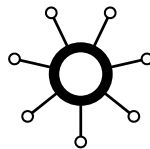


collectd-nagios: future ideas / directions

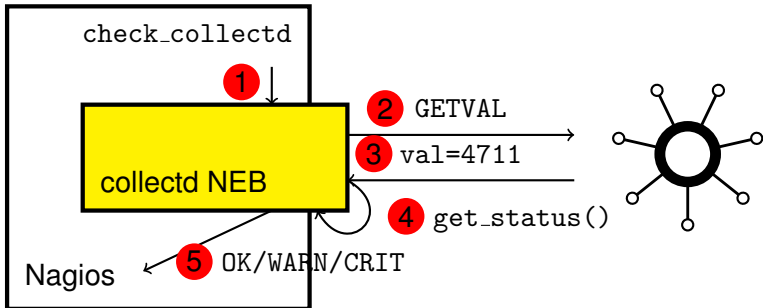
Sebastian “tokkee” Harl <sh@teamix.net>

team(ix) GmbH / **collectd** core team

Nagios Workshop 2011
May 23, 2011, Hannover

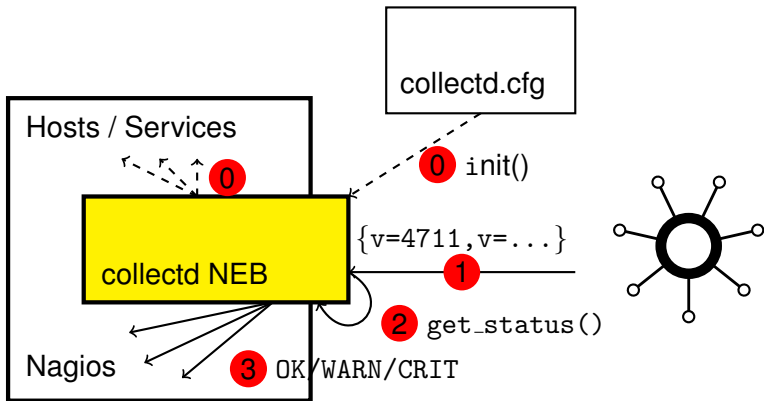


In the following, I propose a few ideas how to improve the interaction between **collectd** and Nagios as introduced at Monitors 2011. The basic idea is to implement a Nagios event broker which communicates with collectd, either using the UNIXSOCK/netcmd interface or the binary network protocol. To provide the most flexibility, it ought to be possible to use the two proposed modes simultaneously in a single instance.



- “active” event broker: “embedded collectd-nagios”
- automatically detect collectd checks (e.g., using a special command name or custom variables) and handle those in the NEB rather than executing the command
- query collectd using the UNIX socket (`unixsock` plugin)
- check thresholds according to, e.g., command parameters or custom variables
- pass active check result to Nagios directly

- → provides the same functionality as `collectd-nagios` in a more efficient way, i.e., checks are scheduled and (actively) executed by Nagios
- → the “Nagios way” of embedding collectd



- “passive” event broker: collectd network client
- configure “interesting” data-sets and thresholds in custom config file (`collectd.cfg`)
- optionally, create appropriate hosts and services on the fly
- act as server for the collectd network plugin
- check thresholds of “interesting” values, possibly after consolidating some values (30 values \times 10 seconds interval \rightarrow 5 minutes interval)
- pass active/passive check results to the Nagios core

- \rightarrow “push modell”
- \rightarrow the “collectd way” of integration into Nagios

- the active mode might block the Nagios process; possibly querying and checking the values should be done asynchronously
- passive mode might benefit from consolidation done on the (collectd) client, e.g., using some filter-chain match taking care of the consolidation and dispatching the consolidated value to the network plugin