



---

**Project:** Build Notes...  
a S.E.E.R. Server for Industrial Deployment

**Versions:** S.E.E.R. - SVN\_SNAPSHOT 2012\_0705 (v. 1.5.2-a | build 26)  
mod\_openopc - v. 3.2-3 (build 61)  
STaRBUC - v. 0.2-7 (build 8)

**Date:** 2012\_0703

**Platform:** Red Hat Enterprise Linux (version 6.x)



---

#### LINKED PROJETS

| <i>Project Name</i> | <i>http://</i>          | <i>Contact</i>   | <i>License</i>      |
|---------------------|-------------------------|------------------|---------------------|
| OpenOPC for Python  | openopc.sourceforge.net | Barry Barnreiter | GNU GPL (Perpetual) |
| mod_openopc         | spinellicreations.com   | Vince Spinelli   | GNU GPL v3          |
| S.E.E.R.            | spinellicreations.com   | Vince Spinelli   | GNU AGPL v3         |

---

#### REGARDING PLATFORM

RHEL is the preferred platform for deployment, although any Linux flavor will suffice nicely. Actually, you don't even need to run Linux to run a S.E.E.R. Server – it'll perform as intended under Windows, BSD, or a true UNIX. That said, the it runs most effectively, and efficiently (and is easiest to deploy) under the Linux flavor of your choice.

That said, our favorite flavor is RHEL. Any of the 5 series and up should work nicely. Servers have been deployed successfully on 5.3 and 5.4, and now 6.0... your mileage may vary given the version, however the 'nuts-and-bolts' of deployment are the same, regardless.

#### AUDIENCE and DISCLAIMER OF ANY and ALL LIABILITY

This guide is intended for experienced Linux SysAdmins and is offered 'as is' without any warranty (stated or implied), guarantee of fitness for any given purpose, or promise of efficacy.

It is offered simply as a record of the steps necessary to deploy such a system, in the hopes that it may prove useful to others seeking to do the same.

## **HARDWARE UTILIZED**

This particular build utilizes an HP DL-180 G6 2U rackmount server, equipped as follows:

- 2x 300GB SAS 15k HDD (Raid 1 Mirror; OS)
- 2x 300GB SAS 15k HDD (Raid 1 Mirror; DB and Active Storage)
- 2x 1 TB SATA II 7.2k HDD (Raid 1 Mirror; Backup and Archive)
- 2x Intel Xeon e5506 Quad-Core 2.13 GHz CPU (8 cores total) non-HT
- 32 GB DDR3-800 MHz ECC Memory
- 750W redundant PSU
- HP p410i Smart Array Hard-RAID Controller (PCI-Express)
- HP p212/Zero Smart Array External SAS Controller (PCI-Express)

Dedicated LT0-4 HP Ultrium Backup Tape Drive

- connected to p212/Zero via External SAS Cable

CyberPower PR1500 1500VA Dedicated Backup UPS / Battery System (pure sine wave out)

- connected to 9-pin D-SUB serial port on DL-180 G6
- requires dedicated 15A 120 Vac circuit
- provides power to the DL-180 G6 and the tape backup drive

I/O

- miscellaneous generic USB keyboard and mouse
- gigabit ethernet drop

## **PARTITIONING AND ALLOCATING SPACE**

From hence forth, we'll refer to a RAID-mirrored pair as if it were a single drive. So, for example, the OS drive mirror of two disks will simply be referred to as "the OS drive".

| DRIVE   | PARTITION | MOUNT POINT  | SIZE   | FILESYSTEM |
|---------|-----------|--------------|--------|------------|
| OS      | /dev/sda1 | /boot        | 200 MB | EXT3       |
| OS      | /dev/sda2 | /            | 280 GB | EXT3       |
| OS      | /dev/sda3 | --none--     | 16 GB  | SWAP       |
| STORAGE | /dev/sdb1 | /mnt/storage | 300 GB | Irix XFS   |
| BACKUP  | /dev/sdc1 | /mnt/backup  | 1 TB   | Irix XFS   |

It is important to choose a speedy and easily managed filesystem for your STORAGE and BACKUP drives. We chose XFS by Irix, however you may choose ReiserFS, JFS, or ZFS. We advise against EXT2 or EXT3 filesystems.

For your OS drive, EXT3 is very resilient to the abuse an operating system and running software will inevitably inflict upon a host drive, and it is our recommendation (the emerging EXT4 notwithstanding – as it is too 'young' at

this point for us to make a judgement on it).

## **A VERY BASIC LINUX OS INSTALL**

Contrary to what you may think at first, we started with a very light-weight OS install.

*Selecting our time zone as an offset from GMT (rather than a timezone observing daylight savings time) will prevent any problems with record duplication when time-shifting points come along.*

You will not need any media applications, animation effects / desktop effects, or the like; nor will you need QT, perl, or many of the other offered goodies.

*Note – some persons consider a disc-burning application (such as Roxio / K3B / or Brasero) to be a 'media' app. In this regard, you should install one of these (provided your server has a cd/and-or/dvd burner).*

Keep an eye out for the Vinagre VNC Viewer app, all of the compatability libraries, development libraries (these are all of the [some-app]-devel.rpm files), and development tools (such as the various compilers). You will need the bulk of these, and those you don't need 'now' are good to have in the 'future'.

We prefer, and therefore went with the Gnome desktop environment. This will eventually be run as a detached desktop (with the 'vncserver' command). Although, in reality, you don't need any desktop at all. A fully configured S.E.E.R. Server should boot into run level '3' (terminal prompt only), and need no attention whatsoever. However, that's after you've set it up – the process of configuring the machine is infinitely easier if you have the GUI available to make all of your file editing and setup easier. Of course, once you're done, you can just set it up to enter run level '3', without launching the vncserver remote desktop, thereby eliminating the GUI.

Once you're up and running (and installation has completed), you can log into the GUI as root (yes, it will likely bark at you, that this is a bad idea – don't worry about it, you're not making a habit of this), and get to the matter of setting up the basics...

### **MOUNTS**

Create directories /mnt/storage and /mnt/backup and chmod permissions to 777.

Edit the file /etc/fstab as follows, in order to auto-mount the STORAGE drive and BACKUP drive to their respective locations.

```
# OS DRIVE
# -----
/dev/sda1  /boot          ext3 defaults 1 2
/dev/sda2  /              ext3 defaults 1 1
/dev/sda3  swap           swap defaults 0 0
tmpfs      /dev/shm       tmpfs defaults 0 0
devpts     /dev/pts       gid=5,mode=620 0 0
sysfs      /sys           sysfs defaults 0 0
```

```

proc          /proc          proc defaults 0 0
#
# STORAGE DRIVE
# -----
/dev/sdb1     /mnt/storage    xfs  defaults 1 1
#
# BACKUP DRIVE
# -----
/dev/sdc1     /mnt/backup      xfs  defaults 1 1
#

```

### SELINUX

It is advised to disable SELinux by editing the config file,  
/etc/selinux/config ...

```

SELINUX=disabled
SELINUXTYPE=targeted

```

Also, temporarily shut down the firewall until we set everything else up and tell you which ports to turn on!

### NETWORKING

Edit the startup file for your ethernet adapter as you wish, and keep in mind this should be a static IP address that you're assigning.

A quick-and-dirty alternative is to shut off auto-connect in the GUI's network connection manager, set the NetworkManager service not to start (command = /sbin/chkconfig --level 12345 NetworkManager off), and then add the following lines to your /etc/rc.d/rc.local file (preferably as the first command added)...

```

ifconfig eth0 down
ifconfig eth0 192.168.1.100 netmask 255.255.255.0 up
route add default gw 192.168.1.1 eth0
ifconfig lo up (may be unnecessary)

```

Additionally, should you have trouble with large numbers of very small packets of information, disabling large frame offload to the network card's controller should resolve this. Simply add the line...

```
ethtool -K eth0 tso off
```

(requires program 'ethtool', may have to install via yum first, then enable.)

Add DNS Nameservers by editing the file /etc/resolv.conf as follows...

```

# NAMESERVERS
nameserver 192.168.1.1
nameserver 192.168.1.25

```

... where nameservers are listed in order of preference.

### YUM REPOSITORIES

You should have, in the least, your distributions update / install repository

(in our case it is the Fermi Scientific Linux [RHEL Clone] sl.repo and sl-updates.repo... for CentOS, it'll be centos.repo, centos-updates.repo, etc...), along with the following...

- AT Rpms (atrpms.repo)
- Fedora Extra Packages for Enterprise Linux (epel.repo)
- RPM Forge (rpmforge.repo)
- RPM Fusion, formerly Livna (rpmfusion.repo)
- Enterprise Linux (elrepo.repo)

### SOME BASIC TASKS / UTILITIES

Via yum, install the following...

- update kernel, kernel-headers, and kernel-devel (reboot is required afterward).
- Add some useful utilities:
  - gftp
  - firefox
  - iftop
  - ethtool
  - sendmail
  - sendmail-cf
  - sendmail-milter
  - clamav
  - cups
  - cups-pdf
  - mt-st
  - k3b (if you have a cd or dvd disc burner)
  - lmsensors
  - gparted
  - vinagre (or some other vnc-viewer)
  - vnc-server

Note: the running kernel we achieved after update was 2.6.32-71.29.1 (Red Hat vanilla).

At this point, you're ready to start gathering and installing the rest of the system as needed.

### ADD REQUIRED STANDARD PACKAGES

Install the following via yum...

- tar (GNU tar) and / or STAR (star by Jorg Schilling)
- we installed both, but specifically use STAR for integration with StaRBUC.
- httpd
- httpd-devel
- php
- php-cli
- php-common
- php-dba

- php-devel
- php-eaccelerator
- php-gd
- php-mssql (only needed if using bartender-plugin compatability)
- php-mysql
- php-odbc
- php-oci8
- php-pdo
- php-pgsql
- php-zts
- sqlite
- mysql
- mysql-devel
- mysql-server

### **ADD REQUIRED NON-STANDARD PACKAGES**

Add the following non-standard packages via yum call direct to rpm on local disk...

- VirtualBox 4.0 (also installs...)
  - mesa-libGLU
  - phonon-backend-gstreamer
  - qt
  - qt-sqlite
  - qt-x11
- CyberPower Power Panel Plus UPS Management Tool 1.1.3-0

Add the following non-standard packages compiled from source on local disk...

- lzo compression library 2.03 (default configure)
- openvpn 2.1\_rc15 (default configure)

### **PYTHON**

RHEL 6.0 and clones ship with python 2.6.5, which is great, as we'll require python 2.6.x or greater. So...

Install via yum...

- python-devel

Install the following non-standard packages compiled from source on local disk...

- setproctitle 1.0 (python ./setup.py build [install])
- pyro 3.8-1 (python ./setup.py build [install])
- mysql-python 1.2-3.c1 (python ./setup.py build [install])

### **CONFIGURE THE UNDERLYING SYSTEM**

So now it's time to go ahead and configure all of the underlying OS jobs and properties. This is all going to depend on how -YOU- want to administer and run your system, so there really isn't any set 'rule' here. These include...

- cron jobs
- system logs / logging method / etc.
- ups backup interaction
- - upon low battery or power event it MUST call for a system shutdown.
- grub bootloader
- - filename = SAMPLE\_boot-grub-grub.conf
- system startup processes
  - anything that we start up later, manually, should not be included here. For example, a sample output of /sbin/chkconfig is included here... specifically, look at level '3', as the system which spawned this printout runs in runlevel '3' only.

```
[root@wheeljack MySQL-python-1.2.3c1]# /sbin/chkconfig --list
NetworkManager      0:off 1:off 2:off 3:off 4:off 5:off 6:off
acpid                0:off 1:off 2:on  3:on  4:on  5:on  6:off
atd                  0:off 1:off 2:off 3:on  4:on  5:on  6:off
auditd              0:off 1:off 2:on  3:on  4:on  5:off 6:off
autofs              0:off 1:off 2:off 3:on  4:on  5:on  6:off
avahi-daemon         0:off 1:off 2:off 3:on  4:on  5:on  6:off
cgconfig             0:off 1:off 2:off 3:off 4:off 5:off 6:off
cgred                0:off 1:off 2:off 3:off 4:off 5:off 6:off
clamav               0:off 1:off 2:off 3:off 4:off 5:off 6:off
crond                0:off 1:off 2:on  3:on  4:on  5:on  6:off
cups                 0:off 1:off 2:on  3:off 4:off 5:off 6:off
dnsmasq              0:off 1:off 2:off 3:off 4:off 5:off 6:off
firstboot            0:off 1:off 2:off 3:off 4:off 5:off 6:off
haldaemon            0:off 1:off 2:off 3:on  4:on  5:on  6:off
httpd                0:off 1:off 2:off 3:off 4:off 5:off 6:off
ip6tables            0:off 1:off 2:on  3:off 4:off 5:off 6:off
iptables             0:off 1:off 2:on  3:on  4:on  5:on  6:off
irqbalance           0:off 1:off 2:off 3:on  4:on  5:on  6:off
kdump                0:off 1:off 2:off 3:off 4:off 5:off 6:off
lm_sensors            0:off 1:off 2:off 3:on  4:on  5:on  6:off
mdmonitor            0:off 1:off 2:on  3:off 4:off 5:off 6:off
messagebus           0:off 1:off 2:on  3:on  4:on  5:on  6:off
microcode_ctl        0:off 1:off 2:on  3:on  4:on  5:on  6:off
mysqld                0:off 1:off 2:off 3:off 4:off 5:off 6:off
netconsole           0:off 1:off 2:off 3:off 4:off 5:off 6:off
netfs                 0:off 1:off 2:off 3:on  4:on  5:on  6:off
network              0:off 1:off 2:on  3:on  4:on  5:on  6:off
nfs                   0:off 1:off 2:off 3:off 4:off 5:off 6:off
nfslock              0:off 1:off 2:off 3:on  4:on  5:on  6:off
ntpd                  0:off 1:off 2:on  3:off 4:off 5:off 6:off
ntpdate              0:off 1:off 2:off 3:off 4:off 5:off 6:off
portreserve          0:off 1:off 2:on  3:on  4:on  5:on  6:off
psacct               0:off 1:off 2:off 3:off 4:off 5:off 6:off
rdisc                0:off 1:off 2:off 3:off 4:off 5:off 6:off
restorecond          0:off 1:off 2:off 3:off 4:off 5:off 6:off
rpcbind              0:off 1:off 2:on  3:on  4:on  5:on  6:off
rpcgssd              0:off 1:off 2:off 3:on  4:on  5:on  6:off
rpcidmapd            0:off 1:off 2:off 3:on  4:on  5:on  6:off
rpcsvcgssd           0:off 1:off 2:off 3:off 4:off 5:off 6:off
rsyslog              0:off 1:off 2:on  3:on  4:on  5:on  6:off
saslauthd            0:off 1:off 2:off 3:off 4:off 5:off 6:off
sendmail             0:off 1:off 2:on  3:off 4:off 5:off 6:off
smartd               0:off 1:off 2:off 3:off 4:off 5:off 6:off
snmpd                0:off 1:off 2:off 3:off 4:off 5:off 6:off
snmptrapd            0:off 1:off 2:off 3:off 4:off 5:off 6:off
sshd                  0:off 1:off 2:on  3:on  4:on  5:on  6:off
stinit               0:off 1:off 2:off 3:off 4:off 5:off 6:off
sysstat              0:off 1:on  2:on  3:on  4:on  5:on  6:off
udev-post            0:off 1:on  2:on  3:on  4:on  5:on  6:off
vboxballoonctrl-service 0:off 1:off 2:on  3:on  4:on  5:on  6:off
```

|                 |       |       |       |       |       |       |       |
|-----------------|-------|-------|-------|-------|-------|-------|-------|
| vboxdrv         | 0:off | 1:off | 2:on  | 3:on  | 4:on  | 5:on  | 6:off |
| vboxweb-service | 0:off | 1:off | 2:on  | 3:on  | 4:on  | 5:on  | 6:off |
| vncserver       | 0:off | 1:off | 2:off | 3:off | 4:off | 5:off | 6:off |
| wpa_supplicant  | 0:off | 1:off | 2:off | 3:off | 4:off | 5:off | 6:off |
| xinetd          | 0:off | 1:off | 2:off | 3:on  | 4:on  | 5:on  | 6:off |

- clamav
- - enable milter for sendmail
- - filename = SAMPLE\_etc-clamd.conf
- - filename = SAMPLE\_etc-clamav-milter.conf
- - filename = SAMPLE\_etc-freshclam.conf
- /etc/hosts
- - filename = SAMPLE\_etc-hosts
- /etc/inittab
- - filename = SAMPLE\_etc-inittab
- - if you want to force runlevel 3, here's the place to do that.
- sendmail and /etc/mail/access
  - remember to enable the milter from clamav, and do we suggest only allowing mail to be relayed from the localhost (127.0.0.1). Typically, you'll push out to the real world through your company's inside-wire-friendly mail-server. (SMTP-Relay)
  - filename = SAMPLE\_etc-mail-access
  - filename = SAMPLE\_etc-mail-sendmail.mc
- ssh and sshd config
- firewall
- - open port 22 TCP and UDP, open ports 25, 80, and 5900 TCP.

## **CONFIGURE SPECIFIC PACKAGES / APPS**

### **CONFIGURE PHP**

You must edit the file /etc/php.ini – attached is a sample of an optimized file for a S.E.E.R. Server...

- filename = SAMPLE\_etc-php.ini

### **CONFIGURE MYSQL**

Edit file /etc/my.cnf – attached is a sample of a highly optimized file for a S.E.E.R. Server, running InnoDB tables...

- filename = SAMPLE\_etc-my.cnf\_ini
- execute command...
- - /usr/bin/mysqladmin -u root password 'your-root-pw'
- - /usr/bin/mysqladmin\_secure\_installation
- - - enter a new password
- - - delete temporary tables
- - - lockdown root access to localhost only.
- - /usr/bin/mysql -u root -p[your-root-pw]
- - - mysql> CREATE USER 'mysql'@'localhost' IDENTIFIED BY 'mysqlpassword';
- - - mysql> GRANT ALL ON \*.\* TO 'mysql'@'localhost';



- - - mysql> FLUSH PRIVILEGES;

### CONFIGURE APACHE

You must edit the file /etc/httpd/conf/httpd.conf, and assuming you've followed the instructions thus far, the provided sample file is an great example of a solid web-server deployment anticipating load.

- filename = SAMPLE\_etc-httpd-conf-httpd.conf

### CONFIGURE LOGGING

The provided scripts should be placed into /opt/syslog, yielding...

- /opt/syslog/log\_virus.sh  
- /opt/syslog/log\_wrapper.sh  
- /opt/syslog/logwatch\_daily.sh

Note that ClamAV calls log\_virus.sh in order to report detections.

```
>> log_virus
>> set 'SERVER', and 'OUTPUTDIR' to suit your needs.

#!/bin/bash
# virus ID
#
SERVER=wheeljack
#
DATESTAMP=`/bin/date +%Y_%m%d_%H:%M:%S`
NAME=virus_detected
OUTPUTDIR=/mnt/backup/server_logs/${SERVER}
FILENAME=${OUTPUTDIR}/${DATESTAMP}_${NAME}.log
#
echo ""
echo "VIRUS DETECTED!"
echo ""
echo "... clamav or the milter has found an infected"
echo "... file or message."
echo ""
#
echo "${SERVER} -- clamav or the milter has found an infected file or
                                     message at ${DATESTAMP}" >> ${FILENAME}
#
#

>> log_wrapper
>> no configuration needed.

>> logwatch_daily
>> no configuration needed (unless command logwatch is in different location).
```

### CONFIGURE POWER FAIL

Regardless of what UPS / Battery backup you use, you should configure the UPS monitoring daemon (in our case Cyberpower Power Panel) to trigger one of two events depending on state: power\_fail.sh or battery\_fail.sh – these scripts are provided for you also.

>> power\_fail.sh

>> edit 'SERVER' and 'OUTPUTDIR' to suit your needs.

```
#!/bin/bash
# power failure identification

SERVER=wheeljack
#
DATESTAMP=`/bin/date +%Y_%m%d_%H:%M:%S`
NAME=power_failure
OUTPUTDIR=/mnt/backup/server_logs/${SERVER}
FILENAME=${OUTPUTDIR}/${DATESTAMP}_${NAME}.log
#
echo ""
echo "POWER FAILURE!"
echo ""
echo "... you're running on batteries for now."
echo "... if you don't shut down manually, the"
echo "... system will auto-shutdown when battery"
echo "... capacity has reached low setpoint."
echo ""
#
echo "${SERVER} backup UPS suffered power failure at ${DATESTAMP}" >> ${FILENAME}
#
```

>> battery\_fail.sh

>> edit 'SERVER' and 'OUTPUTDIR' to suit your needs.  
>> also edit 'MANAGEDSHUTDOWN' to suit your needs, typically this  
will be the path to the custom 'shutdown' script, assuming you  
are not using a normal system shutdown (see next)...

```
#!/bin/bash
# battery failure identification
#
SERVER=wheeljack
#
DATESTAMP=`/bin/date +%Y_%m%d_%H:%M:%S`
NAME=battery_low-fail
OUTPUTDIR=/mnt/backup/server_logs/${SERVER}
FILENAME=${OUTPUTDIR}/${DATESTAMP}_${NAME}.log
MANAGEDSHUTDOWN=/opt/shutdown/shutdown.sh
#
echo ""
echo "BATTERY IS LOW / FAILURE!"
echo ""
echo "... you've been on batteries for too long."
echo "... you've reached the low threshold of"
echo "... capacity and the system is shutting down."
echo "... the system will stay down until manually"
echo "... restarted."
echo ""
#
echo "${SERVER} backup UPS suffered low battery or battery failure at
${DATESTAMP}" >> ${FILENAME}
#
# FORK AND EXECUTE MANAGED SHUTDOWN
#
$MANAGEDSHUTDOWN &
#
```

### CONFIGURE SHUTDOWN (CUSTOM)

Due to the manner in which the system functions, it is essential to shut down key items in a timely, and most importantly, in a specifically ordered fashion.

To help you accomplish this, we've attached /opt/shutdown/shutdown.sh, which you may tweak to your needs.

```

>> shutdown.sh
>> edit 'SERVER', and 'OUTPUTDIR' to suit your needs.
>> review this file in depth prior to use.

#!/bin/bash
# shutdown identification
#
SERVER=wheeljack
#
DATESTAMP=`/bin/date +%Y_%m%d_%H:%M:%S`
NAME=shutdown
OUTPUTDIR=/mnt/backup/server_logs/${SERVER}
FILENAME=${OUTPUTDIR}/${DATESTAMP}_${NAME}.log
PKILL=/usr/bin/pkill
SHUTDOWN=/sbin/shutdown
SLEEP=/bin/sleep
TAR=/bin/tar
# depending on whether you use GNU tar or STAR for backup, this
# will be either the executable for 'tar' or 'star'.
MT=/bin/mt
#
echo ""
echo "SYSTEM SHUTDOWN REQUESTED"
echo ""
echo "... killing all threads,"
echo "... and shutting down the system."
echo ""
#
echo "${SERVER} SHUTDOWN was requested at ${DATESTAMP}" >> ${FILENAME}
#
# TAKE DOWN PROCESSES BEFORE SHUTDOWN THAT DO NOT STOP ON THEIR OWN
#
# STaRBUC
# -----
echo ""
echo "KILL STaRBUC OPS"
echo ""
$PKILL $MT
$PKILL $STAR
$PKILL $MT
#
# syphon
# -----
echo ""
echo "KILL SYPHON OPS"
echo ""
$PKILL syphon
#
# mod_openopc
# -----
echo ""
echo "KILL MOD_OPENOPC OPS"
echo ""
$PKILL mod_openopc
#
# Sun Virtual Box

```

```

# -----
echo ""
echo "GRACEFULLY BRING DOWN VIRTUAL MACHINES"
echo ""
/usr/bin/VBoxManage controlvm "ironhide" acpipowerbutton
echo ""
echo "WAITING 4 MINUTES TO ALLOW VIRTUAL MACHINES TO EXIT"
$SLEEP 240
#
# MYSQL SERVER
# -----
echo ""
echo "KILL MYSQL_INNODB_BACKUP OPS"
echo ""
$PKILL mysql_innodb_backup
echo ""
echo "GRACEFULLY STOP MYSQL DB SERVER"
echo ""
/etc/init.d/mysqld stop
#
#
# HAND OFF TO SYS SHUTDOWN
#
echo ""
echo "PROCEED WITH STANDARD SYSTEM SHUTDOWN FROM HERE..."
echo ""
$SHUTDOWN -h now
#

```

NOTE - also included is a sample config for the CyberPower Power Panel application. It is called "SAMPLE\_etc-pwrstatd.conf", and is used to replace /etc/pwrstatd.conf, for controlling the /etc/init.d/pwrstatd daemon. (( this is vendor specific, your UPS vendor can assist you in writing a config file suitable for your UPS's monitoring daemon )).

### CONFIGURE STaRBUC

STaRBUC comes with its own instructions, which you should follow. Points of note are as follows...

- README: README\_starbuc\_v-0-2-4.txt

Extract to [path-to-apache-web-root]/starbuc

For example... /opt/apache\_pages/starbuc

- edit ./starbuc/config/globaloptions\_starbuc.php per instructions.
- WARNING !!! see section titled "Dealing With Heavy Disk I/O", at the end of this document before setting up STaRBUC, as disk dumps to tape drives consume massive amounts of system resources.

### MOD\_OPENOPC

mod\_openopc comes with its own instructions, which you should follow. Items of note are as follows...

- README-1: README\_mod\_openopc\_doc-guide\_2011-05-03.pdf
- README-2: README\_openopc\_doc-guide\_v-1-1-6.pdf
- none.

#### THIS COULD TAKE ALL NIGHT

tctan comes with its own instructions, which you should follow. Items of note are as follows...

- README: README\_this\_could\_take\_all\_night\_v-0-2.txt

Extract to /opt/this\_could\_take\_all\_night

- none.

#### FIELDGATE DM3 REPORTER

fg\_dm3\_reporter comes with its own instructions, which you should follow. Items of note are as follows...

- README: README\_fieldgate\_dm3\_reporter\_v-0-3.txt

Extract to /opt/fieldgate\_dm3\_reporter

- none.

#### MYSQL INNODB BACKUP

mysql\_innodb\_backup comes with its own instructions, which you should follow. Items of note are as follows...

- README: README\_mysql\_innodb\_backup\_v-0-2-1.txt

Extract to /opt/mysql\_innodb\_backup

- REVISIONCOUNT - typically '3' to '5' is a sufficient value. This is, of course, the number of concurrent copies to keep on file of the backup. So '3' keeps the last 3 days (rolling), for example.
- DATABASES - for a S.E.E.R. Server, this value should be set to...
  - DATABASES:seer mod\_openopc
- UNIQUEID - typically, just the name you've given your server. So, for example, if your server is named "Joker.MyDomain.com", then your UNIQUEID would be simply "Joker"

#### PYTHON COOKBOOK BACKUP (differential backup)

python\_cookbook\_backup comes with its own instructions, which you should follow. Items of note are as follows...

- README: README\_python\_cookbook\_backup.txt

Extract to /opt/python\_cookbook\_backup

- create a destination folder for backups to roll to.
- we will use /mnt/backup/differential\_backup

## SYPHON

syphon comes with its own instructions, which you should follow. Items of note are as follows...

- README: README\_syphon\_v-0-1-6-a.txt
- none.

## S.E.E.R.

S.E.E.R. comes with its own deployment guide, which you should follow. Items of note are as follows...

- README-1: SEER\_-\_DEPLOYMENT\_-\_GUIDE\_-\_v1\_2011\_1228.pdf
- S.E.E.R. Utilizes wkhtmltopdf in order to generate PDF files. Both the source code, as well as precompiled binaries for Linux (32bit and 64 bit) and Windows (32-bit) are included in the folder...
- - /seer\_2/plugins/extensions/wkhtmltopdf/
- Installation is straightforward. For the Linux binaries, simply chmod the files as executable (777), and you can leave them where they are, or copy them somewhere else, such as /opt/wkhtmltopdf/[files]. For the Windows binary, you must install it.
- Once installed, make sure the S.E.E.R. Global options file is edited, such that \$seer\_system\_cmd\_wkhtmltopdf reflects the installation location of wkhtmltopdf. For example...
- - \$seer\_system\_cmd\_wkhtmltopdf = "/opt/wkhtmltopdf\_static/wkhtmltopdf-amd64";
- - OR
- - \$seer\_system\_cmd\_wkhtmltopdf = "C:\Program Files\wkhtmltopdf\wkhtmltopdf.exe";
- S.E.E.R.'s implementation of server-supplied fonts requires that the server running S.E.E.R. (and using wkhtmltopdf to generate PDF files) have fontconfig version 2.8.8 or better installed. Version 2.9.0 is supplied for free in source code form in the folder...
- - /seer\_2/plugins/fonts/fontconfig
- Installation should be carried out in accordance with the instructions, supplied as file...
- - /seer\_2/plugins/fonts/fontconfig/fontconfig\_cblfs\_compilation\_instructions.pdf
- if you have an issue where the mod\_openopc database will not 'create' itself, or you notice that after logging in for the first time as "administrator / administrator" and then clicking "login" again, it re-prompts you for a new username / pw combo, then php is not properly parsing the non-standard \$\_GET variable declarations.
- - edit your php.ini file (/etc/php.ini) and be sure that ";" (semicolon) is the first character inside the quotes for variable "arg\_separator.input".
- - example... arg\_separator.input = "& ; &#38;"
- - change to... arg\_separator.input = "&;"
- - reason:
- - formerly, this variable could be defined such that groups of characters, separated with a space, would be considered delineators. However in recent php builds, every single character is considered a separator, and the spaces cause havoc!
- - restart httpd (apache) with /etc/init.d/httpd restart

## SETTING THE SYSTEM SCHEDULER

System jobs are scheduled with cron (via crontab entries). Edit your

/etc/crontab file accordingly.

Note - we have removed 'syslog' entries from /etc/cron.daily and such, because we're generating our own system logs (see above).

| TASK FREQUENCY | TASK   |
|----------------|--|
| 1x daily       | log roll   |
| 1x weekly      | mod_openopc database maintenance   |
| 1x daily       | seer + mod_openopc database backup<br>(this is the backup 'to-disk')<br>(tape backup performed via STaRBUC)  |
| 1x daily       | differential backup of...<br>- /opt/mod_openopc<br>- /opt/apache_pages/seer_2<br>- /opt/apache_pages/starbuc<br>- /opt/fieldgate_dm3_reporter<br>- /opt/syphon<br>(this effectively 'covers' you should<br>you accidentally delete or edit something,<br>and want to go back to 'yesterday' or<br>'a few days ago' when it worked as<br>intended). |

NOTE: see the sample file...

"SAMPLE\_etc-crontab"

### **CONTROLLING STARTUP**

System startup is controlled via settings entered through chkconfig (see notes about /sbin/chkconfig, above, and the sample file, noted above), as well as entries in rc.local. Edit your /etc/rc.d/rc.local file accordingly (this file is processed last as a part of system bootup).

- set network adapter settings, and launch ethernet
- start clock sync daemon
- start clamav antivirus daemon
- start sendmail
- launch virtual machine(s)
- start http server
- - set permissions for magnetic tape drive(s)
- launch vnc server to run as dead head (headless)
- start auto-launch routine of mod\_openopc
- start auto-launch routine of syphon

NOTE: see the sample file...

"SAMPLE\_etc-rc.d-rc.local"

### **DEALING WITH LONG QUERIES / LARGE AMOUNTS OF HISTORICAL DATA**

Long queries in S.E.E.R. Are the result of large amounts of historical data over a given time period, or simply running a report over a very long period of time. If your hardware (server) can't keep up, you can lengthen the default times for php script execution and database query timeout...

```
>> /etc/php.ini
```

```
max_execution_time = 600      (we suggested 420 seconds, 600 = 10 minutes)
```

Obviously, '600' is just another suggestion; you may increase this value as much as you see fit.

### **DEALING WITH HEAVY DISK I/O**

Heavy disk I/O can result in very high processor 'wa' (I/O Wait) time observed via top and iostat. This can get very ugly when doing a live backup to a high speed device, such as an LTO-4 tape drive.

Ways to mitigate the issue are as follows...

- 1- in your mysql.cnf file, comment out all "log file" entries, effectively telling MySQL not to store an operations log. This is fine in a production environment, assuming you've already tuned your database for your OS and such. You can always temporarily re-enable logging if you have an issue that you want to debug.
- 2- in your fstab (mount control) file, mount the STORAGE and BACKUP partitions with the 'noatime' flag. Whenever a file is opened, even for simply reading, the 'last accessed' time is updated for that file, which results in a WRITE operation for every READ operation... and a double WRITE operation for every (seemingly) single WRITE. For frequent reads, or very small writes, this can make a dramatic difference. DO NOT APPLY THIS OPTION TO YOUR OS DRIVE!!!!
- 3- overall performance loss due to hyperthreading scheduler on Intel systems. Either disable HT in the sys bios, or add the "noht" flag to the end of the kernel declaration line in your /boot/grub/grub.conf file.
- 4- configure STaRBUC and set options in globaloptions file as follows...
  - - \$starbuc\_ENABLE\_GZIP\_COMPRESSION = "NO";
  - - \$starbuc\_USE\_IO\_NICENESS\_YES\_OR\_NO = "YES";
  - - \$starbuc\_ARCHIVER = "STAR";
  - - - valid options are "GNUTAR" and "STAR"... GNU tar will choke on files larger than 68 GB, and slow down on files larger than 5 GB... therefore, we'll use STAR which has no such hangups.