VMRC (Virtual Machine image Repository & Catalog) Server

Revisión 07/04/2014

Latest version of the document: http://www.grycap.upv.es/vmrc

Content

Introduction 3

Assumptions 3

Pre-requisites 3

- Installing the pre-requisites 3
  - Installing the Java JDK 3
  - Installing the Apache Tomcat 4

Deploying the VMRC Web Service 4

Configuring the VMRC Web Service 7

- Change the Default Administrator User and Password 7
- Database Configuration 7
  - Database Configuration for MySQL 7
- Permissions Model 8
- VMI permissions 8
- User permissions 9
- Special Users 9
- Firewall Configuration 10

Operations performed by the Administrator 10

Add a new User to the VMRC Server 10

Delete a User from the VMRC Server.....10

Obtaining a list of Users from the VMRC Server 10

Change a User's ACL.....10

References 11

Introduction

VMRC (Virtual Machine image Repository & Catalog) is client-server system (based on Web Services) to catalog and store Virtual Machine Images (VMI) along with its metadata (OS, applications, etc.). It supports matchmaking to obtain the appropriate VMIs that satisfy a given set of hard (must) requirements and soft (should) requirements.

This document describes the server-side component of the VMRC.

Assumptions

This guide assumes that you will install the VMRC Server on a Linux (or OS X platform). Since this component is a Java-based Web Service, it might run on other platforms.

Pre-requisites

The VMRC system requires the Java Development Kit (JDK) 1.7.0 (or greater version) and the Apache Tomcat 7 application server. Version 7.0.28 is the last version in which the VMRC Server can be deployed successfully. It has been tested that from Tomcat version >= 7.0.29 it no longer works properly.

Installing the pre-requisites

Installing the Java JDK

The easiest way to install the Java JDK in a Linux box is via the package management tools. For example, installing the Java JDK in Ubuntu requires the following command:

```
sudo apt-get install openjdk-7-jdk
```

You can find out where the Java JDK has been installed if you list the /etc/alternatives/java file. Then define the JAVA_HOME variable. Add the following lines at the end of the $HOME/.bashrc file (use your specific configuration):

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk
```

Do not forget to load the environment variable by invoking:

```
source $HOME/.bashrc
```

Installing the Apache Tomcat

Download and install the Apache Tomcat 7 [1] application server (do not forget to choose version <=7.0.28). Choose the binary distribution for simplicity. This way, you will only have to unpack the package into your preferred destination folder:

```
tar zxvf apache-tomcat-7.0.14.tar.gz
```

Define the required environment variables. Add the following lines at the end of the $HOME/.bashrc file (specify your precise configuration).

```
export TOMCAT_HOME=<path_to_the_folder>
```

Do not forget to load the environment variable by invoking:

```
source $HOME/.bashrc
```

Deploying the VMRC Web Service

Make sure you have the environment variables JAVA_HOME and TOMCAT_HOME correctly configured before attempting these steps.

1. Download the VMRC Server package (either binary or source version) from http://www.grycap.upv.es/vmrc
    - If you downloaded the source version, follow the installation procedure detailed in the README.txt file in order to generate the vmrc.war file
    - If you downloaded the binary version, the package includes a previously generated vmrc.war file.
2. About the the *vmrc.war* and *vmrc-web-gui.war* files.
    - The vmrc.war file includes the VMRC server.
    - The vmrc-web-gui.war includes the simple web-based GUI. Installing this component is optional and you might find it still not available, since it will be released in the near future. You can safely skip its installation if you only want programmatic access to the VMRC server.
3. **(Optional)** Configure Tomcat to enable HTTPS access to the VMRC (for enhanced privacy, if required)

- Create a Java Key Store with the keytool utility available in the JDK

```
keytool -genkey -alias tomcat -keyalg RSA -keystore $TOMCAT_HOME/conf/vmrcert
```

You will have to specify the details about the certificate and a password. Use whatever password and information about the certificate you want.

- Include the following connector into the $TOMCAT_HOME/conf/server.xml, in the Service section.

```
<Service name="Catalina">
…
<Connector port="8444" protocol="HTTP/1.1" SSLEnabled="true"
  maxThreads="150" scheme="https" secure="true"
  clientAuth="false" sslProtocol="TLS"
  keystoreFile="/opt/apache-tomcat-7.0.12/conf/vmrcert"
keystorePass="p9i8uj7"/>
…
</Service>
```

Notice that you are referencing the Java key store previously created. Use the password you specified when creating the key store. You have further information about this step in the Apache Tomcat Documentation [2].

4. Deploy the *vmrc.war* and *vmrc-web-gui.war* into Tomcat.
    - Just drop both files into $TOMCAT_HOME/webapps
5. Start Tomcat
    - $TOMCAT_HOME/bin/startup.sh
6. Verify that it has been successfully deployed.
    - Access the VMRC endpoint at:
        i. http://localhost:8080/vmrc/vmrc (HTTP)
        ii. https://localhost:8444/vmrc/vmrc (HTTPS)
    - Access the VMRC Web GUI to ensure it is up and running (notice that you won't be able to access the Web GUI unless you deployed the vmrc-web-gui.war file into Tomcat)
        i. http://localhost:8080/vmrc-web-gui (HTTP)
        ii. https://localhost:8444/vmrc-web-gui (HTTPS)

Notice that the server listens at ports 8080 (http) and 8444 (https). You can change the ports by modifying the $TOMCAT_HOME/conf/server.xml Connector sections.

7. Shutdown the VMRC service

   This can be achieved by shutting down Tomcat.

   - $TOMCAT_HOME/bin/shutdown.sh

Configuring the VMRC Web Service

All the data employed by VMRC is available in the following folder:

$TOMCAT_HOME/webapps/vmrc

From now on, this path will be denoted $VMRC_SERVER_HOME, although you do not need to define such variable.

Change the Default Administrator User and Password

1. Change the default VMRC admin name and/or password
    a. Modify the $VMRC_SERVER_HOME/WEB-INF/classes/vmrc.properties

```
admin_password=passwd1
```

2. Restart Tomcat

```
$TOMCAT_HOME/bin/shutdown.sh
$TOMCAT_HOME/bin/startup.sh
```

Database Configuration

The VMRC Server comes preconfigured with an in-memory HSQLDB database. This means that no database is required to be configured in order to test the functionality of the VMRC catalog. However, once you stop Apache Tomcat, all the data will be gone. Therefore, do not plan to use it for production purposes.

You can configure another database backend (such as MySQL) by changing the configuration in $TOMCAT_HOME/webapps/vmrc/WEB-INF/classes/hibernate.cfg.xml

You should only specify the connection details to the database. The DB schema will be automatically created upon the service startup the next time you restart Apache Tomcat.

Sample configuration is provided on that file for both HSQLDB and MySQL databases. Since Hibernate is employed as the persistence tool, the underlying backend should be supported by Hibernate (typically any SQL-based oriented DB).

Database Configuration for MySQL

To use MySQL as the database backend you can use the following instructions (for Ubuntu):

1. Install MySQL server
    a. sudo apt-get install mysql-server

2. Connect to MySQL as the root user (and the password specified during installation)
    a. mysql -u root -p
3. Create the database and the user that will be used to connect from VMRCServer
    a. mysql> create database vmrc;
    b. mysql> create user vmrc identified by 'password';
    c. mysql> grant all on vmrc.* to vmrc@localhost;
4. Modify $VMRC_SERVER_HOME/WEB-INF/classes/hibernate.cfg.xml to swich from HSQLDB to MySQL
    a. Uncomment the MySQL section and comment the HSQLDB section
    b. Specify the values for:

```
<property
name="hibernate.connection.username">vmrc</property>

<property
name="hibernate.connection.password">password</property>

<property
name="hibernate.connection.url">jdbc:mysql://localhost/vmr
c</property>
```

5. Restart the VMRCServer service
    a. $TOMCAT_HOME/bin/shutdown.sh
    b. $TOMCAT_HOME/bin/startup.sh

Permissions Model

The VMRC Server supports a permission model that enables to authorize specific operations to registered users and on a per-VMI basis.

VMI permissions

Each VMI has one Access Control List (ACL) with the following permissions:

PERMISSION	DESCRIPTION	VALUES	DEF. VALUE
LIST	This VMI will be included in the listings performed by a user with the 'LIST' permission granted	owner | all	all
SEARCH	This VMI will be included in the list of VMIs obtained by the search operation performed by a user with the 'SEARCH' permission granted. If set to *owner*, only the owner of the VMI is allowed to obtain this VMI as the result of a *search* operation.	owner | all	all

UPLOAD	A user with the 'UPLOAD' permission granted will be able to upload a file for this VMI. If set to *owner*, only the owner of the VMI is allowed to obtain this VMI as the result of a *list* operation	owner | all	owner
DOWNLOAD	A user with the 'DOWNLOAD' permission granted will be able to download the file for this VMI. If set to *owner*, only the owner of the VMI is allowed to download the VMI.	owner | all	all
DELETE	A user with the 'DELETE' permission granted will be able to delete this VMI. If set to *owner*, only the owner of the VMI is allowed to delete this VMI.	owner | all	owner
ADD	A user with the 'ADD' permission granted would be able to update the values of the VMI. If set to *owner*, only the VMI owner is allowed to update. (This feature is currently unimplemented.)	owner | all	owner

User permissions

Each user has one ACL with the following permissions:

PERMISSION	DESCRIPTION	VALUES	DEF. VALUE
LIST	The user is allowed to list the VMIs which include the 'LIST' permission.	owner | all | none	all
SEARCH	The user is allowed to search the VMIs which include the 'SEARCH' permission	owner | all |none	all
UPLOAD	The user is allowed to upload files for the VMIs which include the 'UPLOAD' permission	owner | all |none	owner
DOWNLOAD	The user is allowed to download files from the VMIs which include the 'DOWNLOAD' permission	owner | all |none	all
DELETE	The user is allowed to delete the VMIs which include the 'DELETE' permission	owner | all |none	owner
ADD	The user is allowed to register a VMIs in the catalog.	owner | all |none	owner

Note that if the permission *none* is specified, the corresponding operation will not even proceed server-side.

Special Users

ADMIN	
PERMISSION	**VALUE**
ADD	all
LIST	all
SEARCH	all
UPLOAD	all
DOWNLOAD	all
DELETE	all

ANONYMOUS	
PERMISSION	**VALUE**
ADD	none
LIST	all
SEARCH	all
UPLOAD	none
DOWNLOAD	none
DELETE	none

Firewall Configuration

The VMRC Server uses port 21000 for FTP transfers.

Operations performed by the Administrator

All the following commands require using the $VMRC_SERVER_HOME/bin/vmrc-admin.sh tool and admin client-side credentials. In a Windows platform, the admin CLI tool can be invoked as follows:

```
java –cp  vmrc-client.jar org.grycap.vmrc.client.cmd.admin.VMRCAdminCLI ---aduser john johndoe
```

Add a new User to the VMRC Server

To add a new user called john with password johndoe, you have to issue the following command:

```
$VMRC_SERVER_HOME/bin/vmrc-admin.sh --adduser john johndoe
```

Delete a User from the VMRC Server

To delete user john, the following command is required:

```
$VMRC_SERVER_HOME/bin/vmrc-admin.sh --deleteUser john
```

Obtaining a list of Users from the VMRC Server

```
$VMRC_SERVER_HOME/bin/vmrc-admin.sh --listUsers
```

You can obtain extended information by producing XML output.

```
$VMRC_SERVER_HOME/bin/vmrc-admin.sh --listUsers --xml
```

Change a User's ACL

```
$VMRC_SERVER_HOME/bin/vmrc-admin.sh –userAcl username operation perm
```

Where operation = [add|list|upload|search|delete] perm = [all|owner|none]

Please refer to the previous section for a detailed explanation of the permissions model.

References

[1] http://tomcat.apache.org/download-70.cgi

[2] http://tomcat.apache.org/to_mcat-7.0-doc/ssl-howto.html#Prepare_the_Certificate_Keystore