Mod-Gearman
Distributed Monitoring based on the Gearman Framework

Sven Nierlein

24.05.2011
• Introduction
• Common Scenarios
• Configuration
• Performance Data
• Exports
• Tools
• OMD
• Hints
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

Mod-Gearman Worker

PNP4Nagios Worker

Tools:
 send_gearman
 send_multi

Checkresults
 Checks / Events
 Perfdata / Exports

Checkresults
 Checks / Events
 Perfdata

Perfdata

Checkresults
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

Nagios

- hosts=\textit{yes}
- services=\textit{yes}
- eventhandler=\textit{yes}

Worker

- hosts=\textit{yes}
- services=\textit{yes}
- eventhandler=\textit{yes}
Load Balancing

Pros

• Spread load across multiple hosts
Distributed Setup

Worker

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

Pros

- Easy replacement for remote nagios installations
- Central configuration

Nagios

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

Worker

- hosts = yes
- services = yes
- eventhandler = yes
Distributed & Load Balancing

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

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

Worker
hosts=yes
services=yes
eventhandler=yes
hostgroups=remote

Worker
hosts=yes
services=yes
eventhandler=yes

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

**Worker**
- hosts=\textbf{no}
- services=\textbf{no}
- eventhandler=\textbf{no}
- hostgroups=\textbf{remote}

**Worker**
- hosts=\textbf{no}
- services=\textbf{no}
- eventhandler=\textbf{no}
- hostgroups=\textbf{remote}

**Worker**
- hosts=\textbf{yes}
- services=\textbf{yes}
- eventhandler=\textbf{yes}
- hostgroups=remote

**Worker**
- hosts=\textbf{yes}
- services=\textbf{yes}
- eventhandler=\textbf{yes}

**Nagios**
- hosts=\textbf{yes}
- services=\textbf{yes}
- eventhandler=\textbf{yes}
- hostgroups=remote
- perfdata=\textbf{yes}

**PNPWorker**
- hosts=\textbf{yes}
- services=\textbf{yes}
- eventhandler=\textbf{yes}

**Worker**
- hosts=\textbf{yes}
- services=\textbf{yes}
- eventhandler=\textbf{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”
Configuration
Configuration

- NEB configuration should be the sum of all workers
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
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.
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

Currently experimental and limited to a few callbacks:

- NEBCALLBACK_PROCESS_DATA
- NEBCALLBACK_TIMED_EVENT_DATA
- NEBCALLBACK_LOG_DATA
Tools
gearman\_top

- Shows current state of all queues
  - $\text{gearman\_top -H localhost:4730}$

![Gearman Top Output](image)
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: 0.14|...
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 multi.sh:

    ```bash
    #!/bin/bash
    host=$1; shift;
    other=$*
    report="256"
    if [ "$other" != "" ]; then
      report="13"
    fi
    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.cfg host=$host
    ```

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

    ```bash
    $ ./multi.sh               # for all
    $ ./multi.sh -i check17    # for a single check
    ```
OMD

- Mod-Gearman can be enabled via "omd config"
OMD

**Configuration:**

```
etc/mod-gearman/
  ├── nagios.cfg        # loading broker
  ├── perfdata.conf     # perfdata config part of server.cfg
  ├── port.conf         # tcp port for gearmand
  │   └── secret.key    # encryption key
  │       └── server.cfg # neb module config
  │                         └── worker.cfg # gearman worker config
```

**Logfiles**

```
var/log/gearman/
  └── gearmand.log
  └── neb.log
  └── worker.log
```
OMD

- Connect multiple OMD instances
- Share the secret.key
  - Use same secret.key for all connected OMD sites
  - `/omd/sites/<site>/etc/mod-gearman/secret.key`
  - Disable gearmand on remote workers
  - Enter master sites fqdn for nodes and master as GEARMAN_PORT
Hints

- **Always monitor your gearman infrastructure! (check_gearman)**
  - Put gearman infrastructure monitors into the “localservicegroups”.
- **Enable freshness checks**
- **Secure gearmand (ex.: iptables)**
  - gearmand currently has no access control
Resources

- [http://gearman.org/](http://gearman.org/)
- [http://docs.pnp4nagios.org/de/pnp-0.6/modes#gearman_mode](http://docs.pnp4nagios.org/de/pnp-0.6/modes#gearman_mode)
- [http://my-plugin.de/wiki/projects/check_multi/feed_passive](http://my-plugin.de/wiki/projects/check_multi/feed_passive)
- [http://packages.debian.org/de/source/sid/mod-gearman](http://packages.debian.org/de/source/sid/mod-gearman)
Questions?