How to create a plugin (Only for ESIA Infinity)

If you have chosen the ESIA Infinity solution, you can develop your own plugins and add them to Esia!

Development

Prerequisites

The plugin must be a file that can be executed by the server. It can therefore be coded in PHP, Perl, Python, ... And use independent libraries for these languages. Provided that these languages and libraries are installed on the server.

The plugin must **always** return one of the following statuses in numerical form :

Service status	Numeric value
ОК	0
Warning	1
Critical	2
Unknown	3

Output message

The result of the **stdout** of the plugin will be displayed in the \Rightarrow Service status interface.

Example for the check_icmp plugin.

On the command line on the esia server:

сору

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

The result will be :

OK - 10.5.0.1: rta 1.837ms, lost 0%

Will display in the web interface :

ETAT DES SERVICES								
SERVICE	\$	status ^{\$}	DERNIÈRE EXÉCUTION	٥	INFORMATIONS	\$	ACTION	٥
PING		ок	12-03-2019 16:25:17		OK - 10.5.0.1: rta 1.823ms, lost 0%		()	

To create a stdout, in PHP for example, you just need to do a :

сору

echo "mon message de sortie";

Don't forget to return the status in numerical form:

сору

exit(0);

Check out how the other plugins are built!

On your Esia server in the directory /usr/local/esia/plugins/

Plugin nomenclature

In Esia, the names of plugins are always in capital letters and the words separated by underscores. Make sure you give your plugins clear, explicit names so that you know what they do just by reading their name.

Use the following format: <type de plugin>_<protocole>_<plateforme>_<élément testé>

Plugin type	Description
GESA	For plugins that go through a Unity to find the information they are looking for (ESIA Unity)
CHECK	For plugins that directly query a node (mainly ESIA Infinity).
TRAP	For plugins that use SNMP Trap.
MAN	For Management plugins.
CORR	For correlation plugins. These are plugins that work on the results of other plugins.

Examples:

GESA_SNMP_WINDOWS_STORAGE

- GESA: Tests via the Unity box.
- SNMP: uses the SNMP protocol.
- WINDOWS: for all Windows OS.
- STORAGE: tests the storage space used on a drive.

CHECK_HTTPS

- CHECK: Tests directly from the server.
- HTTPS: if an HTTPS server is present

It's important that you follow these rules when you add a new plugin to keep things

consistent.

Adding a plugin to the Esia interface

First, copy the executable of your plugin to the Esia server in the directory /usr/local/esia/plugins/ (Check that the file has execution rights **X**)

On the web interface, go to the menu « Administration 💿 ». Go to « Plugins » then click on the 🔛 at the top right.

	ONJOUR QUENTIN Accueil + Administration	0
	📄 NOEUDS 📽 GROUPES DE NOEUDS 🏷 PLUGINS 🚊 UTILISATEURS 🕮 GROUPES 🛔 DROITS 🕅 LOCALISATION 🍈 CONFIGURATION 🧧 LOC	
	N PLUGINS DATTERNS	
	PLUGINS	> 50 -
-	NOM DU PLUGIN ^ TYPE ⁰ DESCRIPTION ⁰ ACT	rion ^{\$}
€C		
	CHECK_APPLICATION default check l'état d'une chaine applicative	۵
\sim	CHECK_DNS default defa	ê 💼
	uieus_uis *n www.upeiulis.org *s iuu.i *w i *u ∠ ex	~
~	CHECK_ESIA default	
	CHECK_FTP default Test le service FTP. ex: check_ftp -H 10.7.0.6 -p 21 -w 1 -c 2	0
(3)	CHECK_HTTP default Test le service HTTP d'un serveur. ex: check_http -H 10.7.0.9 -p 80 -w 2 -c 3	9 💼
٢	CHECK_HTTPS_CERTIFICATE_EXPIRE default Test à partir du serveur, l'expiration d'un certificat https. Remplacer <url> par l'uri 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.</url>	0
	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	۵ 💼
	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	۵ 🗊

This will open a new window.

AJOUTER UN PLUGIN		×
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)	
Туре	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 <u>windows</u> ,Esx,etc	
Annuler Ajouter		

Configure the new plugin:

- **Plugin name**The name it will have in the interface. Plugin Nomenclature)
- **Access path**The path and name of the executable file from the directory on the server /usr/local/esia/plugins/ (Check that the file has execution rights **X**).

ExampleOn the image above, our executable check_snmp_load.pl is located at the root of the /usr/local/esia/plugins/

- **Type**: This option will be implemented in the future.
- **SNMP trap OID**: This option will be implemented in the future.
- **Settings***: Here you can configure the default options to be passed to the plugin.
- Interval*: Default time between plugin executions in milliseconds (300 = 5 minutes).
- **Priority*** Choose a default priority for the test (See Adding services => Priority of services).
- **Description**What this plugin does.

* Parameters, Interval and Priority are default values. They can be modified on a case-by-case basis for each node if necessary.

Don't forget to click on "Add" at the bottom left to save the plugin.

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

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

Last update: **2023/11/09 18:18**

