

Oracle Database on Amazon EC2

An Oracle White Paper
May 2009

NOTE:

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle

Oracle Database on Amazon EC2

Why Run Oracle Database on Amazon EC2?	1
Getting Started.....	1
Setting Up Your EC2 Instance.....	2
Creating your Database	6
Moving an existing database to EBS	8
Moving to a new EC2 instance	10
Backup	12
Oracle Secure Backup Cloud module	12
EBS Snapshots	13

Oracle Database on Amazon EC2

WHY RUN ORACLE DATABASE ON AMAZON EC2?

Oracle customers can now run Oracle software on Amazon EC2 using existing or new licenses. For database customers, Oracle provides pre-packaged Amazon Machine Images (AMIs) containing Oracle Enterprise Linux 5 and Oracle Database 11g which are supported by Oracle and [Amazon Premium Support](#). For creating new databases, Oracle provides scripts that will have a database ready for use in just a few minutes.

Backing up databases to Amazon S3 is fast and secure with the Oracle Secure Backup Cloud module. The Cloud module includes special licensing for encryption and compression that allow these features to be used without requiring separate licenses for the Advanced Security and Advanced Compression options.

GETTING STARTED

This guide provides information to help you get started with Oracle Database on Amazon EC2 running Linux. The instructions and examples are provided for introductory and evaluation purposes and do not represent best practices for running Oracle in an enterprise.

Examples provided:

- Creating a new database
- Moving an existing database to EBS
- Moving a database to a new EC2 instance
- Creating Backups

Useful Links

If you're new to AWS, the following links provide an introduction.

[Oracle Cloud Computing Portal](#)

AWS services used in this guide

[Elastic Compute Cloud \(EC2\)](#)

[Elastic Block Store \(EBS\)](#)
[Simple Storage Service \(S3\)](#)

Getting Access to AWS

[Sign up for AWS](#)

[Sign up for EC2](#) Note: EC2 includes EBS - there is no separate sign-up for EBS.

[Sign up for S3](#) Required for the Oracle Secure Backup Cloud Module ([doc](#), [download](#))

[Finding your AWS access identifier, secret key, and X.509 certificate](#)

AWS Service Management Tools

Name	Interface	Manage	Credentials Required	Comments
AWS Console	GUI	Multiple AWS services including EC2	AWS user name and password	Nothing to install – works with multiple browsers
ElasticFox	GUI	EC2	AWS access id and secret key	Requires FireFox and plug-in install
S3Fox	GUI	S3	AWS access id and secret key	Requires FireFox and plug-in install
EC2 API Tools	CLI	EC2	AWS X.509 Certificate consisting of both the public and private key files (the cert-xxx.pem and pk-xxx.pem files, respectively)	Requires Java 1.4 or later and small download and install Maximum flexibility to automate and build tools to suite your needs
S3 API	API	S3	AWS access id and secret key	Support for a variety of programming and scripting languages

SETTING UP YOUR EC2 INSTANCE

Create a Keypair

If you don't already have an EC2 keypair for your Oracle instances, [create one](#) . The Oracle AMI will use the keypair to set the root login credentials. When you create a keypair, EC2 stores the public key and provides you with the private key. Save the private key in a secure file. EC2 will only provide the private key once, so if you lose it, you won't be able to login to your instances as root.

```
ec2-add-keypair OraRoot
KEYPAIR OraRoot
xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx
:xx
-----BEGIN RSA PRIVATE KEY-----
```

```
...
-----END RSA PRIVATE KEY-----
```

Cut and paste the private key (including the BEGIN and END RSA PRIVATE KEY lines) into a secure file (OraRoot.pem in the examples).

Configure Security Groups

Security groups allow you to define firewall rules for your EC2 instances. If you haven't already done so, configure a security group for your Oracle database instances. The specific rules you'll need will depend upon how your network security is managed. At a minimum, you will need ssh access (port 22). In order to access the database via clients external to EC2, you'll need to open a port for the database listener. If you plan to use Oracle Enterprise Manager and/or Application Express, open the ports provided by the configuration script at the end of the installation.

```
ec2-add-group Oracle --description "Oracle database
security group"
GROUP Oracle Oracle database security group
```

```
ec2-authorize Oracle --protocol tcp --port-range 22 --
source-subnet 10.0.0.0/24
GROUP Oracle
PERMISSION Oracle ALLOWS tcp 22
22 FROM CIDR 10.0.0.0/24
```

```
ec2-authorize Oracle --protocol tcp --port-range 1521 --
source-subnet 10.0.0.0/24
GROUP Oracle
PERMISSION Oracle ALLOWS tcp 1521
1521 FROM CIDR 10.0.0.0/24
```

Be sure to specify the source subnet. The default is 0.0.0.0/0, which leaves the port wide open to attack. If you are behind a corporate firewall, or just connecting via your router at home, you can usually determine your source IP address using a trusted website. Some corporate networks use different source addresses depending upon the destination address. If this is the case, contact your network security administrator.

Verify your security group settings:

```
ec2-describe-group Oracle
GROUP 1234567890 Oracle Oracle database security
group
PERMISSION 1234567890 Oracle ALLOWS tcp
22 22 FROM CIDR 10.0.0.0/24
PERMISSION 1234567890 Oracle ALLOWS tcp
1521 1521 FROM CIDR 10.0.0.0/24
```

Select an AMI

Oracle provides public AMIs that make it easy to create a new database on EC2. The AMIs come pre-installed with Oracle Enterprise Linux and Oracle Database software. A script runs automatically to guide you through creating a database.

AMIs are available for all database editions (XE, SE, and EE) with Enterprise Edition available for both 32-bit and 64-bit. Oracle AMIs are easily identified by their owner id: **725966715235**.

```
ec2-describe-images --owner 725966715235 --executable-by
all | grep database-ami
IMAGE ami-02cb2f6b /oracle-corporation/database-
ami/64-bit/oracle_11106_SE_SE1_64Bit-
image.manifest.xml 725966715235 available
public x86_64 machine aki-0d9f7b64 ari-369f7b5f
```

```
IMAGE ami-7acb2f13 oracle-corporation/database-
ami/32-bit/oracle_10g_XE_32Bit-image.manifest.xml
725966715235 available public
i386 machine aki-c5e703ac ari-c2e703ab
```

```
IMAGE ami-cecb2fa7 oracle-corporation/database-
ami/32-bit/oracle_11106_EE_32Bit-image.manifest.xml
725966715235 available public
i386 machine aki-c5e703ac ari-c2e703ab
```

```
IMAGE ami-7ecb2f17 oracle-corporation/database-
ami/64-bit/oracle_11106_EE_64Bit-image.manifest.xml
725966715235 available public
x86_64 machine aki-0d9f7b64 ari-369f7b5f
```

Choose an AMI that meets your needs and select an [instance type](#) compatible with the AMI. The example uses a 64-bit EE AMI and an m1.large instance type. You can specify the availability zone here or let AWS choose it for you. Just be sure to use the same availability zone when creating EBS volumes for this instance. When starting your instance, make sure to supply the name of your SSH keypair using the `--key` parameter, otherwise you will not be able to login to your instance.

```
ec2-run-instances ami-7ecb2f17 --key OraRoot --instance-
type m1.large --group Oracle
RESERVATION r-abcdef123 1234567890 Oracle
INSTANCE i-a99999bc ami-
7ecb2f17 pending OraRoot
0 m1.large 2009-03-
20T19:06:51+0000 us-east-1c aki-0d9f7b64
ari-369f7b5f
```

Create EBS volume

Next, we'll create an EBS volume for the database. Choose a size large enough for your database and other related files such as archived logs, disk backups, etc. For the default database created by the Oracle AMI, 20GB is sufficient. Make sure that you specify the same availability zone that your EC2 instance belongs to. In this example, our EC2 instance is running in availability zone us-east-1c.

```
ec2-create-volume --size 20 --availability-zone us-east-
1c
VOLUME vol-abcd1234 20 us-east-1c
```

```
creating 2009-03-20T20:38:28+0000
```

Create Elastic IP

Elastic IP addresses (EIPs) provide static public (Internet routable) and private (AWS routable) addresses and DNS aliases for EC2 instances. If the instance terminates and is replaced with a new one, the EIP can be transferred to a new instance to allow client applications to connect to the database with the same IP address and DNS alias.

Because the public DNS alias resolves to the internal IP address when used within EC2, it can be used in Oracle Net configuration files (tnsnames.ora, listener.ora, and sqlnet.ora) in lieu of the private DNS alias. This greatly simplifies configuration and maintenance since the same DNS alias can be used for both internal and external clients.

Without EIPs, client applications will be unable to connect to the database after replacing a terminated EC2 instance until the Oracle Net configuration has been updated with the new address.

```
ec2-allocate-address  
ADDRESS 172.16.0.0
```

Verify That the Instance is Running

```
ec2-describe-instances i-a99999bc  
RESERVATION    r-abcdef123      1234567890      Oracle  
INSTANCE       i-a99999bc       ami-7ecb2f17    ec2-10-  
0-0-0.compute-1.amazonaws.com    domU-12-34-56-78-9A-  
BC.compute-1.internal      running OraRoot  
0 m1.large 2009-03-  
20T19:06:51+0000      us-east-1c      aki-0d9f7b64  
ari-369f7
```

Associate the EIP with the Database Instance

```
ec2-associate-address 172.16.0.0 --instance i-a99999bc  
ADDRESS 172.16.0.0 i-a99999bc
```

Verify that the DNS Names have Changed

```
ec2-describe-instances i-a99999bc  
RESERVATION    r-abcdef123      1234567890      Oracle  
INSTANCE       i-a99999bc       ami-7ecb2f17    ec2-172-  
16-0-0.compute-1.amazonaws.com domU-12-34-56-78-9A-  
BC.compute-1.internal      running OraRoot 0  
m1.large 2009-03-20T19:06:51+0000      us-east-  
1c      aki-0d9f7b64      ari-369f7b5f
```

Attach the EBS Volume

Choose a device name that is not already used by the instance. As of this writing, instances have between one and four ephemeral devices (/dev/sdb-e) in addition to the root volume on /dev/sda. Choosing /dev/sdf as the first device name for all instance types will avoid conflicts with ephemeral devices.

```
ec2-attach-volume vol-abcd1234 --instance i-a99999bc --  
device /dev/sdf  
ATTACHMENT    vol-abcd1234      i-a99999bc
```


Connect to the Instance

If this is your first time connecting to an EC2 instance and KeyPairs are still a bit of a mystery, this is where they come into play. Provide ssh with the name of the secure file containing the private key from the KeyPair.

```
ssh -i OraRoot.pem root@ec2-172-16-0-0.compute-1.amazonaws.com
```

(Note for users connecting from Windows: see the [EC2 Getting Started Guide Appendix: PuTTY](#) for information on how to connect using PuTTY)

Accept the Oracle License Agreements

You will be prompted to accept the [Oracle Enterprise Linux license agreement](#) and the [Oracle Technology Developer License Terms](#) . Enter 'Y' to accept.

Create a password for the oracle Linux account

Avoid passwords that are easy to guess, even for a quick evaluation. Hackers can and will break an easy password, particularly if the Security Group CIDR is set to 0.0.0.0/0 (currently the default - this is being addressed). See [Tips for Securing your EC2 Instance](#) for information on securing your systems.

Do not create a database now - respond 'n' to the prompt

When prompted to create a database now, enter 'n'. We need to create a filesystem on the EBS volume before creating a database.

Update the oracle account's .bash_profile with the correct ORACLE_HOSTNAME

During instance boot, /etc/init.d/dbora sets the ORACLE_HOSTNAME variable in .bash_profile for the oracle account. Since the Elastic IP isn't associated with the instance at first launch, ORACLE_HOSTNAME will not have the correct value and will cause problems with the Enterprise Manager and APEX configuration tools. Update ORACLE_HOSTNAME to the public hostname provided by the Elastic IP address.

```
PUBLIC_HOSTNAME=`wget -q -O - http://169.254.169.254/latest/meta-data/public-hostname`  
sed -i "s/ORACLE_HOSTNAME=.*/ORACLE_HOSTNAME=$PUBLIC_HOSTNAME/"  
~oracle/.bash_profile
```

Create an ext3 filesystem on the EBS volume (/dev/sdf).

There is no need to create a partition table on the device (there's no harm in it either, but it adds an extra step).

```
yes | mkfs.ext3 /dev/sdf
```

CREATING YOUR DATABASE

Use the Oracle provided script to create a new database:

- Select /dev/sdf as the disk device for the database and Flash Recovery Area

- Choose /u02 as the mount point
- Enter the database name
- Provide passwords for privileged accounts. Use passwords that meet the default password profile. If you enter passwords that do not meet the minimum password complexity set by the profile, the script will run without error, the database will start, but you will not be able to access the privileged accounts. If this happens, connect to the database as SYS and set the passwords.

```
/home/oracle/scripts/run_dbca.sh
```

```
...
```

You can connect to your database as follows:

```
Command Line: sqlplus sys/<your-password> as sysdba
Enterprise Manager Database Console: https://ec2-172-16-0-0.compute-1.amazonaws.com:<port number>/em e.g.
https://ec2-172-16-0-0.compute-1.amazonaws.com:1158/em
Oracle Application Express (APEX): http://ec2-172-16-0-0.compute-1.amazonaws.com:<port number>/apex e.g.
http://ec2-172-16-0-0.compute-1.amazonaws.com:8080/apex
Actual ports numbers can be found in
/u01/app/oracle/product/11.1.0/db_1/install/portlist.ini
file. If this is the first database being created on
this machine, the above example URLs should work.
```

At this point, the database is up and running, there is a listener on port 1521, and the Oracle Enterprise Manager Database Control and Application Express services are running. From here you can login to the database with any database client. The next few steps use sqlplus to connect to the database locally.

Login as oracle

```
ssh oracle@ec2-172-16-0-0.compute-1.amazonaws.com
```

Set environment variables

```
. oraenv
```

Connect to the database

```
sqlplus "/" as sysdba"
```

Prevent archiving to \$OH/dbs

11.1.0.6 databases only: Oracle's database creation script enables archiving and leaves the log archive destination at the default, which is the Flash Recovery Area. In 11.1.0.6, bug# **6373164** causes archive logs to be written to \$ORACLE_HOME/dbs as well as the FRA, which can quickly fill the filesystem. To prevent this, we'll set log_archive_dest_1 to explicitly set the destination to the FRA. Note that this bug is fixed in 11.1.0.7.

```
alter system set
log_archive_dest_1='LOCATION=USE_DB_RECOVERY_FILE_DEST';
```

Add the OEM and APEX ports to the security group in order to access these services

The database creation script also configures Oracle Enterprise Manager Database Control and Application Express (APEX). If you plan to use these services, you'll need to open the corresponding ports in the firewall.

```
ec2-authorize Oracle --protocol tcp --port-range 1158 --
source-subnet 10.0.0.0/24
```

```
GROUP          Oracle
PERMISSION     Oracle  ALLOWS  tcp      1158
1158 FROM CIDR  10.0.0.0/24
```

```
ec2-authorize Oracle --protocol tcp --port-range 8080 --
source-subnet 10.0.0.0/24
```

```
GROUP          Oracle
PERMISSION     Oracle  ALLOWS  tcp      8080
8080 FROM CIDR  10.0.0.0/24
```

Connect to OEM. Accept the certificate.

```
https://ec2-172-16-0-0.compute-1.amazonaws.com:1158/em
```

Connect to APEX administration

```
http://ec2-172-16-0-0.compute-
1.amazonaws.com:8080/apex/apex_admin
```

Username: admin

Password: <password provided during installation>

Connect APEX workspace

```
http://ec2-172-16-0-0.compute-1.amazonaws.com:8080/apex
```

Problems connecting?

If you are unable to connect to the database via the listener, OEM, or APEX, check your Security Group settings. If the settings are correct, you may be encountering an outbound firewall restriction. One way to work around this is to create ssh tunnels. This may violate your corporate security policy, so check with your network security administrator before attempting this.

MOVING AN EXISTING DATABASE TO EBS

The general strategy is simple: replace each ephemeral drive used by the database with EBS drives. There are many ways to approach this with varying degrees of

complexity and downtime. We'll cover the simplest approach here, assuming the database was created on a single ephemeral drive (/u02).

Create an EBS volume to replace the ephemeral disk

The volume must be in the same availability zone as the database and must be large enough to hold the database, Flash Recovery Area, and any other files currently stored on the ephemeral disk.

Determine the EC2 instance id and availability zone

One way to do this is to examine the instance meta-data from within the instance itself.

```
echo `wget -q -O - http://169.254.169.254/latest/meta-  
data/instance-id`  
i-abcd1234  
echo `wget -q -O - 169.254.169.254/latest/meta-  
data/placement/availability-zone/`  
us-east-1c
```

Determine the minimum size required for the EBS volume

```
df -h /u02  
Filesystem                Size  Used Avail Use% Mounted on  
/dev/sdb                   414G  2.0G  391G   1% /u02
```

Create and attach the volume

```
ec2-create-volume --size 20 --availability-zone us-east-  
1c  
VOLUME  vol-1234abcd    20                us-east-1c  
creating          2009-03-22T07:48:37+0000  
ec2-attach-volume vol-1234abcd --instance i-abcd1234 --  
device /dev/sdf  
ATTACHMENT      vol-1234abcd    i-abcd1234  
/dev/sdf        attaching       2009-03-22T07:51:19+0000
```

Create a filesystem on the EBS volume

```
yes | mkfs.ext3 /dev/sdf
```

Mount the EBS volume

```
mount /dev/sdf /mnt
```

Shutdown the database

```
dbshut
```

Find and stop any other processes that have open files on the source filesystem

```
lsof | grep /u02
```

Copy the files to the EBS volume

```
rsync -aR /u02/. /mnt
```

Replace the ephemeral volume with the EBS volume

```
umount /u02  
umount /mnt
```

```
mount /dev/sdf /u02
```

Change the /etc/fstab entry

```
sed -i 's/\dev\sdb/\dev\sdf/' /etc/fstab
```

Restart the database

```
dbstart
```

MOVING TO A NEW EC2 INSTANCE

This section describes how to move an existing database that is already stored on EBS to a new EC2 instance. This procedure can be used to recover from a terminated EC2 instance or to migrate a database to a different instance type. **It is important to note that any changes made to files stored on the ephemeral disks of the former instance will be lost.** This includes home directories, passwords, Linux kernel parameter settings, packages, audit trails, ORACLE_HOME, etc. Persisting these changes is beyond the scope of this article.

Start an instance of the desired type in the same availability zone as the EBS volumes that will be mounted

```
ec2-run-instances ami-7ecb2f17 --key OraRoot --instance-type m1.large --group Oracle --availability-zone us-east-1c
```

```
RESERVATION    r-abcd1234      1234567890    Oracle
INSTANCE       i-ab1234cd      ami-
7ecb2f17      pending OraRoot
0              m1.large        2009-03-
22T21:01:44+0000 us-east-1c      aki-0d9f7b64
ari-369f7b5f
```

Associate the same Elastic IP address as the former instance

```
ec2-associate-address 172.16.0.0 --instance i-ab1234cd
```

Attach EBS volumes

```
ec2-attach-volume vol-abcd1234 --instance i-ab1234cd --device /dev/sdf
```

Connect to the instance

If the new instance was launched using a public Oracle AMI, accept the license agreement and answer 'n' when prompted to create a database. Since the new instance has the same DNS name and IP address as the old instance, but a different host identifier (RSA key), ssh may refuse to connect. If this happens, edit your ssh known_hosts file to allow ssh to connect (remove the entry for the former instance).

Mount the EBS volumes

Use the same mount points as the former instance.

```
mkdir /u02
mount /dev/sdf /u02
```

Update the oracle account's .bash_profile with the correct ORACLE_HOSTNAME (as root)

```
PUBLIC_HOSTNAME=`wget -q -O -`
```

```
http://169.254.169.254/latest/meta-data/public-hostname`
sed -i
"s/ORACLE_HOSTNAME=.*"/ORACLE_HOSTNAME=${PUBLIC_HOSTNAME}/"
~oracle/.bash_profile
```

Login as oracle and set environment variables

```
export ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1
export ORACLE_SID=db1
```

Add the database to /etc/oratab

```
echo $ORACLE_SID:$ORACLE_HOME:Y >> /etc/oratab
```

Create symbolic links to the spfile and password file

```
ln -s /u02/admin/db1/dbs/orapw${ORACLE_SID}
$ORACLE_HOME/dbs
ln -s /u02/admin/db1/dbs/spfile${ORACLE_SID}.ora
$ORACLE_HOME/dbs
```

Start the database

```
sqlplus "/ as sysdba" <<EOF
startup
exit
EOF
```

Reconfigure Enterprise Manager Database Control

```
export ORACLE_HOSTNAME=`wget -q -O -
http://169.254.169.254/latest/meta-data/public-hostname`
export EMKEY_LOCATION=/u02/admin/${ORACLE_SID}/dbs
emca -config dbcontrol db
```

```
STARTED EMCA at Mar 23, 2009 7:53:14 PM
EM Configuration Assistant, Version 11.1.0.7.0
Production
Copyright (c) 2003, 2005, Oracle. All rights reserved.
```

```
...
***** WARNING
*****
```

```
Management Repository has been placed in secure mode
wherein Enterprise Manager data will be encrypted. The
encryption key has been placed in the file:
/u02/admin/db1/dbs/emkey.ora. Please ensure this file
is backed up as the encrypted data will become unusable
if this file is lost.
```

```
*****
***
Enterprise Manager configuration completed successfully
FINISHED EMCA at Mar 23, 2009 7:56:44 PM
```

BACKUP

Oracle Secure Backup Cloud module

[Oracle Secure Backup Cloud module](#) provides a secure means of backing up databases to Amazon S3. Backups can be done from any database that has network access to S3, not just those in EC2.

The OSB Cloud module provides special licensing not included with a standard Oracle Secure Backup license that allows:

- encrypted backups to S3 without requiring an additional license for the Advanced Security Option and
- fast backup compression without requiring an additional license for the Advanced Compression option

To obtain the Oracle Secure Backup Cloud module, [download the install tool from OTN](#). If you're using one of the Oracle AMIs, the install tool is already available in `/home/oracle/scripts/osbws`.

You'll need the following in order to use the install tool:

- OTN account credentials
- AWS identifier and secret key
- S3 account

Install Oracle Secure Backup Cloud Module

Install the Cloud module on an EBS volume so that the binaries and your configuration settings are retained.

```
OSB_BASE=/u02/admin/osb
OSB_WALLET_DIR=$OSB_BASE/wallet
OSB_CONFIG_DIR=$OSB_BASE/config
OSB_LIB_DIR=$OSB_BASE/lib
```

```
mkdir -p $OSB_WALLET_DIR $OSB_CONFIG_DIR $OSB_LIB_DIR
```

```
java -jar osbws_install.jar -AWSID <AWS ID> -AWSKey <AWS
KEY> -otnUser <OTN Username> -otnPass <OTN password> -
walletDir $OSB_WALLET_DIR -configFile
$OSB_CONFIG_DIR/osbcm.conf -libDir $OSB_LIB_DIR
Oracle Secure Backup Database Web-Service Install Tool
OTN userid is valid.
AWS credentials are valid.
S3 user already registered.
Registration ID: xxxxx
S3 Logging Bucket: xxxxx
Create credential oracle.security.client.connect_string1
OSB web-services wallet created in directory
/u02/admin/osb/wallet.
```

```
OSB web-services initialization file
/u02/admin/osb/config/osbcm.conf created.
Downloading OSB Web Services Software Library.
Downloaded 14669832 bytes in 5 seconds. Transfer rate
was 2933966 bytes/second.
Download complete.
Extracted file /u02/admin/osb/lib/libosbws11.so
```

Configure RMAN settings

Use the following RMAN command to configure the SBT channel to use Oracle Secure Backup Cloud Module.

```
CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' PARMS
'SBT_LIBRARY=/u02/admin/osb/lib/libosbws11.so
ENV=(OSB_WS_PFILE=/u02/admin/osb/config/osbcm.conf)';
```

Encrypting Backups

Encrypting backups to S3 is strongly encouraged. Refer to the [Oracle Database Backup and Recovery User's Guide](#) for information on configuring your system for backup encryption.

EBS Snapshots

In configurations similar to the one used in this guide, where database files are stored in filesystems and the filesystems are built directly on an EBS volume (i.e. no LVM, RAID, or ASM), backing up the database with EBS snapshots is as simple as:

- Put the database in hot backup mode
- Take EBS snapshot of each volume
- End hot backup mode

```
alter database begin backup
ec2-create-snapshot vol-abcd1234
alter database end backup
```

These steps can be automated using a script on the database host or on a remote host. In either case, both the database credentials and the EC2 credentials are required. Storing EC2 credentials on EC2 instances is discouraged for security reasons, so configuring the database for remote SYSOPER access and running the script on a remote host is recommended.

Note that this method will NOT work with LVM, RAID (md), or ASM.

There are ways to use EBS snapshots with them, but they are beyond the scope of this guide.



Oracle Database on Amazon EC2

May 2009

Author: John Smiley (Amazon), Bill Hodak (Oracle)

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

oracle.com

Copyright © 2007, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

Other names may be trademarks of their respective owners.