Mod-Gearman
Distributed Monitoring based on the Gearman Framework

Sven Nierlein
18.10.2012
Consol

• Introduction
• Common Scenarios
• Installation
• Configuration
• Performance Data
• Improved Plugin Output
• Exports
• Tools
• Performance
Introduction
Introduction

- **Gearman**
  - Distributes tasks across the network from multiple clients to multiple worker
  - Load balancing
  - Client/Worker supports C, Java, Perl, PHP, Python and Shell
  - Asynchronous
Introduction

Nagios
Mod-Gearman
NEB

Gearman Daemon

Checkresults
Checks / Events
Perfdata / Exports

Perfdata

Checkresults

PNP4Nagios Worker

Tools:
send_gearman
send_multi

Mod-Gearman Worker
Common Scenarios
Load Reduction & Non Blocking

Pros

- Move blocking events away from Nagios core (Eventhandler, on-demand hostchecks)
- Reduce forking overhead from huge nagios core
- Even reduces load when both are on the same host
Load Balancing

Pros

• Spread load across multiple hosts
Distributed Setup

Worker
- hosts\(=\) no
- services\(=\) no
- eventhandler\(=\) no
- hostgroups\(=\) remote

Nagios
- hosts\(=\) yes
- services\(=\) yes
- eventhandler\(=\) yes
- hostgroups\(=\) remote

Worker
- hosts\(=\) yes
- services\(=\) yes
- eventhandler\(=\) yes

Pros
- Easy replacement for remote nagios installations
- Central configuration

www.consol.com
Distributed & Load Balancing

Worker
- hosts=no
- services=no
- eventhandler=no
- hostgroups=remote

Worker
- hosts=no
- services=no
- eventhandler=no
- hostgroups=remote

Nagios
- hosts=yes
- services=yes
- eventhandler=yes
- hostgroups=remote

Worker
- hosts=yes
- services=yes
- eventhandler=yes

Worker
- hosts=yes
- services=yes
- eventhandler=yes

Pros
- Active/active remote sites
Distributed & Load Balancing + Graphing

Worker
- hosts=no
- services=no
- eventhandler=no
- hostgroups=remote

Worker
- hosts=no
- services=no
- eventhandler=no
- hostgroups=remote

Worker
- hosts=no
- services=no
- eventhandler=no
- hostgroups=remote

Nagios
- hosts=yes
- services=yes
- eventhandler=yes
- hostgroups=remote
- perfdata=yes

Worker
- hosts=yes
- services=yes
- eventhandler=yes

Worker
- hosts=yes
- services=yes
- eventhandler=yes

PNPWorker
- hosts=yes
- services=yes
- eventhandler=yes
Check Serialization

Pros

- Useful for non-serializable checks (ex. check_selenium, java checks, etc...)
- "parallelize_check" has been removed in Nagios 3.x
- Works better than "max_concurrent_checks"
Installation

- **Standalone**
  - Packages are available for Centos/Redhat/SLES
    - including Gearmand
  - Mod-Gearman is part of the Debian 7, Wheezy

- **Consol Labs Repository**
  - [https://labs.consol.de/repo/](https://labs.consol.de/repo/)
  - Packages for Mod-Gearman, Gearmand, Thruk, OMD

- **OMD**
  - Mod-Gearman is included in OMD
Configuration
Configuration - NEB Module

- **Load Broker Module**
  - nagios.cfg:
    - `broker_module=../lib/mod_gearman/mod_gearman.o config=/etc/mod-gearman/server.cfg`
Configuration

• NEB configuration should be the sum of all workers

Worker
hosts=yes
services=yes
eventhandler=yes

Worker
hosts=yes
services=yes
eventhandler=yes

Worker
hosts=yes
services=no
eventhandler=yes
hostgroups=remote

Worker
hosts=no
services=no
eventhandler=yes

Nagios
hosts=yes
services=yes
eventhandler=yes

Nagios
hosts=yes
services=yes
eventhandler=yes
hostgroups=remote
Configuration - Common

- config
  - can be used to specify/include config files

- server
  - list of gearmand servers to connect to

- encryption
  - enable/disable encryption

- key
  - plaintext key used for encryption

- keyfile
  - read key from this file
Configuration - Queues

- **services**
  - all servicechecks

- **hosts**
  - all hostchecks

- **hostgroups**
  - list of hostgroups going into a separate queue

- **servicegroups**
  - list of servicegroups going into a separate queue

- **eventhandler**
  - execute eventhandler with Mod-Gearman

- **localhostgroups**
  - list of hostgroups not managed by Mod-Gearman

- **localservicegroups**
  - list of servicegroups not managed by Mod-Gearman

- **do_hostchecks**
  - can be used to manage hostchecks by Nagios
Configuration - Queues

- `localservicegroups?`
  - Let Nagios take care about this check

- `localhostgroups?`
  - Let Nagios take care about this check

- `servicegroups?`
  - Put check in servicegroup queue: `servicegroup_<groupname>`

- `hostgroups?`
  - Put check in hostgroup queue: `hostgroup_<groupname>`

- `hosts=yes?`
  - Put check in generic "hosts" queue

- `services=yes?`
  - Put check in generic "services" queue
set queue by custom variable

- NEB: `queue_custom_variable=worker`
- Nagios:
  ```
  define host {
    ...
    _WORKER hostgroup_test
  }
  ```
- Worker: `hostgroups=test`

http://labs.consol.de/nagios/mod-gearman/#_how_to_set_queue_by_custom_variable
Embedded Perl has serious memory leaks
  - bad for nagios
    - process grows and gets slower and slower
  - ok with Mod-Gearman
    - worker processes will be renewed from time to time

worker:
  - enable_embedded_perl=on
  - enable embedded perl
  - use_embedded_perl_implicitly=off
    - only when explicitly enabled by the script itself
  - #!/usr/bin/perl
    # nagios: +epn
Configuration - Worker

- **identifier**
  - unique name of this worker, defaults to hostname

- **min-worker**
  - minimum number of total worker

- **max-worker**
  - maximum number of total worker

- **spawn-rate**
  - rate at which new worker will be spawned

- **idle-timeout**
  - timeout in seconds before a idling worker exists

- **max-jobs**
  - maximum number of jobs before a worker exists

- **dupserver**
  - useful to send copy of result to other Gearmand server
Performance Data
Performance Data

Config
- Set “perfdata=yes” in your Mod-Gearman neb configuration.
- Set “process_performance_data=1” in your nagios.cfg.
- Adjust gearman options in process_perfdata.cfg and start pnp_gearman_worker.
Improved Plugin Output
Improved Plugin Output

- **STDERR output included:**
  - display worker identifier on errors
  - display stderr output for easy plugin debugging

  ```
  CRITICAL (for 0d 0h 0m 20s)
  CRITICAL: Return code of 255 is out of bounds. (worker: mo)
  
  Execution of /omd/sites/devel/local/lib/nagios/plugins/test1.pl aborted due to compilation errors.]
  ```

- **translated signal names**

  ```
  CRITICAL (for 0d 0h 0m 8s)
  CRITICAL: Return code of 130 is out of bounds. Plugin exited by signal SIGINT. (worker: mo)

  CRITICAL (for 0d 0h 5m 16s)
  CRITICAL: Return code of 139 is out of bounds. Plugin exited by signal SIGSEGV. (worker: mo)
  ```
Exports
Exports

- Export core events and data into gearman queues
- Format is JSON
- Write worker in any language gearman supports (C, Java, Perl, PHP, Python and Shell)
- No need to poll for data all the time

Example

- Syntax:
  export=<queue>:<returncode>:<callback>[,<callback>,...]  

  mod_gearman_neb.cfg:
  export=log_queue:1:NEBCALLBACK_LOG_DATA

Limited to a few callbacks currently:

- NEBCALLBACK_PROCESS_DATA
- NEBCALLBACK_TIMED_EVENT_DATA
- NEBCALLBACK_LOG_DATA
Tools
gearman_top

- Shows current state of all queues
  - `$ gearman_top -H localhost:4730`
check_gearman

- Use as nagios plugin to check Gearmand and worker
  
  - $ ./check_gearman -H localhost
    check_gearman CRITICAL - failed to connect to localhost:4730 - Connection refused
  
  - $ ./check_gearman -H localhost
    check_gearman OK - 0 jobs running and 0 jobs waiting. Version: 0.25|...
send_gearman

- Similar but extended functionality like send_nsca
- Can be used to send passive check result via Mod-Gearman
- Can send active results with --active
- Use --latency, --starttime, --finishtime to preserve those attributes too

```
$ ./bin/send_gearman --server=mo --keyfile=etc/mod-gearman/secret.key \
    --host='localhost' --service='ping' --message='Ping OK' --returncode=0
```
**send_multi**

- **Return multiple results from check_multi**
  - Basically:
    
    $ check_multi -r 256 -f check.cfg | ./bin/send_multi --config=mod_gearman.cfg --host=<host>

  - Better:
    
    ```bash
    #!/bin/bash
    host=$1; shift;
    other=$*
    report="256"
    [ "$other" != "" ] && report="13"

    out='.../libexec/check_by_ssh -H $host -q -C ".../check_multi -f .../multi.cfg -r $report $other" 2>&1'
    rc=$?
    if [ `echo "$out" | grep -c "CHILD"` -eq 0 -o "$other" != "" ]; then
        echo "$out"
        exit $rc
    fi
    echo "$out" | .../send_multi config=.../mod_gearman.conf host=$host
    ```

  - “check_multi -i <subcheck>” allows you to reschedule single checks from a multi.cfg

    $ ./better.sh               # for all
    $ ./better.sh -i check17    # for a single check
• All connections are initiated from the worker/client
• Use gearman_proxy.pl in case where it’s not possible to directly access the gearmand from remote locations
Thruk

- Thruks Dashboard has some Mod-Gearman related Panels
Performance
Performance

- Main reason for Mod-Gearman was making distributed monitoring easy
  - but it’s quite fast too

- all tests done with Livestatus and Mod-Gearman Module loaded

- tests were made on a single virtual machine
Performance

- Debian6 VM 2x2.5GHz with 2GB Ram + 2 external Worker
  - nearly 2,000 active service checks per second!
Questions?
Resources

- http://labs.consol.de/nagios/mod-gearman/
- http://gearman.org/
- http://docs.pnp4nagios.org/de/pnp-0.6/modes#gearman_mode
- http://my-plugin.de/wiki/projects/check_multi/feed_passive
- http://packages.debian.org/de/source/sid/mod-gearman
- http://mod-gearman.org/pkg/