

IBM Information Management software



**IBM® DB2®
Amazon® Machine
Image (AMI)
Re-Bundling Guide**

Thomas Chong (chongt@ca.ibm.com)

**Data Management Emerging Partnership and Technologies
IBM Toronto Lab
Updated 24 March 2009**

Contents

1. INTRODUCTION	- 3 -
1.1 PURPOSE	- 3 -
1.2 PREREQUISITES FOR BUNDLING AN AMI INSTANCE	- 3 -
2. INTEGRATING CUSTOM ISV SOFTWARE	- 4 -
3. AMI RE-BUNDLING INSTRUCTIONS	- 5 -
3.1 LICENSE SCREENS	- 5 -
3.2 RESTORING DB2CONFIG TO ITS INITIAL STATE.....	- 8 -
3.3 RESTORING AWSSYSCONFIG TO ITS INITIAL STATE	- 12 -
3.4 RESTORING AWSCONFIG AND EBSCONFIG TO THEIR INITIAL STATES.....	- 14 -
3.5 THE .VERSION FILE	- 14 -
3.6 THE /ROOT/.BASHRC_AMI SCRIPT	- 14 -
3.7 RESETTING THE STARTUP SERVICES	- 16 -
3.8 RE-BUNDLING, UPLOADING, AND REGISTERING OF AMI	- 17 -
3.9 PUBLICIZING YOUR AMI	- 19 -
4. ADDITIONAL INFORMATION	- 20 -
4.1 FEEDBACK.....	- 20 -

1. Introduction

1.1 Purpose

The purpose of this document is two-fold:

- ▶ To outline the process for integrating software applications into the IBM® DB2® AMIs in the Amazon® Elastic Compute Cloud™ (EC2™)
- ▶ To explain the steps, technical issues, and considerations regarding the re-bundling of Amazon Machine Images (AMIs)

For instructions on the deployment of the AMIs, please refer to:

- ▶ IBM DB2 9.5 Amazon Machine Image (AMI) Get Started Guide, available at: http://download.boulder.ibm.com/ibmdl/pub/software/dw/cloud/udbexp/Get_Started_DB2_AMI.pdf
- ▶ Amazon EC2 guide, available at: <http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/>

The integration instructions contained in this guide can be applied to:

- ▶ DB2 Express-C 9.5 AMI (32-bit / 64-bit)
- ▶ DB2 Express 9.5 AMI (32-bit)
- ▶ DB2 Workgroup AMI (64-bit)

This document is intended for a technical audience and will help independent solution vendors (ISVs) obtain knowledge about integrating their software with the existing DB2 AMIs in Amazon Web Services (AWS).

1.2 Prerequisites for Bundling an AMI Instance

In order to perform bundling and uploading operations on your AMI Instance, you will need the EC2 AMI Tools. By default, both the EC2 AMI and API Tools are installed on all IBM DB2 AMIs. If you had removed the EC2 AMI Tools at an earlier time, they can be downloaded at <http://developer.amazonwebservices.com/connect/entry.jspa?externalID=368>. Likewise, the EC2 API Tools can be downloaded at <http://developer.amazonwebservices.com/connect/entry.jspa?externalID=351>.

Since you will be performing these operations directly within your instance, you will also need to place within the file system of your instance your AWS credentials, such as the X.509 certificate and private key files.

This guide assumes that files mentioned within this document were not removed prior to the usage of this guide.

2. Integrating Custom ISV Software

The purpose of pre-integrating software into the DB2 AMI is to alleviate the application installation, configuration, and integration tasks from the end-user, making the deployment of the applications on DB2 in an AMI easy and painless.

Before continuing on the task of integration an application on a running AMI instance, it is recommended that you create an AMI that captures the current state of the instance. In the case where the application fails to be configured properly or there is a problem with the AMI, a backed-up, working AMI may be deployed so the next iteration of the integration process can start from a clean environment.

Integrating an ISV application into the DB2 AMI is similar to installing and configuring the application to work within an operating system in a normal, physical environment with a few extra considerations.

The following guidelines specify the process for integrating an ISV application into a DB2 AMI:

1. Install all required dependencies needed by ISV application (e.g. Linux® operating system packages)
2. Install the ISV application and configure the default settings of the ISV application to work with DB2
3. Verify that all components of the ISV application are installed and configured successfully. Create, test, or sample data from the ISV application.
4. Develop and execute test plans for this integration task; Ensure the application is able to run properly, able to access the DB2 database, and is fully functional on the EC2 instance
5. Identify tasks to be performed at “first boot” of an AMI instance. E.g. any configuration of the ISV application that requires user input such as user name and password for security.
6. Develop post-install plans based on the tasks identified in the previous step (if applicable). For example, if the ISV software files are dependent on any system or DB2 parameters such as system hostname, domain name, network protocol, DB2 instance name, port number, etc, be prepared to develop customized shell scripts and/or YaST configuration modules to configure the AMI instance accordingly.
7. Set the AMI instance back to a “first boot” state by resetting any configuration scripts and/or YaST configuration modules. This will ensure the scripts/modules will be executed during the initial boot-up of an instance of the newly re-bundled AMI.
8. Re-bundle the file system into an AMI. When you re-bundle an AMI in EC2, the directories `/sys`, `/proc`, `/dev`, `/media`, and `/mnt` are not included. Therefore,

it's recommended that any files that need to be preserved in the newly re-bundled AMI should not be stored in these directories.

3. AMI Re-Bundling Instructions

If you installed an ISV application on top of the DB2 AMI, you may want to restore the AMI back to the initial state. This will allow end-users to go through the original sequence of license agreement acceptance screens and any DB2 and AMI related configuration modules. An outline of the AMI re-bundling steps are listed as follows:

1. Adding additional license text associated with the integrated ISV application to be displayed in the license agreement acceptance screens. See section [3.1 License Screens for details](#).
2. Restoring `awsconfig`, `ebsconfig`, `awssysconfig`, and `db2config` YaST modules into their initial state to allow these modules to be run during the initial boot-up of the re-bundled AMI. See sections [3.2 Restoring `db2config` to its Initial State](#), [3.3 Restoring `awssysconfig` to its Initial State](#), and [3.4 Restoring `awsconfig` and `ebsconfig` to their Initial States](#) for details.
3. Replacing the file `/root/.bashrc` with the content in `/root/.bashrc_ami` to invoke the YaST modules required for DB2 AMI configuration upon first login of the root user. See section [3.6 The `/root/.bashrc_ami` Script](#) for more details.
4. Removing any user specific confidential data such as private keys, certificate files, and RSA public keys from the AMI instance.

Note: If you do not remove any user specific confidential data before re-bundling, they might get included in the file system of the new AMI, potentially exposing them to the public if you choose to make this new AMI public.

5. Packaging up the AMI which has ISV application integrated with DB2. See section [3.8 Re-bundling, Uploading, and Registering of AMI](#) for details.

3.1 License Screens

If you wish to add additional products to the DB2 AMI, you will need to add the corresponding license agreements for your product to the initial license acceptance screen.

1. You will need to have your product license file in .txt file format.

In order to have the license displayed correctly in the license module, you will need the license text to be wrapped around column 65.

2. If the license text file has not been wrapped, you can do so by issuing the `fmt` command on Linux:

```
# fmt -w 65 unwrapped_lic > \  
/var/adm/vm-postinstall/product_license.txt
```

where:

`unwrapped_lic` is the filename of your original unwrapped license text file;
`product_license.txt` file is the filename of the newly wrapped license text file.

3. Name the wrapped license text file as `<product_name>_license.txt`.
4. Copy the wrapped license txt file into `/var/adm/vm-postinstall` directory.

```
# cp <license>.txt /var/adm/vm-postinstall/.
```

5. Open the `db2license` script in `/var/adm/vm-postinstall` directory. The `db2license` script is a shell script which is used to display various license agreements and user acceptance buttons in text mode.
6. Add the following two variables in the `/var/adm/vm-postinstall/db2license` script file:

```
NUMLIC=<total_number_of_licenses>  
BACKTITLE[#]="<License_Agreement_title>"  
LICFILE[#]="${THISPATH}/<name_of_license_file>"
```

Note: # above denotes the sequence in which the license agreements are shown in the license module. It needs to be in ascending order incremented by one each time.

`BACKTITLE[#]` variable starts from `BACKTITLE[0]`, `BACKTITLE[1]`, and so on. `BACKTITLE[0-1]` are assigned to Linux Distribution Agreement and the Novell® SUSE® Linux Enterprise Server (SLES) License agreement by default. These agreements should be shown before the license screens for other ISV applications are integrated in the DB2 AMI.

As for the `LICFILE[#]` variable, it denotes the exact location path of the license agreement txt file associated with the license agreement title assigned to the

above BACKTITLE[#] variable. LICFILE[#] variable starts from LICFILE[1], LICFILE[2], and so on. Similar to BACKTITLE, LICFILE[1] is assigned to Novell SLES license agreement by default. It should not be modified or removed in any case.

Note: Please do not modify or remove the Linux Distribution Agreement, and the Novell SLES 10 License Agreement. These agreements are represented by BACKTITLE[0] and BACKTITLE[1]/LICFILE[1] respectively

For example, to add a new license agreement, add the following blue lines to the db2license script.

```
BACKTITLE[0]="Linux Distribution Statement"
BACKTITLE[1]="Novell SLES10 License Agreement"
BACKTITLE[2]="New product license agreement"
...
LICFILE[1]="${THISPATH}/sles10_license.txt"
LICFILE[2]="${THISPATH}/new_productname_license.txt"
```

Note: Please make sure the provided new_productname_license.txt file has been text-wrapped to 65 columns width.

Once you have added a new license file to the list, you will need to modify the following variable as well.

The value of the NUMLIC variable should be set to the total number of licenses displayed (i.e. LICFILE[#]). In this example, the number should set to 2.

```
NUMLIC=2
```

7. If the license screens have been previously displayed on the instance, the /var/adm/vm-postinstall/wasrun.lck file will be created. This existence of this lock file will prevent the db2license shell script from displaying the license agreement screens again. Remove the /var/adm/vm-postinstall/wasrun.lck file so that the license screens can be displayed in the initial boot-up of the AMI instance.

Checklist:

If you would like to add additional license files for your product to the licensing screens, make sure you have completed all the tasks in the following checklist as explained earlier in the section.

- a. Create a license agreement as a .txt file. Wrap up the paragraph width to 65 columns.
- b. Assign a new variable `BACKTITLE[2]`="Your product license agreement title"
- c. Assign a new variable and point it to the associated license file location:
`LICFILE2="{THISPATH}/your_product_license.txt"`
- d. Repeat step (b) and (c) for each license agreement that needs to be displayed.
- e. Reset the `NUMLIC` variable to the updated total number of licenses.
- f. Save the changes to `/var/adm/vm-postinstall/db2license` script file
- g. When all the changes are made, remove the `/var/adm/vm-postinstall/wasrun.lck` file

3.2 Restoring db2config to its Initial State

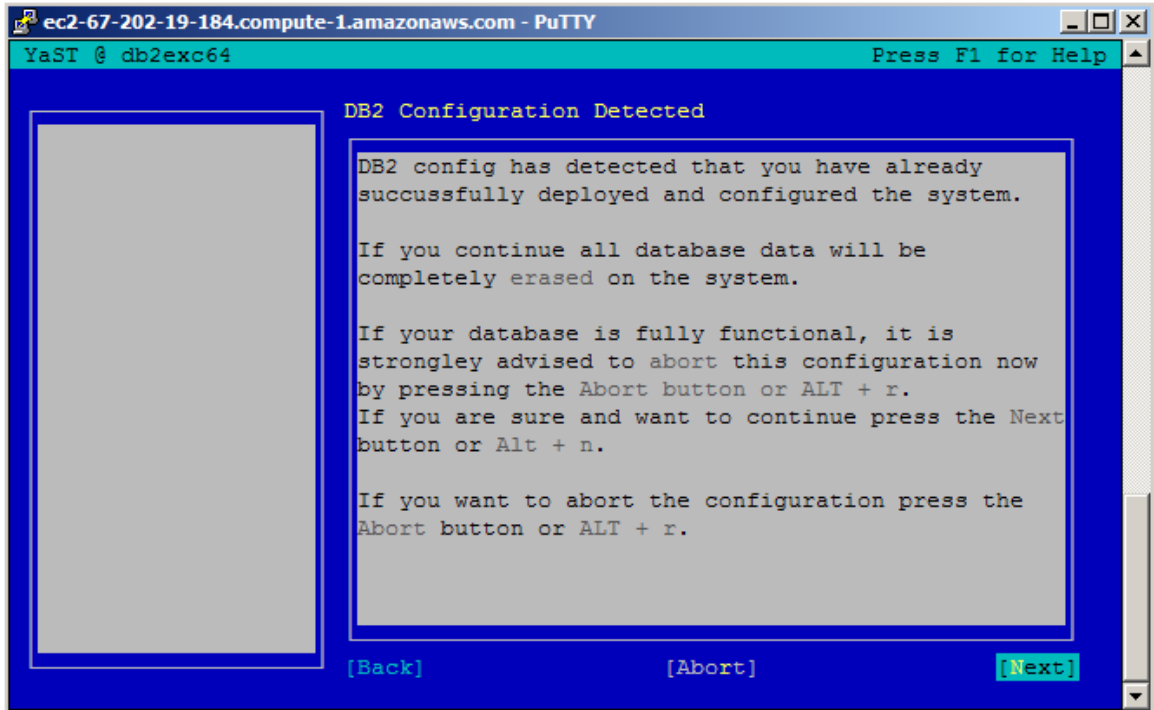
1. Logout all DB2 instance users, Database Administration Server (DAS) users, and fenced users from the system.

Note: This step removes all existing data from the current DB2 instance. If you have any needed DB2 data on the existing instance, please make sure you back-up your data outside of your AMI instance, e.g. to an EBS volume

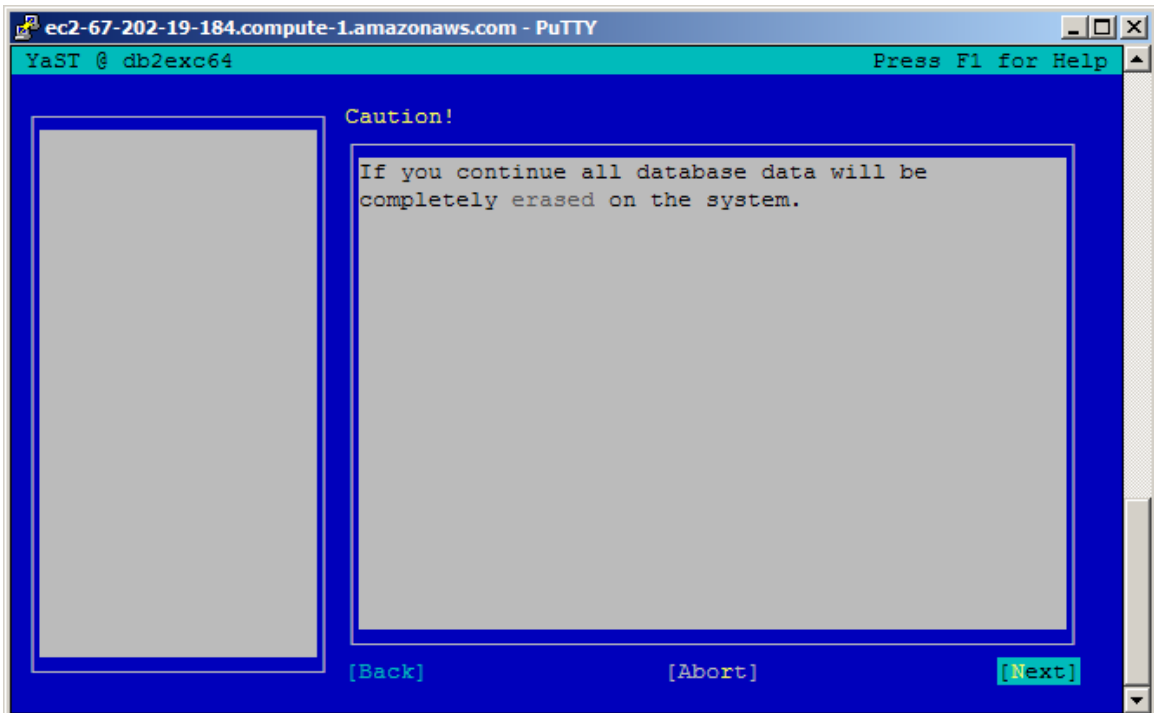
2. Execute the following command in a terminal. This will start the db2config module again.

```
# yast2 db2config
```

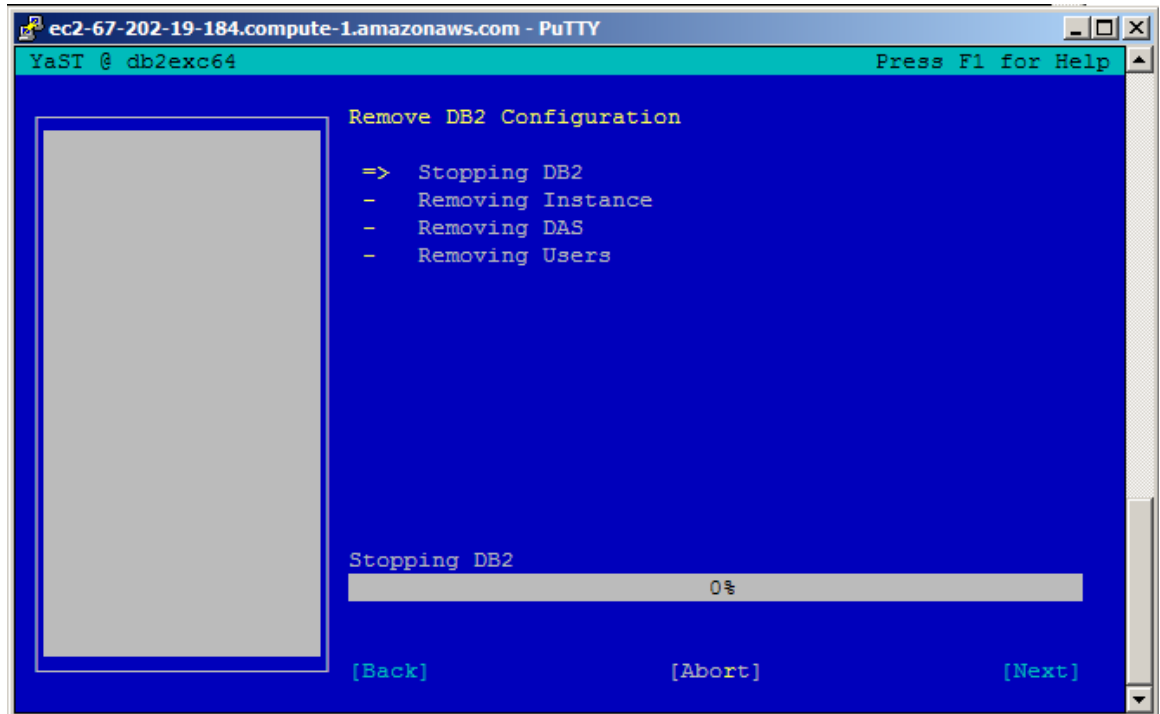
3. The YaST2-db2config module will start. The following screen shows that there is an existing DB2 Configuration. Press **[Next]** to continue.



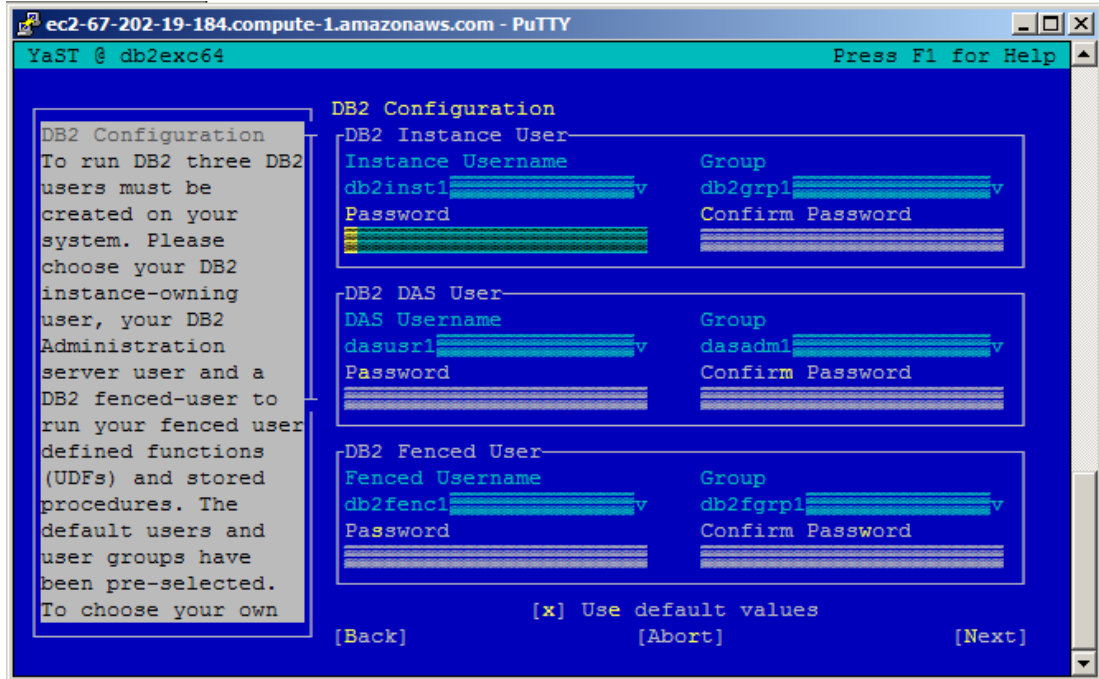
4. The next screen will present a warning stating that the existing DB2 configuration will be erased. Press **[Next]** to continue.



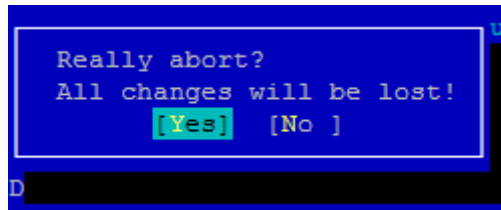
5. The next screen will show the progress of clearing existing DB2 configurations. This will stop the DB2 service and remove all DB2 instances and users on the AMI instance.



6. When the YaST module has finished removing the existing DB2 configurations, the following screen will appear. At this stage, press **[Abort]** to abort the db2config module.



7. A message box will pop up prompting you to confirm the Abort. Click **[Yes]** to exit the module.



8. The db2config module and DB2 configuration on the instance is now reverted back to its original state.
9. Remove any .ok, .error, .logfile* files from /usr/share/YaST2/include/db2config directory by using the following commands. This will clean up the environment so the db2config YaST module can run on a clean environment at the next initial boot-up.

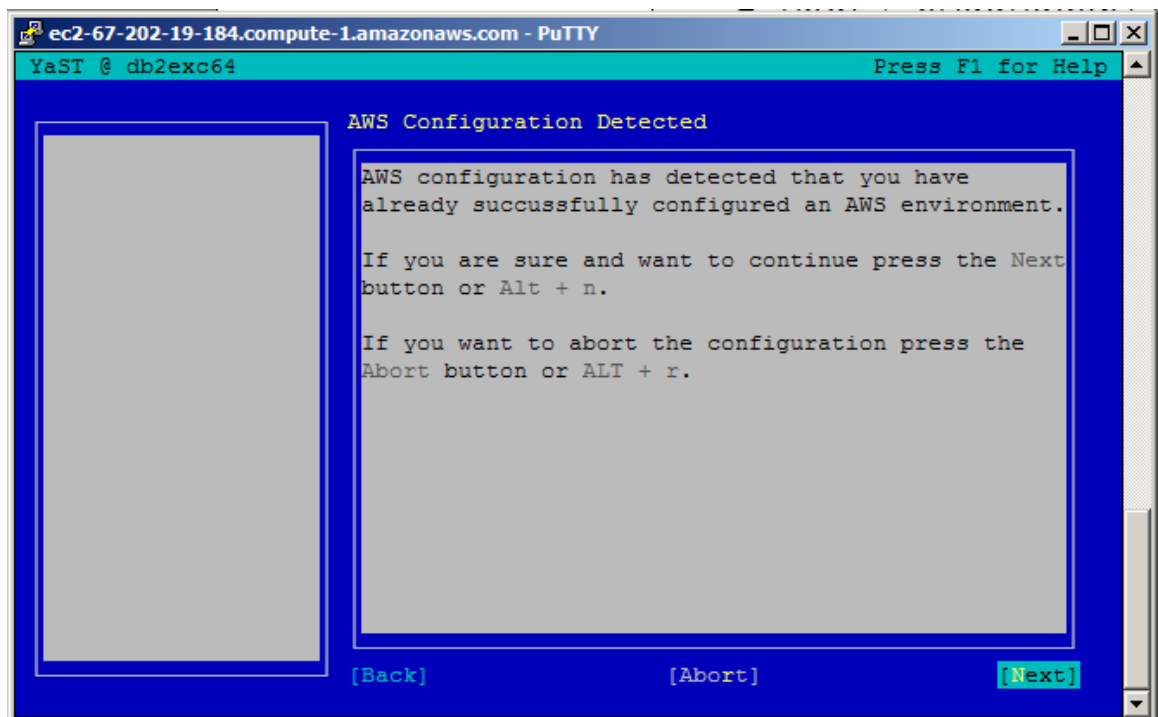
```
# cd /usr/share/YaST2/include/db2config/
# rm .ok .error .logfile*
```

3.3 Restoring awssysconfig to its Initial State

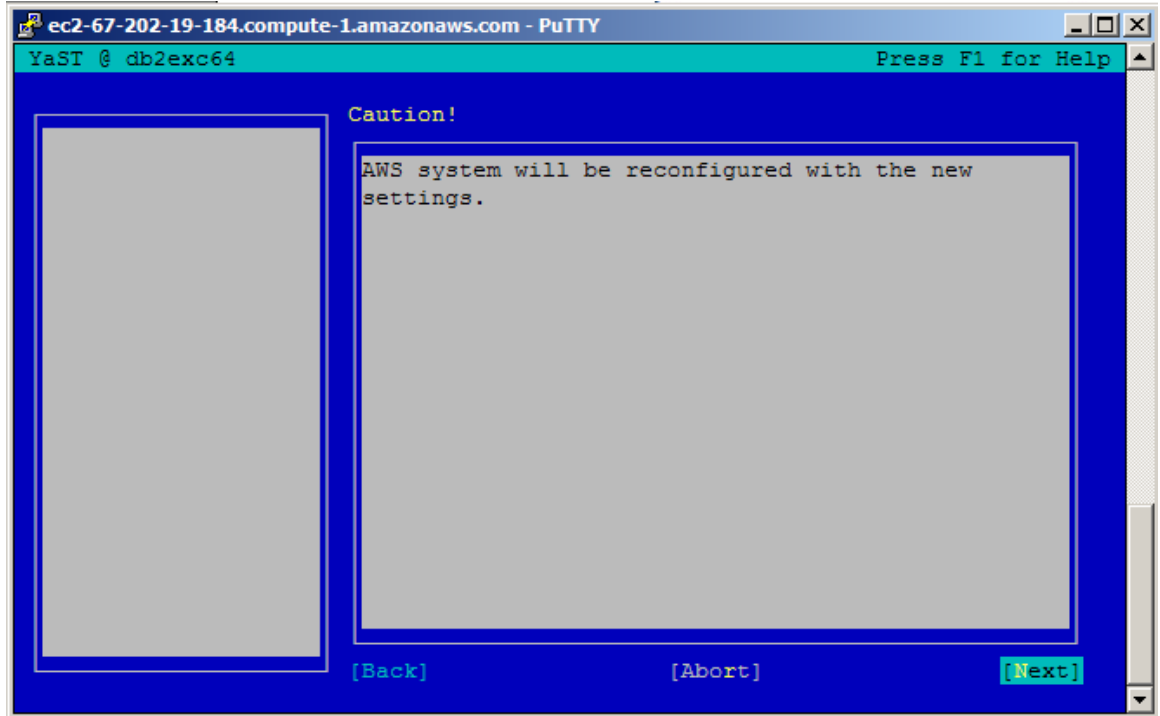
1. Execute the following command on the command line. This will start the awssysconfig module.

```
# yast2 awssysconfig
```

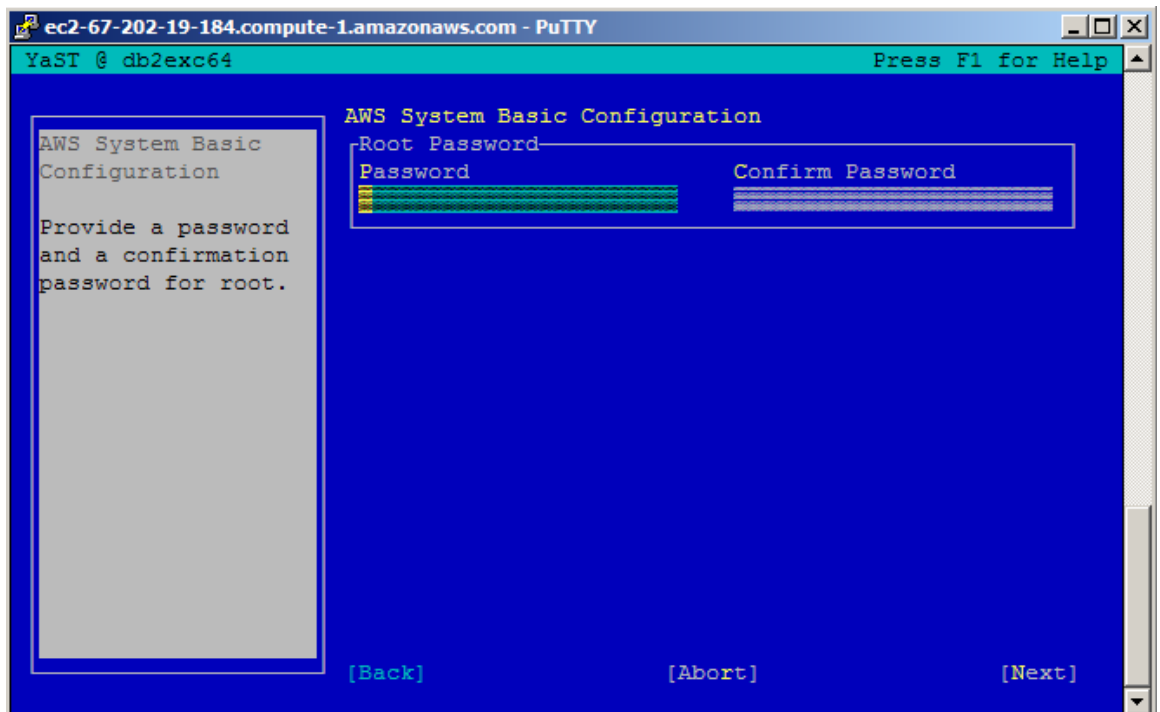
2. The YaST2-awssysconfig module will start. You will see the following screen stating that an existing AWS System configuration is detected. Press **[Next]** to continue.



3. On the next screen, you will be warned that the existing system configuration will be replaced with new setting. Press **[Next]** to continue.



4. The following screen will prompt you to enter a root password for the DB2 AMI instance. Enter **password** as the root password and confirm it by entering it again. Press **[Next]** to continue.



5. After the awssysconfig module has finished running, remove any `.ok/`, `.error/`, and `.logfile*` files from the `/usr/share/YaST2/include/awssysconfig` directory by issuing the following commands. This will clean up the environment so the awssysconfig YaST module can run on a clean environment at the next initial boot-up.

```
# cd /usr/share/YaST2/include/awssysconfig/  
# rm .ok .error .logfile*
```

3.4 Restoring awsconfig and ebsconfig to their Initial States

To restore awsconfig and ebsconfig YaST modules to their original states, remove any `.ok/`, `.error/`, and `.logfile*` files from `/usr/share/YaST2/include/awsconfig` and `/usr/share/YaST2/include/ebsconfig` directories:

```
# cd /usr/share/YaST2/include/awsconfig/  
# rm .ok .error .logfile*  
  
# cd /usr/share/YaST2/include/ebsconfig/  
# rm .ok .error .logfile*
```

This will allow the awsconfig and ebsconfig YaST modules to run on a clean environment upon the next initial boot-up.

3.5 The `.version` file

The `/var/adm/vm-postinstall/.version` file keeps track of the version of the DB2 AMI. If you would like to track the version number and packages installed in the AMI, you may update the `.version` file accordingly.

3.6 The `/root/.bashrc_ami` Script

The `/root/.bashrc` script is executed every time a shell is opened by the root user. The `/root/.bashrc_ami` script is a customized copy of the standard `/root/.bashrc` script. Include all post-install scripts or YaST configuration modules in `/root/.bashrc_ami`. This file needs to be copied to `/root/.bashrc` just before re-bundling the AMI instance.

Follow the below instructions to take a closer look at the `bashrc_ami` file.

1. Open up the `/root/.bashrc_ami` file with a text editor. (For example, `vi` in Linux).
2. The `/root/.bashrc_ami` file contains scripts that will be executed during the first interactive login of the root user. The root user is the only existing user on an AMI before the initial boot-up configuration is completed.

```

If [ "$(echo $- | grep -c i)" == 1 ]; then
    export JAVA_HOME=/usr/java
    export EC2_HOME=/root/ec2
    # Run license agreements screen
    /var/adm/vm-postinstall/db2license
    yast awsconfig
    if [ ! -f /usr/share/YaST2/include/awsconfig/.noebs ];
then
        . /root/.ec2
        yast ebsconfig
    fi
    yast2 awssysconfig
    yast2 db2config
    if [ -f /usr/share/YaST2/include/awssysconfig/.ok -a \
-f /usr/share/YaST2/include/db2config/.ok ]; then
        rcapache2 restart
#This sets up the swap space for large instances
fdisk /dev/sdc << EOF
n
p
1
1
+16G
t
82
w
EOF
mkswap /dev/sdc1
echo "/dev/sdc1 swap swap defaults 0 0" >> /etc/fstab
swapon -a
#end swap setup
clear
echo -e "***** THE DB2 9.5 INSTANCE \
IS NOW READY *****\n"
    mv /root/.bashrc /root/.bashrc_ami
    cp /root/.ec2 /root/.bashrc
else
    echo -e "***** To run the
configuration again, login again *****\n"
fi
fi

```

The diagram shows a code block with several red boxes and arrows pointing to specific lines, labeled A through E:

- A** points to `export JAVA_HOME=/usr/java`
- B** points to `export EC2_HOME=/root/ec2`
- C** points to the `if [! -f /usr/share/YaST2/include/awsconfig/.noebs];` line.
- D** points to the `fdisk /dev/sdc << EOF` line.
- E** points to `mv /root/.bashrc /root/.bashrc_ami`

When a user launches an AMI instance and login as root user, the `/root/.bashrc` script will kick-off the following tasks:

- a. Two environment variables are set, namely the **JAVA_HOME** environment variable and the **EC2_HOME** variable, both of which are required for the EC2 API tools to work properly. Both environment variables are set to the respective location of the two programs.
 - b. The license agreement screens will be displayed to the user by the `/var/adm/vm-postinstall/db2license` script. The user must accept these license agreements to proceed.
 - c. The `awsconfig`, `ebsconfig`, `awssysconfig`, and `db2config` YaST modules are executed to complete the initial configuration of the AMI. When all the DB2 AMI configuration modules are completed, the `apache2` service will be restarted. The initial configuration of the AMI is finished at this point.
 - d. This section is only applicable to the IBM DB2 64 bit AMIs. It is responsible for creating swap space for large instances, since swap space for 64 bit instances are not automatically created. As such, 32 bit instances do not require this excerpt of code.
 - e. By renaming the `/root/.bashrc` file to `/root/.bashrc_ami` and then overwriting the content of `/root/.bashrc` with `/root/.ec2`, it prevents the above configuration tasks to be re-run at the next login of the root user, after the initial boot-up and login. These configuration tasks are not required since the environment is already setup.
3. If you would like to add any additional steps to the AMI initialization stage, please add the steps to the `/root/.bashrc_ami` file after stage (d), where all DB2 AMI related customized configuration modules are executed. Remember to copy the `/root/.bashrc_ami` over to `/root/.bashrc` before re-bundling.

Tip: If the installed ISV application is dependent on the DB2 instance/databases, the ISV application custom configuration steps should be executed after `yast2-db2config` module. The `db2config` module creates and configures a DB2 instance and database. Any steps dependent on DB2 databases should be done after `db2config`.

3.7 Resetting the Startup Services

Once you have finished modifying the `/root/.bashrc_ami` file, you will need to reset the start-up services. In the original start-up configuration of the DB2 AMI, certain services are not yet run at boot-up time, and as such, preserving the state of the original instance requires removal of certain startup services as listed below.

For DB2 AMIs, issue the following command:

```
# chkconfig -d db2
db2          0:off  1:off  2:off  3:off  4:off  5:off  6:off
```

This will remove the start-up service “db2” from run level 3 (which should be our current run level).

3.8 Re-Bundling, Uploading, and Registering of AMI

At this point, the file system of the AMI instance should be ready for bundling. To bundle the instance, go into the corresponding directory by issuing the command:

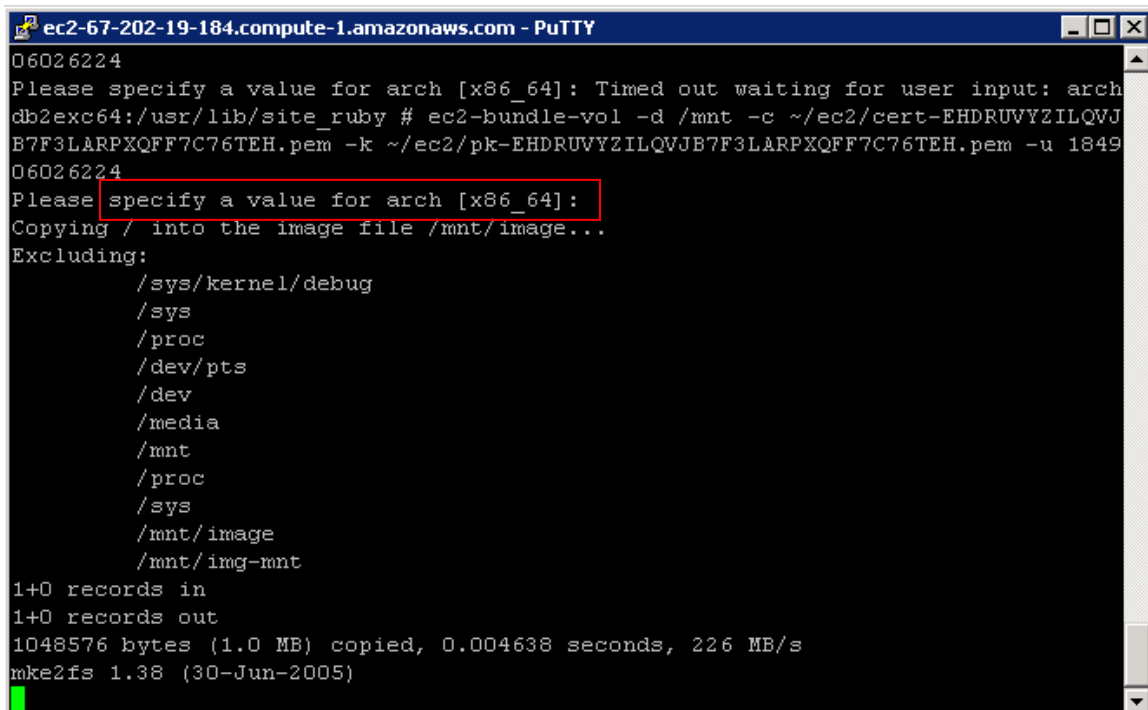
```
cd /usr/lib/site_ruby
```

To bundle the instance, issue the command::

```
ec2-bundle-vol -d /mnt -k <private key location> -c  
<certificate location> -u <AWS account ID>
```

The `-d` option specifies the local directory the bundle will be saved to. It is advisable to keep the certificate and private key files in the `/mnt` directory since volume bundling will exclude the `/mnt` directory, thus preventing the new AMI from bundling confidential information.

When prompted for a specifying “a value for arch”, press enter to choose the default value (this value will change depending on whether you are using the 32-bit or 64-bit versions).



```
ec2-67-202-19-184.compute-1.amazonaws.com - PuTTY
06026224
Please specify a value for arch [x86_64]: Timed out waiting for user input: arch
db2exc64:/usr/lib/site_ruby # ec2-bundle-vol -d /mnt -c ~/ec2/cert-EHDRUVYZILQVJ
B7F3LARPXQFF7C76TEH.pem -k ~/ec2/pk-EHDRUVYZILQVJB7F3LARPXQFF7C76TEH.pem -u 1849
06026224
Please specify a value for arch [x86_64]:
Copying / into the image file /mnt/image...
Excluding:
  /sys/kernel/debug
  /sys
  /proc
  /dev/pts
  /dev
  /media
  /mnt
  /proc
  /sys
  /mnt/image
  /mnt/img-mnt
1+0 records in
1+0 records out
1048576 bytes (1.0 MB) copied, 0.004638 seconds, 226 MB/s
mke2fs 1.38 (30-Jun-2005)
```

The bundling will now start and the excluded directories are shown. Bundling will take some time, as this step involves compressing the entire file system into a format that is compatible with Amazon EC2.

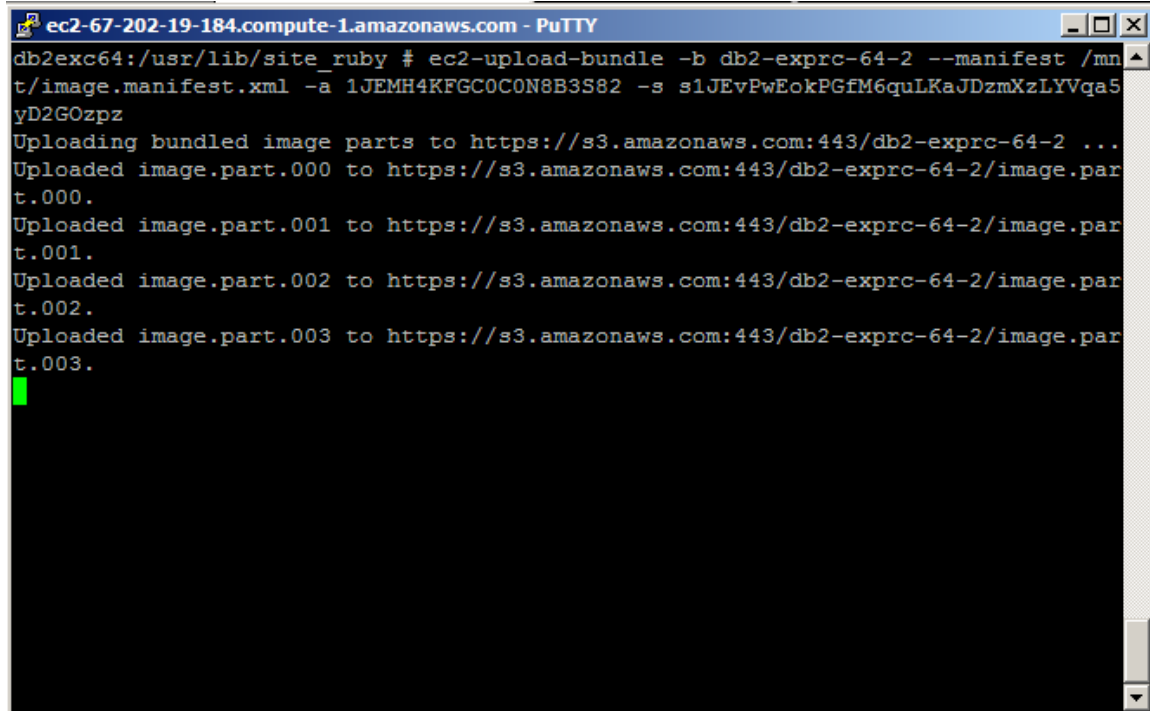
The bundle will be saved in `/mnt`. After bundling completes, it is time to upload the bundle to Amazon S3 storage by issuing the following command

```
ec2-upload-bundle -b <bucket name> --manifest <manifest location>  
-a <Amazon access key ID> -s <Amazon secret access key>
```

The `-b` option specifies the bucket (or folder) location that the bundle will be stored on the Amazon S3 storage. Buckets are automatically created if they do not already exist. The `--manifest` option points to the `image.manifest.xml` file stored locally on your instance.

Please refer to section 2.2 in *IBM DB2 9.5 Amazon Machine Image (AMI) Get Started Guide* regarding Amazon access key ID and secret access key. This document is available at:

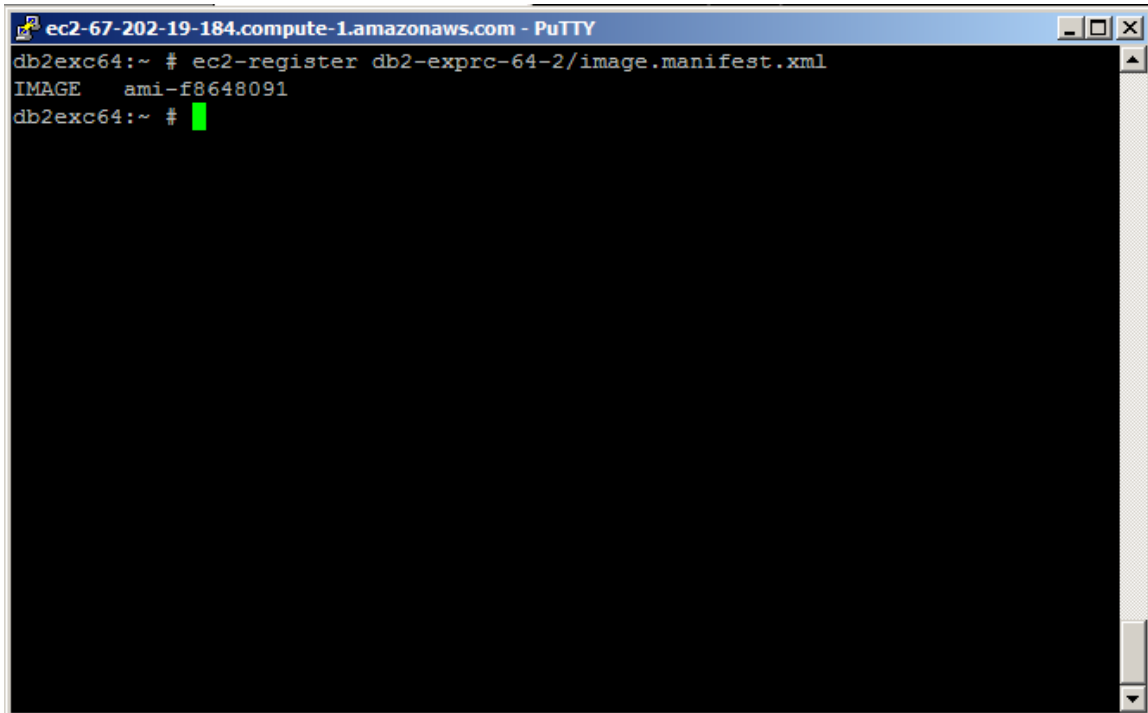
http://download.boulder.ibm.com/ibmdl/pub/software/dw/cloud/udbexp/Get_Started_DB2_AMI.pdf



```
ec2-67-202-19-184.compute-1.amazonaws.com - PuTTY  
db2exc64:/usr/lib/site_ruby # ec2-upload-bundle -b db2-exprc-64-2 --manifest /mnt/  
image.manifest.xml -a 1JEMH4KFGC0C0N8B3S82 -s s1JEvPwEokPGfM6quLKaJDzmXzLYVqa5  
yD2GOzpz  
Uploading bundled image parts to https://s3.amazonaws.com:443/db2-exprc-64-2 ...  
Uploaded image.part.000 to https://s3.amazonaws.com:443/db2-exprc-64-2/image.par  
t.000.  
Uploaded image.part.001 to https://s3.amazonaws.com:443/db2-exprc-64-2/image.par  
t.001.  
Uploaded image.part.002 to https://s3.amazonaws.com:443/db2-exprc-64-2/image.par  
t.002.  
Uploaded image.part.003 to https://s3.amazonaws.com:443/db2-exprc-64-2/image.par  
t.003.  
█
```

Once the bundle has been successfully uploaded to Amazon S3 storage, the final step is to register the image by issuing the following command:

```
ec2-register mybucket/image.manifest.xml
```



```
ec2-67-202-19-184.compute-1.amazonaws.com - PuTTY
db2exc64:~ # ec2-register db2-exprc-64-2/image.manifest.xml
IMAGE    ami-f8648091
db2exc64:~ #
```

Your AMI should now be registered in Amazon. To list your AMIs, issue the following command:

```
ec2-describe-images -o self
```

3.9 Publicizing your AMI

By default, AMIs that users create will be private. When an AMI is private, no other AWS user will be able to see your AMI. To publicize an AMI, type:

```
ec2-modify-image-attribute <ami_id> --launch-permission -a all
```

Once the AMI is publicized, all AWS users will be able to see the AMI, and launch instances from their own accounts. Before publicizing your AMI, be sure to secure your AMI by removing all confidential information on your AMI such as certificates and private keys.

4. Additional Information

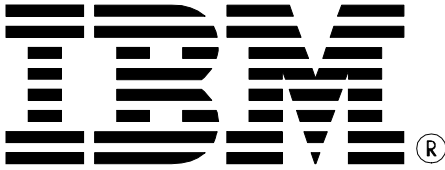
4.1 Feedback

Thank you for using the DB2 AMI.

We welcome your feedback about your experience on the DB2 AMI. If you notice any errors or discrepancies, or would like to comment on your experience on the DB2 AMI, please send your feedback to db2x@ca.ibm.com, with the name of the DB2 AMI in the subject line.

If you are an ISV and would like to engage our team for technical enablement in creating an embedded solution based on the DB2 AMI, please contact us by sending an email to db2x@ca.ibm.com with subject line “Embed DB2 AMI”.

For technical questions on DB2 9.5, please refer to the DB2 Information Center at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/index.jsp>.



© Copyright IBM Corporation 2009
All Rights Reserved.

IBM Canada
8200 Warden Avenue
Markham, ON
L6G 1C7
Canada

Printed in Canada
03/2009

IBM, IBM (logo), DB2 are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Java is a trademark of Sun Microsystems, Inc in the United States, other countries, or both.

Amazon Web Services, AWS, Amazon Elastic Compute Cloud, EC2, Amazon Simple Storage Service, Amazon S3 are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries

Other company, product, or service names may be trademarks or service marks of others.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates. The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

The information in this publication is provided AS IS without warranty. Such information was obtained from publicly available sources, is current as of March 2009, and is subject to change. Any performance data included in the paper was obtained in the specific operating environment and is provided as an illustration. Performance in other operating environments may vary. More specific information about the capabilities of products described should be obtained from the suppliers of those products.