

Grouper health check endpoint (healthcheck, status, diagnostics)

Wiki Home	Grouper Release Announcements	Grouper Guides	Grouper Deployment Guide	Community Contributions	Internal Developer Resources
---------------------------	---	--------------------------------	--	---	--

- [Introduction](#)
- [Endpoints](#)
 - [Grouper demo examples](#)
 - [Trivial option](#)
 - [DB option](#)
 - [Subject sources](#)
 - [Daemon jobs](#)
 - [Exclude/include jobs by URL param](#)
 - [All tests](#)
- [Error example](#)
- [Sample configuration](#)

Introduction

There is a health check endpoint in both the UI and WS, that can be used to monitor whether Grouper is up and running, with varying levels of its components. This can include memory in the running process, connection to the Grouper Registry DB, subject sources lookups, and that [Grouper loader](#) jobs are successfully executing. If everything is ok, a 200 HTTP code will be returned, else 500. A description of the issue will be returned as well. The point is that this URL can be pointed to be web monitoring software like Nagios, Big Brother, BMC, etc.

There is general information displayed on success as well, the server name, number of WS requests (since server started), the last error (if recent), etc.

There isn't any sensitive information in these calls, but if you want to lock them down, do that in your servlet container or web server (or don't map the servlet in the WS web.xml). You could restrict to your PC and nagios server source IP addresses for example.

Each test is configurable to restrict it (without causing an error) in grouper.properties (grouper-ws.properties prior to 2.2). If you want to customize the number of minutes since a SUCCESS should be detected in loader jobs, you can do that as well. These settings are in grouper.properties (grouper-ws.properties prior to 2.2).

Note, there is a lot of intelligent caching here so that repeated hits do not do queries each time.

Endpoints

WS open access, UI accessible but may require authentication:

- `https://<context_uri>/status?diagnosticType=<trivial|db|sources|daemonJobsOnly|all>`

UI anonymous access (when using Apache in the container):

- `https://<context_uri>/status_grouper/status?diagnosticType=<trivial|db|sources|daemonJobsOnly|all>`

Grouper demo examples

https://grouperdemo.internet2.edu/grouper_v4/status?diagnosticType=trivial

https://grouperdemo.internet2.edu/status_grouper_v4/status?diagnosticType=all

Trivial option

Use this to do checks often, or when there is a cluster, you can use this on all nodes, and a deeper check on one node only

<https://url.to.grouper.edu/grouper-ws/status?diagnosticType=trivial>

Note, this is a success, but since there was an error recently, it is displayed

```
Server: mchyzer-PC, grouperVersion: 1.6.0, up since: 2010/05/17 02:19, 0 requests
SUCCESS memoryTest: Allocating 100000 bytes to an array to make sure not out of memory (11ms elapsed)
```

```
Diagnostics errors since start: 3 (11ms elapsed)
```

```
Last diagnostics error date: 2010/05/17 02:23:27
```

```
Last diagnostics error message:
```

```
There was an error in the diagnostic task DiagnosticLoaderJobTest, Loader job
```

```
CHANGE_LOG_changeLogTempToChangeLog
```

```
:Cant find a success since: 2010/05/17 01:38:50.000, expecting one in the last 30 minutes
```

```
java.lang.RuntimeException: Cant find a success since: 2010/05/17 01:38:50.000, expecting one in the last 30 minutes
```

```
at edu.internet2.middleware.grouper.ws.status.DiagnosticLoaderJobTest.doTask(DiagnosticLoaderJobTest.java:103)
at edu.internet2.middleware.grouper.ws.status.DiagnosticTask.executeTask(DiagnosticTask.java:44)
at edu.internet2.middleware.grouper.ws.status.GrouperStatusServlet.doGet(GrouperStatusServlet.java:129)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:617)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:717)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:290)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:206)
at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:233)
at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:191)
at org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:433)
at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:128)
at org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:102)
at org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:109)
at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:293)
at org.apache.coyote.http11.Http11Processor.process(Http11Processor.java:849)
at org.apache.coyote.http11.Http11Protocol$Http11ConnectionHandler.process(Http11Protocol.java:583)
at org.apache.tomcat.util.net.JIoEndpoint$Worker.run(JIoEndpoint.java:454)
at java.lang.Thread.run(Thread.java:619)
```

DB option

This will do a lightweight query to the registry, and the memory query

<https://url.to.grouper.edu/grouper-ws/status?diagnosticType=db>

```
Server: mchyzer-PC, grouperVersion: 1.6.0, up since: 2010/05/17 02:19, 0 requests
SUCCESS memoryTest: Allocating 100000 bytes to an array to make sure not out of memory (20ms elapsed)
SUCCESS dbTest_grouper: Retrieved object from database (28ms elapsed)
```

```
Diagnostics errors since start: 3 (28ms elapsed)
```

Subject sources

This will do a find by ID on all sources, and the DB test, and the memory test. Note that the same subject.properties settings in each source that configure the Grouper startup settings will apply here as well. i.e. you can skip a source, or set the ID to search for.

```
subjectApi.source.myConfig.param.findSubjectByIdOnCheckConfig.value = true|false
subjectApi.source.myConfig.param.subjectIdToFindOnCheckConfig.value = someSubjectIdWhichMightExistOrWhatever
subjectApi.source.myConfig.param.findSubjectByIdentifiedOnCheckConfig.value = true|false
subjectApi.source.myConfig.param.subjectIdentifierToFindOnCheckConfig.value =
someSubjectIdentifierWhichMightExistOrWhatever
subjectApi.source.myConfig.param.findSubjectByStringOnCheckConfig.value = true|false
subjectApi.source.myConfig.param.stringToFindOnCheckConfig.value = someStringWhichMightExistOrWhatever
```

<https://url.to.grouper.edu/grouper-ws/status?diagnosticType=sources>

```
Server: mchyzer-PC, grouperVersion: 1.6.0, up since: 2010/05/17 02:19, 0 requests
SUCCESS memoryTest: Allocating 100000 bytes to an array to make sure not out of memory (37ms elapsed)
SUCCESS dbTest_grouper: Retrieved object from database (40ms elapsed)
SUCCESS source_g:gsa: Searched for subject by id: grouperTestSubjectByIdOnStartupASDFGHJ (42ms elapsed)
SUCCESS source_jdbc: Searched for subject by id: grouperTestSubjectByIdOnStartupASDFGHJ (45ms elapsed)
SUCCESS source_g:isa: Searched for subject by id: grouperTestSubjectByIdOnStartupASDFGHJ (45ms elapsed)
```

```
Diagnostics errors since start: 3 (45ms elapsed)
```

Daemon jobs

Diagnostic type "daemonJobsOnly" will only check the status of daemon and loader jobs. In v4.10+ Grouper diagnostics will report success based on the schedule of the job. Jobs that run every minute, hour, day, week, month, year makes threshold: 30 min, 150 min, 52 hours, 8 days, 33 days, and 367 days (unless there is an override in the config).

<https://url.to.grouper.edu/grouper-ws/status?diagnosticType=daemonJobsOnly>

```
Server: mchyzer-pc, grouperVersion: 2.2.2, up since: 2016/01/31 15:14, 0 requests
SUCCESS loader_CHANGE_LOG_changeLogTempToChangeLog: Not checking, there was a success from before: 2016/01/31
15:14:50.000, expecting one in the last 30 minutes (65ms elapsed)
SUCCESS loader_MAINTENANCE_cleanLogs: Not checking, there was a success from before: 2016/01/31 11:45:13.000,
expecting one in the last 3120 minutes (65ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_syncGroups: Not checking, there was a success from before: 2016/01/31 15:14:
00.000, expecting one in the last 30 minutes (66ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_grouperRules: Not checking, there was a success from before: 2016/01/31 15:
14:02.000, expecting one in the last 30 minutes (66ms elapsed)
SUCCESS loader_SQL_SIMPLE_loader:owner__9178d7d636de49d6b271d12ca351dc19: Not checking, there was a success
from before: 2016/01/31 13:40:04.000, expecting one in the last 3120 minutes (66ms elapsed)
```

```
Diagnostics errors since start: 0 (66ms elapsed)
```

Exclude/include jobs by URL param

You can includeOnly jobs in the URL by comma separated param (2.2.3+ and 2.2.2.api.patch.6)

https://url.to.grouper.edu/grouper/status?diagnosticType=daemonJobsOnly&includeOnly=loader_MAINTENANCE_cleanLogs,loader_CHANGE_LOG_consumer_syncGroups,loader_SQL_SIMPLE_loader:owner__9178d7d636de49d6b271d12ca351dc19

```
SUCCESS loader_CHANGE_LOG_changeLogTempToChangeLog: Loader job CHANGE_LOG_changeLogTempToChangeLog ignored in
config since URL param contains includeOnly which doesn't have 'loader_CHANGE_LOG_changeLogTempToChangeLog'
(46ms elapsed)
SUCCESS loader_MAINTENANCE_cleanLogs: Not checking, there was a success from before: 2016/01/31 11:45:13.000,
expecting one in the last 3120 minutes (46ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_syncGroups: Not checking, there was a success from before: 2016/01/31 15:14:
00.000, expecting one in the last 30 minutes (46ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_grouperRules: Loader job CHANGE_LOG_consumer_grouperRules ignored in config
since URL param contains includeOnly which doesn't have 'loader_CHANGE_LOG_consumer_grouperRules' (46ms elapsed)
SUCCESS loader_SQL_SIMPLE_loader:owner__9178d7d636de49d6b271d12ca351dc19: Not checking, there was a success
from before: 2016/01/31 13:40:04.000, expecting one in the last 3120 minutes (46ms elapsed)
```

You can exclude jobs in the URL by comma separated param (2.2.3+ and 2.2.2.api.patch.6)

https://url.to.grouper.edu/grouper/status?diagnosticType=daemonJobsOnly&exclude=loader_MAINTENANCE_cleanLogs,loader_CHANGE_LOG_consumer_syncGroups,loader_SQL_SIMPLE_loader:owner__9178d7d636de49d6b271d12ca351dc19

```
SUCCESS loader_CHANGE_LOG_changeLogTempToChangeLog: Not checking, there was a success from before: 2016/01/31 15:14:50.000, expecting one in the last 30 minutes (31ms elapsed)
SUCCESS loader_MAINTENANCE_cleanLogs: Loader job MAINTENANCE_cleanLogs ignored in config since URL param contains exclude which has 'loader_MAINTENANCE_cleanLogs' (31ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_syncGroups: Loader job CHANGE_LOG_consumer_syncGroups ignored in config since URL param contains exclude which has 'loader_CHANGE_LOG_consumer_syncGroups' (31ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_grouperRules: Not checking, there was a success from before: 2016/01/31 15:14:02.000, expecting one in the last 30 minutes (31ms elapsed)
SUCCESS loader_SQL_SIMPLE_loader:owner__9178d7d636de49d6b271d12ca351dc19: Loader job SQL_SIMPLE_loader:owner__9178d7d636de49d6b271d12ca351dc19 ignored in config since URL param contains exclude which has 'loader_SQL_SIMPLE_loader:owner__9178d7d636de49d6b271d12ca351dc19' (31ms elapsed)
```

All tests

"all" will test all loader jobs (for a success within a certain threshold), do a find by ID on all sources, and the DB test, and the memory test. By default all loader jobs will look for a success within the last 25 hours. The exception is change log jobs which look for a success within the last 30 minutes. This is configurable in the grouper-ws.properties

<https://url.to.grouper.edu/grouper-ws/status?diagnosticType=all>

```
Server: mchzyer-PC, grouperVersion: 1.6.0, up since: 2010/05/17 02:45, 0 requests
SUCCESS memoryTest: Allocating 100000 bytes to an array to make sure not out of memory (6055ms elapsed)
SUCCESS dbTest_grouper: Retrieved object from database (6076ms elapsed)
SUCCESS source_g:gsa: Searched for subject by id: grouperTestSubjectByIdOnStartupASDFGHJ (6077ms elapsed)
SUCCESS source_jdbc: Searched for subject by id: grouperTestSubjectByIdOnStartupASDFGHJ (6091ms elapsed)
SUCCESS source_g:isa: Searched for subject by id: grouperTestSubjectByIdOnStartupASDFGHJ (6091ms elapsed)
SUCCESS loader_CHANGE_LOG_changeLogTempToChangeLog: Loader job CHANGE_LOG_changeLogTempToChangeLog ignored in config (6091ms elapsed)
SUCCESS loader_MAINTENANCE_grouperReport: Loader job MAINTENANCE_grouperReport ignored in config (6091ms elapsed)
SUCCESS loader_MAINTENANCE_cleanLogs: Found the most recent success: 2010/05/17 02:39:00.000, expecting one in the last 1500 minutes (6122ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_chrisTest: Loader job CHANGE_LOG_consumer_chrisTest ignored in config (6122ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_chrisTest: Loader job CHANGE_LOG_consumer_chrisTest ignored in config (6122ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_xmpp: Loader job CHANGE_LOG_consumer_xmpp ignored in config (6122ms elapsed)
SUCCESS loader_CHANGE_LOG_consumer_xmpp: Loader job CHANGE_LOG_consumer_xmpp ignored in config (6122ms elapsed)
SUCCESS loader_SQL_GROUP_LIST_aStem:aGroup2__f74068fd47124b079ea0c750354f6935: Found the most recent success: 2010/05/17 02:39:00.000, expecting one in the last 1500 minutes (6125ms elapsed)
SUCCESS loader_SQL_SIMPLE_aStem:aGroup__a186d80e0fe946b78dba45d16a2a1be7: Found the most recent success: 2010/05/17 02:39:00.000, expecting one in the last 1500 minutes (6132ms elapsed)
SUCCESS loader_ATTR_SQL_SIMPLE_penn:community:employee:orgPermissions:orgs__a8c2933dd66945af9755372efa9141b5: Found the most recent success: 2010/05/17 02:39:00.000, expecting one in the last 1500 minutes (6135ms elapsed)

Diagnostics errors since start: 0 (6135ms elapsed)
```

Error example

HTTP Status 500 -

type Exception report

message

description The server encountered an internal error () that prevented it from fulfilling this request.

exception

java.lang.RuntimeException:

There was an error in the diagnostic task DiagnosticLoaderJobTest, Loader job
CHANGE_LOG_changeLogTempToChangeLog

:Cant find a success since: 2010/05/17 01:38:50.000, expecting one in the last 30 minutes
edu.internet2.middleware.grouper.ws.status.GrouperStatusServlet.doGet(GrouperStatusServlet.java:191)
javax.servlet.http.HttpServlet.service(HttpServlet.java:617)
javax.servlet.http.HttpServlet.service(HttpServlet.java:717)

root cause

java.lang.RuntimeException: Cant find a success since: 2010/05/17 01:38:50.000, expecting one in the last 30 minutes
edu.internet2.middleware.grouper.ws.status.DiagnosticLoaderJobTest.doTask(DiagnosticLoaderJobTest.java:103)
edu.internet2.middleware.grouper.ws.status.DiagnosticTask.executeTask(DiagnosticTask.java:44)
edu.internet2.middleware.grouper.ws.status.GrouperStatusServlet.doGet(GrouperStatusServlet.java:129)
javax.servlet.http.HttpServlet.service(HttpServlet.java:617)
javax.servlet.http.HttpServlet.service(HttpServlet.java:717)

note The full stack trace of the root cause is available in the Apache Tomcat/6.0.20 logs.

Sample configuration

grouper.properties

```

# if ignore tests. Note, in job names, invalid chars need to be replaced with underscore (e.g. colon)
# anything in this regex: [^a-zA-Z0-9._-]
ws.diagnostic.ignore.memoryTest = false
ws.diagnostic.ignore.dbTest_grouper = false
ws.diagnostic.ignore.source_jdbc = false
ws.diagnostic.ignore.loader_CHANGE_LOG_changeLogTempToChangeLog = false
ws.diagnostic.ignore.loader_OTHER_JOB_syncAllPitTables = false
ws.diagnostic.ignore.loader_MAINTENANCE_grouperReport = false

# number of minute that can go by without a success before an error is thrown
ws.diagnostic.defaultMinutesSinceLastSuccess = 3120
ws.diagnostic.defaultMinutesChangeLog = 30
ws.diagnostic.minutesSinceLastSuccess.loader_SQL_GROUP_LIST__aStem_aGroup2 = 60
ws.diagnostic.minutesSinceLastSuccess.loader_OTHER_JOB_usduDaemon = 14400

# list groups which should check the size, in this case, "employee" or "students" in the key name is a variable
# {valueType: "group", required: true, regex: "^ws\\.diagnostic\\.checkGroupSize\\.([a-zA-Z0-9._-]+)\\.group\\.$"}
# ws.diagnostic.checkGroupSize.students.groupName = community:students

# min group size of known groups
# {valueType: "integer", required: true, regex: "^ws\\.diagnostic\\.checkGroupSize\\.([a-zA-Z0-9._-]+)\\.minSize\\.$"}
# ws.diagnostic.checkGroupSize.students.minSize = 18000

# if a change log consumer hasn't had a success but it is running and progress is being made, treat as a success
# {valueType: "boolean", required: true}
ws.diagnostic.successIfChangeLogConsumerProgress = true

# allow diagnostics from these IP ranges, e.g. 1.2.3.4/32 or 2.3.4.5/24, comma separated, leave blank if
# available from everywhere
# {valueType: "string", multiple: true}
ws.diagnostic.sourceIpAddresses =

# if status details should be sent to the client or just logged
# {valueType: "boolean", required: true}
ws.diagnostic.sendDetailsInResponse = true

```