

# Essais du système de débogage des réseaux Globalping

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 28 août 2024

<https://www.bortzmeyer.org/globalping.html>

---

Si vous lisez ce blog régulièrement, vous savez que j'insiste souvent pour que, lorsqu'on teste un service réseau, on ne le fasse pas que depuis un seul point de mesure. L'Internet est vaste, et varié! Il faut donc utiliser plusieurs points de mesure. Si on ne travaille pas chez Google, on n'a probablement à sa disposition qu'un petit nombre de points à sa disposition et on utilise donc un système réparti de mesure. Vous le savez, je suis un grand fan des sondes RIPE Atlas <<https://atlas.ripe.net/>> mais il est toujours bon de regarder les alternatives comme Globalping <<https://www.jsdelivr.com/globalping>>, qui fait l'objet de cet article.

Alors, je ne vais pas arrêter d'utiliser le système des sondes RIPE Atlas <<https://atlas.ripe.net/>>, qui a bien plus de points de mesure et, surtout, qui est géré par une organisation sans but lucratif, contrôlée par ses membres. Mais la diversité sur l'Internet est une bonne chose, et il faut connaître les autres possibilités. Donc, Globalping <<https://www.jsdelivr.com/globalping>> est un réseau de points de mesures actives, utilisant l'infrastructure du CDN jsDelivr <<https://www.jsdelivr.com/>>, mesures qu'on peut déclencher soi-même. Je vous laisse regarder le site Web pour en savoir plus, je vais juste tester un peu.

Les mesures peuvent être déclenchées par l'interface Web <<https://www.jsdelivr.com/globalping>>, par un programme qu'on installe localement (n'étant pas fan de `curl install.sh | sudo bash`, je ne l'ai pas testé) ou par une API. Celle-ci semble bien fichue et bien documentée <<https://www.jsdelivr.com/docs/api.globalping.io>> donc je vais faire les essais avec elle. Dans tous les cas, c'est gratuit et il n'y a pas besoin de se créer un compte, ce qui est très rafraichissant dans un monde où les marketeux insistent pour que vous laissiez vos données personnelles pour tout et n'importe quoi. Je suppose que ces accès « anonymes » ont des limites, par exemple en nombre de mesures déclenchées, mais je n'ai pas étudié la question en profondeur.

Donc, commençons avec l'API. On fait une requête HTTP à `api.globalping.io` et la requête est évidemment un petit bout de JSON. Voici un exemple d'un script shell qui appelle curl pour demander une mesure IPv6 vers ce blog :

```
#!/bin/sh

curl --compressed --write-out '\nReturn code: %{http_code}\n' \
  --data-binary @- \
  --header "Content-Type: application/json" \
  https://api.globalping.io/v1/measurements \
<<EOF
{
  "limit": 5,
  "locations": [],
  "target": "www.bortzmeyer.org",
  "type": "ping",
  "measurementOptions": {
    "packets": 3,
    "ipVersion": 6
  }
}
EOF
```

Le script va vous afficher l'identificateur de cette mesure, vous pouvez la récupérer en ajoutant `https://api.globalping.io/v1/measurements/` devant. Ce sera du JSON, qu'on peut traiter avec les moyens classiques, par exemple `jq`, comme avec les sondes Atlas <[https://labs.ripe.net/author/stephane\\_bortzmeyer/processing-ripe-atlas-results-with-jq/](https://labs.ripe.net/author/stephane_bortzmeyer/processing-ripe-atlas-results-with-jq/)> :

```
% jq '"Average: " + (.results[].result.stats.avg | tostring) + " ms"' chDqJ8xB9HxCrpCe.json
"Average: 225.948 ms"
"Average: 223.1 ms"
"Average: 139.932 ms"
"Average: 35.985 ms"
"Average: 2.802 ms"
```

Ici, on affiche le temps d'aller-retour moyen pour chaque sonde. Je ne crois pas que le JSON produit soit documenté mais il est assez clair comme cela.

J'aime bien les éventuels messages d'erreur, très précis :

```
{
  "error": {
    "type": "validation_error",
    "message": "Parameter validation failed.",
    "params": {
      "measurementOptions.protocol": "\"measurementOptions.protocol\" is not allowed"
    }
  },
  "links": {
    "documentation": "https://www.jsdelivr.com/docs/api.globalping.io#post-/v1/measurements"
  }
}
```

On peut aussi faire des mesures DNS, par exemple avec cette demande en JSON :

---

<https://www.bortzmeyer.org/globalping.html>

```
{
  "limit": 5,
  "locations": [],
  "target": "qwant.com",
  "type": "dns",
  "measurementOptions": {
  }
}
```

Et si on demande dans le fichier résultant les données obtenues :

```
% jq '.results[].result.answers[].value' eghOnwP7HffbmGVa.json
"54.38.0.163"
"141.95.150.143"
"141.94.211.182"
"54.38.0.163"
"141.94.211.182"
...
```

Mais, apparemment, on ne peut pas récupérer la réponse DNS complète, seulement les parties analysées par leur logiciel.

Et avec HTTP? Demandons :

```
{
  "limit": 5,
  "locations": [],
  "target": "www.aemarielle.com",
  "type": "http",
  "measurementOptions": {
    "protocol": "HTTPS",
    "request": {
      "path": "/"
    }
  }
}
```

Et on obtient un beau fichier JSON dont on peut extraire, par exemple :

```
% jq '"Total time: " + (.results[].result.timings.total | tostring) + " ms"' iLBZulTOiyqXzQF1.json
"Total time: 1481 ms"
"Total time: 1157 ms"
"Total time: 421 ms"
"Total time: 1512 ms"
"Total time: 888 ms"
```

Et si on veut participer et avoir sa propre sonde? (J'ai oublié de dire que tout est apparemment en logiciel libre.) La solution suggérée est d'installer une sonde logicielle via Docker ou Podman. Bon, pour être franc, avec la commande indiquée, je n'ai pas réussi à le faire avec Podman :

---

<https://www.bortzmeyer.org/globalping.html>

```
WARN[0000] Using cgroups-v1 which is deprecated in favor of cgroups-v2 with Podman v5 and will be removed in the future
Error: failed to get new shm lock manager: failed to create 2048 locks in /libpod_lock: no such file or directory
```

Ou bien, sur une autre machine :

```
Error: short-name "globalping/globalping-probe" did not resolve to an alias and no unqualified-search registries are defined
```

On peut aussi demander une sonde matérielle mais il faut payer dix dollars par mois ce qui m'a semblé excessif.