

\$KEY\$ Frontend Traffic by SSH (pour le modèle HAproxy-by-SSH__frontend)

Sommaire

Contexte

Paramétrage

- Données utilisées provenant du modèle
 - Données communes pour les checks des modèles
 - Données spécifiques pour ce check
 - Les données DFE (Duplicate Foreach)
- Données utilisées provenant du check
 - Données globales
 - Propriétés de l'hôte

Résultat

- Exemple
- Interprétation des données
 - Statut
 - Résultat
 - Résultat long

Métriques

- Interprétation des métriques
- Exemple

Erreurs et pré-requis

- Erreurs de connexion (communes à tous les checks)
 - UNKNOWN – SSH authentication failed.
 - UNKNOWN – SSH private key could not be loaded
 - UNKNOWN HAProxy statistics query failed

Contexte

Le check **\$KEY\$ Frontend Traffic by SSH** permet de superviser l'activité réseau d'un frontend HAProxy.

Il collecte et analyse différents indicateurs de trafic afin de suivre l'utilisation du frontend et de détecter rapidement une anomalie ou une surcharge de celui-ci.

En fonction de l'activité des utilisateurs et des applications publiées, un frontend peut générer une charge réseau importante. Une évolution inhabituelle des métriques de trafic peut révéler :

- Une surcharge de connexions utilisateurs
- Une consommation réseau anormale
- Une attaque DDoS ou un trafic malveillant
- Un dysfonctionnement applicatif provoquant une hausse des requêtes
- Une saturation progressive des ressources du frontend.

Ce check permet donc :

- De surveiller le trafic entrant et sortant du frontend
- De suivre le taux de requêtes et de connexions actives
- D'estimer le nombre d'utilisateurs actuellement connectés
- D'identifier rapidement une montée en charge inhabituelle

Pour résumer, ce check est un indicateur clé pour assurer le bon fonctionnement et la stabilité du trafic applicatif exposé par HAProxy.

Statut	Nom de check	Résultat	Résultat Long																	
	[front_webservers] Frontend Status by SSH	 Frontend 'front_webservers' is healthy.	HAProxy Frontend Status :																	
			<table><tr><th>Name</th><th>Status</th><th>User Stats</th><th>Queue</th><th colspan="2">Backend(s)</th></tr><tr><td>front_webservers</td><td>OPEN</td><td>0.0% (0/262122)</td><td>—</td><td colspan="2">• backend_webservers_0</td></tr></table>						Name	Status	User Stats	Queue	Backend(s)		front_webservers	OPEN	0.0% (0/262122)	—	• backend_webservers_0	
			Name	Status	User Stats	Queue	Backend(s)													
			front_webservers	OPEN	0.0% (0/262122)	—	• backend_webservers_0													
			HTTP Response Breakdown :																	
<table><tr><th>Errors</th><th>1XX</th><th>2XX</th><th>3XX</th><th>4XX</th><th>5XX</th></tr><tr><td>Quantity</td><td>0</td><td>313</td><td>542104</td><td>552159</td><td>10206</td></tr></table>						Errors	1XX	2XX	3XX	4XX	5XX	Quantity	0	313	542104	552159	10206			
Errors	1XX	2XX	3XX	4XX	5XX															
Quantity	0	313	542104	552159	10206															

Paramétrage

Le check utilise la ligne de commande suivante :

```
$GLOBAL_HAPROXY-BY-SSH_SHINKEN_PLUGINDIR$/check_haproxy_SSH.py
-H "$HOSTADDRESS$"
-u "$_HOSTHAPROXY-BY-SSH__SSH-USER$"
-k "$_HOSTHAPROXY-BY-SSH__SSH-KEY$"
-x "$_HOSTHAPROXY-BY-SSH__SSH-PASSPHRASE$"
-p "$_HOSTHAPROXY-BY-SSH__SSH-PORT$"
--mode frontend_traffic
-d "$_HOSTHAPROXY-BY-SSH__EXCLUDE-PAUSED-DRAIN$"
-n "$ARG1$"
-w "$_HOSTHAPROXY-BY-SSH__FRONTEND__REQUEST-RATE-WARN$"
-c "$_HOSTHAPROXY-BY-SSH__FRONTEND__REQUEST-RATE-CRIT$"
```

Données utilisées provenant du modèle

Données communes pour les checks des modèles

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut à l'installation de Shinken	Description
HAPROXY-BY-SSH__SSH-KEY	l'Hôte (<i>Onglet Données</i>)	--	\$HAPROXY-BY-SSH__SSH-KEY\$	/var/lib/shinken/.ssh/id_rsa	Chemin vers la clé SSH privé de l'utilisateur shinken , sur le serveur hébergeant le Poller qui exécutera le check. Cette clé doit être présente dans les clés autorisées du compte utilisateur utilisé pour se connecter sur le serveur linux supervisé (voir la donnée SSH_USER ci dessous).
HAPROXY-BY-SSH__SSH-PASSPHRASE	l'Hôte (<i>Onglet Données</i>)	--	\$HAPROXY-BY-SSH__SSH-PASSPHRASE\$	SSH_KEY_PASSPHRASE\$	Phrase secrète utilisée pour déchiffrer la clé privée de l'utilisateur (<i>si celle-ci est protégée par une passphrase</i>). La clé privée déchiffré est ensuite utilisée pour authentifier l'utilisateur.
HAPROXY-BY-SSH__SSH-PORT	l'Hôte (<i>Onglet Données</i>)	--	\$HAPROXY-BY-SSH__SSH-PORT\$	22	Port de connexion SSH.
HAPROXY-BY-SSH__SSH-USER	l'Hôte (<i>Onglet Données</i>)	--	\$HAPROXY-BY-SSH__SSH-USER\$	shinken-user	Nom de l'utilisateur pour se connecter sur le serveur supervisé.

Données spécifiques pour ce check

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut à l'installation de Shinken	Description
HAPROXY-BY-SSH__FRONTEND__REQUEST-RATE-CRIT	l'Hôte (<i>Onglet Données</i>)	--	3000	3000	Cette donnée définit le nombre de requêtes reçu par seconde à partir duquel le check passe en CRITIQUE . Il suffit que la valeur franchisse ce seuil pour que le check change d'état.

HAPROXY-BY-SSH__FRONTEND__REQUEST-RATE-WARN	l'Hôte (Onglet Données)	--	2000	2000	Cette donnée définit le nombre de requêtes reçu par seconde à partir duquel le check passe en ATTENTION. Il suffit que la valeur franchisse ce seuil pour que le check change d'état.
---	------------------------------	----	------	------	---

Les données DFE (Duplicate Foreach)

Nom	Modifiable sur	Unités	Défaut	Valeur par défaut à l'installation de Shinken	Description
HAPROXY-BY-SSH__FRONTENDS	l'Hôte (Onglet Données)	---	frontend1\$(frontend1)\$	frontend1\$(frontend1)\$	Nom du ou des frontends (<i>séparés par une virgule</i>), exemple "main\$(main)\$,app\$(app)\$". Check(s) impacté(s) : [KEY\$] Frontend Status by SSH (pour le modèle HAproxy-by-SSH__frontend). [KEY\$] Frontend traffic by SSH (pour le modèle HAproxy-by-SSH__frontend). [KEY\$] Frontend Response by SSH (pour le modèle HAproxy-by-SSH__frontend).

Données utilisées provenant du check

Pas de données utilisées provenant du check

Données globales

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut à l'installation	Description
USERPLUGINS_DIR	Non modifiable (Sauf Admin Shinken)	--	/var/lib/shinken-user/libexec	/var/lib/shinken-user/libexec	Chemin absolu contenant les sondes installées par Shinken.
GLOBAL_HAPROXY-BY-SSH_SHINKEN_VENDOR	Non modifiable (Sauf Admin Shinken)	--	shinken-additional-packs	shinken-additional-packs	Dossier fourni par Shinken.
GLOBAL_HAPROXY-BY-SSH_SHINKEN_PACKNAME	Non modifiable (Sauf Admin Shinken)		HAproxy-by-SSH__shinken	HAproxy-by-SSH__shinken	Dossier contenant les sondes.
GLOBAL_HAPROXY-BY-SSH_SHINKEN_PLUGINSDIR	Non modifiable (Sauf Admin Shinken)	--	\$USERPLUGINS_DIR\$ /\$GLOBAL_HAPROXY-BY-SSH_SHINKEN_VENDOR\$ /\$GLOBAL_HAPROXY-BY-SSH_SHINKEN_PACKNAME\$	/var/lib/shinken-user/libexec/shinken-additional-packs/HAproxy-by-SSH__shinken	Chemin absolu du dossier contenant les sondes du pack HAproxy-by-SSH__shinken (<i>non modifiable</i>).

Propriétés de l'hôte

Nom	Modifiable sur	Unité	Défaut	Valeur par défaut	Description
HOSTADDRESS	l'Hôte (Onglet Général)	--	Nom de l'hôte	Nom de l'hôte	Adresse de l'hôte

Résultat

Exemple

Statut	Nom de check	Résultat	Résultat Long										
	[front_webserver] Frontend Status by SSH	 Frontend 'front_webserver' is healthy.	HAProxy Frontend Status :										
			<table><tr><th>Name</th><th>Status</th><th>User Stats</th><th>Queue</th><th>Backend(s)</th></tr><tr><td>front_webserver</td><td>OPEN</td><td>0.0% (0/262122)</td><td>—</td><td>• backend_webserver_0</td></tr></table>	Name	Status	User Stats	Queue	Backend(s)	front_webserver	OPEN	0.0% (0/262122)	—	• backend_webserver_0
			Name	Status	User Stats	Queue	Backend(s)						
			front_webserver	OPEN	0.0% (0/262122)	—	• backend_webserver_0						
			HTTP Response Breakdown :										
<table><tr><th>Errors</th><th>1XX</th><th>2XX</th><th>3XX</th><th>4XX</th><th>5XX</th></tr><tr><td>Quantity</td><td>0</td><td>313</td><td>542104</td><td>552159</td><td>10206</td></tr></table>	Errors	1XX	2XX	3XX	4XX	5XX	Quantity	0	313	542104	552159	10206	
Errors	1XX	2XX	3XX	4XX	5XX								
Quantity	0	313	542104	552159	10206								

Interprétation des données

Statut

Il peut prendre 4 états OK / CRITIQUE / ATTENTION / INCONNU .

- Le statut va dépendre du retour de sonde et de la configuration spécifique du check pour les données suivantes :
 - HAPROXY-BY-SSH_FRONTEND_REQUEST-RATE-WARN
 - HAPROXY-BY-SSH_FRONTEND_REQUEST-RATE-CRIT
- Voici un tableau récapitulatif du statut attendu suivant le retour de sonde :

Situation	Statut	Exemple																		
Le frontend sélectionné est dans l'état "STOP".	CRITIQUE	Statut Nom de check Résultat  [main] Frontend Traffic by SSH CRITICAL Collected traffic counters for frontend main, which is in a critical state.	Résultat Long HAProxy Frontend Traffic: main : <table><tr><th>Frontend</th><th>Inbound</th><th>Outbound</th><th>Req Rate</th><th>Conn. Rate</th><th>Active Users</th><th>Frontend State</th></tr><tr><td>main</td><td>102.6 MB</td><td>312.6 MB</td><td>—</td><td>—</td><td>0</td><td>STOP</td></tr></table> Active users are approximated with the current frontend sessions count reported by HAProxy ("scur").				Frontend	Inbound	Outbound	Req Rate	Conn. Rate	Active Users	Frontend State	main	102.6 MB	312.6 MB	—	—	0	STOP
Frontend	Inbound	Outbound	Req Rate	Conn. Rate	Active Users	Frontend State														
main	102.6 MB	312.6 MB	—	—	0	STOP														
- Le nombre de requêtes par seconde dépasse la valeur de : - HAPROXY-BY-SSH_FRONTEND_REQUEST-RATE-CRIT	CRITIQUE	Statut Nom de check Résultat  [main] Frontend Traffic by SSH CRITICAL Frontend main request rate is critical (2280/s).	Résultat Long HAProxy Frontend Traffic: main : <table><tr><th>Frontend</th><th>Inbound</th><th>Outbound</th><th>Req Rate</th><th>Conn. Rate</th><th>Active Users</th><th>Frontend State</th></tr><tr><td>main</td><td>136.5 MB</td><td>415.5 MB</td><td>2280/s</td><td>—</td><td>2400</td><td>OPEN</td></tr></table> Active users are approximated with the current frontend sessions count reported by HAProxy ("scur"). Request-rate thresholds are evaluated with "req_rate" (requests per second).				Frontend	Inbound	Outbound	Req Rate	Conn. Rate	Active Users	Frontend State	main	136.5 MB	415.5 MB	2280/s	—	2400	OPEN
Frontend	Inbound	Outbound	Req Rate	Conn. Rate	Active Users	Frontend State														
main	136.5 MB	415.5 MB	2280/s	—	2400	OPEN														
- Le nombre de requêtes par seconde dépasse la valeur de : - HAPROXY-BY-SSH_FRONTEND_REQUEST-RATE-WARN	ATTENTION	Statut Nom de check Résultat  [main] Frontend Traffic by SSH WARNING Frontend main request rate is over warning threshold (2444/s).	Résultat Long HAProxy Frontend Traffic: main : <table><tr><th>Frontend</th><th>Inbound</th><th>Outbound</th><th>Req Rate</th><th>Conn. Rate</th><th>Active Users</th><th>Frontend State</th></tr><tr><td>main</td><td>128.1 MB</td><td>389.8 MB</td><td>2444/s</td><td>—</td><td>2400</td><td>OPEN</td></tr></table> Active users are approximated with the current frontend sessions count reported by HAProxy ("scur"). Request-rate thresholds are evaluated with "req_rate" (requests per second).				Frontend	Inbound	Outbound	Req Rate	Conn. Rate	Active Users	Frontend State	main	128.1 MB	389.8 MB	2444/s	—	2400	OPEN
Frontend	Inbound	Outbound	Req Rate	Conn. Rate	Active Users	Frontend State														
main	128.1 MB	389.8 MB	2444/s	—	2400	OPEN														

Résultat

Le résultat court affiche, en une ligne, un résumé du trafic. Ce retour compact permet de pouvoir visualiser l'information même avec la taille des lignes réduites dans l'interface de visualisation.

Résultat long

Le résultat long affiche un tableau regroupant l'ensemble des informations sur le trafic sur le HAProxy.

Métriques

Interprétation des métriques

Nom de la métrique	Unité	Description	Seuil d'avertissement	Seuil critique
in_(frontend)	B	Retourne la valeur du flux en Bits par seconde transitants en entrée du HAProxy		
out_(frontend)	B	Retourne la valeur du flux en Bits par seconde transitants en sortie du HAProxy		

req_rate_(frontend)	--	Affiche le nombre de requêtes par minutes reçu par le HAProxy		
scur_(frontend)	--			

Exemple

Métrique	Valeur	Seuil d'avertissement	Seuil critique
bin_main	0B		
bout_main	0B		
req_rate_main	0		
scur_main	0		

Erreurs et pré-requis

Erreurs de connexion (communes à tous les checks)

UNKNOWN – SSH authentication failed.

L'authentification par clé SSH n'as pas pu s'effectuer.

Statut	Nom de check	Résultat	Résultat Long
	Global Status by SSH	UNKNOWN SSH authentication failed.	-

Résolution :

Possibles raisons :

- L'utilisateur utilisé n'existe pas
- La paire utilisateur / clef publique n'est pas autorisée à se connecter sur la machine supervisée.
- La clé publique n'a pas été déposée dans le "**authorized_keys**" côté serveur.

UNKNOWN – SSH private key could not be loaded

La clé SSH a utiliser sur le poller de Shinken n'as pas été trouvée.

Statut	Nom de check	Résultat	Résultat Long
	Global Status by SSH	UNKNOWN SSH private key could not be loaded.	-

Résolution :

Vérifier sur les pollers si la clé privé indiqué dans la donnée "\$_HOSTHAPROXY-BY-SSH__SSH-KEY\$" est bien présente et accessible par l'utilisateur shinken.

UNKNOWN HAProxy statistics query failed

L'utilisateur utilisé sur le serveur HAProxy n'appartient pas au groupe HAproxy ou le socket HAProxy n'est pas lancé.

Statut	Nom de check	Résultat	Résultat Long
	Global Status by SSH	UNKNOWN HAProxy statistics query failed.	-

Résolution :

La commande ci-dessous permet de lister les utilisateurs du groupe haproxy :

```
cat /etc/group | grep haproxy
```

Pour ajouter un utilisateur au groupe haproxy :

```
usermod -a -G haproxy user-service-shinken
```

Pour vérifier si le socket existe bien :

```
ls -l /var/run/haproxy.sock
```