hgontijo**](https://github.com/hgontijo) That is odd. Refreshes resulting from the refresh timer should execute the same code path as the initial load or refreshes triggered by changes to the files.

By the way: you probably do not need to set your refresh interval to 10s. The file\_sd SD mechanism automatically detects changes to any of the watched target files, so the refresh interval is only a safety guard in case a watched update ever gets lost due to some bug or something. Then it will still eventually get the new target state. Normally you should be able to keep it at 5 minutes or so.

[**@juliusv**](https://github.com/juliusv) the error message happens when I configure refresh\_interval under file\_sd\_configs and restart (stop/start) the service. As you suggested, I'll go without refresh\_interval since the file change detection works fine.

# Prometheus won't read file\_sd\_config · Issue #909

· prometheus/prometheus https://github.com/prometheus/prometheus/issues/909

I don't see errors upon my prometheus, version 0.17.0 (branch: stable, revision: 6e8d4e9). How do I check for an error?

I'm losing my mind why for example:

<http://s.natalian.org/2016-04-13/prometheus.yml>  
<http://s.natalian.org/2016-04-13/targets.yml>

Is not working.

## file\_sd\_configs

https://s.natalian.org/2016-04-13/prometheus.yml

scrape\_configs:

- job\_name: 'foo'

scrape\_interval: 10s

scrape\_timeout: 10s

metrics\_path: /probe

params:

module: [http\_2xx] # Look for a HTTP 200 response.

file\_sd\_configs:

- names:

- targets.yml

relabel\_configs:

- source\_labels: [\_\_address\_\_]

regex: (.\*)(:80)?

target\_label: \_\_param\_target

replacement: ${1}

- source\_labels: [\_\_param\_target]

regex: (.\*)

target\_label: instance

replacement: ${1}

- source\_labels: []

regex: .\*

target\_label: \_\_address\_\_

replacement: 10.8.1.113:9115 # Blackbox exporter.

https://s.natalian.org/2016-04-13/targets.yml

targets.yml

- targets:

- host1.example.com:9103

- host2.example.com:9103

- host3.example.com:9103

- host4.example.com:9103

- host5.example.com:9103

labels:

job: zookeeper