

# 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

A screenshot of a GitHub repository page for 'chriselswede/hanachecker'. The page shows a list of files: README.md, hanachecker.pdf, hanachecker.py (which has a red arrow pointing to it), and hanachecker\_configurationfile\_example.txt.

**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](#)**

# 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/HCI/home/.hdb/dewdfglp00835/SSFS_HDB.DAT
KEY FILE       : /usr/sap/HCI/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 Potential:
```

# 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

User Name\*:   Disable ODBC/JDBC access

Authentication

Password  Confirm\*:

Granted Roles		System Privileges	Object Privileges	Analytic Privileges	Package Privileges
Role		Grantor			
MONITORING		SYSTEM			
PUBLIC		SYS			

# 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
<b>-zf</b>	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	"
<b>-ct</b>	Check Types	<b>M</b> = Mini-Checks, <b>I</b> = Internal Mini-Checks, <b>S</b> = Security Mini-Checks, <b>T</b> = Trace Mini-Checks, <b>P</b> = Parameter Checks (see some slides below), <b>C</b> = Call Stacks Mini-Checks (see some slides below), <b>R</b> = SQL Recommendations (see some slides below), <b>A</b> = ABAP Mini-Checks (see some slides below)	"

### Example:

```
xscadm@atgvm1s866:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip
-ct M,S -M0551 christian.hansen01@sap.com -M0712 chriselswede@gmail.com -S0120
christian.hansen01@sap.com -k T1KEY
"
Mini Check ID M0712 Description: Internal statistics server errors (last day)
Value: 334 Expectation: 0 C: X SAPNote: 2147247" is sent to chriselswede@gmail.com
"
Mini Check ID M0551 Description: Proper setup of timezone table TTZZ Value: no
Expectation: yes C: X SAPNote: 1791342" is sent to christian.hansen01@sap.com
"
Mini Check ID S0120 Description: SYSTEM user deactivated Value: no Expectation: yes C: X" is sent to christian.hansen01@sap.com
```

### One of the emails:

HANAChecker: Potential Critical Situation(s) @XSC!



SAP HANA Database System Administrator <xscadm@atgvm1s866>
To ✓ Hansen, Christian

Mini Check ID M0551 Description: Proper setup of timezone table TTZZ
Value: no Expectation: yes C: X SAPNote: 1791342

# HANAChecker – Input Mini-Check File (not needed)



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

- Note: Always use the mini-check file that corresponds to your SAP HANA Revision and download these files regularly from SAP Note [1969700](#) 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)	"
<b>Possible Files:</b>			
	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)		
	HANA_ABAP_MiniChecks.txt (see some slides below)		AXXXX A0510

**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 sends 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. for each check where the column “C” is “X”

For this to work an email client, e.g. mailx (see [this page](#) for more information), mail, or mutt, must have been set up and configured (e.g. the sender's email and the internal smtp email server)

In HANAChecker this can be controlled with the following flags:

Flag	Details	Explanation	Default	
<b>-en</b>	email client	For example mail, mailx, mutt, ...	mailx	
<b>-ens</b>	sender's email	To explicitly specify sender's email address	"	(configured is used)
<b>-enm</b>	mail server	To explicitly specify the internal smtp email server	"	(configured is used)

### Test: If

```
echo -e "$(date)\\n$(hostname)\\n\\nTEST" | mailx -s "TEST mailx" <your email address>
```

sends you an email, then the above flags are not needed when running HANAChecker, e.g.:

```
xscadm@atgvmls866:/tmp> echo -e "$(date)\\n$(hostname)\\n\\nTEST" | mailx -s "TEST mailx" christian.hansen01@sap.com
```



TEST mailx  
 SAP HANA Database  
 To  Hansen, Christian  
 Wed Jul 28 16:44:00 CEST 2021  
 atgvmls866  
 TEST

# 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 sent to lena@du.my
```

# HANAChecker – Only One Email



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

The `-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
<code>-oe</code>		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 Value: 2 Expectation: 0 C: X SAPNote: 2124112
Mini Check ID M1150 Description: Pinned statements in SQL cache (%) Host: mo-fc8d991e0 Value: 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.mv
```

# 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
<code>-cg</code>		mini-check groups	Groupings of mini-checks with responsible email addresses associated	

**Example:** Here [lena@du.my](mailto:lena@du.my) gets emails for every potential critical mini-check between M1140 and M1149 and [chris@du.my](mailto: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_MinChecks_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  Value: 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  Value: 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](mailto: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 internal 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 priority 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 default 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](mailto: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: -ct C

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

```
hsiadm@atgvm1s7071:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct C -ca 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: atgvm1s7071 Port: 30003 Count: 21 Active 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: atgvm1s7071 Port: 30003 Count: 31 Active 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](mailto:chris@me.com) gets an email if C1013 is potential critical and has more than 0.01 average active threads

```
hsiadm@atgvm1s7071:/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: atgvm1s7071 Port: 30003 Count: 21 Active 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
<b>-se</b>	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](mailto:chris@me.com) gets emails for every SQL statement with recommendations:

```
hsiadm@atgvm1s7071:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct R -se chris@me.com -en chris@comp.com,smtp.intra.comp.com -k T1SYSKEY
"
SQL statement d6fd6678833f9a2e25e7b53239c50e9a is one of the most expensive statements in the SQL cache and there 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 there 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
```

# HANAChecker – Different Emails Example



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

```
hsiadm@atgvm1s7071:/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: atgvm1s7071 Port: 30003 Count: 30  
Active Threads: 0.0148 LastOccurrence: 2021/03/17 05:08:27 C: X SAPNote: 1999998 TraceText: Synchronization::Mutex::unlock" is sent to lina@me.com
```

```
...  
SQL statement d6fd6678833f9a2e25e7b53239c50e9a is one of the most expensive statements in the SQL cache and there 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 'resource_tracking' has default value 'false', is configured to 'true' in ALTER SYSTEM ALTER CONFIGURATION ('global.ini', '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 – ABAP Mini Checks



**HANAChecker also covers the ABAP mini checks: -ct A**

The ABAP schema must be specified

Flag	Details	Explanation	Default
-abs	ABAP schema	this sets the ABAP schema before HANA_ABAP_MiniChecks.txt is executed	(not set → it must be set)

**Example:** Here [chris@me.com](mailto:chris@me.com) gets emails for all potential critical ABAP mini-checks

```
oqladm@ls80010:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct A -abs SAPQH1 -ca chris@me.com
-en chris@comp.com,smtp.intra.comp.com -k SYSTEMKEY
"
Mini Check ID A0510 Description: Parameter: dbs/hdb/cmd_buffersize Value: wrong (1048576 instead of 2097152 to
10485760) Expectation: like 'okay%' C: X SAPNote: 2600030" is sent to chris@me.com
"
Mini Check ID A0520 Description: Parameter: rsdb/tbi_buffer_area_MB Value: wrong (192 instead of >= 1000) Exp
ection: like 'okay%' C: X SAPNote: 2600030" is sent to chris@me.com
"
Mini Check ID A0521 Description: Parameter: rsdb/tbi_dir_entries Value: wrong (13200 instead of >= 20000) Exp
ection: like 'okay%' C: X SAPNote: 2600030" is sent to chris@me.com
"
Mini Check ID A0900 Description: SQLM activated Value: no Expectation: yes C: X SAPNote: 1885926" is sent t
o chris@me.com
"
Mini Check ID A1702 Description: System change option set to modifiable Value: yes Expectation: no C: X" is
sent to chris@me.com
```

# HANAChecker – Another Different Emails Example



**Example:** Here [chris@me.com](mailto:chris@me.com) gets an email if A0510 is potential critical, [john@me.com](mailto:john@me.com) gets email an email if T9000 is potential critical, [sue@me.com](mailto:sue@me.com) gets an email if M1220 is potential critical, and [mila@me.com](mailto:mila@me.com) gets an email if S0120 is potential critical.

```
oqladm@ls80010:/tmp/HANAChecker> python hanachecker.py -zf SQLstatements.zip -ct M,S,T,A -abs SAPQH1 -A0510 chris@me.com
-T9000 john@me.com -M1220 sue@me.com -S0120 mila@me.com -en chris@comp.com,smtp.intra.comp.com -k SYSTEMKEY
"
Mini Check ID T9000 Area: General Description: Max. trace lines per hour Host: ls80010 Port: 30003 Count: 10715 Last occurrence: 2021/03/20 02:00:00 C: X SAPNote: 2380176" is sent to john@me.com
"
Mini Check ID A0510 Description: Parameter: dbs/hdb/cmd_buffersize Value: wrong (1048576 instead of 2097152 to 10485760
) Expectation: like 'okay%' C: X SAPNote: 2600030" is sent to chris@me.com
"
Mini Check ID S0120 Description: SYSTEM user deactivated Value: no Expectation: yes C: X" is sent to mila@me.com
"
Mini Check ID M1220 Description: QCM conversion tables Value: 1 Expectation: 0 C: X SAPNote: 9385" is sent to sue@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_MiniChecks_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: >= 122.03 C: X SAPNote: 2021789" is sent to chris@du.my
"Mini Check ID M0115 Description: Service startup time variation (s) Host: mo-fc8d991e0 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: >= 122.03 C: X SAPNote: 2021789" is sent to peter@du.my
"Mini Check ID M0115 Description: Service startup time variation (s) Host: mo-fc8d991e0 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 except some checks

Example:

Flag	Unit	Details	Explanation	Default
<b>-ic</b>		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_MinChecks
_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
```

# 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
<b>-hci</b>	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_MiniChecks_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!



## 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
-or	days	retention	logs in the path specified with -od are only saved for this number of days	-1 (not used)
-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_Minichcks_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 you put a dummy email for the catch all flag (-ca), set the one email flag (-oe) to true and specify that you don't want any output to std out (-so) then HANAChecker can be used to simply log the checks without receiving any email:

Example:

```
xscadm@atgvm1s866:/tmp/HANAChecker> rm /tmp/hanachecker_output/hanacheckerlog_2022-11-23_20-*  
xscadm@atgvm1s866:/tmp/HANAChecker> python hanachecker.py -zf SQLStatements.zip -ct M,S -ca dummy@dum.com -oe true -so 0 -k T1KEY  
HANAChecker started on python 3.7.12  
xscadm@atgvm1s866:/tmp/HANAChecker> more /tmp/hanachecker_output/hanacheckerlog_2022-11-23_20-54-53.txt  
"  
Mini Check ID M0314 Description: I/O read reload throughput avg. (MB/s) Host: atgvm1s866 Value: 49.82 Expectation: >= 200.00 C: X  
Mini Check ID M0371 Description: Unused space in packed LOBs vs. data (%) Value: 11.50 Expectation: <= 7.00 C: X SAPNote: 2220627  
Mini Check ID M0420 Description: Large heap allocators (current) Value: Pool/RowEngine/QueryExecution/SearchAlloc (2 GB) Expectation:  
Mini Check ID M0551 Description: Proper setup of timezone table TTZZ Value: no Expectation: yes C: X SAPNote: 1791342  
Mini Check ID M0552 Description: Number of current timezone alerts Value: 2 Expectation: 0 C: X SAPNote: 1791342  
Mini Check ID M0712 Description: Internal statistics server errors (short-term) Value: 288 Expectation: 0 C: X SAPNote: 2147247  
Mini Check ID M0730 Description: Pending e-mails older than 3 days Value: 40055 Expectation: <= 100 C: X SAPNote: 2133799  
Mini Check ID M0752 Description: Historic thread call stacks interval (s) Value: 300 Expectation: 299 C: X SAPNote: 2147247
```

# 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

**SYSTEMDB@PQL (SYSTEM) SiteA-SystemDB**

User	User Parameters
<b>HANACHECKER</b>	

**PQL@PQL (SYSTEM) SiteA-T1**

User	User Parameters
<b>HANACHECKER</b>	

(for privileges,  
see earlier slides)

**SYSTEMDB@PQL (SYSTEM) SiteA-SystemDB**

Overview	Landscape	Alerts	Performance	Volumes	Configuration
Services	Hosts	Redistribution	System Replication	Host: <All>	
Active	Host	Port	Service	SQL Port	
[atgls90010]	30001	nameserver	30013		
[atgls90010]	30010	compileserver			

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   |
```

**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 SDBCHECKERKEY -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 SAPNote: 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 SDBCHECKERKEY -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 SAPNote: 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_configurationfile.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 peter@comp.com
```

Note: for multiple configuration files, see some slides below

...

# HANAChecker – Configuration File Example (1/3)



hanachecker\_configurationfile\_example.txt - Notepad

File Edit Format View Help

An Example of a HANA CHECKER Configuration File:

#-----#

The hdbuserstore key

-k T1KEY

The .zip file is located in the same directory  
as hanachecker.py  
-zf SQLstatements.zip

Will run the normal mini-checks and the  
security mini-checks

-ct M,S

Recievers will only get one email per execution  
-oe true

Recievers will always get one email per execution  
(even if their checks were not critical)

-as true

Chris is responsible for HANA revisions and for  
that everything is up running at "OurCompany, OC"  
-cg M0001-M0199,chris@oc.com

Lena is responsible for the Operating System  
-cg M0200-M0299,lena@oc.com

Ole is responsible for the Disk

-cg M0300-M0399,ole@oc.com

Emma is the Memory expert

-cg M0400-M0499,emma@oc.com

Per is responsible for the Tables

-cg M0500-M0599,per@oc.com

Cindy is responsible for Traces, Dumps and Logs

-cg M0600-M0699,cindy@oc.com

Clark makes sure that the Statistics Server,  
and Transaction and Threads run smoothly

-cg M0700-M0899,clark@oc.com

Loren is the Backup expert

-cg M0900-M0999,loren@oc.com

Tom checks that all Locks and SQLs are OK

-cg M1000-M1199,tom@oc.com

Jen takes care of the Applications

-cg M1200-M1299,jen@oc.com

Tim is Security responsible

-cg M1300-M1399,tim@oc.com

Mark handles the Network

-cg M1500-M1599,mark@oc.com

John checks the XS Engine, and Nameserver

-cg M1600-M1799,john@oc.com

Michael is responsible for the System Replication

-cg M1800-M1899,michael@oc.com

Ann checks Objects, BW and Consistency

-cg M1900-M2199,ann@oc.com

Oscar handles SDA/SDI

-cg M2200-M2299,oscar@oc.com

Administration is Chris' task,  
and he also gets the rest

-cg M2300-M3000,chris@oc.com

Tim is responsible for Security, so he checks  
everything from the Security Minichecks

-cg S0001-S2000,tim@oc.com

Gustav is the boss, so he wants to  
see ALL Critical Checks

-ca gustav@oc.com

# HANAChecker – Configuration File Example (2/3)



First part of  
the output:

```
xscadm@atgvm1s866:/tmp/HANAChecker> python hanachecker.py -ff hanachecker_configurationfile_example.txt
"HANAChecker was executed 2021-07-28 18:48:28 on XSC with hanachecker.py -ff hanachecker_configurationfile_example.txt
If any of the mini and/or parameter checks that you are responsible for seem critical, you will be notified now.

Mini Check ID M0474 Description: Heap allocators with large allocation share Value: Pool/RowEngine/QueryExecution/SearchAlloc (20 %) Expectation: none C: X SAPNote: 1999997" is sent to emma@oc.com
"HANAChecker was executed 2021-07-28 18:48:28 on XSC with hanachecker.py -ff hanachecker_configurationfile_example.txt
If any of the mini and/or parameter checks that you are responsible for seem critical, you will be notified now.
" is sent to lena@oc.com
"HANAChecker was executed 2021-07-28 18:48:28 on XSC with hanachecker.py -ff hanachecker_configurationfile_example.txt
If any of the mini and/or parameter checks that you are responsible for seem critical, you will be notified now.

Mini Check ID M0312 Description: I/O read latency data max. (ms, last day) Host: atgvm1s866 Value: 29.70 Expectation: <= 20.00 C: X SAPNote: 1999930" is sent to ole@oc.com
"HANAChecker was executed 2021-07-28 18:48:28 on XSC with hanachecker.py -ff hanachecker_configurationfile_example.txt
If any of the mini and/or parameter checks that you are responsible for seem critical, you will be notified now.

Mini Check ID M0012 Description: Revision level Value: 52.00 Expectation: >= 122.03 C: X SAPNote: 2617548
Mini Check ID M2320 Description: Time since last hdbcons execution (h) Value: 0 Expectation: >= 24 C : X SAPNote: 2222218" is sent to chris@oc.com
```

# HANAChecker – Configuration File Example (3/3)



I exchanged all email addresses in the example to my own email address, then HANAChecker send me this:

HANAChecker: Potential Critical Situation(s) @XSC!



SAP HANA Database System Administrator <xscadm@atgvm1s866.wdf.s  
To Hansen, Christian

[Reply](#) [Reply All](#) [Forward](#) ·  
Mi 28.07.2021 19:05:37

HANAChecker was executed 2021-07-28 19:05:37 on XSC with  
hanachecker.py -ff hanachecker\_configurationfile\_example\_only\_me.txt  
If any of the mini and/or parameter checks that you are responsible for seem critical, you will be notified now.

Mini Check ID M0012 Description: Revision level Value: 52.00 Expectation: >= 122.03 C: X SAPNote: 2617548

Mini Check ID M0312 Description: I/O read latency data max. (ms, last day) Host: atgvm1s866 Value: 29.70 Expectation: <= 20.00 C: X SAPNote: 1999930

Mini Check ID M0474 Description: Heap allocators with large allocation share Value: Pool/RowEngine/QueryExecution/SearchAlloc (21 %) Expectation: none C: X SAPNote: 1999997

Mini Check ID M0551 Description: Proper setup of timezone table TTZZ Value: no Expectation: yes C: X SAPNote: 1791342

Mini Check ID M0552 Description: Number of current timezone alerts Value: 2 Expectation: 0 C: X SAPNote: 1791342

Mini Check ID M0641 Description: Number of trace files (last day) Host: atgvm1s866 Value: 39 Expectation: <= 30 C: X SAPNote: 2119087

Mini Check ID M0646 Description: Number of statement OOM trace files (last day) Host: atgvm1s866 Value: 1 Expectation: 0 C: X SAPNote

# HANAChecker – Configuration Files & Config Output



**HANAChecker can be controlled with a list of configuration files (files listed later in the -ff list will overwrite flags from files listed earlier in the list, flags on the command line will overwrite the configuration files)**

Flag	Unit	Details	Explanation	Default
-ff		comma separated list of flag files	full paths to configuration files	
-oc	true/false	output configuration	logs all set parameters and where the flags were set	false

## Example:

```
xscadm@atgvmls866:/tmp/HANAChecker> more hanacheckerconfig.txt
The hdbuserstore key
-k T1KEY
The .zip file
-zf SQLStatements.zip
normal and security mini-checks
-ct M,S
only get one email per execution
-oe true
always one email per execution
-as true
Chris
-cg M0001-M0199,chris@oc.com
Lena
-cg M0200-M0299,lena@oc.com
Ole
-cg M0300-M0399,ole@oc.com
Tim
-cg S0001-S2000,tim@oc.com
Gustav
-ca gustav@oc.com
```

```
xscadm@atgvmls866:/tmp/HANAChecker> more hanacheckerconfig2.txt
Chris --> Lena
-cg M0001-M0199,lena@oc.com
Lena
-cg M0200-M0299,lena@oc.com
Ole --> Lena
-cg M0300-M0399,ole@oc.com
Tim
-cg S0001-S2000,tim@oc.com
Gustav --> Chris
-ca chris@oc.com
```

The example continues on the next slide

# HANAChecker – Configuration Files & Config Output



## Example (continued):

```
xscadm@atgvm1s866:/tmp/HANAChecker> python hanachecker.py -ff hanacheckerconfig.txt,hanacheckerconfig2.txt -ca chris@oc.com -oc true
"HANAChecker was executed 2021-09-02 17:46:52 with
-as      true from hanacheckerconfig.txt
-ff      hanacheckerconfig.txt,hanacheckerconfig2.txt from command line
-cq      M0001-M0199,lena@oc.com from hanacheckerconfig2.txt      M0200-M0299,lena@oc.com from hanacheckerconfig2.txt      M0300-M0399,o
le@oc.com from hanacheckerconfig2.txt      S0001-S2000,tim@oc.com from hanacheckerconfig2.txt
-k       T1KEY from hanacheckerconfig.txt
-zf      SQLStatements.zip from hanacheckerconfig.txt
-ca      chris@oc.com from command line
-oc      true from command line
-ct      M,S from hanacheckerconfig.txt
-oe      true from hanacheckerconfig.txt on XSC with
hanachecker.py -ff hanacheckerconfig.txt,hanacheckerconfig2.txt -ca chris@oc.com -oc true
If any of the mini and/or parameter checks from following check files:./tmp_sql_statements/HANA_Configuration_MiniChecks_2.00.053+.tx
t,./tmp_sql_statements/HANA_Security_MiniChecks_2.00.030+.txt
that you are responsible for seem critical, you will be notified now.
```

# HANAChecker – HANA System Replication, HSR (1/2)



**On a Secondary in a HSR setup one can use -oi to let HANAChecker wait for a takeover**

Flag	Unit	Details	Explanation	Default
-oi	days	Online Check Interval (days)	<p>&lt; 0: HANAChecker does not check if online or secondary</p> <p>= 0: if not online or not primary HANAChecker will abort</p> <p>&gt; 0: days it waits before it checks if DB is online and primary again</p> <p>Note: For &gt; 0, you might have to use cron with a lock (see the HANASitter pdf, "HANASitter &amp; CRON" slide)</p>	-1 (not used)

## Example:

Here the HANAChecker is started on a system that is a Secondary in a HSR setup, with online check interval 1 day (-oi), and every time it does an online check it will also clean up hanachecker logs based on -or:

```
ha2adm@atgvm1s7055:/tmp/HANAChecker> python hanachecker.py -ff hanachecker_configurationfile_example_only_me.txt -oi 1 -or 10 -k T1KEY
HANAChecker started on python 3.8.12
Double check that the configuration file only contains ascii characters!
Online Check , 2022-10-27 04:43:52 , - , True , True , # index services: 2, # running services: 10 out of 10
Primary Check , 2022-10-27 04:43:54 , - , True , False ,
One of the online checks found out that this HANA instance, 00, is not online or not master.
0 hanachecker log files were removed (based on the flag -or) even though HANA is offline
HANAChecker will now have a 1 days break and check again if this Instance is online, or master, after the break.
```

(example continues on the next slide)

# HANAChecker – HANA System Replication, HSR (2/2)



## Example (continued from previous slide):

A take over is performed (different terminal):

```
ha2adm@atgvm1s7055:/tmp/HANAChecker> hdbnsutil -sr_takeover  
done.
```

Then the HANAChecker that runs on the previous Secondary can now start with the check tasks:

```
Online Check      , 2022-10-27 04:46:59      ,      -      , True      , True      , # index services: 2, # running services: 10 out of 10  
Primary Check    , 2022-10-27 04:47:01      ,      -      , True      , True      ,  
Master Check     , 2022-10-27 04:47:03      ,      -      , True      , True      , Nameserver actual role = master  
"HANAChecker was executed 2022-10-27 04:48:15 on HA2 with  
hanachecker.py -ff hanachecker_configurationfile_example_only_me.txt -oi 1 -or 10 -k T1KEY  
If any of the mini and/or parameter checks from following check files:./tmp_sql_statements/HANA_Configuration_MiniChecks_2.00.059.01+.txt,./tmp_sql_statements/HANA_Security_MiniChecks_2.00.030+.txt  
that you are responsible for seem critical, you will be notified now.  
  
Mini Check ID M0250 Description: Max. used disk size (%) Host: atgvm1s7055 Value: 97 Expectation: <= 80 C: X SAPNote: 1870858  
Mini Check ID M0312 Description: I/O read latency data max. (ms, short-term) Host: atgvm1s7055 Value: 124.48 Expectation: <= 20.00 C: X SAPNote:  
Mini Check ID S0520 Description: Data volume encryption enabled value: no Expectation: yes C: X  
Mini Check ID S0620 Description: Users with EXPORT or IMPORT privilege Value: 38 Expectation: <= 5 C: X  
Mini Check ID S1010 Description: Adjusted authentication methods Value: no Expectation: yes C: X" is sent to christian.hansen01@sap.com  
0 hanachecker log files were removed (based on the flag -or)  
ha2adm@atgvm1s7055:/tmp/HANAChecker>
```

If you use cron, this locks will now disappear, so next cron-cycle a new HANAChecker instance can now start (see details in hanasitter.pdf)

# HANAChecker on HANACloud (1/3)



## With -cdb HANAChecker can run a HANACloud instance in BTP

Flag	Details	Explanation	Default
<b>-cdb</b>	cloud data base	if a HANACloud database name is specified, HANAChecker assumes the key (-k) is pointing to a HANACloud instance	" (HANAChecker assumes the key (-k) points to the on-prem HANA instance of the current server)

**Prerequisites:** See next two slides

**Example:** Here HANAChecker is executed from the a server (host: atgvm1s866) where a HANA on-prem is installed, however HANAChecker will run on a HANACloud instance, "CHC", and an email will be send with potential critical checks from the mini checks and the security mini checks:

```
xscadm@atgvm1s866:/tmp/HANAChecker> python hanachecker.py -k CHCKEY -cdb CHC -ct M,S -zf SQLStatements.zip -ca christian.hansen01@sap.com -oe true -as true
HANAChecker started on python 3.7.12
"HANAChecker was executed 2022-11-20 23:57:51 on CHC@CHC with
hanachecker.py -k CHCKEY -cdb CHC -ct M,S -zf SQLStatements.zip -ca christian.hansen01@sap.com -oe true -as true.
If any of the mini and/or parameter checks from following check files:./tmp_sql_statements/HANA_Configuration_MiniChecks_SHC.txt,./tmp_sql_statements/HANA_Security_MiniChecks_SHC.txt
that you are responsible for seem critical, you will be notified now.

Mini Check ID M0215 Description: Hosts with outdated CPU type Value: 1 Expectation: 0 C: X SAPNote: 2399995
Mini Check ID M0229 Description: Discrepancy between NUMA nodes and sockets Value: yes Expectation: no C: X SAPNote: 2470289
Mini Check ID M0263 Description: Asynchronous I/O queue size limit Value: 262144 Expectation: 18446744073709551615 C: X SAPNote: 1868829
Mini Check ID M0264 Description: Virtual memory map count limit Value: 135217728 Expectation: 2147483647 C: X SAPNote: 2600030
Mini Check ID M0270 Description: Unknown hardware components Value: 2 Expectation: 0 C: X SAPNote: 1828631
Mini Check ID M0364 Description: Redo log per day vs. data (%, mid-term) Value: 33.46 Expectation: <= 20.00 C: X SAPNote: 1999930
Mini Check ID M0385 Description: Savepoint vol. per day vs. data (%, mid-term) Value: 371.91 Expectation: <= 150.00 C: X SAPNote: 2100009
Mini Check ID M0551 Description: Proper setup of timezone table TTZZ Value: no Expectation: yes C: X SAPNote: 1791342
Mini Check ID M0552 Description: Number of current timezone alerts Value: 1 Expectation: 0 C: X SAPNote: 1791342
Mini Check ID M0577 Description: Tables with special unload priority Value: 7 Expectation: 0 C: X SAPNote: 2127458
Mini Check ID M0644 Description: Size of largest trace file (MB) Host: 9941364e-c0d5-4861-9aa9-d0d40e5a9964 Value: 91.78 Expectation: <= 70.00 C: X SAPNote: 2119087
Mini Check ID M0744 Description: Total SQL cache share of statistics server (%) Host: 9941364e-c0d5-4861-9aa9-d0d40e5a9964 Value: 58.85 Expectation: <= 20.00 C: X SAPNote: 2124112
```

# HANAChecker on HANACloud (2/3)



## Prerequisites (1/2):

HANAChecker must be executed on a server where a HANA on-prem is installed (for now)

To make HANAChecker run on a HANACloud instance, create the key with the SQL End Point found in BTP:

The screenshot shows the SAP HANA Database Instances interface in the SAP Cloud Platform (BTP). A specific database instance named 'CHC' is selected. The 'Actions' dropdown menu is open, and the 'Copy SQL Endpoint' option is highlighted with a red arrow. To the right, a terminal window displays the command output:

```
xscadm@atgvm1s866:/tmp/HANAChecker> hdbuserstore LIST CHCKEY  
KEY CHCKEY  
  ENV : 9941364e-c0d5-4861-9aa9-d0d40e5a9964.hana.prod-eu11.hanacloud.ondemand.com:443  
  USER: DBADMIN
```

(For simplicity I here use DBADMIN, but might be preferable to use a dedicated HANACHECKER user)

Note: Prerequisites continues on the next slide!

# HANAChecker on HANACloud (3/3)



## Prerequisites (2/2):

The target DB (i.e. the HANACloud instance in BTP) must allow the IP address of the host from where HANAChecker is executed

This can be configured from the SAP HANA Cloud Central:

(it is also possible to "Allow all IP addresses",

but this is not recommended from a security point of view)

[Manage SAP HANA Cloud](#)

SAP HANA Cloud Central interface showing the 'All Instances' list. A context menu is open over an instance named 'Chris' HANA Cloud'. The menu items are: 'Manage Configuration' (highlighted with a red arrow), 'Copy SQL Endpoint', 'Copy Instance ID', and 'Copy Configuration'. The 'Manage Configuration' item has a sub-menu with three options: 'Organization: \* THISCOMP', 'Space: \* DBSPACE', 'Alert:' (empty), and 'Type:' (empty). Below the menu, there is a table with columns: Status, Name, Storage, Memory, Compute, Actions. The status is 'RUNNING', name is 'Chris' HANA Cloud', storage is '45 GB', memory is '160 GB', compute is '3 vCPUs', and actions include a '...' button.

```
xscadm@atgvmls:~:/usr/sap/XSC> ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc
    link/ether 00:50:56:a5:e3:d7 brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.255/22 brd 10.0.3.255 scope global eth0
        valid_lft forever preferred_lft forever
```

'Connections' tab in SAP HANA Cloud Central. It shows a list of 'Trusted sources or their IP addresses that can access the instance'. Under 'Allowed Connections', it says 'Allow specific IP addresses and IP ranges (in addition to BTP)'. There are four entries listed: '18.185.27.129', '3.122.202.117', '3.65.47.134', and '192.168.6'. The last entry '192.168.6' is highlighted with a red box and a red arrow points from it to the corresponding line in the terminal output above.

# HANAChecker – Real Life Example



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

A screenshot of three Outlook email messages. Each message has a yellow envelope icon and the subject line "HANAChecker: A Potential Critical Mini-Check".

- The first message is about "Mini Check ID 1170 Description: Average database request time (ms) Host: Value: 6.91 Expectation: <= 2.00 Potential Critical: X SAP Note: 2000002".
- The second message is about "Mini Check ID 750 Description: Stat. server tables with retention < 42 days Host: Value: 14 Expectation: 0 Potential Critical: X SAP Note: 2147247".
- The third message is about "Mini Check ID 2113 Description: Last global table consistency check (days) Host: Value: never Expectation: <= 32.00 Potential Critical: X SAP Note: 2116157".