

---

**Name** create-jvm-options– creates options in the Java configuration or profiler element of the `domain.xml` file.

**Synopsis** `create-jvm-options`  
[`--help`]  
[`--target target`]  
[`--profiler={true|false}]` (`jvm_option_name=jvm_option_value`)  
[:`jvm_option_name=jvm_option_value*`]

**Description** The `create-jvm-options` subcommand creates command-line options that are passed to the Java™ application launcher when Enterprise Server is started. These options are stored in the Java configuration element or profiler element of the `domain.xml` file. If options are created for a profiler, they are used to record the settings that are required to start a particular profiler.

**Note** – In the syntax, there can be more than one option, separated by a colon.

This subcommand is supported in remote mode only.

The `create-jvm-options` subcommand creates the following options:

- Java system properties. System properties are set through the `-D` option of the Java application launcher, for example:
  - `-Djava.security.manager`
  - `-Denvironment=Production`
- Startup parameters for the Virtual Machine for the Java platform (Java Virtual Machine or JVM™ machine).<sup>1</sup> Startup parameters are preceded by the dash character (`-`), for example:
  - `--XX:PermSize=size`
  - `-Xmx1024m`
  - `-client`
  - `-d64`

In the following situations, you must terminate the command with a double-dash (`--`) before the options:

- You are specifying *only one* system property with just a name and no value, for example, `-Djava.security.manager`.
- You are specifying a startup parameter for the JVM machine without a value, for example, `-Xmx1024m`

**Note** – If the command specifies an option that already exists, the command does not re-create the option.

---

<sup>1</sup> The terms "Java Virtual Machine" and "JVM" mean a Virtual Machine for the Java platform.

The addition of some options requires a restart of the domain administration server (DAS) to become effective. Other options are set immediately in the environment of the DAS and do not require a restart.

Whether a restart is required depends on the type of the option.

- Restart is not required for Java system properties whose names do *not* start with -Djava. or -Djavax. (including the trailing period). For example, restart is *not* required for the Java system property -Denvironment=Production.
- Restart is required for the following options:
  - System properties whose names start with -Djava. or -Djavax. (including the trailing period), for example, -Djava.security.manager
  - Startup parameters for the JVM machine, for example, -Xmx1024m, -client or -d64

To restart the DAS, use the `restart-domain(1)` command.

For more information about the Java application launcher, see the reference page for the operating system that you are using:

- Solaris™ Operating System (Solaris OS) and Linux: *java - the Java application launcher* (<http://java.sun.com/javase/6/docs/technotes/tools/solaris/java.html>)
- Windows: *java - the Java application launcher* (<http://java.sun.com/javase/6/docs/technotes/tools/windows/java.html>)

**Options** `--help`

Displays the help text for the subcommand.

`--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 subcommand runs successfully and the option is silently ignored.

`--profiler`

Indicates whether the JVM options are for the profiler. The profiler must exist for this option to be true. Default is false.

**Operands** `jvm_option_name`

The left side of the equal sign (=) is the JVM option name. The right side of the equal sign (=) is the JVM option value. A colon (:) is a delimiter for multiple options.

**Note** – If an option name or option value contains a colon, the backslash (\) must be used to escape the colon in the name or value. Other characters might also require an escape character. For more information about escape characters in subcommand options, see the [asadmin\(1M\)](#) man page.

**Examples** **EXAMPLE 1** Setting Java System Properties

JVM machine options must start with a dash (-). Use the backslash (\) to escape the dash delimiter.

```
asadmin> create-jvm-options --interactive=true
--passwordfile passwords.txt --terse=false
--user admin --host localhost --port 4848 --target server
\--Dunixlocation=/root/example:
\--Dvariable=$HOME:
\--Dwindowslocation=d:\\sun\\appserver:
\--Doption1=-value1
created 4 option(s)
```

Command create-jvm-options executed successfully

**EXAMPLE 2** Setting a Single Startup Parameter for the JVM Machine

This example sets the maximum available heap size to 1024. Because this example is setting a startup parameter for the JVM machine without a value, the command is terminated by a double dash -- before the parameter.

```
asadmin> create-jvm-options -- -Xmx1024m
created 1 option(s)
Command create-jvm-options executed successfully.
```

**EXAMPLE 3** Setting Multiple Startup Parameters for the JVM Machine

This example sets startup parameters for the JVM machine. The maximum available heap size is set to 1024. Information about garbage collection is to be provided.

```
asadmin> create-jvm-options "-Xmx1024m:-XX\:+PrintGCDetails"
created 1 option(s)
Command create-jvm-options executed successfully.
```

Because one parameter already existed before the command was run, the command reports that only one option was set.

<b>Exit Status</b>	0	subcommand executed successfully
	1	error in executing the subcommand

**See Also** [delete-jvm-options\(1\)](#), [list-jvm-options\(1\)](#), [restart-domain\(1\)](#)  
[asadmin\(1M\)](#)