

Comment créer un plugin (Uniquement pour ESIA Infinity)

Si vous avez optez pour la solution ESIA Infinity, vous pouvez développer vos propres plugins et les ajouter dans Esia !

Développement

Prérequis

Le plugin doit être un fichier qui peut être exécuté par le serveur. Il peut donc être codé en PHP, Perl, Python, ... Et utilisé des librairies indépendantes pour ces langages. À condition que ces langages et librairies soient installés sur le serveur.

Le plugin doit **toujours** renvoyer un des status suivants sous forme numérique :

Status du service	Valeur numérique
OK	0
Warning	1
Critical	2
Unknown	3

Message de sortie

Le résultat du **stdout** du plugin sera affiché dans l'interface ⇒ État des services.

Exemple pour le plugin check_icmp.

En ligne de commande sur le serveur esia :

copy

```
/usr/local/esia/plugins# ./check_icmp -H 10.5.0.1 -w 200,50% -c 500,100%
```

Aura pour résultat :

```
OK - 10.5.0.1: rta 1.837ms, lost 0%
```

Affichera dans l'interface web :

ETAT DES SERVICES				INFORMATIONS	ACTION
SERVICE	STATUS	DERNIÈRE EXÉCUTION			
PING	OK	12-03-2019 16:25:17	OK - 10.5.0.1: rta 1.823ms, lost 0%		 

Pour créer un stdout, en PHP par exemple, il suffit de faire un :

[copy](#)

```
echo "mon message de sortie";
```

Ne pas oublier de renvoyer le status sous forme numérique :

[copy](#)

```
exit(0);
```

Allez voir comment les autres plugins sont construits !

Sur votre serveur Esia dans le répertoire `/usr/local/esia/plugins/`

Nomenclature des plugins

Dans Esia, les noms des plugins sont toujours en lettres capitales et les mots séparés par des underscores. Veillez à donner des noms clairs et explicites à vos plugins de manière à savoir ce qu'ils font rien qu'en lisant leur nom.

Respectez le canevas suivant : `<type de plugin>_<protocole>_<plateforme>_<élément testé>`

Type de plugin	Description
GESA	Pour les plugins qui passent au travers d'une Unity pour trouver l'information recherchée. (ESIA Unity)
CHECK	Pour les plugins qui interrogent directement un nœud. (Principalement ESIA Infinity)
TRAP	Pour les plugins qui utilisent les Trap SNMP.
MAN	Pour les plugins de Management.
CORR	Pour les plugins de Corrélation. Ce sont des plugins qui travaillent sur les résultats d'autres plugins.

Exemples :

`GESA_SNMP_WINDOWS_STORAGE`

- GESA : Teste via le boitier Unity.
- SNMP : utilise le protocole SNMP.

- WINDOWS : pour tous les OS Windows.
- STORAGE : teste l'espace de stockage utilisé sur un lecteur.

CHECK_HTTPS

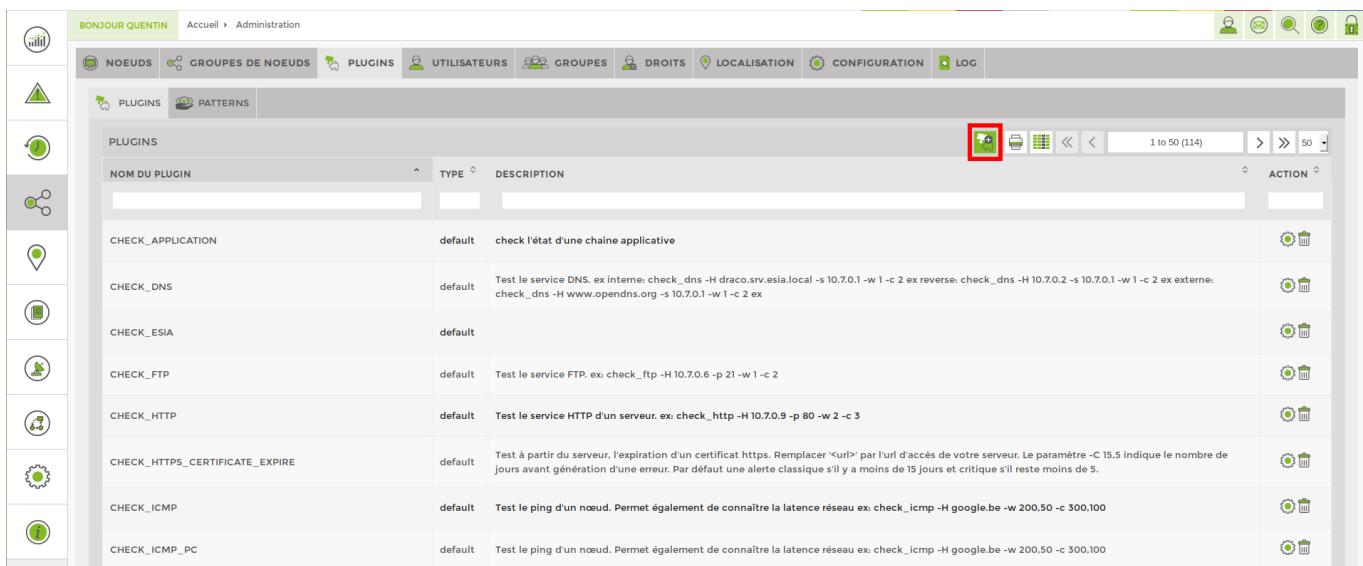
- CHECK : Teste directement depuis le serveur
- HTTPS : si un serveur HTTPS est présent

Il est important que vous respectiez ces règles quand vous ajoutez un nouveau plugin pour garder un ensemble cohérent.

Ajouter un plugin dans l'interface Esia

D'abord, copiez l'exécutable de votre plugin sur le serveur Esia dans le répertoire `/usr/local/esia/plugins/` (Vérifiez que le fichier a les droits d'exécution **X**)

Sur l'interface web, rendez-vous dans le menu « Administration  ». Allez dans « Plugins » puis cliquez sur l'icône  en haut à droite.



NOM DU PLUGIN	TYPE	DESCRIPTION	ACTION
CHECK_APPLICATION	default	check l'état d'une chaîne applicative	 
CHECK_DNS	default	Test le service DNS. ex interne: check_dns -H draco.srv.esia.local -s 10.7.0.1 -w 1 -c 2 ex reverse: check_dns -H 10.7.0.2 -s 10.7.0.1 -w 1 -c 2 ex externe: check_dns -H www.opendns.org -s 10.7.0.1 -w 1 -c 2	 
CHECK_ESIA	default		 
CHECK_FTP	default	Test le service FTP. ex: check_ftp -H 10.7.0.6 -p 21 -w 1 -c 2	 
CHECK_HTTP	default	Test le service HTTP d'un serveur. ex: check_http -H 10.7.0.9 -p 80 -w 2 -c 3	 
CHECK_HTTPS_CERTIFICATE_EXPIRE	default	Test à partir du serveur, l'expiration d'un certificat https. Remplacer <url> par l'url d'accès de votre serveur. Le paramètre -C 15.5 indique le nombre de jours avant génération d'une erreur. Par défaut une alerte classique s'il y a moins de 15 jours et critique s'il reste moins de 5.	 
CHECK_ICMP	default	Test le ping d'un nœud. Permet également de connaître la latence réseau ex: check_icmp -H google.be -w 200.50 -c 300,100	 
CHECK_ICMP_PC	default	Test le ping d'un nœud. Permet également de connaître la latence réseau ex: check_icmp -H google.be -w 200.50 -c 300,100	 

Cela vous ouvrira une nouvelle fenêtre.

AJOUTER UN PLUGIN X

Nom du plugin	CHECK_SNMP_LOAD
Chemin d'accès	check_snmp_load.pl
L'exécutable doit se trouver dans le répertoire /usr/local/esia/plugins/ (vérifiez que le fichier est exécutable)	
Type	pooling
OID de la trap SNMP	
Paramètres	-H \$IP -C \$SNMP_COM -w 80 -c 90
Intervalle par défaut entre 2 exécutions (en millisecondes)	300
Priorité	7
Description	Récupère la charge CPU utilisée. Fonctionne sous Linux, Windows, Esx, etc

[Annuler](#) [Ajouter](#)

Configurez le nouveau plugin :

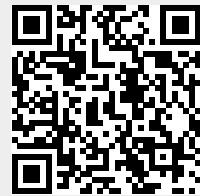
- **Nom du plugin** : Le nom qu'il aura dans l'interface. (Veuillez à respecter la [Nomenclature des plugins](#))
- **Chemin d'accès** : Le chemin et le nom du fichier exécutable depuis le répertoire sur le serveur /usr/local/esia/plugins/ (Vérifiez que le fichier a les droits d'exécution **X**).
Exemple : sur l'image ci-dessus notre exécutable check_snmp_load.pl se situe à la racine du répertoire /usr/local/esia/plugins/
- **Type** : *Cette option sera implémentée dans le futur.*
- **OID de la trap SNMP** : *Cette option sera implémentée dans le futur.*
- **Paramètres*** : Ici vous pouvez configurer les options par défaut à passer au plugin.
- **Intervalle*** : Temps par défaut entre les exécutions du plugin en millisecondes (300 = 5 minutes).
- **Priorité*** : Choisissez une priorité par défaut pour le test (Voir [Ajouter des services => Priorité des services](#)).

- **Description** : Que fait ce plugin.

* : Paramètres, Intervalle et Priorité sont des valeurs par défaut. Elles pourront être modifiées au cas par cas sur chaque nœud si nécessaire.

N'oubliez pas de cliquer sur « Ajouter » en bas à gauche pour sauvegarder le plugin.

From:
<https://wiki.esia-sa.com/> - **Esia Wiki**



Permanent link:
https://wiki.esia-sa.com/advanced/creer_plugin

Last update: **2023/03/03 15:26**