

Version : **2023.01**.

Dernière mise-à-jour : 2023/12/04 09:22

SER306 - Journalisation, Supervision et Clustering

Contenu du Module

- **SER306 - Journalisation, Supervision et Clustering**
 - Contenu du Module
 - Configuration des journaux
 - java.util.logging
 - log4j
 - Supervision
 - JMeter
 - Interface JMX
 - JConsole
 - Clustering avec Tomcat
 - Préparation
 - Le Cluster de Répartition de Charge avec Apache et mod_jk
 - Le Cluster de Répartition de Charge avec Apache et mod_proxy_ajp
 - Le Cluster en mode Maître/Esclave
 - Maintenir l'Etat des Clients
 - Préparation
 - Sessions Persistantes sur Système de Fichiers

Configuration des journaux

java.util.logging

Par défaut, Tomcat utilise le framework **java.util.logging** pour produire ses journaux.

Ce système de journalisation utilise le fichier **\$CATALINA_HOME/conf/logging.properties** :

```
[root@centos7 bin]# cat $CATALINA_HOME/conf/logging.properties
# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements.  See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License.  You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

handlers = 1catalina.org.apache.juli.AsyncFileHandler, 2localhost.org.apache.juli.AsyncFileHandler,
3manager.org.apache.juli.AsyncFileHandler, 4host-manager.org.apache.juli.AsyncFileHandler,
java.util.logging.ConsoleHandler

.handlers = 1catalina.org.apache.juli.AsyncFileHandler, java.util.logging.ConsoleHandler

#####
# Handler specific properties.
# Describes specific configuration info for Handlers.
#####
```

```
1catalina.org.apache.juli.AsyncFileHandler.level = FINE
1catalina.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
1catalina.org.apache.juli.AsyncFileHandler.prefix = catalina.

2localhost.org.apache.juli.AsyncFileHandler.level = FINE
2localhost.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
2localhost.org.apache.juli.AsyncFileHandler.prefix = localhost.

3manager.org.apache.juli.AsyncFileHandler.level = FINE
3manager.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
3manager.org.apache.juli.AsyncFileHandler.prefix = manager.

4host-manager.org.apache.juli.AsyncFileHandler.level = FINE
4host-manager.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
4host-manager.org.apache.juli.AsyncFileHandler.prefix = host-manager.

java.util.logging.ConsoleHandler.level = FINE
java.util.logging.ConsoleHandler.formatter = org.apache.juli.OneLineFormatter

#####
# Facility specific properties.
# Provides extra control for each logger.
#####

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].handlers =
2localhost.org.apache.juli.AsyncFileHandler

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/manager].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/manager].handlers =
3manager.org.apache.juli.AsyncFileHandler

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/host-manager].level = INFO
```

```
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/host-manager].handlers = 4host-  
manager.org.apache.juli.AsyncFileHandler
```

```
# For example, set the org.apache.catalina.util.LifecycleBase logger to log  
# each component that extends LifecycleBase changing state:  
#org.apache.catalina.util.LifecycleBase.level = FINE  
  
# To see debug messages in TldLocationsCache, uncomment the following line:  
#org.apache.jasper.compiler.TldLocationsCache.level = FINE
```

Dans ce fichier on constate la directive **handlers** :

```
...  
handlers = 1catalina.org.apache.juli.AsyncFileHandler, 2localhost.org.apache.juli.AsyncFileHandler,  
3manager.org.apache.juli.AsyncFileHandler, 4host-manager.org.apache.juli.AsyncFileHandler,  
java.util.logging.ConsoleHandler  
...
```

Il existe deux types de **handlers** :

- **org.apache.juli.AsyncFileHandler** qui écrit dans un fichier texte,
- **java.util.logging.ConsoleHandler** qui écrit sur la sortie standard.

Dans la déclaration des handlers, il faut spécifier un nom. Dans la déclaration ci-dessus, les noms sont :

- 1catalina,
- 2localhost,
- 3manager,
- 4host-manager.

La directive suivante permet de référencer le gestionnaire principal pour le serveur lui-même :

```
...  
.handlers = 1catalina.org.apache.juli.AsyncFileHandler, java.util.logging.ConsoleHandler
```

...

Chaque handler doit ensuite être configuré :

```
1catalina.org.apache.juli.AsyncFileHandler.level = FINE
1catalina.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
1catalina.org.apache.juli.AsyncFileHandler.prefix = catalina.

2localhost.org.apache.juli.AsyncFileHandler.level = FINE
2localhost.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
2localhost.org.apache.juli.AsyncFileHandler.prefix = localhost.

3manager.org.apache.juli.AsyncFileHandler.level = FINE
3manager.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
3manager.org.apache.juli.AsyncFileHandler.prefix = manager.

4host-manager.org.apache.juli.AsyncFileHandler.level = FINE
4host-manager.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
4host-manager.org.apache.juli.AsyncFileHandler.prefix = host-manager.

java.util.logging.ConsoleHandler.level = FINE
java.util.logging.ConsoleHandler.formatter = org.apache.juli.OneLineFormatter
```

Les attributs communs à la classe `org.apache.juli.AsyncFileHandler` et `java.util.logging.ConsoleHandler` sont les suivants :

Attribut	Description
level	Spécifie le niveau de journalisation. Les niveaux sont SEVERE, CONFIG, INFO, WARN, FINE, FINEST ou ALL
formatter	Spécifie la classe utilisée pour formater le journal soit par défaut java.util.logging.SimpleFormatter soit java.util.logging.XMLFormatter pour générer une sortie au format XML

Les attributs spécifiques à la classe `org.apache.juli.AsyncFileHandler` sont les suivants :

Attribut	Description
prefix	Spécifie le nom du fichier

Attribut	Description
suffix	Spécifie l'extension du fichier
directory	Spécifie le répertoire de stockage des journaux

La rotation des journaux est journalier à 00h00. Le nom du journal aura don la forme suivante : **nom.AAAA.MM.JJ**.

La section suivante du fichier fournit un niveau de contrôle supplémentaire :

```
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].handlers =
2localhost.org.apache.juli.AsyncFileHandler

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/manager].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/manager].handlers =
3manager.org.apache.juli.AsyncFileHandler

org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/host-manager].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/host-manager].handlers = 4host-
manager.org.apache.juli.AsyncFileHandler
```

Par exemple :

- [Catalina].[localhost] fait référence au hôte localhost,
- [Catalina].[localhost].[/manager] fait référence à l'application manager.

Il est à noter que les configurations spécifiques aux applications peuvent être incluses soit dans le fichier **\$CATALINA_HOME/conf/logger.properties** soit dans un fichier logger.properties qui se trouve dans le répertoire **WEB-INF/classes** de l'application concernée.



Important : Pour mettre en place le debugging, utilisez le niveau **FINEST** ou **ALL**.

log4j

Il existe une alternative au framework **java.util.logging**, appelé **log4j**.

Téléchargez **log4j-1.2.17.zip** :

```
[root@centos7 ~]# wget http://apache.mirrors.ovh.net/ftp.apache.org/dist/logging/log4j/1.2.17/log4j-1.2.17.zip
```

Décompressez le fichier zip téléchargé :

```
[root@centos7 ~]# unzip log4j-1.2.17.zip
```

Copiez ensuite le fichier **log4j-1.2.17.jar** du répertoire **/root/apache-log4j-1.2.17/log4j-1.2.17.jar** vers le répertoire **\$CATALINA_HOME/lib** en le renommant en **log4j.jar** :

```
[root@centos7 ~]# cp apache-log4j-1.2.17/log4j-1.2.17.jar $CATALINA_HOME/lib/log4j.jar
```

Téléchargez ensuite les fichiers **tomcat-juli-adapters.jar** et **tomcat-juli.jar** du dépôt **extras** d'Apache :

```
[root@centos7 ~]# wget https://archive.apache.org/dist/tomcat/tomcat-8/v8.0.36/bin/extras/tomcat-juli-adapters.jar
[root@centos7 ~]# wget https://archive.apache.org/dist/tomcat/tomcat-8/v8.0.36/bin/extras/tomcat-juli.jar
```

Copiez le fichier **tomcat-juli-adapters.jar** vers **\$CATALINA_HOME/lib/** :

```
[root@centos7 ~]# cp tomcat-juli-adapters.jar $CATALINA_HOME/lib/
```

Remplacez le fichier **\$CATALINA_HOME/bin/tomcat-juli.jar** par le fichier téléchargé du même nom **/root/tomcat-juli.jar** :

```
[root@centos7 ~]# cp tomcat-juli.jar $CATALINA_HOME/bin/
cp: overwrite '/usr/tomcat8/bin/tomcat-juli.jar'? o
```

Créez maintenant le fichier **\$CATALINA_HOME/lib/log4j.properties** qui configure log4j de façon à ce que celui-ci produise dans un premier temps les

mêmes journaux aux mêmes formats que le framework **java.util.logging** :

```
[root@centos7 ~]# vi $CATALINA_HOME/lib/log4j.properties
[root@centos7 ~]# cat $CATALINA_HOME/lib/log4j.properties
log4j.rootLogger = INFO, CATALINA

# Define all the appenders
log4j.appender.CATALINA = org.apache.log4j.DailyRollingFileAppender
log4j.appender.CATALINA.File = ${catalina.base}/logs/catalina
log4j.appender.CATALINA.Append = true
log4j.appender.CATALINA.Encoding = UTF-8
# Roll-over the log once per day
log4j.appender.CATALINA.DatePattern = '.'yyyy-MM-dd'.log'
log4j.appender.CATALINA.layout = org.apache.log4j.PatternLayout
log4j.appender.CATALINA.layout.ConversionPattern = %d [%t] %-5p %c- %m%n

log4j.appender.LOCALHOST = org.apache.log4j.DailyRollingFileAppender
log4j.appender.LOCALHOST.File = ${catalina.base}/logs/localhost
log4j.appender.LOCALHOST.Append = true
log4j.appender.LOCALHOST.Encoding = UTF-8
log4j.appender.LOCALHOST.DatePattern = '.'yyyy-MM-dd'.log'
log4j.appender.LOCALHOST.layout = org.apache.log4j.PatternLayout
log4j.appender.LOCALHOST.layout.ConversionPattern = %d [%t] %-5p %c- %m%n

log4j.appender.MANAGER = org.apache.log4j.DailyRollingFileAppender
log4j.appender.MANAGER.File = ${catalina.base}/logs/manager
log4j.appender.MANAGER.Append = true
log4j.appender.MANAGER.Encoding = UTF-8
log4j.appender.MANAGER.DatePattern = '.'yyyy-MM-dd'.log'
log4j.appender.MANAGER.layout = org.apache.log4j.PatternLayout
log4j.appender.MANAGER.layout.ConversionPattern = %d [%t] %-5p %c- %m%n

log4j.appender.HOST-MANAGER = org.apache.log4j.DailyRollingFileAppender
log4j.appender.HOST-MANAGER.File = ${catalina.base}/logs/host-manager
```

```
log4j.appender.HOST-MANAGER.Append = true
log4j.appender.HOST-MANAGER.Encoding = UTF-8
log4j.appender.HOST-MANAGER.DatePattern = '.'yyyy-MM-dd'.log'
log4j.appender.HOST-MANAGER.layout = org.apache.log4j.PatternLayout
log4j.appender.HOST-MANAGER.layout.ConversionPattern = %d [%t] %-5p %c- %m%n

log4j.appender.CONSOLE = org.apache.log4j.ConsoleAppender
log4j.appender.CONSOLE.Encoding = UTF-8
log4j.appender.CONSOLE.layout = org.apache.log4j.PatternLayout
log4j.appender.CONSOLE.layout.ConversionPattern = %d [%t] %-5p %c- %m%n

# Configure which loggers log to which appenders
log4j.logger.org.apache.catalina.core.ContainerBase.[Catalina].[localhost] = INFO, LOCALHOST
log4j.logger.org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/manager] =\
    INFO, MANAGER
log4j.logger.org.apache.catalina.core.ContainerBase.[Catalina].[localhost].[/host-manager] =\
    INFO, HOST-MANAGER
```

Déplacez le fichier **\$CATALINA_HOME/conf/logging.properties**, utilisé par le framework **java.util.logging** :

```
[root@centos7 ~]# mv $CATALINA_HOME/conf/logging.properties /root
```

Dernièrement, redémarrez le serveur Tomcat :

```
[root@centos7 bin]# ./shutdown.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
[root@centos7 bin]# ./startup.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
```

```
Using JRE_HOME:      /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:     /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
```

Pour plus d'information concernant la journalisation, consultez cette [page](#).

Supervision

JMeter

Ouvrez un terminal dans l'interface graphique de votre VM.

La gestion de la montée en charge de Tomcat peut être faite avec le produit libre **JMeter**. Pour l'obtenir, il convient de le télécharger :

```
[root@centos7 ~]# wget https://archive.apache.org/dist/jmeter/binaries/apache-jmeter-3.0.tgz
```

Décompressez l'archive dans **\$CATALINA_HOME/JMeter** :

```
[root@centos7 ~]# mkdir $CATALINA_HOME/JMeter
[root@centos7 ~]# mv apache-jmeter-3.0.tgz $CATALINA_HOME/JMeter
[root@centos7 ~]# cd $CATALINA_HOME/JMeter
[root@centos7 JMeter]# tar xvf apache-jmeter-3.0.tgz
```

L'arborescence obtenu est :

```
[root@centos7 JMeter]# ls
apache-jmeter-3.0  apache-jmeter-3.0.tgz
[root@centos7 JMeter]# cd apache-jmeter-3.0/
[root@centos7 apache-jmeter-3.0]# ls
bin  docs  extras  lib  LICENSE  licenses  NOTICE  printable_docs  README
```

Définissez maintenant la variable **\$JMeter_HOME** :

```
[root@centos7 apache-jmeter-3.0]# JMeter_HOME=/usr/tomcat8/JMeter/apache-jmeter-3.0
[root@centos7 apache-jmeter-3.0]# export JMeter_HOME
[root@centos7 apache-jmeter-3.0]# echo $JMeter_HOME
/usr/tomcat8/JMeter/apache-jmeter-3.0
```

Lancez ensuite JMeter dans un terminal de votre VM en mode graphique :

```
[root@redhat JMeter]# /usr/tomcat8/JMeter/apache-jmeter-3.0/bin/jmeter
```

Testez cet outil en utilisant les différents fichiers mis-à-disposition lors de l'installation de JMeter :

```
[root@centos7 apache-jmeter-3.0]# updatedb
[root@centos7 apache-jmeter-3.0]# locate .jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/examples/CSVSample.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/examples/PerformanceTestPlanMemoryThread.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/BeanShellSampler.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/build-adv-web-test-plan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/build-ftp-test-plan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/build-ldap-ext-test-plan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/build-ldap-test-plan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/build-web-test-plan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/build-webservice-test-plan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/jdbc.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/mongodb.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/recording-with-think-time.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/bin/templates/recording.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/extras/Test.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/AssertionTestPlan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/AuthManagerTestPlan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/ForEachTest2.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/HeaderManagerTestPlan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/InterleaveTestPlan.jmx
```

```
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/InterleaveTestPlan2.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/JDBC-Pre-Post-Processor.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/JMSPointToPoint.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/LoopTestPlan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/OnceOnlyTestPlan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/ProxyServerTestPlan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/RegEx-User-Parameters.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/SimpleTestPlan.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/URLRewritingExample.jmx
/usr/tomcat8/JMeter/apache-jmeter-3.0/printable_docs/demos/forEachTestPlan.jmx
```



Important : Pour plus d'information concernant cet outil, consultez la page <http://jmeter.apache.org/changes.html>.

Interface JMX

L'interface JMX est un outil complémentaire à JMeter car il est capable de montrer :

- le comportement interne du serveur,
- le nombre de connexions JDBC disponibles à un instant t,
- le nombre de threads occupés dans un connecteur.

JMX ou *Java Management eXtensions* est un API Java qui extrait des informations des **MBeans**. Les MBeans sont créés dans le fichier `$CATALINA_HOME/conf/server.xml` en utilisant des éléments `<Listener>` :

```
...
<Listener className="org.apache.catalina.startup.VersionLoggerListener" />
<Listener className="org.apache.catalina.security.SecurityListener" />
<Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />
<Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener" />
<Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener" />
```

```
<Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener" />
...
```

Revenez à l'authentification en utilisant le fichier **tomcat-users.xml** en éditant le fichier **\$CATALINA_HOME/conf/server.xml** :

```
[root@centos7 bin]# vi $CATALINA_HOME/conf/server.xml
[root@centos7 bin]# cat $CATALINA_HOME/conf/server.xml
...
    <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
        resourceName="UserDatabase" digest="sha" />
...
    <!-- <Realm className="org.apache.catalina.realm.JNDIRealm"
        connectionURL="ldap://localhost:389"
        connectionName="cn=Manager,o=fenestros.loc"
        connectionPassword="fenestros"
        roleBase="ou=roles,o=fenestros.loc"
        roleName="cn"
        roleSearch="(uniqueMember={0})"
        userPassword="userPassword"
        userPattern="cn={0},ou=utilisateurs,o=fenestros.loc" /> -->
...
```

Éditez ensuite le fichier **\$CATALINA_HOME/conf/tomcat-users.xml** :

```
[root@centos7 bin]# vi $CATALINA_HOME/conf/tomcat-users.xml
[root@centos7 bin]# cat $CATALINA_HOME/conf/tomcat-users.xml
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users xmlns="http://tomcat.apache.org/xml"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
    version="1.0">
    <role rolename="tomcat"/>
    <role rolename="role1"/>
    <role rolename="manager-script"/>
```

```
<role rolename="manager-gui"/>
<role rolename="manager-jmx"/>
<user username="tomcat" password="tomcat" roles="tomcat"/>
<user username="both" password="tomcat" roles="tomcat,role1"/>
<user username="role1" password="tomcat" roles="role1"/>
<user username="admin"
password="75affcc6adf7ec7cd21f6479b8fb87b89a550b2602e613715d57d862b15c7c17$1$015b4b320315c87572918dbcc08b65a240d0
a750" roles="manager-script,manager-gui,manager-jmx"/>
</tomcat-users>
```

Redémarrez le serveur Tomcat :

```
[root@centos7 apache-jmeter-3.0]# cd $CATALINA_HOME/bin
[root@centos7 bin]# ./shutdown.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:        /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
[root@centos7 bin]# ./startup.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:        /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
```

Testez que l'authentification fonctionne correctement :

```
[root@centos7 bin]# lynx --dump -auth admin:fenestros "http://www.i2tch.loc:8080/manager/text/serverinfo"
OK - Server info
Tomcat Version: Apache Tomcat/8.0.36
OS Name: Linux
OS Version: 3.10.0-1062.4.1.el7.x86_64
```

```
OS Architecture: amd64
JVM Version: 1.8.0_232-b09
JVM Vendor: Oracle Corporation
```

L'application **manager** propose un client JMX sous la forme d'un proxy JMX, **jmxproxy**.

Saisissez donc la commande suivante :

```
[root@centos7 bin]# lynx --dump -auth admin:fenestros "http://www.i2tch.loc:8080/manager/jmxproxy/?qry=*:*" |
more
OK - Number of results: 174

Name: Catalina:type=Service
modelerType: org.apache.catalina.mbeans.ServiceMBean
stateName: STARTED
connectorNames: Array[javax.management.ObjectName] of length 3
    Catalina:type=Connector,port=8080
    Catalina:type=Connector,port=8443
    Catalina:type=Connector,port=8009
name: Catalina
managedResource: StandardService[Catalina]

Name: Catalina:j2eeType=Servlet,WebModule=//localhost/examples,name=stock,J2EEAp
plication=none,J2EESever=none
modelerType: org.apache.catalina.mbeans.ContainerMBean
maxTime: 0
requestCount: 0
stateManageable: false
servletClass: async.AsyncStockServlet
countAllocated: 0
available: 0
backgroundProcessorDelay: -1
processingTime: 0
loadOnStartup: -1
```

```
singleThreadModel: false
loadTime: 2
stateName: STARTED
minTime: 9223372036854775807
classLoadTime: 2
--More--
```

Saisissez maintenant la commande suivante :

```
[root@centos7 bin]# lynx --dump -auth admin:fenestros
"http://www.i2tch.loc:8080/manager/jmxproxy/?qry=Catalina:type=Connector,*" | more
OK - Number of results: 3

Name: Catalina:type=Connector,port=8009
modelerType: org.apache.catalina.mbeans.ConnectorMBean
maxPostSize: 2097152
scheme: http
acceptCount: 100
secure: false
threadPriority: 5
ajpFlush: true
maxSavePostSize: 4096
proxyPort: 0
protocol: AJP/1.3
maxParameterCount: 10000
useIPVHosts: false
stateName: STARTED
redirectPort: 8443
allowTrace: false
protocolHandlerClassName: org.apache.coyote.ajp.AjpNioProtocol
maxThreads: 200
connectionTimeout: -1
tcpNoDelay: true
useBodyEncodingForURI: false
```

```
connectionLinger: -1
processorCache: 200
keepAliveTimeout: -1
localPort: 8009
enableLookups: false
packetSize: 8192
--More--
```

Saisissez maintenant la commande suivante :

```
[root@centos7 bin]# lynx --dump -auth admin:fenestros
"http://www.i2tch.loc:8080/manager/jmxproxy/?qry=Catalina:type=ThreadPool,*"
OK - Number of results: 3

Name: Catalina:type=ThreadPool,name="http-nio-8443"
modelerType: org.apache.tomcat.util.modeler.BaseModelMBean
currentThreadsBusy: 0
selectorTimeout: 1000
paused: false
socketProperties: org.apache.tomcat.util.net.SocketProperties@346f7a2a
useCometTimeout: true
currentThreadCount: 0
usePolling: true
maxThreads: 200
tcpNoDelay: true
algorithm: SunX509
maxKeepAliveRequests: 100
keepAliveTimeout: 60000
localPort: 8443
pollerThreadCount: 1
acceptorThreadCount: 1
soTimeout: 60000
daemon: true
minSpareThreads: 10
```

```
acceptorThreadPriority: 5
backlog: 100
maxHeaderCount: 100
port: 8443
keystoreType: JKS
name: http-nio-8443
soLinger: -1
sslProtocol: TLS
sessionTimeout: 86400
useComet: true
clientAuth: false
connectionCount: 1
threadPriority: 5
executorTerminationTimeoutMillis: 5000
running: true
sslEnabledProtocolsArray: Array[java.lang.String] of length 0
ciphers: HIGH:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!kRSA
sSLEnabled: false
selectorPool: org.apache.tomcat.util.net.NioSelectorPool@8fecc2b
maxConnections: 10000
keepAliveCount: 0
deferAccept: false
useSendfile: true
oomParachute: 1048576
bindOnInit: true
pollerThreadPriority: 5
keystoreFile: /root/.keystore
useServerCipherSuitesOrder:

Name: Catalina:type=ThreadPool,name="ajp-nio-8009"
modelerType: org.apache.tomcat.util.modeler.BaseModelMBean
currentThreadsBusy: 0
selectorTimeout: 1000
paused: false
```

```
socketProperties: org.apache.tomcat.util.net.SocketProperties@50b03315
useCometTimeout: true
currentThreadCount: 0
usePolling: true
maxThreads: 200
tcpNoDelay: true
algorithm: SunX509
maxKeepAliveRequests: 100
keepAliveTimeout: -1
localPort: 8009
pollerThreadCount: 1
acceptorThreadCount: 1
soTimeout: -1
daemon: true
minSpareThreads: 10
acceptorThreadPriority: 5
backlog: 100
maxHeaderCount: 100
port: 8009
keystoreType: JKS
name: ajp-nio-8009
soLinger: -1
sslProtocol: TLS
sessionTimeout: 86400
useComet: true
clientAuth: false
connectionCount: 1
threadPriority: 5
executorTerminationTimeoutMillis: 5000
running: true
sslEnabledProtocolsArray: Array[java.lang.String] of length 0
ciphers: HIGH:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!kRSA
sSLEnabled: false
selectorPool: org.apache.tomcat.util.net.NioSelectorPool@3b104969
```

```
maxConnections: 10000
keepAliveCount: 0
deferAccept: false
useSendfile: false
oomParachute: 1048576
bindOnInit: true
pollerThreadPriority: 5
keystoreFile: /root/.keystore
useServerCipherSuitesOrder:

Name: Catalina:type=ThreadPool,name="http-nio-8080"
modelerType: org.apache.tomcat.util.modeler.BaseModelMBean
currentThreadsBusy: 2
selectorTimeout: 1000
paused: false
socketProperties: org.apache.tomcat.util.net.SocketProperties@182321ae
useCometTimeout: true
currentThreadCount: 8
usePolling: true
maxThreads: 200
tcpNoDelay: true
algorithm: SunX509
maxKeepAliveRequests: 100
keepAliveTimeout: 20000
localPort: 8080
pollerThreadCount: 1
acceptorThreadCount: 1
soTimeout: 20000
daemon: true
minSpareThreads: 10
acceptorThreadPriority: 5
backlog: 100
maxHeaderCount: 100
port: 8080
```

```
keystoreType: JKS
name: http-nio-8080
soLinger: -1
sslProtocol: TLS
sessionTimeout: 86400
useComet: true
clientAuth: false
connectionCount: 3
threadPriority: 5
executorTerminationTimeoutMillis: 5000
running: true
sslEnabledProtocolsArray: Array[java.lang.String] of length 0
ciphers: HIGH:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!kRSA
sSLEnabled: false
selectorPool: org.apache.tomcat.util.net.NioSelectorPool@596e32c2
maxConnections: 10000
keepAliveCount: 1
deferAccept: false
useSendfile: true
oomParachute: 1048576
bindOnInit: true
pollerThreadPriority: 5
keystoreFile: /root/.keystore
useServerCipherSuitesOrder:
```

Notez la valeur de **maxThreads** dans la section **name="http-nio-8080"** :

```
...
maxThreads: 200
...
```

Pour modifier la valeur de maxThreads, il faut créer le fichier \$CATALINA_HOME/bin/setenv.sh :

```
[root@centos7 bin]# vi setenv.sh
```

```
[root@centos7 bin]# cat setenv.sh
export JAVA_OPTS="-Dcom.sun.management.jmxremote=true -Dcom.sun.management.jmxremote.ssl=false -
Dcom.sun.management.jmxremote.authenticate=false"
[root@centos7 bin]# chmod ugo+x setenv.sh
[root@centos7 bin]# ls -l setenv.sh
-rwxr-xr-x 1 root root 208 Jun 28 23:40 setenv.sh
```

Redémarrez le serveur Tomcat pour une prise en compte du fichier setenv.sh :

```
[root@centos7 bin]# ./shutdown.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
[root@centos7 bin]# ./startup.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
```

Maintenant saisissez l'URL suivant dans un navigateur en mode graphique pour modifier la valeur de MaxThreads :

```
http://www.i2tch.loc:8080/manager/jmxproxy/?set=Catalina:type=ThreadPool,name="http-nio-8080"&att=maxThreads&val=300
```



Important : Le navigateur vous demandera de renseigner un utilisateur et un mot de passe : admin/fenestros.

Vous obtiendrez un résultat similaire à celui-ci :

OK - Attribute set

Saisissez la commande suivante pour vérifier la prise en compte de la modification :

```
[root@centos7 bin]# lynx --dump -auth admin:fenestros
"http://www.i2tch.loc:8080/manager/jmxproxy/?qry=Catalina:type=ThreadPool,*"
...
Name: Catalina:type=ThreadPool,name="http-nio-8080"
modelerType: org.apache.tomcat.util.modeler.BaseModelMBean
currentThreadsBusy: 1
selectorTimeout: 1000
paused: false
socketProperties: org.apache.tomcat.util.net.SocketProperties@37cfc160
useCometTimeout: true
currentThreadCount: 10
usePolling: true
maxThreads: 300
tcpNoDelay: true
algorithm: SunX509
maxKeepAliveRequests: 100
keepAliveTimeout: 20000
localPort: 8080
pollerThreadCount: 1
acceptorThreadCount: 1
soTimeout: 20000
daemon: true
minSpareThreads: 10
acceptorThreadPriority: 5
backlog: 100
maxHeaderCount: 100
port: 8080
keystoreType: JKS
name: http-nio-8080
soLinger: -1
```

```
sslProtocol: TLS
sessionTimeout: 86400
useComet: true
clientAuth: false
connectionCount: 2
threadPriority: 5
executorTerminationTimeoutMillis: 5000
running: true
sslEnabledProtocolsArray: Array[java.lang.String] of length 0
ciphers: HIGH:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!kRSA
sSLEnabled: false
selectorPool: org.apache.tomcat.util.net.NioSelectorPool@65151909
maxConnections: 10000
keepAliveCount: 1
deferAccept: false
useSendfile: true
oomParachute: 1048576
bindOnInit: true
pollerThreadPriority: 5
keystoreFile: /root/.keystore
useServerCipherSuitesOrder:
```

JConsole

Pour utiliser JConsole, commencez par créer le fichier des utilisateurs et des mots de passe :

```
[root@centos7 bin]# cd ../conf
[root@centos7 conf]# vi jmxremote.access
[root@centos7 conf]# cat jmxremote.access
administrator    readwrite
operator         readonly
[root@centos7 conf]# vi jmxremote.password
[root@centos7 conf]# cat jmxremote.password
```

```
administrator fenestros
operator tomcat
[root@centos7 conf]# chmod 600 jmxremote.password
```

Dans un premier temps vous allez mettre en place une connexion anonyme à JConsole. Éditez donc votre fichier **\$CATALINA_HOME/bin/setenv.sh** :

```
[root@centos7 conf]# cd ../bin
[root@centos7 bin]# vi setenv.sh
[root@centos7 bin]# cat setenv.sh
export JAVA_OPTS="-Dcom.sun.management.jmxremote=true -Dcom.sun.management.jmxremote.ssl=false -
Dcom.sun.management.jmxremote.authenticate=false -Dcom.sun.management.jmxremote.port=9004"
```

Redémarrez maintenant le serveur Tomcat :

```
[root@centos7 bin]# ./shutdown.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:        /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
[root@centos7 bin]# ./startup.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:        /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
```

Lancez la commande **jconsole** dans un terminal de l'interface graphique de votre VM. Cochez **Remote Process** et utilisez l'adresse **localhost:9004** **sans** stipuler un utilisateur et un mot de passe.

Pour mettre en place une autorisation en utilisant les fichiers **jmxremote.access** et **jmxremote.password**, il convient d'éditer de nouveau le fichier **\$CATALINA_HOME/bin/setenv.sh** :

```
[root@centos7 bin]# vi setenv.sh
[root@centos7 bin]# cat setenv.sh
export JAVA_OPTS="-Dcom.sun.management.jmxremote=true -Dcom.sun.management.jmxremote.ssl=false -
Dcom.sun.management.jmxremote.port=9004 -
Dcom.sun.management.jmxremote.password.file=$CATALINA_HOME/conf/jmxremote.password -
Dcom.sun.management.jmxremote.access.file=$CATALINA_HOME/conf/jmxremote.access"
```

Redémarrez le serveur Tomcat :

```
[root@centos7 bin]# ./shutdown.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
[root@centos7 bin]# ./startup.sh
Using CATALINA_BASE:   /usr/tomcat8
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
```

Lancez la commande **jconsole** dans un terminal de l'interface graphique de votre VM. Cochez **Remote Process** et utilisez l'adresse **localhost:9004** en stipulant un utilisateur et un mot de passe dans les fichiers **jmxremote.access** et **jmxremote.password** respectivement.

Clustering avec Tomcat

Préparation

Créez maintenant deux répertoires en dessous de \$CATALINA_HOME :

```
[root@centos7 ~]# mkdir $CATALINA_HOME/tomcat1 $CATALINA_HOME/tomcat2
```

Arrêtez le serveur Tomcat et copiez les répertoires \$CATALINA_HOME/conf, \$CATALINA_HOME/logs, \$CATALINA_HOME/temp, \$CATALINA_HOME/webapps, \$CATALINA_HOME/work dans les répertoires \$CATALINA_HOME/tomcat1 et \$CATALINA_HOME/tomcat2 :

```
[root@centos7 ~]# cd $CATALINA_HOME
[root@centos7 tomcat8]# cp -rp conf/ tomcat1/
[root@centos7 tomcat8]# cp -rp logs/ tomcat1
[root@centos7 tomcat8]# cp -rp temp/ tomcat1
[root@centos7 tomcat8]# cp -rp webapps/ tomcat1
[root@centos7 tomcat8]# cp -rp work/ tomcat1
[root@centos7 tomcat8]# cp -rp conf/ tomcat2/
[root@centos7 tomcat8]# cp -rp logs/ tomcat2/
[root@centos7 tomcat8]# cp -rp temp/ tomcat2/
[root@centos7 tomcat8]# cp -rp webapps/ tomcat2/
[root@centos7 tomcat8]# cp -rp work/ tomcat2/
```

Supprimez les répertoires \$CATALINA_HOME/conf, \$CATALINA_HOME/logs, \$CATALINA_HOME/temp, \$CATALINA_HOME/webapps, \$CATALINA_HOME/work :

```
[root@centos7 tomcat8]# rm -rf conf/ logs/ temp/ webapps/ work/
```

Supprimez maintenant le fichier **\$CATALINA_HOME/bin/setenv.sh** :

```
[root@centos7 tomcat8]# rm -rf bin/setenv.sh
```

Créez maintenant les scripts de démarrage et d'arrêt de chaque instance de Tomcat :

```
[root@centos7 tomcat8]# cd bin
[root@centos7 bin]# vi startTomcat1
[root@centos7 bin]# cat startTomcat1
#!/bin/bash
export CATALINA_BASE=/usr/tomcat8/tomcat1
```

```
. $CATALINA_HOME/bin/startup.sh

[root@centos7 bin]# vi stopTomcat1
[root@centos7 bin]# cat stopTomcat1
#!/bin/bash
export CATALINA_BASE=/usr/tomcat8/tomcat1
. $CATALINA_HOME/bin/shutdown.sh

[root@centos7 bin]# vi startTomcat2
[root@centos7 bin]# cat startTomcat2
export CATALINA_BASE=/usr/tomcat8/tomcat2
. $CATALINA_HOME/bin/startup.sh

[root@centos7 bin]# vi stopTomcat2
[root@centos7 bin]# cat stopTomcat2
#!/bin/bash
export CATALINA_BASE=/usr/tomcat8/tomcat2
. $CATALINA_HOME/bin/shutdown.sh
```

Rendez les scripts exécutables :

```
[root@centos7 bin]# chmod a+x startTomcat1
[root@centos7 bin]# chmod a+x startTomcat2
[root@centos7 bin]# chmod a+x stopTomcat1
[root@centos7 bin]# chmod a+x stopTomcat2
[root@centos7 bin]# ls -l | grep startT
-rwxr-xr-x 1 root root 86 Jun 29 01:47 startTomcat1
-rwxr-xr-x 1 root root 86 Jun 29 01:51 startTomcat2
[root@centos7 bin]# ls -l | grep stopT
-rwxr-xr-x 1 root root 87 Jun 29 01:50 stopTomcat1
-rwxr-xr-x 1 root root 87 Jun 29 01:52 stopTomcat2
```

Modifiez les ports dans le fichier server.xml de chaque installation de Tomcat en utilisant VI :

```
[root@centos7 bin]# vi /usr/tomcat8/tomcat1/conf/server.xml
[root@centos7 bin]# vi /usr/tomcat8/tomcat2/conf/server.xml
```

Les commandes VI suivantes peuvent vous aider :

```
Pour le fichier /usr/tomcat8/tomcat1/conf/server.xml :
:g/8080/s//8180/g
:g/8009/s//8109/g
:g/8005/s//8105/g
:g/8443/s//8143/g
Pour le fichier /usr/tomcat8/tomcat2/conf/server.xml :
:g/8080/s//8280/g
:g/8009/s//8209/g
:g/8005/s//8205/g
:g/8443/s//8243/g
```

Démarrez les deux instances de Tomcat :

```
[root@centos7 bin]# ./startTomcat1
Using CATALINA_BASE:   /usr/tomcat8/tomcat1
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat1/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
[root@centos7 bin]# ps aux | grep tomcat
root      25696 30.0  4.4 2399312 67900 pts/0    Sl   02:47   0:04 /usr/lib/jvm/jre-1.8.0-
openjdk-1.8.0.232.b09-0.el7_7.x86_64/bin/java -
Djava.util.logging.config.file=/usr/tomcat8/tomcat1/conf/logging.properties -
Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djdk.tls.ephemeralDHKeySize=2048 -
Djava.endorsed.dirs=/usr/tomcat8/endorsed -classpath /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-
juli.jar -Dcatalina.base=/usr/tomcat8/tomcat1 -Dcatalina.home=/usr/tomcat8 -
Djava.io.tmpdir=/usr/tomcat8/tomcat1/temp org.apache.catalina.startup.Bootstrap start
root      25785  0.0  0.0 112644   968 pts/0    R+   02:47   0:00 grep --color=auto tomcat
```

```
[root@centos7 bin]# ./startTomcat2
Using CATALINA_BASE:   /usr/tomcat8/tomcat2
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat2/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
[root@centos7 bin]# ps aux | grep tomcat
root      25696 32.1  5.2 2403492 80468 pts/0    Sl   02:47   0:07 /usr/lib/jvm/jre-1.8.0-
openjdk-1.8.0.232.b09-0.el7_7.x86_64/bin/java -
Djava.util.logging.config.file=/usr/tomcat8/tomcat1/conf/logging.properties -
Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djdk.tls.ephemeralDHKeySize=2048 -
Djava.endorsed.dirs=/usr/tomcat8/endorsed -classpath /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-
juli.jar -Dcatalina.base=/usr/tomcat8/tomcat1 -Dcatalina.home=/usr/tomcat8 -
Djava.io.tmpdir=/usr/tomcat8/tomcat1/temp org.apache.catalina.startup.Bootstrap start
root      25817 32.6  2.4 2381580 37172 pts/0    Sl   02:47   0:00 /usr/lib/jvm/jre-1.8.0-
openjdk-1.8.0.232.b09-0.el7_7.x86_64/bin/java -
Djava.util.logging.config.file=/usr/tomcat8/tomcat2/conf/logging.properties -
Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djdk.tls.ephemeralDHKeySize=2048 -
Djava.endorsed.dirs=/usr/tomcat8/endorsed -classpath /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-
juli.jar -Dcatalina.base=/usr/tomcat8/tomcat2 -Dcatalina.home=/usr/tomcat8 -
Djava.io.tmpdir=/usr/tomcat8/tomcat2/temp org.apache.catalina.startup.Bootstrap start
root      25843  0.0  0.0 112644   968 pts/0     S+   02:47   0:00 grep --color=auto tomcat
```

Vérifiez maintenant que les deux instances peuvent être arrêtés :

```
[root@centos7 bin]# ./stopTomcat2
Using CATALINA_BASE:   /usr/tomcat8/tomcat2
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat2/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
[root@centos7 bin]# ./stopTomcat1
Using CATALINA_BASE:   /usr/tomcat8/tomcat1
```

```
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat1/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
[root@centos7 bin]# ps aux | grep tomcat
root      27318  0.0  0.0 112644   964 pts/0    R+   02:52   0:00 grep --color=auto tomcat
```

Le Cluster de Répartition de Charge avec Apache et mod_jk

Modifiez le fichier **/etc/httpd/conf/workers.properties** :

```
[root@centos7 bin]# vi /etc/httpd/conf/workers.properties
[root@centos7 bin]# cat /etc/httpd/conf/workers.properties
worker.list=balancer

worker.tomcat1.type=ajp13
worker.tomcat1.host=10.0.3.51
worker.tomcat1.port=8109
worker.tomcat1.lbfactor=1

worker.tomcat2.type=ajp13
worker.tomcat2.host=10.0.3.51
worker.tomcat2.port=8209
worker.tomcat2.lbfactor=1

worker.balancer.type=lb
worker.balancer.balance_workers=tomcat1,tomcat2
worker.balancer.sticky_session=1
```

Modifiez la section concernant Tomcat dans le fichier **/etc/httpd/conf/httpd.conf** et commentez la ligne **IncludeOptional conf.d/*.conf** :

```
[root@centos7 bin]# vi /etc/httpd/conf/httpd.conf
[root@centos7 bin]# tail /etc/httpd/conf/httpd.conf
```

```
#
# Load config files in the "/etc/httpd/conf.d" directory, if any.
# IncludeOptional conf.d/*.conf

LoadModule jk_module      modules/mod_jk.so
JkWorkersFile  conf/workers.properties
JkLogFile     logs/mod_jk.log
JkLogLevel    info
JkMount       /docs/*      balancer
JkMount       /docs        balancer
```

Modifiez la section <Engine> du fichier **\$CATALINA_HOME/tomcat1/conf/server.xml** :

```
[root@centos7 bin]# vi $CATALINA_HOME/tomcat1/conf/server.xml
...
    <Engine name="Catalina" defaultHost="localhost" jvmRoute="tomcat1">
...

```

Modifiez ensuite la section <Engine> du fichier **\$CATALINA_HOME/tomcat2/conf/server.xml** :

```
[root@centos7 bin]# vi $CATALINA_HOME/tomcat2/conf/server.xml
...
    <Engine name="Catalina" defaultHost="localhost" jvmRoute="tomcat2">
...

```

Pour pouvoir tester la configuration, remplacer les fichiers index.html de chaque application **docs** afin de pouvoir identifier quelle instance répond à des requêtes :

```
[root@centos7 bin]# mv $CATALINA_HOME/tomcat1/webapps/docs/index.html
$CATALINA_HOME/tomcat1/webapps/docs/index.old
[root@centos7 bin]# vi $CATALINA_HOME/tomcat1/webapps/docs/index.html
[root@centos7 bin]# cat $CATALINA_HOME/tomcat1/webapps/docs/index.html
<html>
<title>Tomcat1</title>
```

```
<body>
<center>This is Tomcat1</center>
</body>
</html>
[root@centos7 bin]# mv $CATALINA_HOME/tomcat2/webapps/docs/index.html
$CATALINA_HOME/tomcat2/webapps/docs/index.old
[root@centos7 bin]# vi $CATALINA_HOME/tomcat2/webapps/docs/index.html
[root@centos7 bin]# cat $CATALINA_HOME/tomcat2/webapps/docs/index.html
<html>
<title>Tomcat2</title>
<body>
<center>This is Tomcat2</center>
</body>
</html>
[root@centos7 bin]#
```

Redémarrez le service httpd.service :

```
[root@centos7 bin]# systemctl restart httpd.service
```

Démarrez les deux instances de Tomcat :

```
[root@centos7 bin]# ./startTomcat1
Using CATALINA_BASE:   /usr/tomcat8/tomcat1
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat1/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
[root@centos7 bin]# ./startTomcat2
Using CATALINA_BASE:   /usr/tomcat8/tomcat2
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat2/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
```

```
Using CLASSPATH:      /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started.
```

Utilisez Lynx pour vous connecter à l'application **docs** :

```
[root@centos7 httpd]# lynx --dump http://www.i2tch.loc/docs
                This is Tomcat2
```

```
[root@centos7 httpd]# lynx --dump http://www.i2tch.loc/docs
                This is Tomcat2
```

```
[root@centos7 httpd]# lynx --dump http://www.i2tch.loc/docs
                This is Tomcat2
```



Attention : Notez que l'affinité de session est activée par défaut par le module AJP.

Arrêtez maintenant l'instance **tomcat2** :

```
[root@centos7 bin]# ./stopTomcat2
Using CATALINA_BASE:   /usr/tomcat8/tomcat2
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat2/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
```

Connectez-vous de nouveau à l'application **docs** :

```
[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
                This is Tomcat1
```

```
[root@centos7 bin]#
```



Important - Notez que c'est maintenant l'instance tomcat1 qui répond.

Le Cluster de Répartition de Charge avec Apache et mod_proxy_ajp

Vérifiez que les lignes **LoadModule proxy_ajp_module modules/mod_proxy_ajp.so**, **LoadModule proxy_balancer_module modules/mod_proxy_balancer.so** et **LoadModule proxy_module modules/mod_proxy.so** soient présentes dans le fichier **/etc/httpd/conf.modules.d/00-proxy.conf** :

```
[root@centos7 bin]# cat /etc/httpd/conf.modules.d/00-proxy.conf
# This file configures all the proxy modules:
LoadModule proxy_module modules/mod_proxy.so
LoadModule lbmethod_bybusyness_module modules/mod_lbmethod_bybusyness.so
LoadModule lbmethod_byrequests_module modules/mod_lbmethod_byrequests.so
LoadModule lbmethod_bytraffic_module modules/mod_lbmethod_bytraffic.so
LoadModule lbmethod_heartbeat_module modules/mod_lbmethod_heartbeat.so
LoadModule proxy_ajp_module modules/mod_proxy_ajp.so
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_express_module modules/mod_proxy_express.so
LoadModule proxy_fcgi_module modules/mod_proxy_fcgi.so
LoadModule proxy_fdpass_module modules/mod_proxy_fdpass.so
LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule proxy_scgi_module modules/mod_proxy_scgi.so
LoadModule proxy_wstunnel_module modules/mod_proxy_wstunnel.so
```

Modifiez le fichier **/etc/httpd/conf/httpd.conf** :

```
[root@centos7 bin]# tail -n 15 /etc/httpd/conf/httpd.conf

#LoadModule jk_module      modules/mod_jk.so
#JkWorkersFile  conf/workers.properties
#JkLogFile logs/mod_jk.log
#JkLogLevel info
#JkMount        /docs/*      balancer
#JkMount        /docs        balancer

ProxyTimeout 300

<Proxy balancer://tomcat8-docs>
    BalancerMember ajp://localhost:8109/docs route=tomcat1
    BalancerMember ajp://localhost:8209/docs route=tomcat2
</Proxy>

ProxyPass        /docs      balancer://tomcat8-docs
ProxyPassReverse  /docs      balancer://tomcat8-docs
```

Redémarrez le serveur httpd :

```
[root@centos7 bin]# systemctl restart httpd.service
```

Démarrez l'instance tomcat2 de Tomcat :

```
[root@centos7 bin]# ./startTomcat2
Using CATALINA_BASE:   /usr/tomcat8/tomcat2
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat2/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
Tomcat started
```

Utilisez Lynx pour vous connecter à l'application **docs** :

```
[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat1

[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat2

[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat1

[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat2
```



Attention : Notez que l'affinité de session n'est pas activée par défaut par le module proxy.

Afin de mettre en place l'affinité de session, il convient d'utiliser un cookie appelé **ROUTEID**.

Modifiez le fichier **/etc/httpd/conf/httpd.conf** ainsi :

```
...
#IncludeOptional conf.d/*.conf

#LoadModule      jk_module      modules/mod_jk.so
#JkWorkersFile   conf/workers.properties
#JkLogFile        logs/mod_jk.log
#JkLogLevel       info
```

```
#JkMount          /docs/* balancer
#JkMount          /docs  balancer
Header add Set-Cookie "ROUTEID=.%{BALANCER_WORKER_ROUTE}e; path=/" env=BALANCER_ROUTE_CHANGED
<Proxy balancer://tomcat8-docs>
    BalancerMember ajp://localhost:8109/docs route=tomcat1
    BalancerMember ajp://localhost:8209/docs route=tomcat2
    ProxySet stickysession=ROUTEID
</Proxy>

ProxyPass          /docs  balancer://tomcat8-docs
ProxyPassReverse   /docs  balancer://tomcat8-docs
```

Testez ensuite l'affinité de session en utilisant un navigateur graphique.

Pour plus d'information concernant l'utilisation de mod_proxy, consultez [cette page](#)

Le Cluster en mode Maître/Esclave

La configuration en mode **Maître/Esclave** utilise le module **mod_jk**. Editez donc votre fichier **/etc/httpd/conf/httpd.conf** :

```
[root@centos7 bin]# vi /etc/httpd/conf/httpd.conf
[root@centos7 bin]# tail -n 20 /etc/httpd/conf/httpd.conf
EnableSendfile on

# Supplemental configuration
#
# Load config files in the "/etc/httpd/conf.d" directory, if any.
# IncludeOptional conf.d/*.conf

LoadModule jk_module      modules/mod_jk.so
JkWorkersFile  conf/workers.properties
JkLogFile     logs/mod_jk.log
JkLogLevel    info
```

```
JkMount    /docs/*    balancer
JkMount    /docs      balancer

#<Proxy balancer://tomcat8-docs>
#  BalancerMember ajp://localhost:8109/docs route=tomcat1
#  BalancerMember ajp://localhost:8209/docs route=tomcat2
#</Proxy>

#ProxyPass      /docs      balancer://tomcat8-docs
#ProxyPassReverse /docs      balancer://tomcat8-docs
```

Éditez ensuite le fichier **/etc/httpd/conf/workers.properties** :

```
[root@centos7 bin]# vi /etc/httpd/conf/workers.properties
[root@centos7 bin]# cat /etc/httpd/conf/workers.properties
worker.list=tomcat1,tomcat2,balancer

worker.tomcat1.type=ajp13
worker.tomcat1.host=10.0.3.51
worker.tomcat1.port=8109
# Indique que tomcat2 doit prendre le relais en cas de défaillance de tomcat1
worker.tomcat1.redirect=tomcat2

worker.tomcat2.type=ajp13
worker.tomcat2.host=10.0.3.51
worker.tomcat2.port=8209
# Indique que l'instance tomcat2 est un esclave
worker.tomcat2.activation=disabled

worker.balancer.type=lb
worker.balancer.balance_workers=tomcat1,tomcat2
```

Redémarrez le serveur httpd :

```
[root@centos7 bin]# systemctl restart httpd
```

Utilisez Lynx pour vous connecter à l'application **docs** :

```
[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat1
```

```
[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat1
```

```
[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat1
```

```
[root@centos7 bin]#
```

Arrêtez l'instance tomcat1 :

```
[root@centos7 bin]# ./stopTomcat1
Using CATALINA_BASE:   /usr/tomcat8/tomcat1
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat1/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
```

Utilisez de nouveau Lynx pour vous connecter à l'application **docs** :

```
[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
      This is Tomcat2
```

```
[root@centos7 bin]# lynx --dump http://www.i2tch.loc/docs
This is Tomcat2
```

```
[root@centos7 bin]#
```



Attention : Notez que le basculement est automatique en cas de défaillance de l'instance tomcat1.

Maintenir l'Etat des Clients

Préparation

Editez le fichier **web.xml** de l'application **/docs** de chaque instance de Tomcat en incluant la directive **<distributable/>** :

```
[root@centos7 bin]# vi $CATALINA_HOME/tomcat1/webapps/docs/WEB-INF/web.xml
[root@centos7 bin]# vi $CATALINA_HOME/tomcat2/webapps/docs/WEB-INF/web.xml
[root@centos7 bin]# tail $CATALINA_HOME/tomcat2/webapps/docs/WEB-INF/web.xml
    http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd"
    version="3.1"
    metadata-complete="true">

    <display-name>Tomcat Documentation</display-name>
    <description>
        Tomcat Documentation.
    </description>
    <distributable/>
</web-app>
[root@centos7 bin]# tail $CATALINA_HOME/tomcat1/webapps/docs/WEB-INF/web.xml
    http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd"
```

```
version="3.1"
metadata-complete="true">

<display-name>Tomcat Documentation</display-name>
<description>
    Tomcat Documentation.
</description>
<distributable/>
</web-app>
```

Créez les fichiers **`$CATALINA_HOME/tomcat1/webapps/docs/session.jsp`** et **`$CATALINA_HOME/tomcat2/webapps/docs/session.jsp`** :

```
[root@centos7 bin]# vi $CATALINA_HOME/tomcat1/webapps/docs/session.jsp
[root@centos7 bin]# cat $CATALINA_HOME/tomcat1/webapps/docs/session.jsp
<%@page language="java" %>
<html>
<body>
<h3>
Session : <%= session.getId() %>
</h3>
</body>
</html>
```

```
[root@centos7 bin]# vi $CATALINA_HOME/tomcat2/webapps/docs/session.jsp
[root@centos7 bin]# cat $CATALINA_HOME/tomcat2/webapps/docs/session.jsp
<%@page language="java" %>
<html>
<body>
<h3>
Session : <%= session.getId() %>
</h3>
</body>
</html>
```

Décommentez la ligne suivante dans les fichiers **server.xml** :

```
[root@centos7 bin]# vi $CATALINA_HOME/tomcat1/conf/server.xml
...
    <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"/>
...
[root@centos7 bin]# vi $CATALINA_HOME/tomcat2/conf/server.xml
...
    <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"/>
...
```

Sessions Persistentes sur Système de Fichiers

Editez maintenant les fichiers **\$CATALINA_HOME/tomcat1/conf/context.xml** et **\$CATALINA_HOME/tomcat2/conf/context.xml** en ajoutant la section suivante :

```
<Manager className="org.apache.catalina.session.PersistentManager" >
  <Store className="org.apache.catalina.session.FileStore"
    directory="/tmp/sessions/" />
</Manager>
```

Vous obtiendrez un résultat similaire à celui-ci :

```
[root@centos7 bin]# cat /usr/tomcat8/tomcat1/conf/context.xml
<?xml version='1.0' encoding='utf-8'?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements.  See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License.  You may obtain a copy of the License at
```

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-->

<!-- The contents of this file will be loaded for each web application -->

<Context>

```
<Manager className="org.apache.catalina.session.PersistentManager" >
  <Store className="org.apache.catalina.session.FileStore"
    directory="/tmp/sessions/" />
</Manager>
```

```
<!-- Default set of monitored resources. If one of these changes, the -->
<!-- web application will be reloaded. -->
```

```
<WatchedResource>WEB-INF/web.xml</WatchedResource>
```

```
<WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>
```

```
<!-- Uncomment this to disable session persistence across Tomcat restarts -->
```

```
<!--
```

```
<Manager pathname="" />
```

```
-->
```

```
<!-- Uncomment this to enable Comet connection tacking (provides events
on session expiration as well as webapp lifecycle) -->
```

```
<!--
```

```
<Valve className="org.apache.catalina.valves.CometConnectionManagerValve" />
```

```
-->
```

</Context>

Créez le répertoire **/tmp/sessions** pour contenir les fichiers de sessions :

```
[root@centos7 bin]# mkdir /tmp/sessions
```

En utilisant votre navigateur graphique, saisissez l'URL suivante :

```
http://www.i2tch.loc/docs/session.jsp
```

Vous obtiendrez une résultat similaire à l'exemple suivant :

```
Session : 7DA9FEE977543F1F574DADFA7B1FADD0.tomcat1
```

ou

```
Session : 7DA9FEE977543F1F574DADFA7B1FADD0.tomcat2
```

Selon l'instance de Tomcat qui a répondu, arrêtez cette instance :

```
[root@centos7 bin]# ./stopTomcat1
Using CATALINA_BASE:   /usr/tomcat8/tomcat1
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat1/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.232.b09-0.el7_7.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
```

ou

```
[root@centos7 bin]# ./stopTomcat2
Using CATALINA_BASE:   /usr/tomcat8/tomcat2
Using CATALINA_HOME:   /usr/tomcat8
Using CATALINA_TMPDIR: /usr/tomcat8/tomcat2/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk-1.8.0.91-1.b14.el7_2.x86_64
Using CLASSPATH:       /usr/tomcat8/bin/bootstrap.jar:/usr/tomcat8/bin/tomcat-juli.jar
```

Contrôlez le contenu du répertoire **/tmp/sessions** :

```
[root@centos7 bin]# ls -l /tmp/sessions
total 4
-rw-r--r-- 1 root root 263 Jul  5 23:32 7DA9FEE977543F1F574DADFA7B1FADD0.tomcat1.session
```

Revenez à votre navigateur Web graphique et rafraîchissez la page. Vous obtiendrez un résultat démontrant que la session est resté la même malgré le fait que c'est l'autre instance de Tomcat qui vous a répondu.

```
Session : 7DA9FEE977543F1F574DADFA7B1FADD0.tomcat1
```

ou

```
Session : 7DA9FEE977543F1F574DADFA7B1FADD0.tomcat2
```