

Name generate-jvm-report– shows the threads, classes, memory, and loggers for a given target instance.

Synopsis generate-jvm-report
[--terse={true|false}] [--echo={true|false}]
[--interactive={true|false}] [--host *host*]
[--port *port*] [--secure| -s] [--user *admin_user*]
[--passwordfile *filename*] [--help]
[--type =summary|memory|class|thread|log] [*target*]

Description This command creates a report that shows the threads (dump of stack trace), classes, memory, or loggers for a given target instance, including the domain administration server (DAS). This command works only with the application server instance processes.

This command provides an alternative to sending Ctrl+Break or kill -3 signals to application server processes to obtain a stack trace for processes that are hanging.

The information in the report is obtained from managed beans (MBeans) and MXBeans that are provided in the Java™ Platform, Standard Edition (Java SE) or JDK™ software with which Sun GlassFish™ Enterprise Server is being used.

If Enterprise Server is running in the Java Runtime Environment (JRE™) software from JDK release 6 or Java SE 6, additional information is provided, for example:

- System load on the available processors
- Object monitors that are currently held or requested by a thread
- Lock objects that a thread is holding, for example, ReentrantLock objects and ReentrantReadWriteLock objects

If the JRE software cannot provide this information, the report contains the text NOT_AVAILABLE.

This command is supported in remote mode only. The command does not work if the target server instance is not running.

Options

- t --terse
Indicates that any output data must be very concise, typically avoiding human-friendly sentences and favoring well-formatted data for consumption by a script. Default is false.
- e --echo
If set to true, the command-line statement is echoed on the standard output. Default is false.
- I --interactive
If set to true (default), only the required password options are prompted.
- H --host
The machine name where the domain administration server is running. The default value is localhost.

-p --port

The HTTP port or HTTPS port for administration. This port is the port in the URL that you specify in your web browser to manage the domain, for example, `http://localhost:4848`.

The default port number for administration is 4848.

-s --secure

If set to true, uses SSL/TLS to communicate with the domain administration server.

The default is false.

-u --user

The user name of the authorized administrative user of the domain administration server.

If you have authenticated to a domain by using the `asadmin login` command, you need not specify the `--user` option for subsequent operations on the domain.

--passwordfile

Specifies the name, including the full path, of a file that contains the password entries in a specific format.

The entry for a password must have the `AS_ADMIN_` prefix followed by the password name in uppercase letters. For example, to specify the password for the domain administration server, use an entry with the following format:

```
AS_ADMIN_PASSWORD=password
```

In this example, *password* is the actual administrator password.

The following other passwords can also be specified:

- `AS_ADMIN_MAPPEDPASSWORD`
- `AS_ADMIN_USERPASSWORD`
- `AS_ADMIN_ALIASPASSWORD`

All remote commands must specify the administration password to authenticate to the domain administration server. The password can be specified by one of the following means:

- Through the `--passwordfile` option
- Through the `asadmin login` command
- Interactively at the command prompt

The `asadmin login` command can be used only to specify the administration password. For other passwords that remote commands require, use the `--passwordfile` option or specify them at the command prompt.

After authenticating to a domain by using the `asadmin login` command, you need not specify the administration password through the `--passwordfile` option for subsequent

operations on the domain. However, only the `AS_ADMIN_PASSWORD` option is not required. You still must provide the other passwords, for example, `AS_ADMIN_USERPASSWORD`, when required by individual commands, such as `update-file-user`.

For security reasons, a password that is specified as an environment variable is not read by the `asadmin` command.

The default value for `AS_ADMIN_MASTERPASSWORD` is `changeit`.

`--help`

Displays the help text for the command.

`--type`

The type of report that is to be generated.

`summary`

Displays summary information about the threads, classes, and memory (default).

`memory`

Provides information about heap and non-heap memory consumption, memory pools, and garbage collection statistics for a given target instance.

`class`

Provides information about the class loader for a given target instance.

`thread`

Provides information about threads running and the thread dump (stack trace) for a given target instance.

`log`

Provides information about the loggers that are registered in the Virtual Machine for the Java platform (Java Virtual Machine or JVM™ machine).¹

Operands *target*

Do not specify this option. This option is retained for compatibility with other releases. If you specify this option, a syntax error does not occur. Instead, the command runs successfully and the option is silently ignored.

Examples **EXAMPLE 1** Obtaining Summary Information About Threads, Classes, and Memory

```
asadmin> generate-jvm-report --user admin --passwordfile passwords.txt
--type summary server1
```

Operating System Information:

Name of the Operating System: Linux

Binary Architecture name of the Operating System: i386, Version:

2.6.9-22.ELsmp

Number of processors available on the Operating System: 2

...

...

¹ The terms "Java Virtual Machine" and "JVM" mean a Virtual Machine for the Java platform.

EXAMPLE 1 Obtaining Summary Information About Threads, Classes, and Memory *(Continued)*

```
...
user.language = en
user.name = root
user.timezone = America/Los_Angeles
Command generate-jvm-report executed successfully
```

EXAMPLE 2 Obtaining Information About Memory Usage

```
asadmin> generate-jvm-report --type=memory
Memory Pool Name: Eden Space
Memory that Java Virtual Machine initially requested to the
Operating System: 2,097,152 Bytes
Memory that Java Virtual Machine is guaranteed to receive from the
Operating System: 9,895,936 Bytes
Maximum Memory that Java Virtual Machine may get from the
Operating System: 168,427,520 Bytes
Note that this is not guaranteed.
Memory that Java Virtual Machine uses at this time: 7,159,784 Bytes
...
Memory Pool Name: Survivor Space
Memory that Java Virtual Machine initially requested to the
Operating System: 65,536 Bytes
Memory that Java Virtual Machine is guaranteed to receive from the
Operating System: 262,144 Bytes
Maximum Memory that Java Virtual Machine may get from the
Operating System: 5,242,880 Bytes.
Note that this is not guaranteed.
Memory that Java Virtual Machine uses at this time: 35,208 Bytes
...
Name of the Garbage Collector: MarkSweepCompact
Number of collections occurred using this garbage collector: 0 Bytes
Garbage Collection Time: 0 Seconds 0 Milliseconds
Name of the Garbage Collector: Copy
Number of collections occurred using this garbage collector: 47 Bytes
Garbage Collection Time: 1 Seconds 395 Milliseconds

Heap Memory Usage:
Memory that Java Virtual Machine initially requested to the
Operating System: 0 Bytes
Memory that Java Virtual Machine is guaranteed to receive from the
Operating System: 30,728,192 Bytes
Maximum Memory that Java Virtual Machine may get from the
Operating System: 531,628,032 Bytes.
Note that this is not guaranteed.
Memory that Java Virtual Machine uses at this time: 25,434,432 Bytes

Non-heap Memory Usage:
```

EXAMPLE 2 Obtaining Information About Memory Usage *(Continued)*

Memory that Java Virtual Machine initially requested to the
Operating System: 29,523,968 Bytes
Memory that Java Virtual Machine is guaranteed to receive from the
Operating System: 32,833,536 Bytes
Maximum Memory that Java Virtual Machine may get from the
Operating System: 121,634,816 Bytes
Note that this is not guaranteed.
Memory that Java Virtual Machine uses at this time: 22,920,624 Bytes

Approximate number of objects for which finalization is pending: 0

Command `generate-jvm-report` executed successfully.

EXAMPLE 3 Obtaining Information About Running Threads

```
asadmin> generate-jvm-report --type=thread
Full Java Thread Dump Java HotSpot(TM) Client VM 1.5.0_14-b03 Sun Microsystems Inc.
Number of threads: 39
Number of daemon threads: 33
Peak live thread count since the Java virtual machine started or peak was reset: 44
Is support for thread contention monitoring available on this JVM? [true]
Is thread contention monitoring enabled? [false]. If false, some thread
synchronization statistics are not be available.
Is support for CPU time measurement for any thread available on this JVM? [true]
Is thread CPU time measurement enabled? [true]. If false, thread execution times
are not available for any thread.
-----
Thread Execution Information:
-----
Thread "RMI ConnectionExpiration-[129.146.11.147:61218]" thread-id: 84 thread-state:
TIMED_WAITING
    at: java.lang.Thread.sleep(Native Method)
    at: sun.rmi.transport.tcp.TCPChannel$Reaper.run(TCPChannel.java:446)
    at: java.lang.Thread.run(Thread.java:595)
Thread Synchronization Statistics:
-----
Number of times this thread was blocked (to enter/reenter a Monitor): 0
Number of times this thread waited for a notification (i.e. it was in WAITING or
TIMED_WAITING state): 0
Total CPU time for this thread: 0 seconds 131,855 nanoseconds.
User-level CPU time for this thread: 0 seconds 131,855 nanoseconds.
Object Monitors currently held or requested by this thread: NOT_AVAILABLE
Ownable Synchronizers (e.g. ReentrantLock and ReentrantReadWriteLock) held by
this thread: NOT_AVAILABLE
-----
```

EXAMPLE 3 Obtaining Information About Running Threads *(Continued)*

```

...
Thread Execution Information:
-----
Thread "Reference Handler" thread-id: 2 thread-state: WAITING Waiting on lock:
java.lang.ref.Reference$Lock@f63055
    at: java.lang.Object.wait(Native Method)
    at: java.lang.Object.wait(Object.java:474)
    at: java.lang.ref.Reference$ReferenceHandler.run(Reference.java:116)
Thread Synchronization Statistics:
-----
Number of times this thread was blocked (to enter/reenter a Monitor): 318
Number of times this thread waited for a notification (i.e. it was in WAITING or
TIMED_WAITING state): 43
Total CPU time for this thread: 0 seconds 26,004,119 nanoseconds.
User-level CPU time for this thread: 0 seconds 26,004,119 nanoseconds.
Object Monitors currently held or requested by this thread: NOT_AVAILABLE
Ownable Synchronizers (e.g. ReentrantLock and ReentrantReadWriteLock) held by this
thread: NOT_AVAILABLE
No deadlock found

```

Command generate-jvm-report executed successfully.

EXAMPLE 4 Obtaining Information About a Class Loader

```

asadmin> generate-jvm-report --type=class
Class loading and unloading in the Java Virtual Machine:
Number of classes currently loaded in the Java Virtual Machine: 2,798
Number of classes loaded in the Java Virtual Machine since the startup: 2,798
Number of classes unloaded from the Java Virtual Machine: 0
Just-in-time (JIT) compilation information in the Java Virtual Machine:
Java Virtual Machine compilation monitoring allowed: true
Name of the Just-in-time (JIT) compiler: HotSpot Client Compiler
Total time spent in compilation: 0 Hours 0 Minutes 2 Seconds

```

Command generate-jvm-report executed successfully.

EXAMPLE 5 Obtaining Information About Loggers

```

asadmin> generate-jvm-report --type=log
Effective logging properties file:
[/home/someuser/glassfishv3-prelude/glassfish/domains/domain1/config/
logging.properties].
If null, it indicates JRE standard file.
Number of loggers currently registered in the JVM: [35]. Details follow:
If the level is blank, it is inherited from parent logger

```

EXAMPLE 5 Obtaining Information About Loggers (Continued)

```
Parent logger is the nearest existing parent logger
Logger Name | Logging Level | Parent Logger Name
-----
|INFO|root
-----
GRIZZLY||root
-----
Grizzly||root
-----
com.sun.enterprise.naming||root
-----
com.sun.enterprise.v3.admin.commands||root
-----
com.sun.enterprise.v3.server.DynamicReloadService||root
-----
com.sun.enterprise.v3.server.DynamicReloader||root
-----
com.sun.grizzly.util.http.HttpRequestURIDecoder||root
-----
com.sun.jmx.remote.opt.util||root
-----
global||root
-----
grizzly||root
-----
javax.enterprise.resource.jta||root
-----
javax.enterprise.resource.resourceadapter||root
-----
...
sun.rmi.transport.misc||root
-----
sun.rmi.transport.tcp||root
-----
sun.rmi.transport.tcp.proxy||sun.rmi.transport.tcp
-----
```

Command generate-jvm-report executed successfully.

| | | |
|-------------|---|--------------------------------|
| Exit Status | 0 | command executed successfully |
| | 1 | error in executing the command |