HP

Monitoring and troubleshooting

HP Monitoring

- monitor to make sure it's running
- publish messages are getting consumed as expected

```
[root@rabbitmq0 bin]# erl -sname foo -cookie coo
Erlang/OTP 25 [erts-13.2.2.1] [source] [64-bit] [smp:6:6] [ds:6:6:10] [async-threads:1] [jit:ns]

Eshell V13.2.2.1  (abort with ^G)
(foo@rabbitmq0)1> net_adm:names().
{ok,[{"rabbit",25672},{"foo",34291}]}
(foo@rabbitmq0)2> net_adm:localhost().
"rabbitmq0"
```

Start a new Erlang shell

Error: badrpc,nodedown and other Erlang-induced problems

HP Monitoring

- Checking that RabbitMQ is alive with AMQP simulation checks

```
#telnet  localhost  5762
```

- Issue AMQP commands : determine capable of servicing requests

```
[root@rabbitmq0 bin]# telnet rabbitmq0 5672
Trying 172.17.0.2...
Connected to rabbitmq0.
Escape character is '^]'.
AMQP

AMQP ✓ Connection closed by foreign host.
[root@rabbitmq0 bin]#
```

```
2023-08-10 09:34:20.582989+05:30 [info] <0.3035.0> accepting AMQP connection <0.3035.0> (172.17.0.2:54254 -> 172.17.0.2:5672)
2023-08-10 09:34:20.892143+05:30 [error] <0.3035.0> closing AMQP connection <0.3035.0> (172.17.0.2:54254 -> 172.17.0.2:5672):
2023-08-10 09:34:20.892143+05:30 [error] <0.3035.0> {bad_version,{13,10,13,10}}
```

HP

Monitoring - REST API

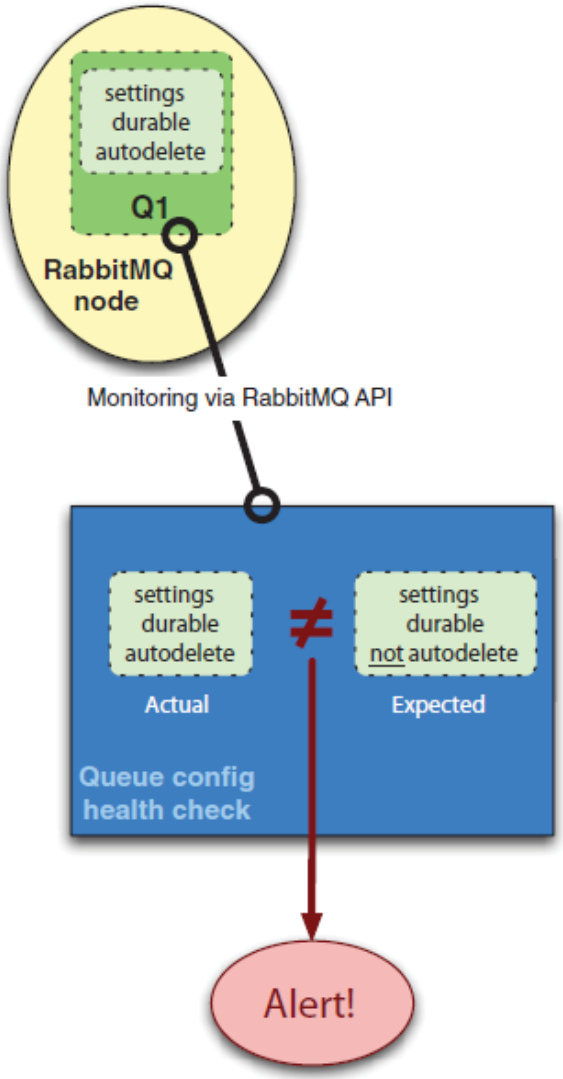
- an API call that tests the health of the Rabbit server internally
- aliveness-test:
  1. Create a new queue to receive the test message.
  2. Publish a test message with the name of the queue as the routing key into the default exchange.
  3. Consume the message when it arrives in the queue, and error out if it doesn't arrive.

```
[root@rabbitmq0 /]# curl -i http://guest:guest@rabbitmq0:15672/api/aliveness-test/%2F
HTTP/1.1 200 OK
cache-control: no-cache
content-length: 15
content-security-policy: script-src 'self' 'unsafe-eval' 'unsafe-inline'; object-src 'self'
content-type: application/json
date: Thu, 10 Aug 2023 02:24:59 GMT
server: Cowboy
vary: accept, accept-encoding, origin

{"status":"ok"}[root@rabbitmq0 /]#
```

HP

configuration changes



HP

Rabbit management API

```
$ curl -i -u guest:guest http://localhost:55672/api/queues/%2F/my_queue
```

```
Cache-Control: no-cache

{
    "memory":8400,
    "idle_since":"2011-8-16 17:24:46",
    "exclusive_consumer_pid":"",
    "exclusive_consumer_tag":"",
    "messages_ready":0,
    "messages_unacknowledged":0,
    "messages":0,
    "consumers":0,
    "backing_queue_status":
    {
        "q1":0,
        ...
    },
```

1 Queue memory usage

2 Message counts

Rabbit management API

```
    "name":"my_queue",
    "vhost":"/",
    "durable":true,
    "auto_delete":false,
    "arguments":
    {
    },
    "node":"rabbit@Phantome"
}
```

3 Queue configuration

HP

Monitoring - cluster status

- Monitor Memory usage
- number of reasons why RabbitMQ can use too much memory and run into the maximum memory cap set in the Rabbit configuration file.
  - app has a bug
  - using RabbitMQ to route large data
  - Rmq a bug that causes a slow memory leak.

HP

Monitoring - cluster status

```
$ curl -i -u guest:guest http://localhost:55672/api/nodes
```

```
Content-Type: application/json
Content-Length: 4254
Cache-Control: no-cache
[
     {
          "name":"rabbit@Phantome",
          "type":"disc",
          "running":true,
          ...
          "mem_used":31537360,
          "mem_limit":1675968512,
          "mem_alarm":false,
...
```

HP

Monitoring queue levels

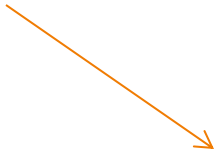
/api/queues/<vhost>/<queue_name>.

```
...
"messages_ready":0,
"messages_unacknowledged":0,
"messages":0,
...
```

HP

Accessing statistics

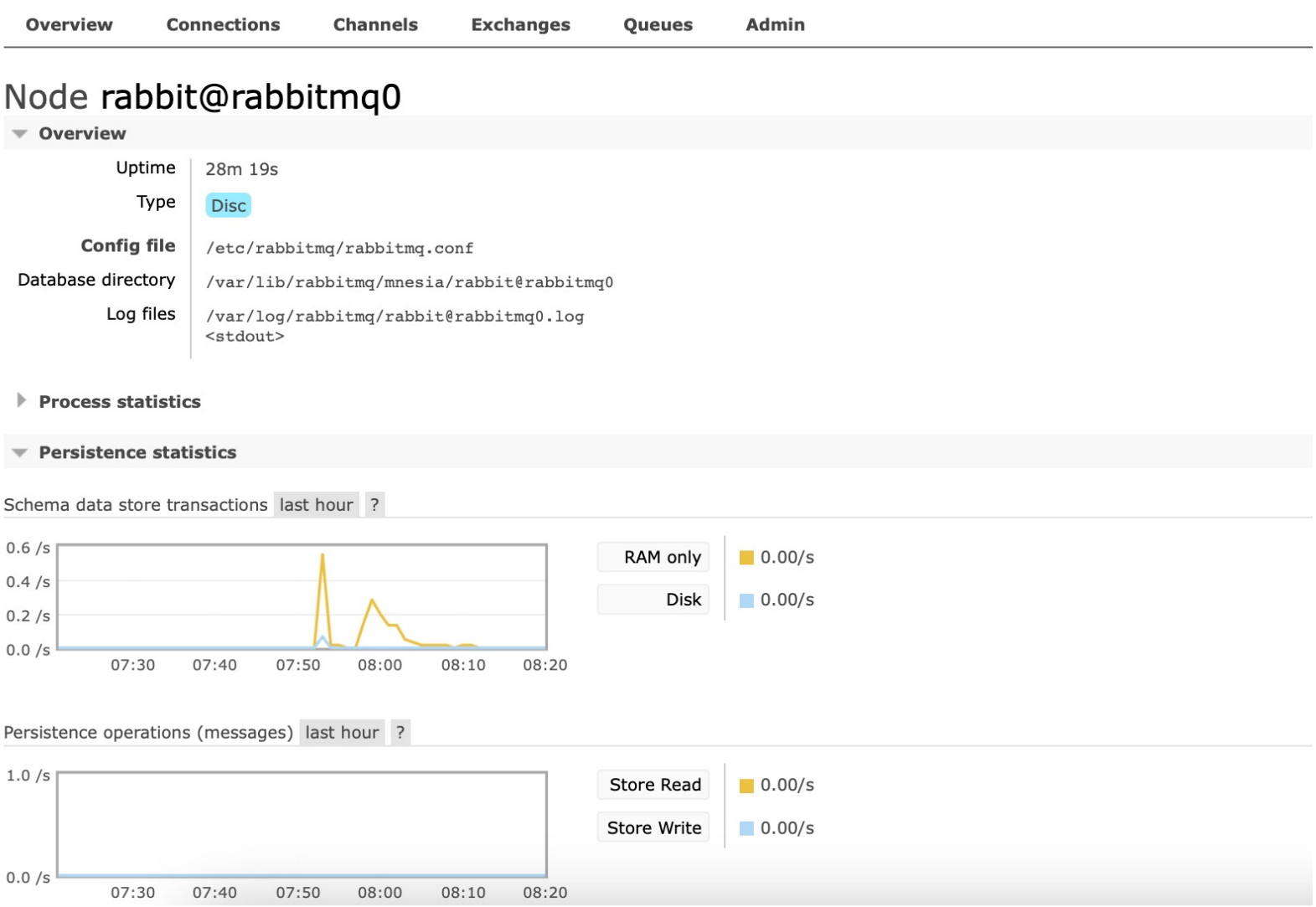
- use **rabbitmqctl** to list the total message count in your queues
- */api/queues/<vhost>/<queue_name>*.



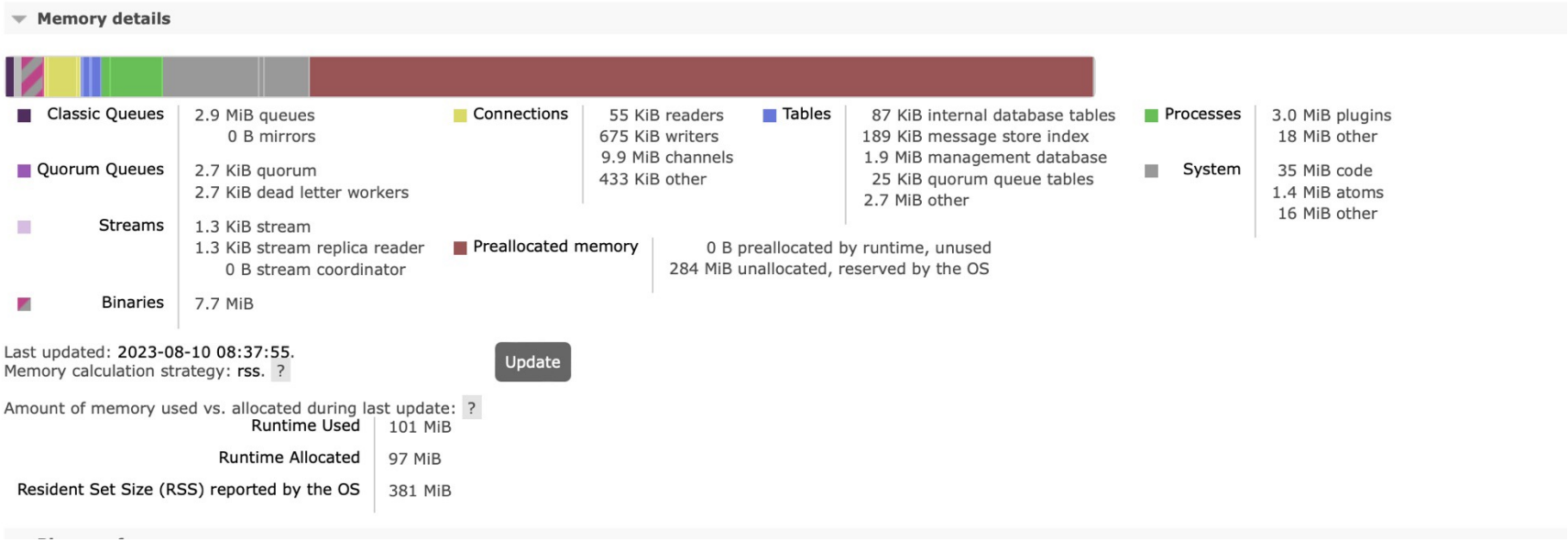
```
Memory Used (bytes): 9104
Consumer Count: 3
Messages:
    Unack'd: 3
    Ready: 4
    Total: 7
```

HP

Using Web UI



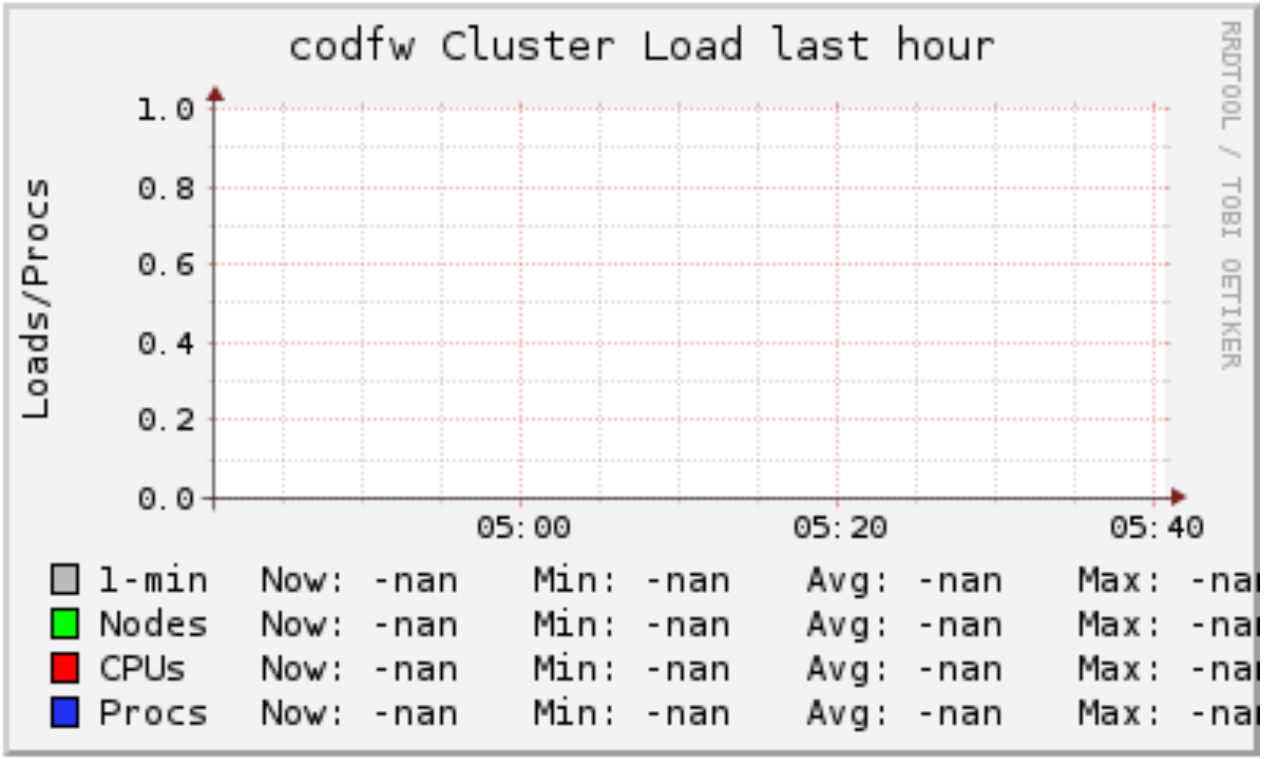
Memory Usage



By correlating breakdown categories with other metrics, e.g. the number of concurrent connections or enqueued messages, it becomes possible to detect problems that stem from an application-specific behavior (e.g. connection leaks or ever growing queues without consumers).

Monitoring

- You can integrate with Nagio and Ganglias



HP

Tracing RabbitMQ

- firehose tracing tool
  - rabbitmqctl trace_on -p [virtual host]
- can be enabled and disabled at runtime

RabbitMQ has a "firehose" feature, where the administrator can turn on (on a per-node, per-vhost basis) an exchange to which publish- and delivery-notifications should be CCed.

The firehose publishes messages to the topic exchange `amq.rabbitmq.trace`

Plugins

HP

Plugins

- To enable plugins

```
rabbitmq-plugins enable plugin-name
```

- To disable plugins

```
rabbitmq-plugins disable plugin-name
```

- list of which plugins are enabled

```
rabbitmq-plugins list
```

HP

LDAP Plugin

- Provides authentication and authorisation by deferring to an external LDAP server.
- Steps:
  - Changes in rabbit.config
  - enable the plugin, and then configure it

```
rabbitmq-plugins enable rabbitmq_auth_backend_ldap
```

HP

Configuration

- LDAP first, and then fall back to the internal database.

```
# try LDAP first
auth_backends.1 = ldap
# fall back to the internal database
auth_backends.2 = internal
```

- LDAP first subsequent authorisation against the internal database

```
# use LDAP for authentication first
 auth_backends.1.authn = ldap
# use internal database for authorisation
auth_backends.1.authz = internal
# fall back to the internal database
 auth_backends.2 = internal
```

HP

Basic configuration - rabbitmq.conf

```
auth_backends.1 = ldap


auth_ldap.servers.1  = my-ldap-server
auth_ldap.user_dn_pattern = cn=${username},ou=People,dc=example,dc=com
auth_ldap.use_ssl    = false
auth_ldap.port       = 389
auth_ldap.log        = false
```

STOMP

HP

STOMP

- The Simple (or Streaming) Text Orientated Messaging Protocol.
- provides an interoperable wire format so that STOMP clients can communicate with any STOMP message broker to provide easy and widespread messaging interoperability among many languages, platforms and brokers.

HP STOMP

- enable it

```
rabbitmq-plugins enable rabbitmq_stomp
```

- listener port to 12345

```
[
  {rabbitmq_stomp, [{tcp_listeners, [12345]}]}
].
```

Lab: Monitoring and Troubleshooting