

# receiver-module-webservice - /push\_check\_result

Le module écoute des requêtes HTTP pour ensuite effectuer des actions sur les hôtes/clusters ou checks concernés.

Un simple curl ou appel HTTP dans votre programme suffit pour envoyer des actions à Shinken.

## Soumettre le résultat de check passif sur un hôte ou un check

Paramètres de l'appel

Nom	Description
Méthode HTTP utilisé	POST
time_stamp	Date à laquelle a eu lieu la mesure afin que Shinken puisse déterminer la date de la "Dernière Vérification".
host_name	Nom de l'hôte cible
service_description	Nom du check cible, ou alors mettre le paramètre à vide si le résultat est pour l'hôte
return_code	Check: 0 => OK, 1 => WARNING, 2 => CRITICAL, 3 => UNKNOWN Hôte: 0 => OK, 1,2,3 => CRITIQUE
output	Résultat du check

Exemple:

```
curl -u user:password -X POST -d "time_stamp=$(date +%s)&host_name=host-checked&service_description=service-checked&return_code=0&output=Everything OK" http://shinken-srv:7760/push_check_result
```

Comme pour l'écriture d'un check, le résultat long doit être séparé du résultat par un retour à la ligne à l'aide du caractère "\n".

L'exemple suivant permet d'envoyer un résultat et un résultat long:

```
curl -u user:password -X POST -d '$host_name=host-checked&service_description=service-checked&return_code=0&output=short_update\nlong_output' http://shinken-srv:7760/push_check_result
```



On note dans l'exemple précédent que la chaîne de caractères passée dans le curl pour les données POST est passée avec :

```
$'chaîne'
```

et non

```
"chaîne"
```

pour ordonner au check d'interpréter les caractères d'échappement ANSI et passer un véritable '\n' au lieu de '\' suivi de 'n'.

Cette astuce fonctionne si le shell utilisé est bash et peut ne pas être utilisable dans d'autres shells, ou si la requête est envoyée via un autre outil ou via un script dans un autre langage.