A Support Diagnostic for Checkmk

CHECKMK CONFERENCE #6 – MUNICH, APRIL 29, 2020

Walter Fisch
Head of Support
tribe29
Why a support diagnostic?

Expedites problem diagnosis and resolution by

- basic system information available upfront in standard format
- reduce number of requests "please send me additional data"
- simplified data gathering by problem type
- visibility of dependencies and incompatibilities
- basic usage statistics (optional)
Architecture

- Generate diagnostic data via WATO
- WATO automation to create data bundle via background job from selected site
- Cleanup up job
- No automated upload to tribe29
- Upload tarfile to
  - https://support.checkmk.com/
  - https://cloud.checkmk.com/
  - Your support partner
Security

- The role cmkadmin has permissions to create and read the diagnostic data
- Full transparency of collected data
- Transfer manually to tribe29 via SSL
- Highlighting and masking sensible data (level needs to be determined)
- Encryption of the data bundle
Pre-defined data collection, most ‘de-selectable’

- **General information**
  - Compulsory
  - Recommended but optional

- **Component-specific**
  - Agents
  - CMC
  - GUI
  - Event Console
  - Notifications
## Data to collect: General information

<table>
<thead>
<tr>
<th>General information</th>
<th>Recommended, but optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS version</td>
<td>mkp find</td>
</tr>
<tr>
<td>Checkmk version and edition</td>
<td>mkp list</td>
</tr>
<tr>
<td>Python version and path</td>
<td>find -L ~/local</td>
</tr>
<tr>
<td>Date and time</td>
<td></td>
</tr>
<tr>
<td>omd config show</td>
<td></td>
</tr>
<tr>
<td>Performance reports</td>
<td></td>
</tr>
<tr>
<td>Analyze configuration</td>
<td></td>
</tr>
</tbody>
</table>
### Data to collect by affected component (I)

<table>
<thead>
<tr>
<th>Event console</th>
<th>Agents</th>
<th>GUI</th>
</tr>
</thead>
<tbody>
<tr>
<td>• EC Log (var/log/mkeventd.log)</td>
<td>• cmk -D HOST</td>
<td>• var/log/web.log</td>
</tr>
<tr>
<td>• EC Configuration (etc/check_mk/mkeventd.mk)</td>
<td>• cmk -d HOST</td>
<td>• var/log/apache/access.log</td>
</tr>
<tr>
<td></td>
<td>• cmk -snmpwalk HOST</td>
<td>• var/log/apache/error.log</td>
</tr>
<tr>
<td></td>
<td>• cmk -v -debug --checks CHECK HOST</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Debug output special Agents</td>
<td></td>
</tr>
</tbody>
</table>
Data to collect by affected component (II)

Notifications

- var/log/mknotifyd.log
- var/log/mknotifyd.state
- var/log/notify.log
- var/log/cmc.log
- etc/check_mk/conf.d/wato/notifications.mk
- etc/check_mk/multisite.d/sites.mk

CMC / Livestatus

- var/log/cmc.log
- var/log/liveproxyd.log
- var/log/liveproxyd.state
- var/check_mk/core/config
- var/check_mk/core/state
- Number of helpers
- Livestatus proxy parameters
Graphical configuration in WATO: Selection

- Select side
- Choose diagnostic data
- Compulsory
  - OS and Checkmk version
  - Python version and path
- Recommended, but optional
- Issue specific
  - some de-selectable
  - maskable
Graphical configuration in WATO: Download

Background job details

- Progress
- Data collected
- Archive location
- Download button for the archive
Alternative: CLI (command-line interface)

- CLI is limited to the local site
- Options to collect additional data

```bash
daily@debian:~$ cmk --create-diagnostics-dump
Collect diagnostics information:
  General: OS, Checkmk version and edition, Time, Core, Python version and paths
Created diagnostics dump:
  '/var/check_mk/diagnostics/4bad43ff-1569-4d63-9bf8-31b0b13b6efa.tar.gz'
daily@debian:~$ tar -xvf 4bad43ff-1569-4d63-9bf8-31b0b13b6efa.tar.gz

daily@debian:~$ cat general.json
{
}
daily@debian:~$ 
```
Roadmap

- Support Diagnostic is part of Checkmk 1.7
- Diagnostic bundle covers multiple sites
- Upload via form to tribe29
- Encryption of the data bundle
- Masking sensible data
- Enhance issue types
Feedback

Please use https://checkmk.de/contact.php - feedback
Thank you

tribe29 GmbH
Kellerstraße 29
81667 München
Deutschland

Web — tribe29.com
E-Mail — mail@tribe29.com