

Apache Harmony

{scrollbar}

top

Running Geronimo on Harmony

This article describes the ways and known issues to run [Geronimo v2.1](#) using [Apache Harmony](#) as a JVM.

This article has the following structure:

- [#General note](#)
- [#Unresolved issues](#)
- [#Geronimo configuration](#)
 - [#Configure the server to use Harmony RMI Registry provider for JNDI](#)
 - [#Use TLS instead of SSL](#)
 - [#Change default keystore type](#)
 - [#Customize the server startup](#)
 - [#Configure Deployer to use Harmony RMI Registry provider for JNDI](#)
- [#Harmony configuration](#)
 - [#Use DRLVM](#)

General note

This article was written using the following environment:

[Geronimo v2.1 Jetty](#) full J2EE version ([geronimo-jetty6-javaee5-2.1-SNAPSHOT-bin.zip](#))

[Harmony](#) snapshot r617157 (Jan 31 2008), Windows/ia32/msvc 1310, debug build

[Microsoft Windows XP Professional Service Pack 2](#), 32-bit, English

With other versions and in a different environment, the behavior may differ.

Unresolved issues

Here's the brief summary of all the unresolved issues that require attention:

- [HARMONY-5191](#) – [Harmony](#) doesn't have [SSL](#) implementation (investigated, [workarounded](#)).
- [HARMONY-5462](#) – [Harmony](#) doesn't support `-javaagent` VM option (investigated, [workarounded](#)).
- [HARMONY-4014](#) – [Geronimo](#) takes up all the CPU resources available (investigated, [workarounded](#)).
- [HARMONY-5523](#) – [Geronimo](#) Console displays memory usage information incorrectly (investigated, not critical).
- Application fails to terminate properly – pressing `Ctrl-C` starts shutdown process that is never finished (uninvestigated).

Geronimo configuration

It's recommended that you backup your `var/config/config.xml` file before editing it – you may need the original file if you want to run your [Geronimo](#) installation on [Sun Java](#) or if you make a mistake while editing and damage the file beyond repair.

Configure the server to use Harmony RMI Registry provider for JNDI

[Harmony](#) places [JNDI providers](#) in `org.apache.harmony.jndi` package tree, not in `com.sun.jndi` like [Sun Java](#) does, so you have to point [Geronimo](#) into the right direction to locate and properly access the [RMI](#) Registry provider.

For that, edit `var/config/config.xml` file, adjust **NamingProperties** GBean configuration as follows:

```
xmlsolid <gbean name="NamingProperties"> <attribute name="namingFactoryUrlPkgs">org.apache.harmony.jndi.provider</attribute> <attribute name="namingProviderUrl">rmi:/${ServerHostname}:${NamingPort + PortOffset}</attribute> </gbean>
```

Use TLS instead of SSL

[Harmony](#) doesn't have [SSL](#) implementation (see [HARMONY-5191](#)), so the following statements have to be added to `var/config/config.xml` file to instruct [Geronimo](#) to use [TLS](#) instead of [SSL](#):

```
xmlsolid <module name="org.apache.geronimo.configs/j2ee-corba-yoko/2.1-SNAPSHOT/car"> <gbean name="CORBASSSLConfig"> <attribute name="protocol">TLS</attribute> </gbean> <gbean name="NameServer"> ...
```

Change default keystore type

[Geronimo](#) uses a JKS keystore that [Harmony](#) doesn't support, see [GERONIMO-3757](#) for details. So you have to generate another keystore, of PKCS12, BKS (default for [Harmony](#)), or another type [Harmony](#) has support for, and to instruct [Geronimo](#) to use it.

To convert the existing [Geronimo](#) JKS keystore to PKCS12 type, you could use a simple Java program like [JKStoPKCS12.java](#). The program needs to be run on [Sun Java](#), as it supports both JKS and PKCS12 keystores. If you need a BKS keystore, you'd need to convert your PKCS12 keystore again, with [PKCS12toBKS.java](#) running on [Harmony](#) – as [Sun Java](#) doesn't support BKS and [Harmony](#) doesn't support JKS.

Alternatively, you could use the pregenerated PKCS12 or BKS keystore attached to this article: [geronimo.pkcs12](#), [geronimo.bks](#).

To instruct [Geronimo](#) to use alternate keystore, copy your keystore file to `var/security/keystores/` directory and add the following to `var/config/config.xml` file:

```
xmlsolid <module name="org.apache.geronimo.framework/server-security-config/2.1-SNAPSHOT/car"> <gbean name="geronimo-default"> <attribute name="keystorePath">var/security/keystores/geronimo.pkcs12</attribute> <attribute name="keystoreType">PKCS12</attribute> </gbean> </module>
```

Alternatively, you may just overwrite you keystore file over the `var/security/keystores/geronimo-default`, then you may omit the `keystorePath` tag in the config. However, this is not recommended, as you may need the original keystore if you need to run your [Geronimo](#) installation on [Sun Java](#). If you decide to overwrite, don't forget to backup your original keystore. Just in case you have no backup and need the original keystore, you may use the one attached to this article, just in case: [geronimo.jks](#).

Customize the server startup

You may either edit the [Geronimo](#) startup scripts or run the server from the command line, but in either case you should consider the following adjustments:

1. [Harmony](#) doesn't accept the `-javaagent` option ([HARMONY-5462](#)), so you have to omit it. Or, you may remove `bin/jpa.jar` file from [Geronimo](#) distribution – this way the `-javaagent` option would be omitted by the [Geronimo](#) startup script.
2. [Geronimo v2.1](#) needs access to Internet for applications like [SPECjAppServer2004](#) to be deployed – it tries to fetch XML schemas from <http://java.sun.com> site. So if you're going to deploy applications like that, and you're behind a firewall, you should add the appropriate `-Dhttp.proxyHost=` and `-Dhttp.proxyPort=` options to the [Geronimo](#) command line. This problem is caused by [OPENEJB-700](#) bug and should disappear after that bug is fixed and the fix is propagated to a new version of [Geronimo](#).
3. [SPECjAppServer2004](#) and other applications may require additional specification of [JNDI](#) naming conventions for EJB, like this: `-Dopenejb.jndiname.failoncollision=true -Dopenejb.jndiname.format={ejbName}`
4. [Harmony](#) generally needs more memory. So make sure to add the corresponding `-Xms` and `-Xmx` options.

Your final startup line may look like this:

```
solid
java -Xms256M -Xmx512M -Djava.endorsed.dirs=lib/endorsed -Dopenejb.jndiname.failoncollision=true -Dopenejb.jndiname.format={ejbName} -Dhttp.proxyHost=your.proxy -Dhttp.proxyPort=8080 -jar bin/server.jar
```

Configure Deployer to use Harmony RMI Registry provider for JNDI

The [Geronimo](#) Deployer doesn't use the `var/config/config.xml` file (see [above](#)), so the [JNDI](#) configuration needs to be supplied through the system properties, like this:

```
solid
java -Djava.naming.factory.url.pkgs=org.apache.harmony.jndi.provider -jar bin/deployer.jar ...
```

Harmony configuration

Use DRLVM

You should use [DRLVM](#) ([Harmony](#) VM) to run [Geronimo](#).

[Harmony](#) also supports [IBM J9](#) VM, but [Geronimo](#) doesn't work with it, as [Geronimo](#) relies on `java.util.concurrent` package that requires a proper implementation of `sun.misc.Unsafe` class that is missing in [IBM J9](#) VM.