

Health Check command

shinken-healthcheck

This tool is used to check the state of your Shinken Enterprise installation and configuration.

If launch on the arbiter daemon, will check all the daemons of your architecture (simulate shinken-global-healthcheck)

If launch from a distant server (scheduler, poller, etc...) only the local daemons will be check (simulate shinken-local-healthcheck)

Usage:

shinken-healthcheck

Options:

Name	Description
--version	Show program's version number and exit.
-h, --help	Show this help message and exit.
-f, --file	Write output that can be send to the Shinken Solutions team. Support file is available at /root/shinken-healthcheck_date_hour

Sample Output

```
#####
This tool is used to check the state of your Shinken Enterprise (02.03.03.U01) installation and configuration
Note: Global check launch as launch from a arbiter server
#####
Versions:
  Original installed version: 02.03.03.U01-137.fr
  Updated version           : 02.03.03.U01-137.fr
#####
[.....] 100%
Architecture
-----
|Realm All/|
-----
      |In All|
      -----
- localhost (127.0.0.1):
  ^^^^^^^^^^^^^^^^^^^^^
    [receiver: receiver-1]
      AT RISK: receiver-1 is defined with the localhost address. It is a problem in distributed
mode. Please configure it with the LAN IP/FQDN address instead
      OK:      Connection to daemon is OK at port 7773
      OK:      Daemon version is: 02.03.03.U01-137.fr
    [reactionner: reactionner-master]
      AT RISK: reactionner-master is defined with the localhost address. It is a problem in
distributed mode. Please configure it with the LAN IP/FQDN address instead
      OK:      Connection to daemon is OK at port 7769
      OK:      Daemon version is: 02.03.03.U01-137.fr
    [broker: broker-master]
      AT RISK: broker-master is defined with the localhost address. It is a problem in
distributed mode. Please configure it with the LAN IP/FQDN address instead
      OK:      Connection to daemon is OK at port 7772
      OK:      Daemon version is: 02.03.03.U01-137.fr
  Modules
      OK:      Name: Livestatus           Type: livestatus
      OK:      Name: Simple-log          Type: simple-log
      OK:      Name: WebUI               Type: webui
```

```

                OK:      Name: Graphite-Perfdata      Type: graphite_perfdata
                OK:      Name: sla                   Type: sla
- 127.0.0.1 (127.0.0.1):
^^^^^^^^^^^^^^^^^^^^
    [poller: poller-central]
        OK:      Connection to daemon is OK at port 7771
        OK:      Daemon version is: 02.03.03.U01-137.fr
- localhost (192.168.56.101):
^^^^^^^^^^^^^^^^^^^^
    [scheduler: scheduler-central]
        OK:      Connection to daemon is OK at port 7768
        OK:      Daemon version is: 02.03.03.U01-137.fr
        Modules
            OK:      Name: PickleRetentionFile      Type: pickle_retention_file
-----
|Realm All/Extra/|
-----

-----
|In Extra|
-----

- 192.168.56.102 (192.168.56.102):
^^^^^^^^^^^^^^^^^^^^
    [poller: poller-distant]
        AT RISK: Cannot get module details from your daemon (HTTP Error 404: Not Found).
Please update it.
            OK:      Connection to daemon is OK at port 7771
-----
|Realm All/Extra/Extra2/|
-----

-----
|In Extra2|
-----

- 192.168.56.102 (192.168.56.102):
^^^^^^^^^^^^^^^^^^^^
    [scheduler: scheduler-distant]
        AT RISK: Cannot get module details from your daemon (HTTP Error 404: Not
Found). Please update it.
            OK:      Connection to daemon is OK at port 7768
- localhost (127.0.0.1):
^^^^^^^^^^^^^^^^^^^^
    [arbiter: synchronizer-master]
        AT RISK: synchronizer-master is defined with the localhost address. It is a
problem in distributed mode. Please configure it with the LAN IP/FQDN address instead
            OK:      Connection to daemon is OK at port 7765
            OK:      Daemon version is: 02.03.03.U01-137.fr
        Modules
            OK:      Name: Cfg_password              Type: cfg_password_webui
            OK:      Name: syncui-import             Type: syncui-import
            OK:      Name: cfg-file-shinken          Type: cfg-file-import
            OK:      Name: active-dir                Type: active-dir-import
            OK:      Name: sync-vmware               Type: sync-vmware
            OK:      Name: cfg-file-nagios           Type: cfg-file-import
            OK:      Name: discovery-import          Type: discovery-import
            OK:      Name: ip-tag-dmz                Type: sync_ip_tag
            OK:      Name: sync-regexptag            Type: sync-regexptag
    [arbiter: arbiter-master]
        AT RISK: arbiter-master is defined with the localhost address. It is a
problem in distributed mode. Please configure it with the LAN IP/FQDN address instead
            OK:      Connection to daemon is OK at port 7770
            OK:      Daemon version is: 02.03.03.U01-137.fr
        Modules
            OK:      Name: synchronizer-import        Type: synchronizer-import
Licence key
    ERROR: The licence key is invalid
    ERROR: No licence key. Trial mode Node limits : 20
Local libraries
    [gevent]
        OK:      Library gevent is available. Version: 0.13.8
    [pycurl]
        OK:      Library pycurl is available. Version: libcurl/7.19.7 NSS/3.16.2.3 Basic ECC zlib/1.2.3 libidn
/1.18 libssh2/1.4.2

```

```

[pymongo]
  OK:      Library pymongo is available. Version: 2.9.2
[ldap]
  OK:      Library ldap is available. Version: 2.3.10
Modules
  [WebUI]
    OK:      Auth_secret is a custom variable
    OK:      Mongodb server is available at: mongodb://localhost/?safe=true
  [Synchronizer]
    OK:      Mongodb server is available at: mongodb://localhost/?safe=false
    OK:      Auth_secret is a custom variable
    OK:      The master_key is a custom variable
Storage
  [graphite]
    OK:      Graphite server 127.0.0.1 is available with 1 top level elements
    OK:      Graphite server localhost:2003 is available

```

Output parts

The healthcheck output is separated into several parts:

- **Architecture:** about your daemons configuration
- **Libs:** about installed libraries need by Shinken Enterprise
- **Licence key:** all about your licence key
- **Modules:** information about UIs
- **Storage:** information about the mongodb and graphite storage system

There are 3 level for output:

- **OK:** all is valid
- **At Risk:** it works, but something can go wrong in the future if you don't fix something
- **Error:** this part is in error and must be fix now

shinken-local-healthcheck

This tool is used to check the state of your local Shinken Enterprise installation and configuration.

With this command, only the local installation and daemons will be checked.

Usage:

```
shinken-local-healthcheck
```

Options:

Name	Description
--version	Show program's version number and exit.
-h, --help	Show this help message and exit.
-f, --file	Write output that can be send to the Shinken Solutions team. Support file is available at /root/shinken-healthcheck_date_hour

Output parts

The healthcheck output is separated into several parts:

- **Architecture:** about your daemons configuration

- **Libs:** about installed libraries need by Shinken Enterprise
- **Licence key:** all about your licence key
- **Modules:** information about UIs, if launch on a broker daemon
- **Storage:** information about the mongodb and graphite storage system, if need by daemons

There are 3 level for output:

- **OK:** all is valid
- **At Risk:** it works, but something can go wrong in the future if you don't fix something
- **Error:** this part is in error and must be fix now

shinken-global-healthcheck

This tool is used to check the state of your local Shinken Enterprise installation and configuration.

With this command, all the local and distant daemons will be checked. This should be launched only for the central (synchronizer/arbiter) daemon.

Usage:

```
shinken-global-healthcheck
```

Options:

Name	Description
--version	Show program's version number and exit.
-h, --help	Show this help message and exit.
-f, --file	Write output that can be send to the Shinken Solutions team. Support file is available at /root/shinken-healthcheck_date_hour

Output parts

The healthcheck output is separated into several parts:

- **Architecture:** about your daemons configuration
- **Libs:** about installed libraries need by Shinken Enterprise
- **Licence key:** all about your licence key
- **Modules:** information about UIs, if launch on a broker daemon
- **Storage:** information about the mongodb and graphite storage system, if need by daemons

There are 3 level for output:

- **OK:** all is valid
- **At Risk:** it works, but something can go wrong in the future if you don't fix something
- **Error:** this part is in error and must be fix now