

SAP HANA Monitoring

A Nagios Plugin for monitoring SAP HANA. This turns your Nagios into a powerful monitoring for SAP HANA.

This Plugin allows to monitor

- CPU Usage
- Memory Usage
- Volume Usage
- License (Valid and Usage)
- Alerts
- Backups
- Running services
- Replication status

This Plugin is written in Java and uses the following libraries:

- commons-cli 1.3.1 (To get Command Line Input)
- ngdbc 1.100.0 (JDBC library from SAP)
- joda-time 2.7 (for time functions with java <= 1.6)

Connections to SAP HANA are always done by JDBC based on the ngdbc library. The Monitoring is done by SQL Queries.

Requirements

- Working Monitoring Solution (Nagios compatible)
- Working SAP HANA System (HANA 1.0 and HANA 2.0 are supported)
- Database user with access to SYS.* and _SYS_STATISTICS.* views
- Connectivity from Monitoring host to SAP HANA host by tcp port 3< INSTANCE >15 (default).
- Java (built and tested with 1.7)

Install

To install just copy the jar file to your libexec folder.

```
cp JAR-File.jar /your/nagios/libexec/folder/  
cp check_hana /your/nagios/libexec/folder/
```

Adjust the paths in the file `check_hana` according to your installation.

Usage

Add your commands to Nagios like this:

```
define command {  
    command_name    hana_check  
    command_line    /your/nagios/libexec/folder/check_hana -H $HOSTADDRESS$ -U $_HOSTDB_USER$ -P '$_HOSTDB_PWD$'  
    -i $_HOSTSAP_SYSNR$ -C $ARG1$  
}  
  
define command {  
    command_name    hana_check_backup  
    command_line    /your/nagios/libexec/folder/check_hana -H $HOSTADDRESS$ -U $_HOSTDB_USER$ -P '$_HOSTDB_PWD$'  
    -i $_HOSTSAP_SYSNR$ -C $ARG1$ -T $ARG2$  
}  
  
define command {  
    command_name    hana_check_threshold  
    command_line    /your/nagios/libexec/folder/check_hana -H $HOSTADDRESS$ -U $_HOSTDB_USER$ -P '$_HOSTDB_PWD$'
```

```
-i $_HOSTSAP_SYSNR$ -C $ARG1$ -w $ARG2$ -c $ARG3$  
}
```

These sample commands use Host Macros. Please check the Nagios documentation if you like to get details about Host Macros.

Monitoring Commands explained

SAP HANA Alerts `alert`

This Command will check for SAP HANA Alerts. If alerts are found this command will return CRITICAL. To not report alerts more than once, the timestamp of the last Alert will be stored. The next run will continue from this timestamp. If no Alert is found it returns state OK.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C alert
```

SAP HANA Data Backup `backupdata`

This Command will check for SAP HANA Data Backups. It checks for Backups successful since given Time `-T` in Seconds. If there are some Backups not successful the state will be CRITICAL. If no backups found the state will also be CRITICAL. Default Value for `-T` is 86400 seconds.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C backupdata -T 86400
```

SAP HANA Data Backup `successdatabackup`

Check for successful Data backups. Return state CRITICAL if there is no backup with state "successful" since given time from `-T`. Default Value for `-T` is 86400 seconds.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C successdatabackup -T 86400
```

SAP HANA Log Backup `backupdataalongrunner`

This Command will check for SAP HANA Data Long Running Backups. It checks for Backups running since given Time `-T` in Seconds. If there are some Backups the state will be CRITICAL. Default Value for `-T` is 86400 seconds.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C backupdataalongrunner -T
```

SAP HANA Log Backup `backuplog`

This Command will check for SAP HANA Log Backups. It checks for Backups successful since given Time `-T` in Seconds. If there are some Backups not successful the state will be CRITICAL. If no backups are found the state will also be CRITICAL. Default Value for `-T` is 900 seconds.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C backuplog -T 900
```

SAP HANA Log Backup `backuploglongrunner`

This Command will check for SAP HANA Log Long Running Backups. It checks for Backups running since given Time `-T` in Seconds. If there are some Backups the state will be CRITICAL. Default Value for `-T` is 86400 seconds.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C backuploglongrunner -T 900
```

SAP HANA CPU Usage `cpu`

This Command will check for CPU usage on SAP HANA Systems. You can use thresholds to set the acceptable Usage. Thresholds are in percent. This Check is similar to the SAP HANA CPU Usage in HANA Studio.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C cpu -w 80 -c 90
```

SAP HANA Used Memory `mem`

This Command will check for Memory Usage on SAP HANA Systems. You can use thresholds to set the acceptable Usage. Thresholds are in percent. This Check is similar to the SAP HANA Used Memory in HANA Studio.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C mem -w 80 -c 90
```

SAP HANA Resident Memory `memresident`

This Command will check for Resident Memory Usage on SAP HANA Systems. You can use thresholds to set the acceptable Usage. Thresholds are in percent. This Check is similar to the Resident Memory in HANA Studio.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C memresident -w 80 -c 90
```

SAP HANA License `license`

This Command will check if the license is valid on the SAP HANA System. You can use thresholds to set the warning and critical limit. Thresholds are in days.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C license -w 20 -c 15
```

SAP HANA License Usage `licusage`

This Command will check if the amount of used memory is still valid for the installed license on the SAP HANA System. You can use thresholds to set the acceptable Usage. Thresholds are in percent.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C licusage -w 80 -c 90
```

SAP HANA Data Disk Usage `data`

This Command will check if the Data Volume Size is OK. You can use thresholds to set the acceptable Usage. Thresholds are in percent. This Check is similar to the Data Volume Size in HANA Studio.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C data -w 80 -c 90
```

SAP HANA Log Disk Usage `log`

This Command will check if the Log Volume Size is OK. You can use thresholds to set the acceptable Usage. Thresholds are in percent. This Check is similar to the Log Volume Size in HANA Studio.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C log -w 80 -c 90
```

SAP HANA Trace Disk Usage `trace`

This Command will check if the Trace Volume Size is OK. You can use thresholds to set the acceptable Usage. Thresholds are in percent. This Check is similar to the Trace Volume Size in HANA Studio.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C trace -w 80 -c 90
```

SAP HANA Services `services`

This Command will check if the services for SAP HANA are running. The state CRITICAL will be returned if one of the default mandatory services (nameserver, indexserver and preprocessor) are not running. If other services are not running the state WARNING will be returned. You can overwrite the mandatory services by using `-m`.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C services  
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C services -m nameserver,indexserver
```

SAP HANA Replication **replication**

This Command will check if your HANA System has a working replication and the given host is defined as secondary.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C replication -S secondary-host
```

SAP HANA Version **version**

This Command will check the SAP HANA Version. It reports state OK when the connection was successful.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -i 00 -C version
```

Nice to know

This plugin also supports SAP HANA multitenant Installations. Just append the database name with **-d**.

Example:

```
./check_hana -H hana-host -U SYSTEM -P password -p 30115 -C successlogbackup -T 1800 -d DATABASE
```

Commands supporting Multitenancy

- successdatabackup
- successlogbackup
- memresident
- services
- data
- log
- trace