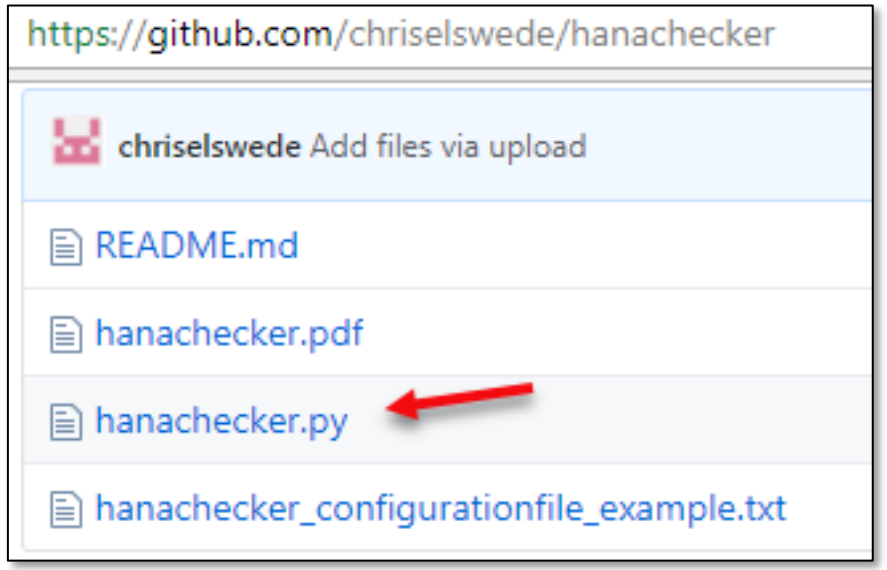
Monitoring

HANAChecker – SAP Note 1999993

SAP Note 1999993 presents a tool that runs the mini-checks (SAP Note 1969700) and sends out emails in case of potential critical issues

1999993 - How-To: Interpreting SAP HANA Mini Check Results

- It is a python script to be downloaded from https://github.com/chriselswede/hanachecker
- It is intended to be executed as <sid>adm on your SAP HANA Server
- It connects via host, port and DB user, provided in hdbuserstore
- It sends out emails via an intern smtp server (to avoid firewall issues) with xmail



NOTE: You have to install the linux program "sendmail" and add a line similar to `DSsmtp.intra.ourcompany.com` in the file sendmail.cf in /etc/mail/, see this page

Monitoring

HANAChecker – using hdbuserstore

Host, port and DB user needs to be provided in the hdbuserstore:

```
hsiadm@dewdfglp00835:/> hdbuserstore SET HANACHECKER1KEY dewdfglp00835:30015 HANACHECKER1 PassWord1
hsiadm@dewdfglp00835:/> hdbuserstore LIST
DATA FILE       : /usr/sap/HSI/home/.hdb/dewdfglp00835/SSFS_HDB.DAT
KEY FILE        : /usr/sap/HSI/home/.hdb/dewdfglp00835/SSFS_HDB.KEY

KEY HANACHECKER1KEY
  ENV : dewdfglp00835:30015
  USER: HANACHECKER1
```

Then the hanachecker can connect using the info stored in hdbuserstore:

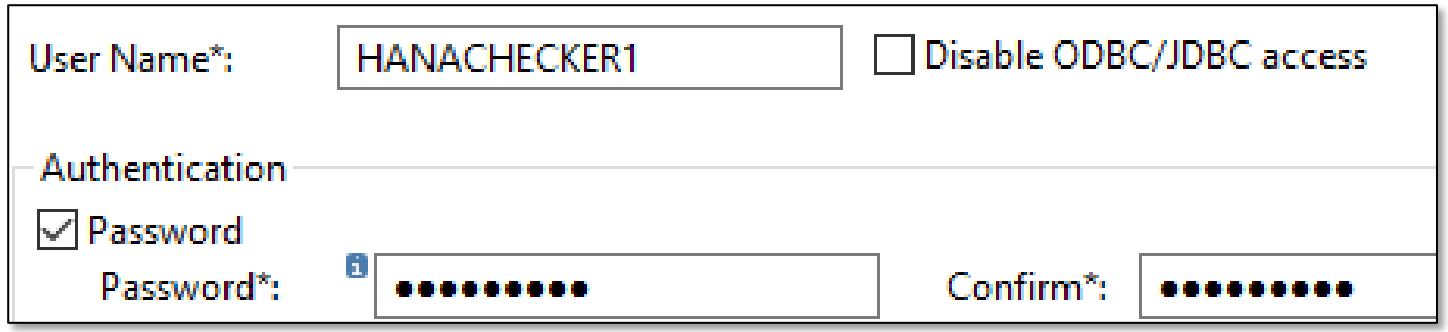
```
hsiadm@dewdfglp00835:/tmp/HANAChecker> whoami
hsiadm
hsiadm@dewdfglp00835:/tmp/HANAChecker> python hanachecker.py -k HANACHECKER1KEY -ff hanachecker_configurationfile.txt
"Mini Check ID 235  Description: Hosts with varying physical memory size  Host:   Value: yes  Expectation: no  Potenti
```

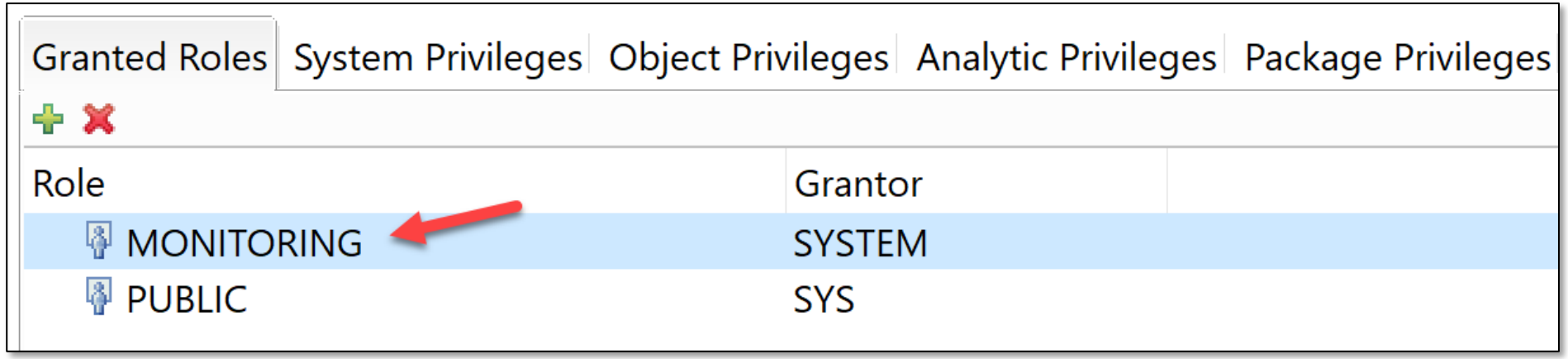
Monitoring

HANAChecker – needs privileges

The DB user that hanachecker uses to connect with needs proper privileges to run the mini-checks

The DB user that hanachecker uses only needs to read in the statistics server tables





HANAChecker – Input 1969700 ZIP File

HANAChecker can take the SQLStatements.zip file as input

- The .zip file has to be downloaded from SAP Note 1969700
- HANAChecker will use the corresponding check files based on your HANA revision

Flag	Details	Explanation	Default
-zf	Full path to SQLStatements.zip	Full path of the SQLStatements.zip file (SAP Note 1969700). Cannot be used together with –mf and must be used together with -ct	‘’
-ct	Check Types	M = Mini-Checks, I = Internal Mini-Checks, S = Security Mini-Checks, T = Trace Mini-Checks, P = Parameter Checks (see some slides below), C = Call Stacks Mini-Checks (see some slides below), R = SQL Recommendations (see some slides below)	‘’

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct M,I,S,T -M1142
lena@comp.com -S0120 chris@du.my -T0101 per@du.my -en chris@comp.com,smtp.intra.comp.com

"Mini Check ID S0120  Description: SYSTEM user deactivated  Value: no  Expectation: yes  C: X"
 is sent to chris@du.my
"Mini Check ID T0101  Area: Statistics server  Description: Unique constraint violation  Host:
 mo-fc8d991e0  Port: 30003  Count: 24  LastOccurrence: 2018/03/07 08:33:41  C: X  SAPNote: 214
7247  TraceText: plan plan6992814@mo-fc8d991e0:30003 failed with rc 301; unique constraint vio
latedTrexUpdate failed on table '_SYS_STATISTICS:HOST_LOAD_HISTORY_HOST_BASE' with error: uniq
ue constraint violation in self check for table  _SYS_STATISTICS:HOST_LOAD_HISTORY_HOST_BASEen,
 constraint='$trexexternalkey$', udiv='2018-03-07 08:33:38;12,mo-fc8d991e0;2018-02-23 07:03:27
.215', pos=708, indexname=_SYS_TREE_CS_#150584_#0_#P0, rc=55" is sent to per@du.my
"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: mo-fc8d991e0  Val
ue: 2  Expectation: 0  C: X  SAPNote: 2124112" is sent to lena@comp.com
```

HANAChecker – Input Mini-Check File

HANAChecker takes a mini-check SQL statement file as input (not needed if you use –zf and -ct)

- The mini-check files have to be downloaded from SAP Note 1969700
- Note: Always use the mini-check file that corresponds to your SAP HANA Revision and download these files regularly to always have the latest versions as SAP Note 1969700 is being updated frequently

Flag	Details	Explanation	Default
-mf	mini-check file	Full path of the mini-check file (Cannot be used together with -zf)	''

Possible Files:	CHID	Example
HANA_Configuration_MiniChecks_<revision>.txt	MXXXX	M0231
HANA_Configuration_MiniChecks_Internal_<revision>.txt	IXXXX	I0076
HANA_Security_MiniChecks.txt	SXXXX	S1045
HANA_TraceFiles_MiniChecks.txt	TXXXX	T1200
HANA_Configuration_Parameters_<revision>.txt (see some slides below)		
HANA_Threads_Callstacks_MiniChecks_<revision>.txt (see some slides below)	CXXXX	C1013
HANA_SQL_SQLCache_TopLists_<revision>.txt (see some slides below)		

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py
-mf HANA_Configuration_MiniChecks_1.00.120+.txt -en c
hris@comp.com,smtp.comp.com -M1115 chris@du.my
```

HANAChecker – Email Notification

HANAChecker can send out emails for all critical mini-checks

The HANAChecker executes the mini-checks and sends out emails for every mini-check that are “Potential Critical”, i.e. the column “C” has an “X” in it

For this to work (see this page for more information):

- an internal smtp server has to be used, e.g. ourcompany.intra.com,
- the Linux program “sendmail” has to be installed and
- a line like DSsmtp.intra.ourcompany.com in file sendmail.cf in /etc/mail/ has to be added

The sender’s email address and the internal smtp email server has to be provided:

Flag	Unit	Details	Explanation	Default
-en		email-notification	<sender’s email>,<mail server>	‘’

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configur
ation_MiniChecks_1.00.120+.txt -en chris@comp.com,smtp.intra.comp.com
-M1115 chris@du.my
```

HANAChecker – Email Mapping

HANAChecker sends out emails to the addresses mapped for the mini-checks

If the flag -<CHID> is specified for a email address and the mini-check with that check id (CHID) is potential critical then an email is send to that email address

Flag	Unit	Details	Explanation	Default
-<CHID>		mini-check to email address	If that particular mini-check specified by the flag is potential critical an email is sent to the addresses specified by the value of the flag	

Example (mini-checks):

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChecks_1.00.120+.txt
 -en chris@comp.com,smtp.intra.comp.com -M1142 chris@du.my -M1150 per@du.my,lena@du.my

"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: mo-fc8d991e0  Value: 2
 Expectation: 0  C: X  SAPNote: 2124112" is sent to chris@du.my
"Mini Check ID M1150  Description: Pinned statements in SQL cache (%)  Host: mo-fc8d991e0  Value: 27
.83  Expectation: <= 20.00  C: X  SAPNote: 2124112" is sent to per@du.my
"Mini Check ID M1150  Description: Pinned statements in SQL cache (%)  Host: mo-fc8d991e0  Value: 27
.83  Expectation: <= 20.00  C: X  SAPNote: 2124112" is sent to lena@du.my
```

Example (security mini-checks):

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Security_MiniChecks.txt -en chris@comp.
com,smtp.intra.comp.com -S0120 lena@du.my

"Mini Check ID S0120  Description: SYSTEM user deactivated  Value: no  Expectation: yes  C: X" is sen
t to lena@du.my
```

HANAChecker – Only One Email

HANAChecker can be told to only send one email per email address

The flag -oe flag tells HANAChecker that if one email address is supposed to get notified by many mini-check warnings, they are sent in only one email

Flag	Unit	Details	Explanation	Default
-oe		one email per address	true: only one email is sent per email address, false: one email is sent per critical mini check	false

Example: Here lena@du.my gets only one email, including warnings from mini-check M1142, and M1150

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChecks_1.00.
120+.txt -en chris@comp.com,smtp.intra.comp.com -M1142 lena@du.my -M1150 lena@du.my -oe true

"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: mo-fc8d991e0  V
alue: 2  Expectation: 0  C: X  SAPNote: 2124112
Mini Check ID M1150  Description: Pinned statements in SQL cache (%)  Host: mo-fc8d991e0  Va
lue: 27.28  Expectation: <= 20.00  C: X  SAPNote: 2124112" is sent to lena@du.my
```

HANAChecker – Always Send a Notification Email

HANAChecker can send an email to all unique email addresses even if none of the mini-checks for those addresses seemed critical – just to notify that HANAChecker was executed

Flag	Unit	Details	Explanation	Default
-as		always send at least a notification email	true: all email addresses will be send at least a notification email, even if none of the mini-checks assigned to the emails were potential critical	false

Example: Here all emails, chris@du.my, lena@du.my, and per@du.my, gets the notification email. Note that Per also get the notification email so he will know that the HANAChecker ran even though none of his mini-checks were critical.

```
oqladm@ls80010:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct M -en chris@comp.com,smtp.intra.comp.com
-M1142 chris@du.my,lena@du.my -M1150 per@du.my,lena@du.my -as true
"HANACecker was executed 2018-06-08 21:42:22 on OQL. If any of the mini-checks that you are responsible for seem critical,
 you will be notified now." is sent to lena@du.my
"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: ls80010  Value: 1  Expectation: 0  C: X  SAPN
ote: 2124112" is sent to lena@du.my
"HANACecker was executed 2018-06-08 21:42:22 on OQL. If any of the mini-checks that you are responsible for seem critical,
 you will be notified now." is sent to per@du.my
"HANACecker was executed 2018-06-08 21:42:22 on OQL. If any of the mini-checks that you are responsible for seem critical,
 you will be notified now." is sent to chris@du.my
"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: ls80010  Value: 1  Expectation: 0  C: X  SAPN
ote: 2124112" is sent to chris@du.my
```

HANAChecker – Always Send One Email

If both the -as and the -oe flags are set to true then all email addresses get one email and one email only; the email includes always the notification that HANAChecker ran and then, if any of the mini-checks were critical, those mini-check notifications will follow

Example: Here all email addresses, chris@du.my, lena@du.my, and per@du.my, get one email

```
oqladm@ls80010:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct M -en chris@comp.com,smtp.
intra.comp.com -M1142 chris@du.my,lena@du.my -M1150 per@du.my,lena@du.my -as true -oe true
"HANACecker was executed 2018-06-08 21:44:40 on OQL. If any of the mini-checks that you are responsible for
 seem critical, you will be notified now.
Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: ls80010  Value: 1  Expectation:
 0  C: X  SAPNote: 2124112" is sent to lena@du.my
"HANACecker was executed 2018-06-08 21:44:40 on OQL. If any of the mini-checks that you are responsible for
 seem critical, you will be notified now." is sent to per@du.my
"HANACecker was executed 2018-06-08 21:44:40 on OQL. If any of the mini-checks that you are responsible for
 seem critical, you will be notified now.
Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: ls80010  Value: 1  Expectation:
 0  C: X  SAPNote: 2124112" is sent to chris@du.my
```

HANAChecker – Email Grouping

HANAChecker sends out emails to the addresses grouped for the checks

The flag -cg can specify ranges of mini-checks and map an email address for each of these ranges

Flag	Unit	Details	Explanation	Default
-cg		mini-check groups	Groupings of mini-checks with responsible email addresses associated	

Example: Here lena@du.my gets emails for every potential critical mini-check between M1140 and M1149 and chris@du.my gets emails for every potential critical mini-check between M1150 and M1159

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChecks_1.00.120
+.txt -en chris@comp.com,smtp.intra.comp.com -cg M1140-M1149,lena@du.my,M1150-M1159,chris@du.my

"Mini Check ID M1150  Description: Pinned statements in SQL cache (%)  Host: mo-fc8d991e0  Valu
e: 27.28  Expectation: <= 20.00  C: X  SAPNote: 2124112" is sent to chris@du.my
"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: mo-fc8d991e0  Valu
e: 2  Expectation: 0  C: X  SAPNote: 2124112" is sent to lena@du.my
```

HANAChecker – Parameter Email (1/3)

HANAChecker sends out emails to the addresses listed after the flag -pe if there are parameters mentioned by HANA_Configuration_Parameters (SAP Note 1969700)

Flag	Unit	Details	Explanation	Default
-pe		parameter emails	a comma seperated list of emails that catches all parameter checks, this only makes sense if HANA_Configuration_Parameters is included in the input, either with -mf, or -ct P	not used
-is	true/false	ignore check_why_set	parameters that have been set without that there is any recommended value, will be ignored	false

Example: Here per@al.com gets emails for every potential critical parameter issue and if M0208 is potential critical:

```
pqladm@atgls90010:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct M,P
-pe per@al.com -M0208 per@al.com -oe true -k TCHECK
"
Mini Check ID M0208  Description: Supported operating system  Host: atgls90010  Value: n
o (SUSE Linux Enterprise Server 12 SP3)  Expectation: yes  C: X  SAPNote: 2235581

Parameter 'global_auditing_state' in configuration file 'global.ini' and in section 'aud
iting configuration' has an internal default value, is not configured, but the recommend
ation is 'true'. For more information see SAP Note 863362. This has priority 2.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS

Parameter 'default_statement_concurrency_limit' in configuration file 'global.ini' and i
n section 'execution' has default value '0', is not configured, but the recommendation i
s '16 to 40'. For more information see SAP Note 2222250. This has priority 1.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS  CpuThrea
ds: 80
```

HANAChecker – Parameter Email (2/3)

Example: Here chris@dum.com gets emails for every potential critical SAP HANA Parameter issue. We also see that some parameter checked specific scenario, that can be changed in the Modification Section in SQL: HANA_Configuration_Parameters (see also next slide).

```
pqladm@atgls90010:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct P -oe true -pe chris@dum.com -k TCHECK
"
Parameter 'global_auditing_state' in configuration file 'global.ini' and in section 'auditing configuration' has an intern
al default value, is not configured, but the recommendation is 'true'. For more information see SAP Note 863362. This has
priority 2.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS

Parameter 'default_statement_concurrency_limit' in configuration file 'global.ini' and in section 'execution' has default
value '0', is not configured, but the recommendation is '16 to 40'. For more information see SAP Note 2222250. This has pr
iority 1.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS  CpuThreads: 80

Parameter 'log_backup_timeout_s' in configuration file 'global.ini' and in section 'persistence' has default value '7200',
 is not configured, but the recommendation is '300 to 3600'. For more information see SAP Note 1645183. This has priority
3.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS

Parameter 'logshipping_max_retention_size' in configuration file 'global.ini' and in section 'system_replication' has defa
ult value '1048576', is not configured, but the recommendation is '0 to 656384'. For more information see SAP Note 2526877
. This has priority 3.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS  LogVolumeSize: 801
```

HANAChecker – Parameter Email (3/3)

Example: Here per@al.com gets emails for every potential critical SAP HANA Parameter issue. Here SQL: HANA_Configuration_Parameters was changed in the Modification Section (number CPU_THREADS was changed to 50), and the changed version was given as input to the –mf flag:

```
pqladm@atgls90010:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_Parameters_cpu50.txt -pe per@al.com
-oe true -k TCHECK
"
Parameter 'global_auditing_state' in configuration file 'global.ini' and in section 'auditing configuration' has an i
nternal default value, is not configured, but the recommendation is 'true'. For more information see SAP Note 863362.
 This has priority 2.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS

Parameter 'default_statement_concurrency_limit' in configuration file 'global.ini' and in section 'execution' has def
ault value '0', is not configured, but the recommendation is '15 to 25'. For more information see SAP Note 2222250. T
his has priority 1.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS  CpuThreads: 50 (manual)

Parameter 'max_concurrency' in configuration file 'global.ini' and in section 'execution' has default value '0', is n
ot configured, but the recommendation is '50'. For more information see SAP Note 2222250. This has priority 1.
Following scenario has been taken into account:
Revision: 2.00.036  Environment: ABAP,ERP,ESS,MDCTEN,SDA,SINGLE,SYSREP,AUD,XFS  CpuThreads: 50 (manual)

Parameter 'log_backup_timeout_s' in configuration file 'global.ini' and in section 'persistence' has default value '7
```

HANAChecker – Call Stacks Mini Checks

HANAChecker also covers the call stacks mini checks

The default value in the modification section of active threads limit, 0.2, is normally fine, but it is possible to change it with the -at flag

Flag	Details	Explanation	Default
-at	active threads	this sets MIN_ACTIVE_THREADS in modification section	(not used, i.e. default value in modification section 0.2 is used)

Example: Here chris@me.com gets emails for all potential critical call stacks mini-checks that have more than 0.01 average active threads

```
hsiadm@atgvmls7071:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct C -ca chris@me.com -en chr
is@comp.com,smtp.intra.comp.com -k T1SYSKEY -at 0.01
"
Mini Check ID C1013  Area: Locks  Description: Futex lock wait  Host: atgvmls7071  Port: 30003  Count: 21  Acti
ve Threads: 0.0104  LastOccurrence: 2021/03/16 08:13:27  C: X  SAPNote: 1999998  TraceText: ptime::Futex::lock"
 is sent to chris@me.com
"
Mini Check ID C1020  Area: Locks  Description: Unlocking of mutex  Host: atgvmls7071  Port: 30003  Count: 31  A
ctive Threads: 0.0153  LastOccurrence: 2021/03/16 00:23:27  C: X  SAPNote: 1999998  TraceText: Synchronization:
:Mutex::unlock" is sent to chris@me.com
```

Example: Here chris@me.com gets an email if C1013 is potential critical and has more than 0.01 average active threads

```
hsiadm@atgvmls7071:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct C -C1013 chris@me.com -en
chris@comp.com,smtp.intra.comp.com -k T1SYSKEY -at 0.01
"
Mini Check ID C1013  Area: Locks  Description: Futex lock wait  Host: atgvmls7071  Port: 30003  Count: 21  Acti
ve Threads: 0.0104  LastOccurrence: 2021/03/16 08:13:27  C: X  SAPNote: 1999998  TraceText: ptime::Futex::lock"
 is sent to chris@me.com
```

HANAChecker – SQL Recommendations

HANAChecker sends out emails to the addresses listed after the flag -se if there are recommendations in SAP Note 2000002 for the SQL statements given in the output of HANA_SQL_SQLCache_TopLists

Flag	Details	Explanation	Default
-se	sql emails	a comma seperated list of emails that catches all sql statements with recommendation in SAP Note 2000002, this only makes sense if HANA_SQL_SQLCache_TopLists is included in the input, either with -mf, or -ct R	not used

Example: Here chris@me.com gets emails for every SQL statement with recommendations:

```
hsiadm@atgvmls7071:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct R -se chris@me.com -en ch
ris@comp.com,smtp.intra.comp.com -k T1SYSKEY
"
SQL statement d6fd6678833f9a2e25e7b53239c50e9a is one of the most expensive statements in the SQL cache and th
ere is a recommendation available in SAP Note 2000002.
This SQL statement is of type CA, originates from Statistics server, and executed by the Row engine." is sent
to chris@me.com
"
SQL statement 430c496e0fe15c0353c80de1c72caab1 is one of the most expensive statements in the SQL cache and th
ere is a recommendation available in SAP Note 2000002.
This SQL statement is of type SE, originates from Statistics server, and executed by the Row engine." is sent
to chris@me.com
"
SQL statement d3759ce6047b78f61d5fc3be392d0336 is one of the most expensive statements in the SQL cache and th
```

Monitoring

HANAChecker – Different Emails Example

Example: Here chris@me.com gets emails for every SQL statement with recommendation, lina@me.com gets emails for some call stacks mini-checks, peter@me.com gets emails for some security mini-chekcs and ole@me.com gets emails for all configuration parameters that do not follow the recommendations. …

```
hsiadm@atgvmls7071:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct R,C,S,P -se chris@me.com
-cg C1000-C2000,lina@me.com,S0100-S0300,peter@me.com -pe ole@me.com -at 0.01 -en chris@comp.com,smtp.intra.com
p.com -k T1SYSKEY
"
Mini Check ID S0120  Description: SYSTEM user deactivated  Value: no  Expectation: yes  C: X" is sent to peter
@me.com
```

…

```
Mini Check ID C1020  Area: Locks  Description: Unlocking of mutex  Host: atgvmls7071  Port: 30003  Count: 30
Active Threads: 0.0148  LastOccurrence: 2021/03/17 05:08:27  C: X  SAPNote: 1999998  TraceText: Synchronizatio
n::Mutex::unlock" is sent to lina@me.com
```

…

```
SQL statement d6fd6678833f9a2e25e7b53239c50e9a is one of the most expensive statements in the SQL cache and th
ere is a recommendation available in SAP Note 2000002.
This SQL statement is of type CA, originates from Statistics server, and executed by the Row engine." is sent
to chris@me.com
```

…

```
Parameter 'service thread sampling monitor enabled' in configuration file 'global.ini' and in section 'resourc
e_tracking' has default value 'false', is configured to 'true' in ALTER SYSTEM ALTER CONFIGURATION ('global.in
i', 'SYSTEM') SET ('resource_tracking', 'service_thread_sampling_monitor_enabled') = 'true' WITH RECONFIGURE;
layer, but the recommendation is '2114710'. For more information see SAP Note DEFAULT. This has priority 2.
Following scenario has been taken into account:
Revision: 2.00.053  Environment: ABAP,ESS,LCACHE,MDCTEN,MULTI,PROD,SCM" is sent to ole@me.com
```

HANAChecker – Catch All

HANAChecker can send out emails for all potential critical checks to the emails defined as “catch all” emails

Flag	Unit	Details	Explanation	Default
-ca		catch all emails	The email addresses specified receive an email about each potential critical mini-check and parameter check	

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChec
ks_1.00.120+.txt -en chris@comp.com,smtp.intra.comp.com -ca peter@du.my,chris@du.my

"Mini Check ID M0012  Description: Revision level  Value: 121.00  Expectation: >= 12
2.03  C: X  SAPNote: 2021789" is sent to chris@du.my
"Mini Check ID M0115  Description: Service startup time variation (s)  Host: mo-fc8d
991e0  Value: 3299447  Expectation: <= 600  C: X  SAPNote: 2177064" is sent to chris
@du.my
```

…

```
"Mini Check ID M0012  Description: Revision level  Value: 121.00  Expectation: >= 12
2.03  C: X  SAPNote: 2021789" is sent to peter@du.my
"Mini Check ID M0115  Description: Service startup time variation (s)  Host: mo-fc8d
991e0  Value: 3299447  Expectation: <= 600  C: X  SAPNote: 2177064" is sent to peter
@du.my
```

…

HANAChecker – Ignore Checks

While using the “Catch All” flag, you can “Ignore” some checks with the –ic flag, i.e. catch all <u>except</u> some checks

Example:

Flag	Unit	Details	Explanation	Default
-ic		ignore checks	A list of mini-check CHIDs to be ignored by the catch all emails	

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChecks
_1.00.120+.txt -en chris@comp.com,smtp.intra.comp.com -ca chris@du.my -ic M0012,M0209

"Mini Check ID M0115  Description: Service startup time variation (s)  Host: mo-fc8d99
1e0  Value: 3299447  Expectation: <= 600  C: X  SAPNote: 2177064" is sent to chris@du.
my
"Mini Check ID M0551  Description: Proper setup of timezone table TTZZ  Value: no  Exp
ectation: yes  C: X  SAPNote: 1791342" is sent to chris@du.my
```

…

Monitoring

HANAChecker – Different Check Types

One can define different mini-check files with the -mf flag and assign different mini-check types to emails, e.g.

```
-mf <mini-check file>,<security-mini-check file>,…
-cg M0100-M0200,email1@comp.com,S0100-S0200,email2@comp.com
```

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChecks_1.00.120+.txt
,HANA_Configuration_MiniChecks_Internal_1.00.120-1.00.122.99.txt,HANA_Security_MiniChecks.txt,HANA_T
raceFiles_MiniChecks.txt -cg M1100-M1150,chris@du.my,I0010-I1000,john@du.my,S0100-S0125,lena@du.my,T
0099-T0105,per@du.my -en chris@me.com,smtp.intra.me.com
"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: mo-fc8d991e0  Value: 2
 Expectation: 0  C: X  SAPNote: 2124112" is sent to chris@du.my
"Mini Check ID M1150  Description: Pinned statements in SQL cache (%)  Host: mo-fc8d991e0  Value: 27
.93  Expectation: <= 20.00  C: X  SAPNote: 2124112" is sent to chris@du.my
"Mini Check ID T0101  Area: Statistics server  Description: Unique constraint violation  Host: mo-fc
8d991e0  Port: 30003  Count: 24  LastOccurrence: 2018/03/05 06:33:41  C: X  SAPNote: 2147247  TraceT
ext: plan plan5636987@mo-fc8d991e0:30003 failed with rc 301; unique constraint violatedTrexUpdate fa
iled on table '_SYS_STATISTICS:HOST_LOAD_HISTORY_HOST_BASE' with error: unique constraint violation
in self check for table _SYS_STATISTICS:HOST_LOAD_HISTORY_HOST_BASEen, constraint='$trexexternalkey$
', udiv='2018-03-05 06:33:38;12,mo-fc8d991e0;2018-02-22 20:18:42.21', pos=18052, indexname=_SYS_TREE
_CS_#150584_#0_#P0, rc=55" is sent to per@du.my
"Mini Check ID S0120  Description: SYSTEM user deactivated  Value: no  Expectation: yes  C: X" is se
nt to lena@du.my
```

HANAChecker – Interval

Run hanachecker “forever” with the –hci flag

Flag	Unit	Details	Explanation	Default
-hci	Days	hanachecker interval	After these number days hanachecker will restart	-1 (exits)

Example: HANAChecker runs here the mini-checks and sends emails once every day

```
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChec
ks_1.00.120+.txt -en chris@comp.com,smtp.intra.comp.com -M1170 peter@comp.com -hci 1

"Mini Check ID M1170  Description: Average database request time (ms)  Value: 6.91
Expectation: <= 2.00  C: X  SAPNote: 2000002" is sent to peter@comp.com
"Mini Check ID M1170  Description: Average database request time (ms)  Value: 6.91
Expectation: <= 2.00  C: X  SAPNote: 2000002" is sent to peter@comp.com
"Mini Check ID M1170  Description: Average database request time (ms)  Value: 6.91
Expectation: <= 2.00  C: X  SAPNote: 2000002" is sent to peter@comp.com
```

Note: HANAChecker could ofcourse also be scheduled by a cron job – do then NOT use the -hci flag!

HANAChecker – output

To control the output of the hanachecker there are these flags

Flag	Unit	Details	Explanation	Default
-od		output directory	full path of the folder where the hanachecker logs are written	/tmp/hanachecker_output
-so		standard out switch	1: write to std out, 0: do not write to std out	1

Example:
Here an output folder is deleted and then automatically created again by hanachecker and a new log file is written into it:

```
mo-fc8d991e0:/tmp/HANAChecker> rm -r /tmp/hanachecker_output/
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_MiniChec
ks_1.00.120+.txt -en chris@comp.com,smtp.intra.comp.com -M1170 peter@comp.com -so 0

mo-fc8d991e0:/tmp/HANAChecker>
mo-fc8d991e0:/tmp/HANAChecker> more ../hanachecker_output/hanacheckerlog_2018-03-05_
12-37-00.txt
"Mini Check ID M1170  Description: Average database request time (ms)  Value: 6.91
Expectation: <= 2.00  C: X  SAPNote: 2000002" is sent to peter@comp.com
```

HANAChecker – Used To Only Log Checks

If the -en flag is not specified, HANAChecker can be used to simply write all potential critical mini-checks in the log file, without sending out emails:

```
python hanachecker.py -mf HANA_Configuration_MiniChecks_1.00.120+.txt
-ca dummy@me.com -so 0
```

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> rm -r /tmp/hanachecker_output/
mo-fc8d991e0:/tmp/HANAChecker>
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -mf HANA_Configuration_
MiniChecks_1.00.120+.txt -ca lena@comp.com -so 0

mo-fc8d991e0:/tmp/HANAChecker>
mo-fc8d991e0:/tmp/HANAChecker> more ../hanachecker_output/hanacheckerlog_201
8-03-05_12-40-31.txt
"Mini Check ID M0012  Description: Revision level  Value: 121.00  Expectatio
n: >= 122.03  C: X  SAPNote: 2021789"
"Mini Check ID M0115  Description: Service startup time variation (s)  Host:
 mo-fc8d991e0  Value: 3299447  Expectation: <= 600  C: X  SAPNote: 2177064"
"Mini Check ID M0209  Description: Recommended operating system kernel versi
on  Host: mo-fc8d991e0  Value: no (3.0.101-68-default instead of >= 0.108.7)
  Expectation: yes  C: X  SAPNote: 2235581"
"Mini Check ID M0551  Description: Proper setup of timezone table TTZZ  Valu
e: no  Expectation: yes  C: X  SAPNote: 1791342"
```

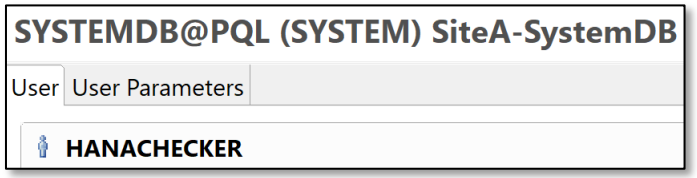
...

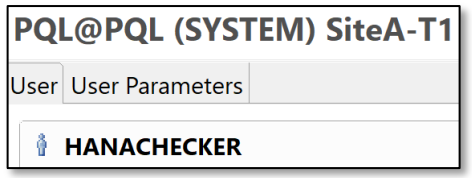
HANAChecker – MDC (1/2)

In a MDC system the hanachecker can check the SystemDB and multiple Tenants with one key

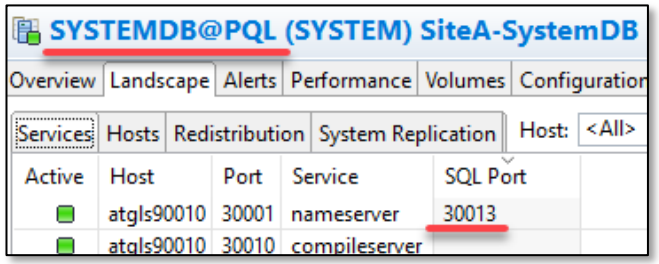
Maintain a user with same user name and same password in multiple DBs in one HANA System

Example:
Here the user HANACLEANER1 with same password was created in both SystemDB and in a Tenant





(for privileges,
see earlier slides)



Then only one key,
for the SystemDB,
was provided in
hdbuserstore

```
pqladm@atgls90010:/tmp> hdbuserstore LIST SDBCHECKERKEY
KEY SDBCHECKERKEY
  ENV : atgls90010:30013
  USER: HANACHECKER
```

Test that this single
key can be used to
access both databases:

```
pqladm@atgls90010:/tmp> hdbsql -j -A -x -U SDBCHECKERKEY -d SYSTEMDB "select * from m_database"
| SYS | DATABASE | HOST       | START_TIME                 | VERSION                  | USAG |
| --- | -------- | ---------- | -------------------------- | ------------------------ | ---- |
| PQL | SYSTEMDB | atgls90010 | 2018-11-29 13:01:39.336000000 | 2.00.034.00.1539746999 | TEST |
pqladm@atgls90010:/tmp>
pqladm@atgls90010:/tmp> hdbsql -j -A -x -U SDBCHECKERKEY -d PQL "select * from m_database"
| SYS | DAT | HOST       | START_TIME                 | VERSION                  | USAG |
| --- | --- | ---------- | -------------------------- | ------------------------ | ---- |
| PQL | PQL | atgls90010 | 2018-11-29 13:01:50.309000000 | 2.00.034.00.1539746999 | TEST |
```

HANAChecker – MDC (2/2)

In a MDC system the hanachecker can check the SystemDB and multiple Tenants with one key

Flag	Unit	Details	Explanation	Default
-dbs		DB key(s)	this can be a list of databases accessed from the system defined by -k (-k can only be one key if -dbs is used)	''

Example:

Here the key SDBCHECKERKEY is used to access the system, then it is specified with -dbs that two databases, SYSTEMDB and PQL, will be checked:

```
pqladm@atgls90010:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct M -M1142 chris@du.my,lena@du.my -M1150 per
@du.my,lena@du.my -as true -oe true -k SDBCHECKERKEY -dbs SYSTEMDB,PQL
"HANACecker was executed 2019-01-23 09:13:50 on SYSTEMDB@PQL with
hanachecker.py -zf SQLStatements.zip -ct M -M1142 chris@du.my,lena@du.my -M1150 per@du.my,lena@du.my -as true -oe true -k SDBC
HECKERKEY -dbs SYSTEMDB,PQL
If any of the mini-checks that you are responsible for seem critical, you will be notified now.
Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: atgls90010  Value: 1  Expectation: 0  C: X  SAPNot
e: 2124112
```

```
"HANACecker was executed 2019-01-23 09:41:15 on PQL@PQL with
hanachecker.py -zf SQLStatements.zip -ct M -M1142 chris@du.my,lena@du.my -M1150 per@du.my,lena@du.my -as true -oe true -k SDBC
HECKERKEY -dbs SYSTEMDB,PQL
If any of the mini-checks that you are responsible for seem critical, you will be notified now.
Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host: atgls90010  Value: 1  Expectation: 0  C: X  SAPNot
e: 2124112
```

HANAChecker – Configuration File

HANAChecker can be controlled with a configuration file
(additional flags given will overwrite flags in the configuration file)

Flag	Unit	Details	Explanation	Default
-ff		flag file	full path to the configuration file	

Example:

```
mo-fc8d991e0:/tmp/HANAChecker> more hanachecker_configurationfile.txt
Comp's HANA CHECKER Configuration File:

-mf HANA_Configuration_MiniChecks_1.00.120+.txt,HANA_Security_MiniChecks.txt
-en chris@comp.com,smtp.intra.comp.com
-cg M1140-M1145,peter@comp.com,M1146-M1150,sara@comp.com
-M1142 lena@comp.com,per@comp.com
-S0120 chris@ourcompany.com
-so 0

mo-fc8d991e0:/tmp/HANAChecker> rm -r ../hanachecker_output/
mo-fc8d991e0:/tmp/HANAChecker> python hanachecker.py -ff hanachecker_configur
ationfile.txt
mo-fc8d991e0:/tmp/HANAChecker> more ../hanachecker_output/hanacheckerlog_2018
-03-05_12-54-16.txt
"Mini Check ID M1142  Description: Table(s) using > 10 % of SQL cache  Host:
mo-fc8d991e0  Value: 2  Expectation: 0  C: X  SAPNote: 2124112" is sent to pe
ter@comp.com
```

…

HANAChecker – Real Life Example

Emails were retrieved from HANAChecker for potential critical mini-checks: