The new checkmk Windows Agent

CHECKMK CONFERENCE #5 - MUNICH, APRIL 29, 2019
Agenda

1. THE OLD WINDOWS AGENT – REACHING ITS LIMITS
2. GREENFIELD REVAMP – THE NEW WINDOWS AGENT
3. ARCHITECTURE OF WINDOWS AGENT 2.0
4. CHANGES FROM A USER’S PERSPECTIVE
5. SHOWCASE
6. GOING FORWARD
The old Windows Agent has reached its limits

- Windows Agent: A reliable work horse, used by a large share of our customers

- But: Has become hard to maintain and extend
  - Still solves low level problems, abstractions available today
  - Linux-based toolchain, hybrid world causes tricky issues
  - Limited diagnostic tools
  - Plug-in set written in multiple languages
THE OLD WINDOWS AGENT

This is the guy to fix this:

Sergey Kipnis
Time for a greenfield revamp...
The new Windows Agent

- Principles of the new Windows Agent
  - Complete rewrite – 30,000 lines of code
  - Available from check `mk` 1.6
  - Simple logic, high quality, modern & clean code
  - Written Windows-native – compilation, installation, only MS-approved APIs
  - Only uses top-level C++ 17-20 code and modern tools
  - Standards-based MSI agent updates – no custom actions
Working on updates
30% complete
Don’t turn off your computer
Greenfield Revamp
New Agent Improvements

- Better config structure (YAML; ini still possible)
- Great diagnostic tools
- Allows automatic migration from old to new Windows Agent
- Easier extensible
- Very high test coverage; integration tests across multiple Windows platforms
Examples
[global]
crash_debug = yes

[ps]
use_wmi = no
full_path = no

[winperf]
counters = 10332:msx_queues
counters = 638:tcp_conn
counters = Terminal Services:ts_sessions
global:
  logging:
    public: yes
    file:
    debug: yes
    windbg: yes
    eventlog: yes

ps:
  enabled: yes
  use_wmi: yes
  full_path: yes

winperf:
  enabled: yes
  exe: agent
  prefix: winperf
  timeout: 10
  counters:
    - 234: phydisk
    - 510: if
    - 638: tcp_conn
    - Terminal Services: ts_sessions
Examples
Diagnostic command: **Help**

Command Line Prompt: **-help**
check_mk_service32.exe [-install|-remove|-help]
  -install  - install as a service
  -remove   - remove service
  -help     - usage

Common Testing:
check_mk_service32.exe <test|test [self seconds]|-exec>
  test  - legacy test
  -test  - short test. If self added, then agent simulates connection from monitoring site with seconds period
  -exec  - exec as app(adhoc)

Realtime Testing:
check_mk_service32.exe <-rt>
  -rt  - test realtime data with all sections and encryption

To Convert Legacy Agent Ini File into Agent Yml file:
check_mk_service32.exe -cvt <ini file> [yaml file]
infile - from Legacy Agent
yamlfile - name of an output file(optional)

To Activate/Deactivate Legacy Agent:
check_mk_service32.exe <-stop_legacy|-start_legacy>
  -stop_legacy  - stop and deactivate legacy agent
  -start_legacy - activate and start legacy agent(only for testing)

To Upgrade Legacy Agent:
check_mk_service32.exe -upgrade [force]
force - optional parameter to force upgrading

To Install Bakery Files, plugins.cap and check_mk.ini, in install folder:
check_mk_service32.exe -cap

To test Sections individually:
check_mk_service32.exe -section <name> [number [trace]]
  name  - allowed only df
  number - not 0: pause between tests in seconds, count of tests are infinite. 0 - test once
  trace - log output on the stdio
  example 'check_mk_service32 -section df 5 trace'
  test section df infinitely long with pause 5 seconds and log output on stdio
Examples

Diagnostic command: Realtime

Command Line Prompt: -rt
Diagnostic command: Individual Sections (df)

Command Line Prompt: -section df 5
To test Sections individually:

```
check_mk_service32.exe -section <name> [number [trace]]
```

name - allowed only df
number - not 0: pause between tests in seconds, count of tests are infinite. 0 - test once
trace - log output on the stdout

eexample 'check_mk_service32 -section df 5 trace'
test section df infinitely long with pause 5 seconds and log output on stdout

C:\Program Files (x86)\check_mk_service>check_mk_agent.exe -section df 5

```
<<df:sep(9)>>

OS  NTFS   716858364  620179960  96678404  87%  C:\
D:\  FAT32  98304   25926   72378  27%   D:\
TOSHIBA_EXT NTFS  976656608  950003572  26653036  98%  F:\
<<df:sep(9)>>

OS  NTFS   716858364  620179960  96678404  87%  C:\
D:\  FAT32  98304   25926   72378  27%   D:\
TOSHIBA_EXT NTFS  976656608  950003572  26653036  98%  F:\
<<df:sep(9)>>

OS  NTFS   716858364  620179960  96678404  87%  C:\
D:\  FAT32  98304   25926   72378  27%   D:\
TOSHIBA_EXT NTFS  976656608  950003572  26653036  98%  F:\
<<df:sep(9)>>

OS  NTFS   716858364  620179960  96678404  87%  C:\
D:\  FAT32  98304   25926   72378  27%   D:\
TOSHIBA_EXT NTFS  976656608  950003572  26653036  98%  F:\
<<df:sep(9)>>

OS  NTFS   716858364  620179960  96678404  87%  C:\
D:\  FAT32  98304   25926   72378  27%   D:\
TOSHIBA_EXT NTFS  976656608  950003572  26653036  98%  F:\

AC

C:\Program Files (x86)\check_mk_service>
Diagnostic command: **Self Diagnostic**

**Command Line Prompt:** `-test self 10`
Successful start of thread
Press any key to stop
Pre Start actions

Plugin Entry C:\ProgramData\CheckMK\Agent\plugins\mk_inventory.vbs changes this mode to SYNC
breakAsync C:\ProgramData\CheckMK\Agent\plugins\mk_inventory.vbs

Plugin Entry C:\ProgramData\CheckMK\Agent\plugins\netstat_an.bat changes this mode to SYNC
breakAsync C:\ProgramData\CheckMK\Agent\plugins\netstat_an.bat

Plugin Entry C:\ProgramData\CheckMK\Agent\plugins\win_license.bat changes this mode to ASYNC

Data 'C:\ProgramData\CheckMK\Agent\plugins\win_license.bat' is obsolete, age is '403712' seconds
restarting async plugin 'C:\ProgramData\CheckMK\Agent\plugins\win_license.bat'

Process 28976 is active
Cma::provider::PluginsProvider::loadConfig: entering
Pre Start actions ended
Cma::world::ExternalPort::ioThreadProc: started

Wait Loop
Starting IO ipv6:false, proposed port:0...
Modes for a key set correctly
Testing mode enabled for RT Thread
Received 3232 bytes
Process 28976 is active
Process 28976 is active
Process 28976 is active
Process 28976 is active
Testing mode enabled for RT Thread
Process 28976 has exit code 0

Thread OFF: C:\ProgramData\CheckMK\Agent\plugins\win_license.bat
Testing mode enabled for RT Thread
Testing mode enabled for RT Thread
Testing mode enabled for RT Thread
Server is going to stop
Cma::srv::ServiceProcessor::stopService: called
Stop request is set
Wait Loop END
Thread is stopped
Shutting down IO...
Stopping execution
Diagnostic command: **Self Diagnostic Part 2**
EXAMPLES

Upgrade from old to new without force

Command Line Prompt: -upgrade

C:\Program Files (x86)\check_mk_service>check_mk_service.exe -upgrade
Starting upgrade process...
Protocol File C:\Program Files (x86)\check_mk_service\upgrade.protocol exists, upgrade not required
End of!
EXAMPLES
Upgrade from old to new with force

Command Line Prompt: -upgrade force
EXAMPLES
Realtime Log Output mit DbgView
<table>
<thead>
<tr>
<th>#</th>
<th>Time</th>
<th>Debug Print</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] Skipping 'vni_webervices' by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] Skipping 'asecxh' by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] Skipping 'asecxh' by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] Skipping 'mrpe' by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] Skipping 'skype' by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] Skipping 'spool' by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] openhardwaremonitor is skipped by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.015</td>
<td>[Trace] openhardwaremonitor is skipped by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>[Trace] openhardwaremonitor is skipped by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>[Trace] openhardwaremonitor is skipped by config</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>Looking for default agent</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>Using file 'C:\Program Files (x86)\check_ak_service\check_ak_agent.exe' for winperf</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>Max Timeout set to valid value 10</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>Exec C:\Program Files (x86)\check_ak_service\check_ak_agent.exe for winperf started</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>waiting futures(only start)</td>
</tr>
<tr>
<td>1</td>
<td>60816.019</td>
<td>async RunStdCmd: 'C:\Program Files (x86)\check_ak_service\check_ak_agent.exe' --runonce winperf mail:\mailslot\Global...</td>
</tr>
<tr>
<td>1</td>
<td>60816.023</td>
<td>futures ready in 2 milliseconds</td>
</tr>
<tr>
<td>1</td>
<td>60816.035</td>
<td>From ip 10.3.1.214</td>
</tr>
<tr>
<td>1</td>
<td>60816.179</td>
<td>Key Index Terminal Services is not found, looking in registry</td>
</tr>
<tr>
<td>1</td>
<td>60816.179</td>
<td>We are using mail with address &quot;\mailslot\Global\WinAgentTest_0&quot;</td>
</tr>
<tr>
<td>1</td>
<td>60816.179</td>
<td>Sending data 'winperf' id is [403995621597167] length [2608]</td>
</tr>
<tr>
<td>1</td>
<td>60816.179</td>
<td>Send to 403995621597167 2608</td>
</tr>
<tr>
<td>1</td>
<td>60816.195</td>
<td>Received 2736 bytes from winperf</td>
</tr>
<tr>
<td>1</td>
<td>60816.195</td>
<td>caa::provider::CheckMk::makeBody: entering</td>
</tr>
<tr>
<td>1</td>
<td>60816.195</td>
<td>Absent/Empty node 'only_from' type is 1</td>
</tr>
<tr>
<td>1</td>
<td>60816.195</td>
<td>The section &quot;systemtime&quot; is disabled in config</td>
</tr>
<tr>
<td>1</td>
<td>60816.195</td>
<td>Nothing</td>
</tr>
<tr>
<td>1</td>
<td>60816.195</td>
<td>Send 3324 from 3324 data to send 3324</td>
</tr>
<tr>
<td>1</td>
<td>60816.195</td>
<td>Send 3324 bytes of data</td>
</tr>
<tr>
<td>1</td>
<td>60816.199</td>
<td>caa::world::ExternalPort::ioThreadProc: one shot ended 1</td>
</tr>
<tr>
<td>1</td>
<td>60816.542</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60817.542</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60818.542</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60819.542</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60820.542</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60821.542</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60821.542</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60822.546</td>
<td>Testing node enabled for RT Thread</td>
</tr>
<tr>
<td>1</td>
<td>60823.546</td>
<td>Testing node enabled for RT Thread</td>
</tr>
</tbody>
</table>
CHANGES FROM A USER PERSPECTIVE

What changes, what doesn’t?

- RELAX, not much changes…
  - Programmed in C++
  - Installation as a Windows service
  - Same system requirements
  - Same checks, same plugins
**Changes from a User Perspective**

Comparison shows near identical resource need

<table>
<thead>
<tr>
<th></th>
<th>OLD</th>
<th>NEW</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Load (avg)</td>
<td>&lt; 1%</td>
<td>&lt; 1%</td>
</tr>
<tr>
<td>CPU Load (max)</td>
<td>8-20%</td>
<td>8-20%</td>
</tr>
<tr>
<td>RAM</td>
<td>2-6 MB</td>
<td>2-3 MB</td>
</tr>
<tr>
<td>Disk Space</td>
<td>2.2 MB</td>
<td>2.2 MB</td>
</tr>
<tr>
<td>Latency</td>
<td>&lt; 1 Sec</td>
<td>&lt; 1 Sec</td>
</tr>
</tbody>
</table>
What about the old agent?

- Windows Server
- Windows 10
- Windows 8 / 8.1
- Windows 7
- Windows Vista
- Windows XP
- Windows ME
- ...

New Windows Agent

Old Windows Agent (still maintained)
GOING FORWARD
Our Windows Agent Roadmap

checkmk 1.6
Everything you saw today...

checkmk 1.7 and on
• First Ideas:
  • Adding Python
  • GUI for Diagnostics
  • Better homogeneity of plugins
  • Extension into other Windows domains
A final thought...

- The new Windows Agent...
  - ...brings a new level of Windows systems monitoring
  - ...offers the quality that checkmk delivers for Linux world

- But...
  - ...the agent is only the first step
  - ...it's your ideas and requirements driving us further!

So let us know!
Any questions?
Thank you!