

# Swarm Logic

## Red Hat Enterprise Linux MySQL Build Standards

Prepared By



**SWARM LOGIC**

INFORMATION SECURITY EXPERTS

# Security Classification - Public

## Notice:

Copyright **Global Information Security Group P/L trading as Swarm Logic** 2010

This work is licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

# Table of Contents

<b>1.0</b>	<b>DOCUMENT ADMINISTRATION .....</b>	<b>4</b>
<b>2.0</b>	<b>INTRODUCTION.....</b>	<b>5</b>
2.1	OVERVIEW .....	5
2.2	STATEMENT OF WARRANTY.....	5
2.3	FEEDBACK .....	5
<b>3.0</b>	<b>OPERATING SYSTEM INSTALLATION.....</b>	<b>6</b>
3.1	CREATE THE VIRTUAL MACHINE .....	6
3.2	CHANGE PASSWORDS .....	6
3.3	CONFIGURE FQDN .....	6
3.4	CONFIGURE RHN .....	7
3.5	PATCH SERVER.....	7
3.6	CONFIGURE ADDITIONAL HDD .....	7
3.7	INSTALL MYSQL.....	7
3.8	CONFIGURE FIREWALL .....	8
<b>4.0</b>	<b>APPENDICES.....</b>	<b>9</b>
4.1	MY.CONF .....	9
4.2	IPTABLES.....	9

# 1.0 DOCUMENT ADMINISTRATION

## Document History

Version	Name	Changes	Date
1.0	Kurt Heinrich	Original	20/09/10

## Distribution List

Version	Name	Issued To	Date

## 2.0 INTRODUCTION

### 2.1 OVERVIEW

This document is provided as a courtesy to the community. It is fairly succinct and assumes you are familiar with Vmware, Linux and template style builds.

It provides details of how to create a reasonably secure MySQL based database server

It is considered suitable for hosting services classified "Commercial in Confidence" or similar

It assumes you have a Vmware template based on our RHEL Vmware Template and all the assumptions in that document/build apply here.

It is assumed that the technical resource using this build guide to (re)construct the server is familiar with Linux installation procedures, is familiar with most common Linux administration tasks and has a good working knowledge of TCP/IP and network routing. As such detailed technical particulars regarding the reasoning for each step of the build procedure have not been included.

**Hardware:** Vmware Virtual Machine

**Service Tag:** N/A

**CPU:** 1

**RAM:** 2 GB Ram

**Disk:** 10 GB

**Network:** 1 x Gigabit Ethernet

**O/S:** RHEL 5.4 X64

#### **Software Requirements**

Red Hat Enterprise Linux 5.4 X64

This template will have ssh and mysqld as the only externally accessible service.

### 2.2 STATEMENT OF WARRANTY

THESE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT

### 2.3 FEEDBACK

We welcome your feedback at [info@swarm-logic.com](mailto:info@swarm-logic.com)

## **3.0 OPERATING SYSTEM INSTALLATION**

### **3.1 CREATE THE VIRTUAL MACHINE**

- Open virtual centre
- Select the VM and templates view
- Locate the Redhat VMware Template and select deploy VM from this template
- Set the name to as appropriate (should be hostname not FQDN, eg host not host.example.com)
- Set the data centre as appropriate, click next
- Select the cluster as appropriate and click next
- Set the datastore as appropriate
- Select thin provisioned, click next
- Select customise using an existing specification
- Select RHEL\_54\_x64\_V1.0 or whatever you have named it, click next
- Set the NIC IP as appropriate, click next
- Select edit virtual hardware and power on after creation, click continue
- Select add, hard disk, click next
- Select create a new virtual disk
- Set the size to 10GB – this assumes an appropriate size based on the amount of content you are hosting
- Enable thin provisioning
- Click next
- Click next
- Click finish
- Click OK
- NB sometimes too many changes to the hardware cause a failure while copying the template, if this occurs make the hardware changes one at a time while the server is offline. (e.g. change Nic/VLAN, then add additional HDD, etc)

### **3.2 CHANGE PASSWORDS**

- Change the password for admin via `passwd admin`
- Change the password for root via `passwd root`

### **3.3 CONFIGURE FQDN**

- `Vi /etc/sysconfig/network`
- After the hostname append your domain suffix with a `.` between. Eg it should state `host.example.com`
- `reboot`

### 3.4 **CONFIGURE RHN**

- rhn\_register
- click next
- enter the username as appropriate
- enter the password as appropriate
- leave the profile as default, click next
- click next
- click next
- click next
- click OK once registered to confirm subscription
- click finish

### 3.5 **PATCH SERVER**

- yum update
- review all patches and install
- NB when rebooting if the VMware tools fail re run vmware-config-tools.pl to recompile. This is typical when kernel is upgraded

### 3.6 **CONFIGURE ADDITIONAL HDD**

- Fdisk /dev/sdb
- n to add a new partition
- select p (primary)
- select 1 (1<sup>st</sup> partition)
- set the start cylinder to 1
- set the last cylinder as default (whole disk)
- w to write partition table
- mkfs -V -t ext3 /dev/sdb1
- mkdir /data
- e2label /dev/sdb1 /data
- vi /etc/fstab and add the following line  
LABEL=/data            /data            ext3  
defaults,nosuid,nodev,noexec    1 2

- The next time the server is rebooted the partition should be mounted under /data
- Reboot the server

NB some people prefer a different mount point, we use /data as we know its customer/user specific data

### 3.7 **INSTALL MYSQL**

- Yum install mysql mysql-server

- Allow dependencies to install
- `chkconfig --level 3 mysqld on`
- `service mysqld start`
- `/usr/bin/mysql_secure_installation`
- The root password is empty by default, press enter
- Set the root password as appropriate
- Remove anonymous users – y
- Disallow remote root login – n
- Remove test database – y
- Reload privilege tables – y
- `Service mysqld stop`
- `mkdir /data/mysql`
- `chmod 755 /data`
- `cp -R -p /var/lib/mysql/* /data/mysql`
- `vi /etc/my.cnf`
- change the following line `datadir=/var/lib/mysql`
- to
- `datadir=/data/mysql`
- `service mysqld start`
- Once access is confirmed and you can list the databases remove the old source files from `/var/lib/mysql`

### **3.8 CONFIGURE FIREWALL**

- `Vi /etc/sysconfig/iptables`
- Edit the file as per the appendix

## 4.0 APPENDICES

### 4.1 *MY.CONF*

```
[mysqld]
#datadir=/var/lib/mysql
datadir=/data/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Default to using old password format for compatibility with mysql 3.x
# clients (those using the mysqlclient10 compatibility package).
old_passwords=1

# Disabling symbolic-links is recommended to prevent assorted security risks;
# to do so, uncomment this line:
# symbolic-links=0

[mysqld_safe]
log-error=/var/log/mysql.log
pid-file=/var/run/mysql/mysql.pid
```

### 4.2 *IPTABLES*

```
# Firewall configuration written by system-config-securitylevel
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A FORWARD -j RH-Firewall-1-INPUT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp --icmp-type any -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state ESTABLISHED,RELATED -j
ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j
ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j
ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
```