

```
#*****
# Wrapper License Properties (Ignored by Community Edition)
#*****
#encoding=UTF-8
wrapper.license.type=DEV
wrapper.license.id=200011010000
wrapper.license.licensee=Software AG
wrapper.license.dev_application=Terracotta Server Array
wrapper.license.features=64bit
wrapper.license.upgrade_term.begin_date=2008-07-30
wrapper.license.upgrade_term.end_date=2014-07-30
wrapper.license.key.1=63fd-fc1e-4a3e-2973
wrapper.license.key.2=78f1-d608-5a93-3c87
wrapper.license.key.3=0117-5193-b43f-4ace
wrapper.license.key.4=b598-6a47-3f98-28d5

#*****
# Wrapper Properties
#*****
# working dir, see
http://wrapper.tanukisoftware.com/doc/english/prop-working-dir.html#working-dir
wrapper.working.dir=c:\SoftwareAg\Terracotta\server\wrapper\work

# Java Application
# When the Wrapper is run as a service, environment variables will be loaded from the
system registry
# rather than from the environment. Users should set their JAVA_HOME property here
or registry.
set.JAVA_HOME=c:\SoftwareAg\jvm\jvm
wrapper.java.command=%JAVA_HOME%/bin/java

# Java Main class. This class must implement the WrapperListener interface
# or guarantee that the WrapperManager class is initialized. Helper
# classes are provided to do this for you. See the Integration section
# of the documentation for details.
wrapper.java.mainclass=org.tanukisoftware.wrapper.WrapperStartStopApp

# Java Classpath (include wrapper.jar) Add class path elements as
# needed starting from 1
wrapper.java.classpath.1=../lib/tc.jar
wrapper.java.classpath.2=../lib/wrapper.jar

# Java Library Path (location of Wrapper.DLL or libwrapper.so)
wrapper.java.library.path.1=../lib
```

```
# Java Additional Parameters
wrapper.java.additional.1=-server
wrapper.java.additional.2=-Dcom.tc.productkey.path=c:\SoftwareAg\Terracotta\Terracotta-license.key
wrapper.java.additional.3=-XX:MaxDirectMemorySize=9223372036854775807
wrapper.java.additional.4=-XX:+HeapDumpOnOutOfMemoryError
wrapper.java.additional.5=-Dcom.sun.management.jmxremote
wrapper.java.additional.6=-Dsun.rmi.dgc.server.gcInterval=3153600000
wrapper.java.additional.7=-Dtc.install-root=../..
```

```
# Initial Java Heap Size (in MB)
wrapper.java.initmemory=2048
```

```
# Maximum Java Heap Size (in MB)
wrapper.java.maxmemory=2048
```

```
# set the name of the TC server you want to start, found in conf/tc-config.xml
set.SERVER_NAME=TcServer1
```

```
# Application parameters. Add parameters as needed starting from 1
# The first application parameter is the name of the class whose main
# method is to be called when the application is launched. The class
# name is followed by the number of parameters to be passed to its main
# method. Then comes the actual parameters.
wrapper.app.parameter.1=com.tc.server.TCServerMain
wrapper.app.parameter.2=4
wrapper.app.parameter.3=-f
wrapper.app.parameter.4=c:\SoftwareAg\Terracotta\server\wrapper\conf\tc-config.xml
wrapper.app.parameter.5=-n
wrapper.app.parameter.6=%SERVER_NAME%
```

```
# The start parameters are followed by the name of the class whose main
# method is to be called to stop the application. The stop class name
# is followed by a flag which controls whether or not the Wrapper should
# wait for all non daemon threads to complete before exiting the JVM.
# The flag is followed by the number of parameters to be passed to the
# stop class's main method. Finally comes the actual parameters.
wrapper.app.parameter.7=com.tc.admin.TCStop
wrapper.app.parameter.8=false
wrapper.app.parameter.9=4
wrapper.app.parameter.10=-f
wrapper.app.parameter.11=c:\SoftwareAg\Terracotta\server\wrapper\conf\tc-config.xml
wrapper.app.parameter.12=-n
```

wrapper.app.parameter.13=%SERVER_NAME%

#####

Wrapper Logging Properties

#####

Format of output for the console. (See docs for formats)

wrapper.console.format=PM

Log Level for console output. (See docs for log levels)

wrapper.console.loglevel=INFO

Log file to use for wrapper output logging.

wrapper.logfile=./logs/wrapper-tsa.log

Format of output for the log file. (See docs for formats)

wrapper.logfile.format=LPTM

Log Level for log file output. (See docs for log levels)

wrapper.logfile.loglevel=INFO

Maximum size that the log file will be allowed to grow to before

the log is rolled. Size is specified in bytes. The default value

of 0, disables log rolling. May abbreviate with the 'k' (kb) or

'm' (mb) suffix. For example: 10m = 10 megabytes.

wrapper.logfile.maxsize=1m

Maximum number of rolled log files which will be allowed before old

files are deleted. The default value of 0 implies no limit.

wrapper.logfile.maxfiles=10

Log Level for sys/event log output. (See docs for log levels)

wrapper.syslog.loglevel=NONE

#####

Wrapper NT Service Properties

#####

WARNING - Do not modify any of these properties when an application

using this configuration file has been installed as a service.

Please uninstall the service before modifying this section. The

service can then be reinstalled.

Name of the service

wrapper.ntservice.name=TerracottaServerArray

Display name of the service

```
wrapper.ntserservice.displayname=Terracotta 4.1 Server1

# Description of the service
wrapper.ntserservice.description=Terracotta 4.1 Server1

# Service dependencies. Add dependencies as needed starting from 1
wrapper.ntserservice.dependency.1=

# Mode in which the service is installed. AUTO_START or DEMAND_START
wrapper.ntserservice.starttype=AUTO_START

# Allow the service to interact with the desktop.
wrapper.ntserservice.interactive=false

# Allows a Java thread dump to be produced when running on a Windows OS
wrapper.ntserservice.console=TRUE

# The account to use when running the service. Use an account name in the form:
#{DomainName}\{UserName}
# If the account belongs to the built-in domain, then you may specify the name in the
form: .\{UserName}
# The default, blank value, will use the LocalSystem account.
# http://wrapper.tanukisoft.com/doc/english/prop-ntserservice-account.html
wrapper.ntserservice.account=

#*****
# Wrapper Ping Service Properties
#*****
# Number of seconds to allow between the wrapper pinging the JVM and the response.
0 means never time out. Defaults to 30 seconds.
# There is a chance on systems for the JVM to hold onto 100% of the CPU for longer
than 30 seconds so we are extending
# the default timeout in order to prevent the wrapper from restarting the AE unnecessar-
ily.
wrapper.ping.timeout=300

# Number of seconds between Wrapper ping requests to the JVM. Must be
# in the range 1 to 3600 seconds. Defaults to 5 seconds.
wrapper.ping.interval=90

#*****
# Wrapper Shutdown and Restart Properties
#*****

# Configure the Wrapper to shutdown the JVM for all exit codes except the exit code
"11"
```

```
wrapper.on_exit.default=SHUTDOWN  
wrapper.on_exit.11=RESTART
```

```
# Number of seconds to allow between the time that the Wrapper asks the  
# JVM to shutdown and the time that the JVM side of the Wrapper responds  
# that it is stopping. 0 means never time out. Defaults to 30 seconds.  
# For details on this property see  
# http://wrapper.tanukisoftware.org/doc/english/prop-shutdown-timeout.html  
wrapper.shutdown.timeout=300
```

```
# Number of seconds to allow between the time that the JVM reports that it  
# is stopped and the time that the JVM process actually terminates. 0  
# means never time out. Defaults to 15 seconds.  
# http://wrapper.tanukisoftware.org/doc/english/prop-jvm-exit-timeout.html  
wrapper.jvm_exit.timeout=250
```