OpenIndiana Small System Server Build
Version 1.25
Jon Green

www.jasspa.com

No Warranty  All the information on this document is published in good faith and for general information purpose only. Jon Green does not make any warranties about the completeness, reliability and accuracy of this information. Any action you take upon the information you find in this document, is strictly at your own risk. Jon Green will not be liable for any losses and/or damages in connection with the use of this document.
Jon Green.
United Kingdom

Mail: Read right to left, from the top of the last column and snake between columns
c.aj
oas@j
mpsno

All the information on this document is published in good faith and for general information purpose only. Jon Green
does not make any warranties about the completeness, reliability and accuracy of this information. Any action you
take upon the information you find in this document, is strictly at your own risk. Jon Green will not be liable for any
losses and/or damages in connection with the use of this document.

Title: OpenIndiana Small System Server Build
Reference: oi_setup
Version: v1.25
Date: 2015/08/16 11:39:06

Typeset with the TeX Live 2012 \LaTeX{} Documentation System.
## Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Who</th>
<th>Description</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015/08/16</td>
<td>JG</td>
<td>Added notes on postfix 587 submission</td>
<td>1.25</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added notes on strengthening DH for Dovecot and Postfix.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added notes on upgrading OI to a later version.</td>
<td></td>
</tr>
<tr>
<td>2014/04/17</td>
<td>JG</td>
<td>Minor corrections and additional information.</td>
<td>1.22</td>
</tr>
<tr>
<td>2014/03/21</td>
<td>JG</td>
<td>Added information on disabling ipv6.</td>
<td>1.18</td>
</tr>
<tr>
<td>2014/03/15</td>
<td>JG</td>
<td>First proof.</td>
<td>1.17</td>
</tr>
<tr>
<td>2014/02/09</td>
<td>JG</td>
<td>Reformatted notes into more useful document.</td>
<td>1.9</td>
</tr>
<tr>
<td>2012/09/08</td>
<td>JG</td>
<td>OpenIndiana server of notes.</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Table 1: Revision History
Contents

1 Background 8
  1.1 Requirements ........................................ 8
  1.2 Domain Name Provision ................................ 8
  1.3 Deciding on a System ................................ 8
  1.4 Mobile Service Connectivity .......................... 9

2 Hardware 11
  2.1 System Assembly ...................................... 12
  2.2 Power On ............................................... 14

3 Architectural Overview 16
  3.1 System Services ....................................... 17

4 OpenIndiana Installation 18
  4.1 Enabling Root Access .................................. 18
  4.2 Running with root privilege ........................... 18
  4.3 Package Manager ...................................... 19
  4.4 Upgrading OpenIndiana ................................ 19

5 Network Setup 21
  5.1 Static IP Address ..................................... 22
  5.2 Network Time .......................................... 23
    5.2.1 Network Time Server ............................... 24
    5.2.2 Disabling ipv6 .................................... 24
  5.3 DNS Server ............................................ 25
    5.3.1 Manually defining DNS files ...................... 28
    5.3.2 Setting up bind .................................. 30
    5.3.3 Bonjour / Zero Configuration Networking ....... 31
    5.3.4 mDNS Service ....................................... 32
  5.4 DHCP Server ........................................... 33
    5.4.1 DHCP Logging ..................................... 34
  5.5 Print Server ........................................... 35
    5.5.1 AirPrint ........................................... 35
  5.6 Samba (SMB Share) .................................... 36
  5.7 Firewall (IP Filter) ................................... 37

6 UPS Protection 41
  6.1 Installing MOXA serial card ......................... 41
  6.2 Installing apcupsd .................................... 42
  6.3 apcupsd logging ...................................... 44
  6.4 apcupsd configuration ............................... 44
6.5 apcupsd starting and stopping ........................................... 44
6.6 apcupsd USB configuration .................................................. 44

7 ZFS File System ................................................................. 45

8 Setting up WAN server ......................................................... 46
  8.1 Zone Preparation ........................................................... 46
    8.1.1 Creating a VNIC ....................................................... 46
  8.2 Zone Creation ............................................................. 47
  8.3 Zone Static IP ............................................................. 48

9 Server Certificate ............................................................ 50

10 E-Mail Service ............................................................... 51
  10.1 Mail packages ............................................................ 51
  10.2 Creating user accounts .................................................. 51
  10.3 Setting up Dovecot ....................................................... 52
    10.3.1 Starting the service ................................................. 55
    10.3.2 Log management ..................................................... 55
  10.4 Removing Sendmail ....................................................... 55
  10.5 Postfix local Mailer ....................................................... 55
  10.6 Global Zone Mailer ....................................................... 58
  10.7 Postfix SMPTS Mailer ..................................................... 58
    10.7.1 Creating a new postfix-smtps service ......................... 58
    10.7.2 Creating postfix-smtps configuration files ............... 61
    10.7.3 Setting up SASL authentication ............................... 61
    10.7.4 Postfix Configuration ............................................. 61
    10.7.5 Starting the service .............................................. 63
    10.7.6 Postfix version number .......................................... 63
  10.8 fetchmail ................................................................. 63
    10.8.1 Creating a new fetchmail service ............................. 63
    10.8.2 Creating fetchmail configuration files .................... 65
    10.8.3 Starting the service .............................................. 65
    10.8.4 Managing logs ..................................................... 65
    10.8.5 TODO ............................................................... 66

11 Web Services ............................................................... 66
  11.1 Web Server packages .................................................... 66
  11.2 Creating the file system ............................................... 66
  11.3 Apache ................................................................. 66
    11.3.1 PHP support ........................................................ 69
    11.3.2 MySQL support ..................................................... 69
    11.3.3 HTTPS services .................................................... 70
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.3.4</td>
<td>WebDAV</td>
<td>72</td>
</tr>
<tr>
<td>11.3.5</td>
<td>Log management</td>
<td>75</td>
</tr>
<tr>
<td>12</td>
<td>Calendar and Address Book Services</td>
<td>76</td>
</tr>
<tr>
<td>12.1</td>
<td>Getting DAViCal and installing</td>
<td>76</td>
</tr>
<tr>
<td>12.2</td>
<td>Setting up Postgres</td>
<td>76</td>
</tr>
<tr>
<td>12.3</td>
<td>Initialising the DAViCal Database</td>
<td>78</td>
</tr>
<tr>
<td>12.4</td>
<td>Importing an existing DAViCal Database</td>
<td>79</td>
</tr>
<tr>
<td>12.5</td>
<td>Remote Server postgres preparation</td>
<td>79</td>
</tr>
<tr>
<td>12.6</td>
<td>Davical Configuration</td>
<td>79</td>
</tr>
<tr>
<td>12.7</td>
<td>Apache Configuration</td>
<td>80</td>
</tr>
<tr>
<td>12.8</td>
<td>DAViCal User Configuration</td>
<td>82</td>
</tr>
<tr>
<td>13</td>
<td>CVS</td>
<td>83</td>
</tr>
<tr>
<td>13.1</td>
<td>User Configuration</td>
<td>84</td>
</tr>
<tr>
<td>14</td>
<td>Backup</td>
<td>85</td>
</tr>
<tr>
<td>15</td>
<td>JASSPA MicroEmacs</td>
<td>91</td>
</tr>
<tr>
<td>16</td>
<td>TeXLive</td>
<td>92</td>
</tr>
<tr>
<td>16.1</td>
<td>TexLive User Setup</td>
<td>93</td>
</tr>
<tr>
<td>17</td>
<td>Client Device Configuration</td>
<td>93</td>
</tr>
<tr>
<td>17.1</td>
<td>Static IP Addresses</td>
<td>93</td>
</tr>
<tr>
<td>17.1.1</td>
<td>OSX Lion DNS server priority</td>
<td>94</td>
</tr>
<tr>
<td>17.1.2</td>
<td>OSX Lion DNS Search Domains</td>
<td>94</td>
</tr>
<tr>
<td>17.2</td>
<td>Mail Server</td>
<td>95</td>
</tr>
<tr>
<td>17.3</td>
<td>Calendar</td>
<td>95</td>
</tr>
<tr>
<td>17.4</td>
<td>Addressbook</td>
<td>95</td>
</tr>
<tr>
<td>17.5</td>
<td>WebDAV</td>
<td>95</td>
</tr>
<tr>
<td>17.6</td>
<td>WebServer</td>
<td>96</td>
</tr>
<tr>
<td>17.7</td>
<td>DAViCal Administrator</td>
<td>96</td>
</tr>
<tr>
<td>17.8</td>
<td>Printing</td>
<td>96</td>
</tr>
<tr>
<td>17.9</td>
<td>CUPS Print Server Administration</td>
<td>96</td>
</tr>
<tr>
<td>17.10</td>
<td>Samba</td>
<td>96</td>
</tr>
<tr>
<td>17.11</td>
<td>Samba Administration (SWAT)</td>
<td>97</td>
</tr>
<tr>
<td>17.12</td>
<td>SSH</td>
<td>97</td>
</tr>
<tr>
<td>18</td>
<td>Conclusion</td>
<td>97</td>
</tr>
</tbody>
</table>
Introduction

This document describes setting up a HP Microserver (N40L) using http://www.openindiana.org for a small business environment with a small number of users. The server provides services as a local server infrastructure for storage and shared file systems in addition to serving a number of iOS and Android mobile devices.

The purpose of the document was primarily as a reminder to myself as to what was set up, however given that I had to search around the web for information then I felt I should perhaps put a little more effort into documenting it more thoroughly so that other people in a similar position to myself may benefit.

The information provided within the document should be considered to be informative only and I accept no liability for errors and omissions. I am not a professional System Administrator but a software professional and have administered my own Sun Solaris systems for many years.

Thanks to Sun Microsystems of old for releasing a great operating system and to the people behind OpenIndiana, their contributors, package owners, package maintainers and others that have all put in a huge amount of time and effort to deliver this distribution.

Jon Green February 2014
1 Background

The company had an existing Solaris 10 (Sparc) infrastructure which had been running for 6 years 24/7 which contained all of the company business information providing services to mobile devices. The system had proved exceptionally reliable but was end of life and power hungry. Sun (Oracle) equipment now appears to be out of reach of the small company, commercial licensing is expensive and Oracle appear to have little interest in the small business. There are no low powered economical systems in Oracle’s range and I had an uneasy experience dealing with them.

1.1 Requirements

The general requirements of the system are defined as follows:

- System must run 24/7, be available and reliable.
- Data integrity is a paramount requirement; the system will host and store business data in addition to providing business services for mobile devices.
- Low power requirements. A system running 24/7 then the power requirements must be low. Speed is not a key requirement.
- System must be secure, security is a concern.
- Provide redundancy and backup solution for critical business data.
- Support LAN services including source control system, file sharing, mail and print services.
- Support WAN services for iOS and Android mobile devices including E-Mail, Address book, Calendar, Web Storage (WebDAV) and HTTP web services.
- System storage minimally defined as 1TB for the storage of existing business critical data, the system should provide sufficient expansion for the next 5 years.
- Life expectancy of the system should be 5 years.
- A low cost system is highly desirable, however cost should not significantly compromise any of the aforementioned requirements.

1.2 Domain Name Provision

For this system then the company has a static IPv4 address and global DNS entries for the domain name www.mydomain.co.uk. The company has a single signed SSL certificate for the domain name www.mydomain.co.uk which will be used for all services. Services provided by www.mydomain.co.uk should be available on the WAN and on the LAN.

1.3 Deciding on a System

The HP N40L was top of the list of base systems, a small low power system which was sold as a complete hardware tested unit. My preference was for a complete system where all of the component parts are tested and are known to be working together. The HP N40L offered a large storage capability in addition to ECC
memory in a compact package at a very competitive price. Given speed was not a over-riding priority there was no other system that came anywhere close to this package which also included Enterprise features.

The selection of operating system was a little more difficult. A primary concern was data integrity so any system must support ZFS, I would also like some form of Zone support to partition WAN facing services. Obviously my preference was Solaris, but the O/S was out of the question with Oracle having little interest in the low end. This left the O/S options as Linux, FreeBSD, OpenIndiana or Apple. OpenSolaris was closed because of licensing when used commercially. Linux was discounted because the ZFS support still looked immature and there was no mature next generation file system. Apple looked like it could possibly be a contender (meaning a Apple Server could be used) but Ten’s Complement who were commercially porting ZFS at the time had not yet delivered a viable solution. The two remaining contenders were FreeBSD and OpenIndiana, my preference was OpenIndiana as this was essentially Solaris; ZFS and Zones are supported but I was not sure how well the whole system would be supported.

My final decision then was HP N40L H/W with OpenIndiana O/S. I was a little sceptical whether I was going to be able to build my desired server configuration and get all of the component S/W and H/W parts to work. At this point in time I had not actively followed the progress of OpenIndiana or Illumos and the only way to find out what it was like was to build a system; if it failed then I could fall back to FreeBSD.

1.4 Mobile Service Connectivity

Mobile service support was a big requirement for the server and all mobile devices should be supported. Any system should not be reliant on third party services such as Google because of privacy concerns. The principle mobile device was iPad although a few Anroid devices existed. Solaris had been previously used to provide the principal services as shown in Figure 1.

![Figure 1: iPad Services](image)

Mail and Notes are supported with IMAP (dovecot, postfix, fetchmail) and SMTP (postfix, dovecot-sasl) e-mail server. These services are interoperable with computer desktops. Calendar, Reminders and Contacts use CalDAV and CardDAV protocols and may be supported with DAViCal running on top of an Apache web server with PHP and Postgres SQL database provision.
The principle iPad office applications (Figure 2) allow documents to be copied via WebDAV services; provided by the Apache web server.

![Figure 2: iPad Applications](image)

Within the LAN then Airprint printing services (Figure 3) must be available on existing legacy printers connected to the LAN.

![Figure 3: iPad Printing](image)

CUPS provides the printing services which may be advertised through DNS via sd-dns using bind. For iOS 7 then mDNS is additionally required which uses avahi as a bridge between CUPS and the mDNS service.
2 Hardware

Having decided on the HP N40L then I decided to use a Solid State Drive (SDD) for the operating system this would not be mirrored as it could be re-generated in the event of a failure. A 2.5” SSD 128GB drive from Crucial was more than big enough. In the HP N40L the 2.5” drive may sit between the 5.25” CD-ROM drive bay and the HDD drive cage just behind the illuminated HP logo. To fit the SSD then an addition Power-Y cable, SATA power adapter and 1M 90° SATA is required. Critical data would be stored on a pair of ZFS mirrored SATA disks, Western Digital 3.5” 3TB Green drives provide this storage.

In hindsight I really should have mirrored the operating system disk, the Crucial SSD failed after 9 months and was replaced under warranty by Crucial. Whilst the disk allocation meant that no critical data was lost it still took a few days to re-build the system. This has been rectified with a new configuration whereby a second SSD has been added (OCZ) and the root file system is now mirrored using ZFS, the OCZ disk is connected by way of the external eSATA connector.

My other regret is that I did not re-flash the BIOS in the Microserver with a community improved BIOS, the internal CD SATA and external eSATA operate as legacy IDE devices rather than SATA which seems to cause a problem with re-silvering the disks. In order for a ZFS re-silvering operation to succeed then one of the CPUs should be disabled (or run something that consumes 100% CPU).

Data backup would be provided by an existing Sun Microsystems DAT 40 SCSI tape drive. An Adaptec Ultra320 29320LPE Ultra320 Single Channel Low-Profile PCI Express SCSI Card was selected to control the tape drive.

An old APC Smart-UPS 620inet was used for the UPS solution and given a new lease of life by replacing the battery. The Keyspan USB serial adapter (USA-19HS) did not play well with the UPS resulting in intermittent communication errors and was later replaced with a Moxa CP-102EL-DB9M 2-port RS-232 low profile PCI Express serial board which proved much more reliable (albeit expensive). The APC USB devices do work with apcupsd although you need to check the apcupsd site to see which ones are supported. The UPS should be matched to the power of the system (which is low in the case of the HP N40L) otherwise the UPS becomes power inefficient.

The hardware itinerary for the system is defined as follows:

<table>
<thead>
<tr>
<th>#</th>
<th>Manufacturer</th>
<th>Part</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1x</td>
<td>HP</td>
<td>ProLiant Microserver G7 Turion II Neo N40L 1.5 GHz 2GB 250GB</td>
<td>Base server.</td>
</tr>
<tr>
<td>1x</td>
<td>Crucial</td>
<td>T2KIT51272BA1339</td>
<td>8GB Kit (4GBx2), 240-pin DIMM Upgrade for a HP - Compaq Proliant MicroServer System</td>
</tr>
<tr>
<td>1x</td>
<td>Crucial</td>
<td>CT128V4SSD2</td>
<td>128GB Crucial v4 SATA 3Gb/s 2.5-inch SSD [Root file system]</td>
</tr>
<tr>
<td>1x</td>
<td>OCZ</td>
<td>VTX4-25SAT3-128G</td>
<td>128GB Vertex 4 SATA 6Gb/s 2.5-inch SSD [Root file system]</td>
</tr>
<tr>
<td>2x</td>
<td>Western Digital</td>
<td>3TB Green SATA 6Gb/s 64M 3.5” HDD</td>
<td>SATA Hard disk drives</td>
</tr>
<tr>
<td>1x</td>
<td>C2G</td>
<td>6in 5.25 Internal Power Y-cable</td>
<td>HDD Power cable splitter.</td>
</tr>
<tr>
<td>2x</td>
<td>StarTech</td>
<td>6in 4pin SATA Power Adapter</td>
<td>SATA HDD power adapter.</td>
</tr>
<tr>
<td>1x</td>
<td>C2G</td>
<td>1m 180° To 90° 7 pin Serial ATA (SATA) Cable</td>
<td>SATA cable 90°</td>
</tr>
<tr>
<td>1x</td>
<td>Unknown</td>
<td>1m eSATA To SATA Cable</td>
<td>External to internal SATA cable.</td>
</tr>
<tr>
<td>1x</td>
<td>Adaptec</td>
<td>Adaptec Ultra320 29320LPE</td>
<td>Ultra320 Single Channel Low-Profile PCI Express SCSI Card [For SCSI DAT Tape Drive]</td>
</tr>
</tbody>
</table>

Table 2: Hardware Itinerary (continued ...)

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
A second failure of the replacement Crucial SSD occurred in February 2014 (again after 9 months) this time the drive remained active but part of the disk could no longer be read or written to without error. The failed drive was replaced with a Toshiba Q Series SSD 128GB drive. The system was re-configured to place the SSD drives in the 3.5” main drive bays using the “Newer Technology Inc - AdptaDrive”, this is a 2.5” to 3.5” SATA Drive Converter Bracket used to mount the SSD drive in a 3.5” bay. The drives appear to work better in this position using SATA rather than legacy ATA.

2.1 System Assembly

Using a anti-static wrist band, the system was disassembled, use the HP “Maintenance and Service Guide” (Part No: 615714-006) if you need some more explicit instructions. The system board was removed and the existing RAM removed and replaced with the 2x4GB memory DIMMs. The RS-232 and SCSI PCIe cards were fitted with their half-height brackets and installed. The RS-232 card fitted in the PCI Express x1 slot and the SCSI card in the PCI Express x16 slot, as show in Figure 4.
The SSD was mounted between the 5.25" CD-ROM bay and the disk cage, a meter long SATA cable was routed round the back of the chassis and connected to the internal SATA connector in the motherboard at the front of the chassis. Power for the SSD was provided with the Power Y-cable connected to a SATA adapter. A second SSD disk was added later; a eSATA to SATA cable was connected to the external SATA connector on the back of the unit and fed through the chassis above the PCI slots into the CD-ROM bay where the second drive is located as shown in Figure 5.
The SSD SATA plugs into the main board as shown in Figure 6 and the system board was replaced inside the chassis.
The Western Digital disks were fitted in the plastic disk carriers and inserted into Slot 1 and Slot 2, the spare 250GB disk that was supplied with the unit was placed in Slot 3.
To keep the system clean then the door was removed and split in half by removing the door lock. A filter was inserted as a sandwich between the 2 halves of the door before re-assembly. This should reduce the dust build up over time and keep the system clean, any such filter should be replaced periodically to ensure that the airflow is not disrupted.

2.2 Power On

The system was completely reassembled and everything was double checked for correct fitting before applying power to the unit. USB keyboard, USB mouse, VGA monitor and ethernet were connected.
The BIOS was updated with the HP recommended update using a USB memory stick (there was an advisory notice from HP supplied with the unit).
There are some details on the Web to upgrade the BIOS to perform a faster disk transfer however this BIOS update was not installed as it was thought that it compromise the reliability of the system. As mentioned in the introduction then it would have been better to have installed this BIOS upgrade to get native SATA disk operation for the CD-ROM and external SATA disk connections.
Figure 6: SSD SATA cable threaded through chassis to system board
3 Architectural Overview

The system is connected to an ADSL modem which provides the gateway to the Internet. The line is assigned a single static IP address with a domain name registered to it. There is a single SSL certificate with name www.mydomain.co.uk.

The basic architecture of the network to be constructed is shown in Figure 7. The server comprises two zones, a global zone which provides local services, a separate zone called www provides the WAN facing services.

![Architectural overview of system](image)

Figure 7: Architectural overview of system

The access point for the network is provided by a DSL router with wired and wireless access points. The DSL router provides a wireless access point which is configured to propagate the local DNS server and not the WAN DNS server provided by the ISP. Local DNS resolution is required so that our public DNS name can be resolved locally on the LAN and ensures that all WAN facing services have the same URL on both the WAN and LAN. DHCP services on the DSL Router are disabled and provided by the local server. The DSL Router still provides a network bridge between the wired and wireless parts of the local network.
3.1 System Services

The services required of the server are shown in Table 3 which outlines where the services are running on the system.

<table>
<thead>
<tr>
<th>Service</th>
<th>Port</th>
<th>Scope</th>
<th>Zone</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSH</td>
<td>22</td>
<td>LAN</td>
<td>Global</td>
<td>Secure sockets for Admin remote login</td>
</tr>
<tr>
<td>SMTP</td>
<td>25</td>
<td>LAN</td>
<td>www</td>
<td>Local Mail submission</td>
</tr>
<tr>
<td>DNS</td>
<td>53</td>
<td>LAN</td>
<td>Global</td>
<td>Domain name services</td>
</tr>
<tr>
<td>DHCP</td>
<td>67,68</td>
<td>LAN</td>
<td>Global</td>
<td>Dynamic host network configuration</td>
</tr>
<tr>
<td>HTTP</td>
<td>80</td>
<td>LAN</td>
<td>www</td>
<td>Local Intranet server</td>
</tr>
<tr>
<td>NFS</td>
<td>111</td>
<td>LAN</td>
<td>Global</td>
<td>UNIX file sharing</td>
</tr>
<tr>
<td>NTP</td>
<td>123</td>
<td>LAN</td>
<td>Global</td>
<td>Network time services</td>
</tr>
<tr>
<td>SMB</td>
<td>137,138,139</td>
<td>LAN</td>
<td>Global</td>
<td>CIFS/Windows file sharing</td>
</tr>
<tr>
<td>HTTPS</td>
<td>443</td>
<td>WAN</td>
<td>www</td>
<td>Extranet HTTPS server</td>
</tr>
<tr>
<td>SMTPS</td>
<td>465</td>
<td>LAN</td>
<td>www</td>
<td>Global Mail submission (SSL)</td>
</tr>
<tr>
<td>submission</td>
<td>587</td>
<td>LAN</td>
<td>www</td>
<td>Global Mail submission (STARTLS)</td>
</tr>
<tr>
<td>CUPS</td>
<td>631</td>
<td>LAN</td>
<td>Global</td>
<td>CUPS administration / IPP Print services</td>
</tr>
<tr>
<td>SWAT</td>
<td>901</td>
<td>LAN</td>
<td>Global</td>
<td>Samba administration</td>
</tr>
<tr>
<td>IMAPS</td>
<td>993</td>
<td>WAN</td>
<td>www</td>
<td>IMAP Mail services (SSL)</td>
</tr>
<tr>
<td>CVS</td>
<td>2401</td>
<td>LAN</td>
<td>Global</td>
<td>Legacy source control system</td>
</tr>
<tr>
<td>MySQL</td>
<td>3306</td>
<td>LAN</td>
<td>Global</td>
<td>MySQL SQL Server (Limited access)</td>
</tr>
<tr>
<td>mDNS</td>
<td>5353</td>
<td>LAN</td>
<td>Global</td>
<td>Bonjour services for iOS Airprint</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>5432</td>
<td>LAN</td>
<td>Global</td>
<td>Postgres SQL Server (Limited access for CalDAV services)</td>
</tr>
<tr>
<td>CalAdmin</td>
<td>8008</td>
<td>LAN</td>
<td>www</td>
<td>CalDAV administration</td>
</tr>
<tr>
<td>WebDAV</td>
<td>8081</td>
<td>WAN</td>
<td>www</td>
<td>HTTPS WebDAV server (SSL)</td>
</tr>
<tr>
<td>CalDAV</td>
<td>8443</td>
<td>WAN</td>
<td>www</td>
<td>Calendar services (SSL)</td>
</tr>
<tr>
<td>CardDAV</td>
<td>8443</td>
<td>WAN</td>
<td>www</td>
<td>Address book services (SSL)</td>
</tr>
</tbody>
</table>

Table 3: Server services

The order in which the basic services were brought up is as follows:

- Static IP address assignment
- NTP time services.
- DNS server
- DHCP Server
- Other services as required.
- Firewall (ipf)

During installation then the external router (DSL modem) should be secured and all external incoming ports closed. It is much easier to bring the server up without installing a server firewall in the first instance and ensure that all of the services are running. Once everything is running then the firewall rules are applied to the server and verified to ensure that they are working. Once the server firewall is in place then the the external router may be configured to connect the WAN services to the server.
4 OpenIndiana Installation

At the start of the OpenIndiana installation then it is assumed that a LAN network exists and provides DHCP and DNS services.

The OpenIndian operating system was downloaded from www.openindia.org. The desktop release was selected with Gnome in preference to the server build as both server and desktop services were required.

The DVD image was selected and copied to DVD using cdrecord on Solaris. Note writing the DVD using Microsoft Windows and an OEM supplied DVD utility failed to write the DVD correctly. The USB image may be a better choice?

The system was booted from a USB CDROM/DVD device and then installed from the desktop onto the SSD occupying the whole disk. Installation took in excess of an hour and it may be left to install on its own after entering the basic system configuration, simply follow the prompts. On completion the DVD drive may be removed and the system rebooted into OpenIndiana from the hard disk.

At this point we now have a fresh install of OpenIndiana with the installation defaults. If you have installed the root file system on a single disk it may be mirrored later.

4.1 Enabling Root Access

It is useful to be able to login as root from the package updater. The root password is immediately expired after installation and you need to choose a new one. To do this:

- Open a Terminal
- Execute "su -" and give the password you chose for your account at installation time. You will be informed that roots password has expired and are prompted to change it; once it has been changed you can exit the su session.

You should be able to login/authenticate as root now. This does not allow root to login via ssh.

4.2 Running with root privilege

In order to set up the system then root privilege is required. sudo is generally used to run commands in a privilege mode by pre-fixing the command with sudo i.e.:

```
sudo svcadm enable network/physical:default
```

For a lot of configuration work then it is easier to run as root all of the time by running a new shell, how you run a root shell will determine the execution path, X-Windows availability etc.

```
sudo zsh
hal$ svcadm enable network/physical:default
```

or

```
hal$ sudo su -
OpenIndiana (powered by illumos) SunOS 5.11 oi_151a7 October 2012
root@hal:~# id
uid =0(root) gid =0(root) groups =0(root),1(other),2(bin),3(sys),4(adm),5(uucp),6(mail),7(ttys),8(lp),9(uucp),12(daemon)
root@hal:~# echo $PATH
/usr/gnu/bin:/usr/bin:/usr/sbin:/sbin
root@hal:~#
```
Running as root is generally frowned upon because it is considered to be much more dangerous and any inadvertent mistake could destroy a system. When running as root then always remember that UNIX is not so forgiving and will do as instructed. UNIX is not going to ask you “Are sure?” and executes any command however silly it might be (this is also true for sudo although the accepted theory seems to be when you write sudo you have explicitly asked for privilege and the associated command has been considered).

4.3 Package Manager

The package manager may be run from the desktop or the command line. Root access should be enabled to run the package manager from the desktop otherwise run with root privileges from a shell i.e.

```bash
$ sudo packagemanager
```

Configure the package manager to pick up additional software that is delivered outside of the OpenIndiana release. From the Package Manager add the Spec Files Extra repositories:

- Publisher -> Add
  URI: http://pkg.openindiana.org/sfe
  Alias: OpenIndianaSFE

- Publisher -> Add
  URI: http://pkg.openindiana.org/sfe-encumbered
  Alias: OpenIndianaSFE-Emumbered

These additional repositories contain some useful packages that may be required later.

4.4 Upgrading OpenIndiana

Upgrading a version of OpenIndiana then we need to upgrade the global zone and any other zones. The following paragraphs show an upgrade from 0.151.1.7 to 0.151.1.9.

Login or sudo to root and check what will be upgraded by the system:

```bash
hal# pfexec pkg image-update -nv
```

If the result is OK then perform the upgrade in the global zone:

```bash
hal# pfexec pkg image-update -v
```

Reboot the system. With the global zone updated then our www zone needs to be updated. Make sure that the zone is not running.

```bash
hal# zoneadm list
global
www
hal#
```

This will list all of the currently running zones, if it is not running we can issue the below command to see all of the zones installed on this system:

```bash
hal# zoneadm list -i
global
www
hal#
```

Now that we know the zone name we must ensure it is not running, to stop the zone do the below, if your zone is currently not running please skip this command.

```bash
hal# zoneadm -z www halt
hal# zoneadm list
global
hal#
```
The www zone has now been stopped.
This will stop our zone and allow us to make changes to it. Now we need to find the location of the zone on
the system, this is performed as follows:

```
hal# zfs list
```

Look for the mountpoint which should be something like the below:

<table>
<thead>
<tr>
<th>NAME</th>
<th>USED</th>
<th>AVAIL</th>
<th>REFER</th>
<th>MOUNTPOINT</th>
</tr>
</thead>
<tbody>
<tr>
<td>rpool</td>
<td>22.9G</td>
<td>75.1G</td>
<td>51K</td>
<td>/rpool</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rpool/zones/www</td>
<td>1.14G</td>
<td>75.1G</td>
<td>33K</td>
<td>/zones/www</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Now update the update the www zone:

```
hal# pkg -R /zones/www/root image - update -v
```

```
Packages to install: 5
Packages to update: 188
Estimated space available: 74.91 GB
Estimated space to be consumed: 1.10 GB
Create boot environment: No
Create backup boot environment: No
Services to change: 3
Rebuild boot archive: No
```

```
Changed packages:
openindiana.org
  compress/xz
    None -> 5.0.3,5.11-0.151.1.9:20140117T204509Z
library/database/gdbm
  None -> 1.6.3,5.11-0.151.1.9:20140117T202525Z
library/desktop/gdk-pixbuf
  None -> 0.5.11,5.11-0.151.1.9:20140117T202422Z
system/kernel
  None -> 0.5.11,5.11-0.151.1.9:20141210T124242Z
library/desktop/gdk-pixbuf
  None -> 0.5.11,5.11-0.151.1.9:20140117T202422Z
library/database/gdbm
  None -> 1.6.3,5.11-0.151.1.9:20140117T202525Z
system/kernel
  None -> 0.5.11,5.11-0.151.1.9:20140117T202422Z
library/database/gdbm
  None -> 1.6.3,5.11-0.151.1.9:20140117T202525Z
SUNWcs
  0.5.11,5.11-0.151.1.7:20121003T225133Z -> 0.5.11,5.11-0.151.1.9:20150504T114725Z
SUNWcsd
  0.5.11,5.11-0.151.1.7:20121003T225201Z -> 0.5.11,5.11-0.151.1.9:20140117T205506Z
compress/bzip2
  1.0.6,5.11-0.151.1.7:20121003T215018Z -> 1.0.6,5.11-0.151.1.9:20140117T201710Z
compress/gzip
```

```
Services:
  restart_fmri:
    svc:/application/desktop-cache/input-method-cache:default
    svc:/application/desktop-cache/pixbuf-loaders-installer:default
    svc:/system/manifest-import:default
```

```
DOWNLOAD     PKGS FILES XFER (MB)
Completed     193/193 8767/8767 151.0/151.0
```

```
PHASE ACTIONS
Removal Phase 8797/8797
Install Phase 9699/9699
Update Phase 8687/8687
```
PHASE ITEMS
Package State Update Phase 381/381
Package Cache Update Phase 188/188
Image State Update Phase 2/2

The following unexpected or editable files and directories were salvaged while executing the requested package operation; they have been moved to the displayed location in the image:

etc/zones -> /zones/www/root/var/pkg/lost+found/etc/zones-20150815T121106Z

NOTE: Please review release notes posted at:

hal#

Now finally boot and login into the zone:

hal# zoneadm -z www boot
zlogin -C www

When logged in to the zone check the OS version. You should see something similar to the below to confirm the upgrade of the zone.

hal# uname -a
SunOS <zonename> 5.11 oi_151a i86pc i386 i86pc

5 Network Setup

In this section we consider the basic network configuration comprising Static IP address, Network Time, DNS and DHCP services which will be managed by the server in the context of the Global zone (this could be another zone if required). Our network and the demands on it are not huge so it was not considered necessary to create a separate zone for these services.

OpenIndiana ships with a DHCP configuration, the first step is to set up static IP addressing. Decide on the IP address allocation that is going to be used in the network. The configuration used in this network is shown in Table 4.

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.8.1</td>
<td>ADSL router and gateway</td>
</tr>
<tr>
<td>192.168.8.2-127</td>
<td>Static IP addressed devices</td>
</tr>
<tr>
<td>192.168.8.128-159</td>
<td>Available DHCP Addresses</td>
</tr>
<tr>
<td>192.168.8.200</td>
<td>Main server (hal) the global zone.</td>
</tr>
<tr>
<td>192.168.8.201</td>
<td>Virtual Machine, zone (www), WAN facing Web and Mail services</td>
</tr>
<tr>
<td>224.0.0.251</td>
<td>The Bonjour multicast address.</td>
</tr>
</tbody>
</table>

Table 4: Static IP Address Allocation
5.1 Static IP Address

The first thing to do is to change from dynamic to static IP addressing. We require the server to have an address of 192.168.8.200. Refer to the following reference on setting up Solaris 11 which was used in the setup:

Run as root with a shell of your choice

```bash
sudo zsh
```

Disable the network auto magic

```bash
hal# svcadm disable network/physical:nwam
hal# svcadm enable network/physical:default
```

Manually set up the networking for the network adapter, this is `bge` in this system:

```bash
hal# ipadm show-if
IFNAME  STATE  CURRENT  PERSISTENT
lo0     ok     -m-v------46  ---

hal# ipadm create-if bge0
hal# dladm show-link
LINK  CLASS  MTU  STATE  BRIDGE  OVER
bge0  phys   1500  up     --      --

hal# ipadm show-if
IFNAME  STATE  CURRENT  PERSISTENT
lo0     ok     -m-v------46  ---
bge0    down   bm------46  46

Create the address to be assigned to the network adapter:

```bash
hal# ipadm create-addr -T static -a 192.168.8.200/24 bge0/v4
hal# ipadm show-addr
ADDROBJ  TYPE  STATE  ADDR
lo0/v4   static  ok   127.0.0.1/8
bge0/v4  static  ok   192.168.8.200/24
lo0/v6   static  ok   ::1/128
hal# netstat -r
Routing Table: IPv4
--------------------  --------------------  ------  -----  ----------  ---------
 hal   hal  UH     2  8    lo0
 192.168.8.0 192.168.8.200 U  2  0  bge0

Routing Table: IPv6
---------------------------  ---------------------------  -----  ---  -------  -----  
 hal   hal  UH     2  574  lo0

Add a default route to the gateway:

```bash
hal# route -p add default 192.168.8.1
add net default: gateway 192.168.8.1
add persistent net default: gateway 192.168.8.1
hal# netstat -r
```

Routing Table: IPv4
--------------------  --------------------  ------  -----  ----------  ---------
default  192.168.8.1  UG     1  0
hal     hal   UH     2  8    lo0
Routing Table: IPv6

<table>
<thead>
<tr>
<th>Destination/Mask</th>
<th>Gateway</th>
<th>Flags</th>
<th>Ref</th>
<th>Use</th>
<th>If</th>
</tr>
</thead>
<tbody>
<tr>
<td>hal</td>
<td>hal</td>
<td>UH</td>
<td>2</td>
<td>574</td>
<td>lo0</td>
</tr>
</tbody>
</table>

Set up the name server and edit /etc/resolv.conf. Add the gateway (DSL Router) and/or DNS servers specified by your ISP.

```
hal# cat /etc/resolv.conf
domain mydomain.co.uk
search mydomain.co.uk
nameserver 192.168.8.1
nameserver 212.23.3.100
nameserver 212.23.6.100
```

Set up the name service switch file /etc/nsswitch.conf for DNS by copying the existing /etc/nsswitch.dns over the file (assuming that LAPD or NIS are not being used).

```
hal# cp /etc/nsswitch.dns /etc/nsswitch.conf
```

Test that names are being resolved.

```
hal# /usr/sbin/host www.zen.co.uk
www.zen.co.uk is an alias for zen.co.uk.
zen.co.uk has address 82.71.140.243
zen.co.uk mail is handled by 10 mailcluster.zen.co.uk.
```

### 5.2 Network Time

The network time services may now be set up. To set up the client then edit the file /etc/inet/ntp.conf. Add the addresses of the NTP servers, typically the NTP service of your ISP is used in preference followed by local NTP pools i.e.

```
#ident "@(#)ntp.server 1.1 09/05/17 SMI"

# Use our ISP Server as preference
server ntp0.zen.co.uk prefer
# Use the UK NTP Pools next
server 0.uk.pool.ntp.org
server 1.uk.pool.ntp.org
server 2.uk.pool.ntp.org
server 3.uk.pool.ntp.org

# Always configure the drift file. It can take days for ntpd to completely
# stabilize and without the drift file, it has to start over on a reboot
# of if ntpd restarts.
driftfile /var/ntp/ntp.drift

# It is always wise to configure at least the loopstats and peerstats files.
# Otherwise when ntpd does something you don’t expect there is no way to
# find out why.
statsdir /var/ntp/ntpsstats/
filegen peerstats file peerstats type day enable
filegen loopstats file loopstats type day enable
```

Enable the NTP daemon

```
hal# svcadm enable network/ntp
```
Check the running status

```
hal# svcs -v ntp
STATE   NSTATE   STIME   CTID   FMRI
online  -       Sep_29 60 svc:/network/ntp:default
```

5.2.1 Network Time Server

For our local network then the server will act as the time server for the whole network. Add the server configuration to the file `/etc/inet/ntp.conf` by adding the following lines to the end of the file:

```
# We are a local time server.
# Broadcast on the local network to the other machines.
broadcast 224.0.1.1 ttl 4
```

Restart the NTP daemon

```
hal# svcadm restart network/ntp
```

Check the running status

```
hal# svcs -v ntp
STATE   NSTATE   STIME   CTID   FMRI
online  -       Sep_29 60 svc:/network/ntp:default
```

The server should now be acting as a NTP server. Static clients on the LAN should now be able to synchronise with the server for their time, mobile clients should directly use Internet time servers.

5.2.2 Disabling ipv6

It may be worth trying to disable ipv6 if there are disconnection problems with SSH or the system appears to be intermittently hanging up for no reason. If ipv6 is to be disabled then the following steps may be followed, I have not managed to find a better way to disable ipv6 without editing the configuration file.

```
hal# ifconfig -a6
lo0: flags=2002000849 <UP,LOOPBACK,_RUNNING,MULTICAST,IPv6,VIRTUAL> mtu 8252 index 1
    inet6 ::1/128
bge0: flags=20002000940 <RUNNING,PROMISC,MULTICAST,IPv6> mtu 1500 index 2
    inet6 ::/0
    ether 0:9c:2:97:51:41
hal# svcs -a |grep network/physical
disabled  Mar_18 svc:/network/physical:nwam
online    Mar_18 svc:/network/physical:default
hal# ipadm show-addr
ADDROBJ TYPE STATE ADDR
lo0/v4 static ok 127.0.0.1/8
bge0/v4 static ok 192.168.8.200/24
lo0/v6 static ok ::1/128
```

Delete the lo0 interface and re-create it.

```
hal# ipadm delete-if lo0
hal# ipadm show-addr
ADDROBJ TYPE STATE ADDR
bge0/v4 static ok 192.168.8.200/24
```

```
hal# ipadm create-addr -T static -a 127.0.0.1/8 lo0/v4
```
hal# ipadm show-addr
<table>
<thead>
<tr>
<th>ADDR OBJ</th>
<th>TYPE</th>
<th>STATE</th>
<th>ADDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>bge0/v4</td>
<td>static</td>
<td>ok</td>
<td>192.168.8.200/24</td>
</tr>
<tr>
<td>lo0/v4</td>
<td>static</td>
<td>ok</td>
<td>127.0.0.1/8</td>
</tr>
</tbody>
</table>

Edit /etc/ipadm/ipadm.conf and comment out the ipv6 entries which are designated with family=26.

```
_ifeqname=bge0;_family=2;
#ifDelete ipv6# _ifeqname=bge0;_family=26;
_ifeqname=bge0;_aobjname=bge0/v4;_ipv4addr=192.168.8.200,;up=yes;
_ifeqname=bge0;_aobjname=bge0/v4;prefixlen=24;
_protocol=ipv4;forwarding=on;
_ifeqname=lo0;_family=2;
#ifDelete ipv6# _ifeqname=lo0;_family=26;
_ifeqname=lo0;_aobjname=lo0/v4;_ipv4addr=127.0.0.1,;up=yes;
_ifeqname=lo0;_aobjname=lo0/v4;prefixlen=8;
```

Reboot the system which should re-start with ipv6 disabled.

```
hal# ipadm show-if
<table>
<thead>
<tr>
<th>IFNAME</th>
<th>STATE</th>
<th>CURRENT</th>
<th>PERSISTENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>lo0</td>
<td>ok</td>
<td>-m-v----4-4-</td>
<td></td>
</tr>
<tr>
<td>bge0</td>
<td>ok</td>
<td>bm-------4-4-</td>
<td></td>
</tr>
</tbody>
</table>
```

```
hal# routeadm -u
hal# routeadm | grep ndp
| disabled | svc://network/routing/ndp:default |
```

The /etc/hosts file may be edited to remove the ipv6 entries

```
# Internet host table
#
#:1:1 hal hal.local localhost loghost
127.0.0.1 hal hal.local localhost loghost hal.mydomain.co.uk
```

### 5.3 DNS Server

One of our requirements is to use the same domain name on both the WAN and LAN networks. In order to do this the LAN must include a DNS server to resolve the domain name `www.mydomain.co.uk` to a local machine.

The DNS server provision is provided by `bind` which is not installed by default. The following link provides the sequence that was followed: [http://www.logiqwest.com/dataCenter/Demos/RunBooks/DNS/DNSsetup.html](http://www.logiqwest.com/dataCenter/Demos/RunBooks/DNS/DNSsetup.html). Refer to the next section for more information on defining these files yourself.

Install the packages:

```
hal# pkg install service/network/dns/bind
```

Create the `named` information directory, there appears to be some conflicting wisdom as whether this should be created in `/var/named` or `/etc/named`. **My preference is `/etc/named` as this is static configuration information which is not volatile and `/etc` is the first place I look for system configuration information.**

```
hal# mkdir -p /etc/named
```

It is suggested that you use `h2n` to generate the DNS files. Download `h2n` from [ftp://ftp.hpl.hp.com/pub/h2n/h2n.tar.gz](ftp://ftp.hpl.hp.com/pub/h2n/h2n.tar.gz) and place in `tmp` directory.
hal# cd /tmp
hal# wget ftp://ftp.hpl.hp.com/pub/h2n/h2n.tar.gz

Extract the files
hal# tar zxvf h2n.tar.gz

Then change to the /etc/named directory and run the script h2n. Any names in the /etc/hosts file will be propagated into the bind configuration files so add any adding machines to the hosts file before running the script – you can remove these machines once DNS is running as the names will be resolved via the DNS server.

hal# cd /etc/named
hal# cp /tmp/h2n-2.56/h2n .
hal# ./h2n -d mydomain.co.uk -n 192.168.8 -u admin@mydomain.co.uk

This generates the files that you can now edit. Now fetch the named.root file from the Internet and copy to the db.cache file.

hal# wget http://www.internic.net/domain/named.root
hal# cp named.root /etc/named/db.cache

The list of files should now look something like this...

hal# ls
boot.cacheonly db.192.168.8 conf.cacheonly
db.cache named.boot db.127.0.0
db.mydomain named.conf

The files db.192.168.8 and db.mydomain may be edited to add nodes to your network. When editing make sure that you define the same names and IP addresses in both files.

File db.192.168.8 will look something like below. For all of the bind files the version field should be updated whenever the file is changed. In the examples then an integer value representing the calendar day, hour and minute is used in the form YYMMDhhmm rather than remembering to increment a number.

$ORIGIN 8.168.192.in-addr.arpa.
$TTL 86400
@ SOA hal.mydomain.co.uk. admin.mydomain.co.uk. ( 
 1208191705 ; Serial 7200 ; Refresh (2 hours)
 120 ; Retry (10 min) 
 604800 ; Expire (1 week)
 86400 ; Default TTL (1 day)
 )

; Name servers listed as forward lookup
@ IN NS hal.mydomain.co.uk.

; A list of machine names and addresses in reverse
200 IN PTR hal.mydomain.co.uk.
201 IN PTR www.mydomain.co.uk.

; Printers
30 IN PTR hplj2200d.mydomain.co.uk.
31 IN PTR hplj2605dn.mydomain.co.uk.

The db.mydomain will look something like below. The sample includes the mail server and Airprint entries

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
```bash
; root@sys:~# svcadm restart network/dns/client

$ORIGIN mydomain.co.uk.
$TTL 86400
@   SOA   hal.mydomain.co.uk admin.mydomain.co.uk ( 
1208181701 ; Version
7200 ; Refresh (2 hours)
120 ; Retry (10 min)
604800 ; Expire (1 week)
86400) ; Default TTL (1 day)

; List the name servers in use. Unresolved entries in other zones
; will go to our ISP's nameserver
@   IN NS   hal.mydomain.co.uk.
; Optional information on the machine type and O/S used for the server.
@   IN HINFO i386 Solaris

; Domain mailing address.
@   IN MX 10 www.mydomain.co.uk.

; A list of machine names and address, first is domain
@   IN A   192.168.8.200
hal   IN A   192.168.8.200
www   IN A   192.168.8.201

; List printers on the network
hpclj2605dn IN A   192.168.8.30
hplj2200d IN A   192.168.8.31

; Alias (canonical) names
mail IN CNAME www
colour IN CNAME hpclj2605dn
mono IN CNAME hplj3015dn

; Set up the name server (hal) and mail server (www)
@   IN TXT "v=spf1 ip4:192.168.8.0/28 a mx ~all"
www IN TXT "v=spf1 a -all";

; Set up DNS records for Airprint
lb._dns-sd._udp IN PTR @
b._dns-sd._udp IN PTR @
dr._dns-sd._udp IN PTR @
db._dns-sd._udp IN PTR @
cf._dns-sd._udp IN PTR @

; Set up printers for Airprint services
_cups._sub._ipp._tcp IN PTR colour._printer._tcp
 universal._sub._ipp._tcp IN PTR colour._printer._tcp

_cups._sub._ipp._tcp IN PTR mono._printer._tcp
 universal._sub._ipp._tcp IN PTR mono._printer._tcp
colour._printer._tcp IN SRV 0 0 631 hal.mydomain.co.uk.
mono._printer._tcp IN SRV 0 0 631 hal.mydomain.co.uk.

; THE FOLLOWING ENTRIES SHOULD BE CONTAINED ON A SINGLE LINE

colour._printer._tcp IN TXT ("txtvers=1" "qtotal=1" "rp-printers/colour"
"adminurl=http://hal:631/printers/colour" "note=Office printer"
"ty=HP LaserJet 2605dn" "product=(HP LaserJet 2605dn)" "transparent=t"
"copies=t" "duplex=T" "color=t" "pdl=application/octet-stream,
application/pdf,application/postscript,image/jpeg,image/png,image/urf"
"
"printer-type=0x8090DC" "URF=W8,SRGB24,CP1,RS600,DM3")
mono._printer._tcp IN TXT ("txtvers=1" "qtotl=1" "rp=printers/mono"
"adminurl=http://hal:631/printers/mono" "note=Basement printer"
"ty=HP LaserJet 3015dn" "product=(HP LaserJet 3015dn)" "transparent=t"
"copies=t" "Duplex=T" "color-f" "pdl=application/octet-stream, application/pdf,application/postscript,image/jpeg,image/png,image/urf"
"printer-type=0x829054" "URF=W8,SRGB24,CP1,RS600,DM3")

Finally the file named.boot should be copied to the /etc directory:

hal# cp /etc/named/named.boot /etc/named.conf

5.3.1 Manually defining DNS files

It is possible to build your own DNS files without using h2n, this must be performed with a lot of care as it is easy to make mistakes. On this system then the DNS files were subsequently updated as follows:

hal# ls
db.127.0.0  db.192.168.8  db.cache
db.localhost  db.mydomain

Where file db.127.0.0 is defined as:

; Reverse pointers for localhost
;
$TTL 86400
$ORIGIN 0.0.127.in-addr.arpa.
@ SOA localhost. root.localhost. (5 ; Serial increment on each edit.
7200 ; refresh (2 hours)
600 ; retry (10 min)
604800 ; expire (1 week)
86400 ; minimum (1 day)
)
IN NS localhost.
1 IN PTR localhost.

File db.192.168.8, db.cache and db.mydomain as defined in the previous section. File db.localhost is defined as:

; Loopback/localhost zone file
;
$TTL 1D
$ORIGIN localhost.
@ IN SOA @ root (4 ; Serial increment on each edit.
7200 ; refresh (2 hours)
600 ; retry (10 min)
604800 ; expire (1 week)
86400 ; minimum (1 day)
)
IN NS @
IN A 127.0.0.1

The file /etc/named.conf is defined as:

// MASTER and CACHING NAME SERVER for mydomain.co.uk
// Changelog:
OpenIndiana Small System Server Build
5.3 DNS Server

// Restart using
// % svcadm restart network/dns/server:default
// % svcadm restart network/dns/client:default
// %svcs -x network/dns/server:default
//
// Checking
// named-checkzone mydomain.co.uk /etc/named/db.mydomain
// named-checkzone localhost /etc/named/db.localhost
// named-checkconf /etc/named.conf
// host -l mydomain.co.uk
// host 192.168.8.1
options {
    // Location of configuration files.
    directory "/etc/named";

    // Version statement - inhibited for security
    version "Not currently available";

    // Optional - disable all transfers, slaves allowed in zones clauses
    allow-transfer {"none"; };

    // Closed DNS - permits only local IPs to issue recursive queries.
    // remove if an Open DNS required to support all users of add additional
    // range.
    allow-recursion {
        192.168.8.0/24;
    };

    // Forward DNS requests to our ISP.
    forwarders {
        212.23.3.100;
        212.23.6.100;
    };

    // Required zone for recursive queries.
    zone "." {  
        type hint;
        file "db.cache";
    };

    // Our local zone.
    zone "mydomain.co.uk" {  
        type master;
        file "db.mydomain";
        allow-update { none; };
    };

    // Required local host zone.
    zone "localhost" {  
        type master;
        file "db.localhost";
        allow-update { none; };
    };

    // Required local host reverse map.
    zone "0.0.127.in-addr.arpa" {  
        type master;
        file "db.127.0.0";
        allow-update { none; };
    };
//
5.3.2 Setting up bind

Bind 9 requires a final initial configuration step, otherwise an error is produced when starting the service. Run the following, this only needs to be done once.

```bash
hal# rndc -confgen -a
```

Check the network files `/etc/nsswitch.conf` which should include DNS entries:

```plaintext
... ipnodes: files dns
hosts: files dns ... 
```

Create or check the file `/etc/defaultdomain`

```bash
hal# vi /etc/defaultdomain
mydomain.co.uk
```

Execute the `domainname` command to set the domain as follows:

```bash
hal# domainname 'cat /etc/defaultdomain'
```

Set up the `/etc/resolv.conf` file, the first is the name of the domain (i.e. `mydomain.co.uk`):

```plaintext
# Localhost
domain mydomain.co.uk
nameserver 192.168.8.200
# ISP (Zen)
nameserver 212.23.3.100
nameserver 212.23.6.100
# Our Router is last resort
nameserver 192.168.8.1
```

The DNS server and client may now be started:

```bash
hal# svcadm enable network/dns/server:default
hal# svcadm enable network/dns/client:default
```

Check that the service is running:

```bash
hal# svcsc -x dns/server:default
svc:/network/dns/server:default (BIND DNS server)
State: online since Mon Aug 13 17:04:17 2012
  See: named(1M)
  See: /var/svc/log/network-dns-server:default.log
Impact: None.
```

If there is a problem then the SVC log may be interrogated:

```bash
hal# tail /var/svc/log/network-dns-server:default.log
[ Aug 13 15:01:19 Disabled. ]
[ Aug 13 17:03:54 Enabled. ]
[ Aug 13 17:03:54 Executing start method
```
Confirm that the DNS addresses are resolving correctly:

```
hal# host hal.mydomain.co.uk
hal.mydomain.co.uk has address 192.168.8.200

hal# host 192.168.8.201
201.8.168.192.in-addr.arpa domain name pointer www.mydomain.co.uk.

hal# host www.mydomain.co.uk
www.mydomain.co.uk has address 192.168.8.201
www.mydomain.co.uk mail is handled by 10 www.mydomain.co.uk.

hal# /usr/sbin/dig hal.mydomain.co.uk
; <<>> DiG 9.6-ESV-R7-P3 <<>> hal.mydomain.co.uk
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24174
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;hal.mydomain.co.uk. IN A

;; ANSWER SECTION:
hal.mydomain.co.uk. 86400 IN A 192.168.8.200

;; AUTHORITY SECTION:
mydomain.co.uk. 86400 IN NS hal.mydomain.co.uk.

Query time: 0 msec
SERVER: 192.168.8.200#53(192.168.8.200)
WHEN: Sat Feb 15 11:09:40 2014
MSG SIZE rcvd: 72
```

If the addresses are failing to resolve then there is a problem in your bind configuration files in /etc/named/.
Look at log files and edit the /etc/named/ files to fix the problem, the version field in the files should be updated when changed.
When the edit is complete then restart the DNS services as follows:

```
hal# svcadm restart network/dns/server:default
hal# svcadm restart network/dns/client
```

### 5.3.3 Bonjour / Zero Configuration Networking

DNS-SD service discovery records may be added to the name server allowing your iOS 5.0 mobile devices
to use the local printers without any configuration. Refer to the following links:

- [http://www.dns-sd.org/ServerSetup.html](http://www.dns-sd.org/ServerSetup.html)
- [http://www.dns-sd.org/ServerStaticSetup.html](http://www.dns-sd.org/ServerStaticSetup.html)

Modification of the name service configuration allows legacy both mono and colour laser printers to added to
the name service to allow network printing. The following was appended to file /etc/named/db.mydomain

---

31

www.jasspa.com

*oi_setup v1.25 2015/08/16 11:39:06*
allowing printing through CUPS (set up later):

```plaintext
; Setup the DNS records for browsing.
lb._dns -sd._udp IN PTR @ ; lb = legacy browse domain
b._dns -sd._udp IN PTR @ ; b = browse domain
dr._dns -sd._udp IN PTR @ ; dr = default reg domain
db._dns -sd._udp IN PTR @ ; db = default browse domain
cf._dns -sd._udp IN PTR @ ;

; Set up CUPS for iPad printing
_cups._sub._ipp._tcp IN PTR colour._printer._tcp
_universal._sub._ipp._tcp IN PTR colour._printer._tcp

colour._printer._tcp IN SRV 0 0 631 hal.mydomain.co.uk.
colour._printer._tcp IN TXT ( "txtvers=1" "qtotl=1"
"rp=printers/colour" "adminurl=http://hal:631/printers/colour"
"note=Office printer"
"ty=HP LaserJet 2605dn" "product=(HP LaserJet 2605dn)"
"transparent=t" "copies=t" "Duplex=T" "color=t"
"pdl=application/octet-stream,application/pdf,application/postscript,
image/jpeg,image/png,image/urf,image/text/plain,image/html"
"printer-type=0x8090DC"
"URF=W8,SRGB24,CP1,RS600")

_cups._sub._ipp._tcp IN PTR mono._printer._tcp
_universal._sub._ipp._tcp IN PTR mono._printer._tcp

mono._printer._tcp IN SRV 0 0 631 hal.mydomain.co.uk.
mono._printer._tcp IN TXT ( "txtvers=1" "qtotl=1"
"rp=printers/mono" "adminurl=http://hal:631/printers/mono"
"note=Basement printer"
"ty=HP LaserJet 3015dn" "product=(HP LaserJet 3015dn)"
"transparent=t" "copies=t" "Duplex=T" "color=f"
"pdl=application/octet-stream,application/pdf,application/postscript,
image/jpeg,image/png,image/urf,image/text/plain,image/html"
"printer-type=0x829054"
"URF=W8,SRGB24,CP1,RS600")
```

Note that the pdl= line should appear on a single line and is not split across lines as shown above. The name services should be re-started after editing

```bash
hal# svcadm restart network/dns/server:default
hal# svcadm restart network/dns/client
```

and printers should become available on the iOS 5.0 device. Note that duplex printing causes problems so when printing from the device then ensue that the Duplex option is OFF.

### 5.3.4 mDNS Service

For iOS 7.0 then DNS-SD service discovery records delivered in DNS are not sufficient and an mDNS service should be set up. The CUPS configuration is covered in more detail later.

The mDNS is quite useful for SSH and may be enabled without causing any problems. Edit the `/etc/-/nsswitch.conf` file which should include mdns entries for both `hosts` and `ipnodes`:

```plaintext`
...
# You must also set up the `/etc/resolv.conf` file for DNS name
# server lookup. See resolv.conf(4). For lookup via mdns
# svc:/network/dns/multicast:default must also be enabled. See mdnsd(1M)
hosts: files dns mdns
```
The multicast mDNS service may then be started:

```
hal# svcadm enable network/dns/multicast:default
```

Once the mDNS service is started then it is possible to connect to other local machines on the network that support Bonjour which have a dynamic address using the mDNS local nomenclature i.e. `hotname.local`. The following paragraphs may be used to confirm that mDNS is fully operational.

e.g. Connect to a host named “MacBook” which is using DHCP on the network and is not defined in DNS connecting using the host name from Solaris:

```
hal\% ssh -X -l user MacBook.local
```

Similarly, MacBook could connect to our server `hal` with the mDNS nomenclature:

```
macbook\% ssh -X -Y -l user hal.local
```

**Note:** with Apple OS X Mavericks using the X Window System XQuartz then SSH into OpenIndiana with the option `-Y` to prevent the X session from timing out.

### 5.4 DHCP Server

The DHCP server package is not installed by default, first download the package and install it, if not already installed.

```
hal# pkg install dhcp dhcpmgr
```

This should now appear as a service.

```
hal# svcs -a | grep dhcp
disabled 9:28:24 svc:/network/dhcp-server:default
```

To configure the DHCP server then it is easier to use the configuration manager `dhcpmgr`. Run from the command line:

```
hal# dhcpmgr
1. text files
2. /var/dhcp
3. Lease policy 1 day, clients can renew their leases
4. Specify DNS domain - this is our DNS server
5. Network Address = 192.168.8.0, Subnet 255.255.255.0
6. Local Area (LAN)
7. Use router discovery protocol
```

**Note:** if you are over-riding the WAN domain name IP look-up within the LAN and there are Apple devices in your network (iOS and OSX) then it is recommended that within the DHCP configuration that the LAN DNS server is advertised only i.e. no external DNS servers are referenced. The Apple devices use a dynamic DNS server ordering and if the device switches to an external DNS server then the local LAN names cannot be resolved. The DNS addresses advertised may be modified from `dhcpmgr`, the DNS addresses may be a subset of those addresses that are defined in `/etc/resolve.conf`.

The address wizard then appears. Configure the lease address range to match your network requirements in this case then 32 addresses starting from 192.168.8.128 have been used.

```
1. Number of IP Addresses = 32; Comment
2. Managed by server = "hal". Starting Address = 192.168.8.128
```
3. Confirm addresses (192.168.8.128 -> 192.168.8.159)
4. Select lease type (hal)
5. Lease type = dynamic

Make sure that "Router" is specified in the DHCP configuration (this field was absent in my case), if omitted then add with dhcpmgr. You can look at the options selected in file /var/dhcp/SUNWfiles1_dhcptab

```bash
hal# cat /var/dhcp/SUNWfiles1_dhcptab
# SUNWfiles1_dhcptab
#
# Do NOT edit this file by hand -- use dhtadm(1M) or dhcpmgr(1M) instead
#
Locale |m |9473040341197127681|: UTCoffst =0:
hal|m |6059593298627002369|: Include = Locale : Timeserv =192.168.8.200: LeaseTIm=86400:
LeaseNeg:DNSdomain="mydomain.co.uk":DNSserv =192.168.8.200:
192.168.8.0|m |3646226848309837826|: Subnet =255.255.255.0:
RDiscvyF=1:Broadcast=192.168.8.255:Router=192.168.8.1:
```

If the **Router** field is absent then restart dhcpmgr and add as follows:

```bash
hal# dhcpmgr
Select Macros
192.168.8.0 => Edit => Properties
Option Name: Router
Option Value: 192.168.8.1 (Send back to router).
=> Add {Ensure "Notify DHCP server of change" is checked.
=> OK.
Finished!
```

**Turn off any existing DHCP service that already exists on the network,** this may be running on any DSL router, then start the DHCP service on the server:

```bash
Start the service
hal# svcadm enable dhcp-server:default
hal# svc -xv dhcp-server:default
svc:/network/dhcp-server:default (DHCP server)
  State: online since 2 September 2012 09:39:36 BST
    See: man -M /usr/share/man -s 1M in.dhcpd
    See: /var/svc/log/network-dhcp-server:default.log
Impact: None.
```

Check that DHCP service is running properly by checking other DHCP enabled devices on the network are able to acquire their leases and are able to connect to the network and resolve addresses. Check that the DNS services are running correctly.

### 5.4.1 DHCP Logging

Add DHCP server logging, see OpenIndiana Wiki which is reproduced here.

Enable logging on the DHCP server

```bash
hal# echo "LOGGING_FACILITY=0" >> /etc/inet/dhcpsvc.conf
hal# svcadm restart dhcp-server
```

Add this line to `/etc/syslog.conf` to enable saving of these messages into a particular file. The two parts must be separated by TAB characters:

```bash
hal# me /etc/syslog.conf
local0.notice /var/log/dhcpsvc
```

touch the file to create it and restart the syslog:
hal# touch /var/log/dhcpsvc
hal# svcadm restart system -log

Note that syslog does not create log files itself and complains if one is not present at the moment of the
daemon’s startup or restart.
Enable log rotation to restrain the disk space requirements:

hal# cat << EOF >> /etc/logadm.conf
### Rotate DHCP/ipmon logs
/var/log/dhcpsvc -C 4 -s 1m -a '/usr/sbin/svcadm refresh system -log'
EOF

This uses the default log rotation engine logadm called from cron, if you use something else (newsyslog,
logrotate.d, etc.) configure that appropriately.

5.5 Print Server

A print server is required for global printing, the print queue is maintained on the server to allow devices to
print. All of the printers in the system are UNIX sympathetic network printers (i.e. they typically support
postscript). It is considered better to print through a server, especially from mobile devices such as laptops
when a long print job can be sent to the print server which deals with the request, the client device can 'fire
and forget’ and then be shut down if required whilst printing continues.

CUPS is used as the print server which needs to be installed. Where Hewlett Packard printers are used then
there are some HP specific filters provided by print/filter/hplip:

pkg install print/cups print/filter/hplip print/filter/ghostscript 
print/cups/system-config-printer

Once installation has completed then edit the CUPS configuration file /etc/cups/cupsd.conf and add a
listener for the printer Web Admin interface if you wish to administer remotely, by default it allows local
administration only.

hal# me /etc/cups/cupsd.conf
# Allow for remote access
Port 631 # Listen on the LAN interface , Port 631
# Comment out the local interface
#Listen localhost:631

Start the cups service(s) as required:

# svcadm enable cups/scheduler:default
# svcadm enable cups/in-lpd:default
# svc -a | grep cups
legacy_run Mar_01 lrc:/etc/rc2_d/S89apcupsd
online Mar_01 svc:/application/cups/scheduler:default
online Mar_01 svc:/application/cups/in-lpd:default

The print services may then be administered via the web interface with URL localhost:631 to set up the
printers.

5.5.1 AirPrint

To enable AirPrint with iOS 7 then mDNS should be enabled. Within the CUPS administration window then
the printer should be made sharable. The printer should be exported using mDNS via avahi. The Airprint
configuration for each printer may be generated with airprint-generate.py which is a python script that

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
interrogates CUPS and generates an Airprint configuration file in /etc/avahi/services. The syntax is relatively straight forward given that we have already set up the named DNS files.

To generate the configuration files then download airprint-generate.py from the web and generate the Airprint files:

```bash
hal# mkdir ~/airprint
hal# cd ~/airprint
hal# wget -O airprint-generate.py --no-check-certificate https://raw.githubusercontent.com/tjfontaine/airprint-generate/master/airprint-generate.py
hal# chmod +x airprint-generate.py
hal# ./airprint-generate.py
```

This will generate a .service file for each printer as follows, this example one is AirPrint-mono.service for a HP black and white laser printer which has been heavily edited:

```xml
<?xml version="1.0" ?>
<!DOCTYPE service-group SYSTEM 'avahi-service.dtd'>
<service-group>
  <name replace-wildcards="yes">AirPrint mono @ %h</name>
  <service>
    <type>_ipp._tcp</type>
    <subtype>_universal._sub._ipp._tcp</subtype>
    <port>631</port>
    <txt-record>txtvers=1</txt-record>
    <txt-record>qtotal=1</txt-record>
    <txt-record>rp= printers/mono</txt-record>
    <txt-record>ty=HP LaserJet 3015</txt-record>
    <txt-record>product=(HP LaserJet 3015)</txt-record>
    <txt-record>adminurl=http://hal.mydomain.co.uk:631/printers/mono</txt-record>
    <txt-record>note= Basement</txt-record>
    <txt-record>printer-state=3</txt-record>
    <txt-record>printer-type=0x829054</txt-record>
    <txt-record>Binary=T</txt-record>
    <txt-record>Color=F</txt-record>
    <txt-record>Transparent=T</txt-record>
    <txt-record>Duplex=T</txt-record>
    <txt-record>Copies=t</txt-record>
    <txt-record>pdl= application/pdf, application/postscript, image/jpeg, image/urf</txt-record>
    <txt-record>URF=W8,SRGB24,CP1,RS600,DM3</txt-record>
  </service>
</service-group>
```

The entries generated by the python script are long and the sd-dns records are short so remove some of the pdl= entries that are cups specific as they are not really needed.

Once the files are tweaked and cleaned up then move the .service files to /etc/avahi/service.

```bash
hal# mv *.service /etc/avahi/service
```

Restart the mDNS and avahi services and Airprint should show up on the network.

```bash
hal# svcadm restart network/dns/multicast:default
hal# svcadm restart system/avahi-bridge-dsd:default
```

### 5.6 Samba (SMB Share)

SMB file sharing may be performed natively by ZFS in OpenIndiana (See OpenIndia web site for further information) or the legacy Samba package from the extra repositories may be installed. In this configuration then we used the legacy Samba.
5.7 Firewall (IP Filter)

OpenIndiana includes an IP filter to protect the system, the package `ipfilter` should be installed, if not already present.

```
hal# pkg install ipfilter
```

Create the file `/etc/ipf/ipf.conf` in an editor that restricts access to the open ports of the system. The file looks something like:

```
#/ipf.conf

# IP Filter rules to be loaded during startup
# See ipf(4) manpage for more information on
# IP Filter rules syntax.
# To Reload rules:
#   $ ipf -Fa -f /etc/ipf/ipf.conf
# To Monitor log:
#   $ ipmon -a
# Summary of IPsec rules
#   $ ipfstat -h -i

# Outgoing - Allow any outbound traffic from this computer (and the response)
# To Reload rules:
#   $ ipf -Fa -f /etc/ipf/ipf.conf
# To Monitor log:
#   $ ipmon -a
# Summary of IPsec rules
#   $ ipfstat -h -i

# To Reload rules:
#   $ ipf -Fa -f /etc/ipf/ipf.conf
# To Monitor log:
#   $ ipmon -a
# Summary of IPsec rules
#   $ ipfstat -h -i

# Outgoing - Allow any outbound traffic from this computer (and the response)
pass out quick on bge0 all keep state

# Allow Looppback traffic
pass in quick on lo0 all
pass out quick on lo0 all

# DNS
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 53 keep state
pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.200 port = 53 keep state

# BOOTP/DHCP Server 0 placed here as may be any address.
pass in quick on bge0 proto udp from any port = 68 to any port = 67

# block from non-routeable addresses
block in quick from 10.0.0.0/8
block in quick from 172.16.0.0/12
block in quick from 192.168.0.0/16

# NDN

# Allow mDNS from local area network
pass in quick on bge0 proto udp from 192.168.8.0/24 to 224.0.0.251 port = 5353

# NTP

# Allow NTP from local area network
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 123 keep state
pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.200 port = 123 keep state

# FTP

# FTP is a TCP based service exclusively. There is no UDP component to FTP.
# FTP is an unusual service in that it utilizes two ports, a 'data' port and a
# 'command' port (also known as the control port). Traditionally these are
# port 21 for the command port and port 20 for the data port. The confusion
# begins however, when we find that depending on the mode, the data port is
# not always on port 20.

# In active mode FTP the client connects from a random unprivileged port (N >
# 1024) to the FTP server's command port, port 21. Then, the client starts
# listening to port N+1 and sends the FTP command PORT N+1 to the FTP server.
# The server will then connect back to the client's specified data port from
# its local data port, which is port 20.
```
In order to resolve the issue of the server initiating the connection to the client, a different method for FTP connections was developed. This is known as passive mode, or PASSV, after the command used by the client to tell the server it is in passive mode.

In passive mode FTP, the client initiates both connections to the server, solving the problem of firewalls filtering the incoming data port connection to the client from the server. When opening an FTP connection, the client opens two random unprivileged ports locally (N > 1024 and N+1). The first port contacts the server on port 21, but instead of issuing a PORT command allowing the server to connect back to its data port, the client will issue the PASV command. The result of this is that the server then opens a random unprivileged port (P > 1024) and sends the PORT P command back to the client. The client then initiates the connection from port N+1 to port P on the server to transfer data.

Active FTP:
command: client >1024 -> server 21
data: client >1024 -> server 20

Passive FTP:
command: client >1024 -> server 21
data: client >1024 -> server >1024

Pass in quick on bge0 proto tcp from 192.168.8.200 port > 1023 to X.X.X.X port = 21 flags S keep state
Passive FTP
Pass in quick on bge0 proto tcp from 194.164.217.96 port > 1023 to X.X.X.X port 51000 =>51501 flags S keep state
Active FTP
Pass in quick on bge0 proto tcp from 194.164.217.96 port = 20 to X.X.X.X port 51000 =>51501 flags S keep state
Pass in quick on bge0 proto tcp from 194.164.217.96 port = 20 to X.X.X.X port > 1023 flags S keep state

Use i pnat instead
map eri0 0/0 -> 0/32 proxy port 21 ftp/tcp

Printer
International Printing Protocol (IPP)
Pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 631 keep state
Pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.200 port = 631 keep state
LPD
Pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 515 keep state
Pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.200 port = 515 keep state
Printers themselves
Pass in quick on bge0 proto tcp from 192.168.8.30/32 to 192.168.8.200 port = 910 keep state
Pass in quick on bge0 proto tcp from 192.168.8.31/32 to 192.168.8.200 port = 910 keep state
Pass in quick on bge0 proto tcp from 192.168.8.200 port > 1023 to X.X.X.X port = 21 flags S keep state

POP3
Pass in quick on bge0 proto tcp from 192.168.8.200 port > 1023 to X.X.X.X port = 21 flags S keep state

Ping
Pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 110 keep state

MySQL/Postgres SQL
Allow Postgres SQL access local domain (iCal server)
Pass in quick on bge0 proto tcp from 192.168.8.201/32 to 192.168.8.200 port = 5432 keep state
Pass in quick on bge0 proto tcp from 192.168.8.201/32 to 192.168.8.200 port = 5306 keep state

DAVical services
Allow access to the DAVICAL server
Port 8443 - CardDAV service with SSL (Internal/External no admin)
Pass in quick on bge0 proto tcp from any to 192.168.8.10 port = 8443 keep state
Pass in quick on bge0 proto tcp from any to 192.168.8.10 port = 8443 keep state
Port 8008 - CalDAV service no SSL (Local domain only)
Pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.10 port = 8008 keep state
Pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.10 port = 8081 keep state

KIl remote login
Allow 17 addresses below 192.168.8.32
Pass in quick on bge0 proto tcp from 192.168.8.0/27 to any port = 177 keep state
Pass in quick on bge0 proto tcp from 192.168.8.0/27 to 192.168.8.200 port = 6000 keep state
Pass in quick on bge0 proto tcp from 192.168.8.0/27 to 192.168.8.200 port = 6000 keep state
We also need the font server on 7100
Pass in quick on bge0 proto tcp from 192.168.8.0/27 to 192.168.8.200 port = 7100 keep state
Pass in quick on bge0 proto tcp from 192.168.8.0/27 to 192.168.8.200 port = 7100 keep state

Samba Access
NETBIO Name Service - used by nmbd
Pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 137 keep state

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
pass in quick on bge0 proto udp from 192.168.8.0/24 to any port = 137 keep state
# NETBIOS Datagram Service - used by nmbd
# pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 138 keep state
pass in quick on bge0 proto tcp from 192.168.8.0/24 to any port = 138 keep state
# NETBIOS Session Service - used by nmbd
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 139 keep state
# Used by nmbd
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 445 keep state
# smtp
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 587 keep state
# imap + SSL/TLS
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 993 keep state
# SMF - Allow ssh inbound but limit to site only
pass in quick on bge0 proto tcp from 192.168.8.0/24 to any port = 22 flags S keep state
# CVS - Allow cvs inbound but limit to site only
# cvspserver
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 2401 keep state
pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.200 port = 2401 keep state
# HTTP
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.0/24 port = 80 keep state
# HTTPS
pass in quick on bge0 proto tcp from 192.168.8.128/25 to any port = 443 keep state
global access
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 443 keep state
# Local access
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 443 keep state
# SunNet/SunRay Services
# FTP
pass in quick on bge0 proto tcp from 192.168.8.128/25 to any port = 69 keep state
# SunRay
pass in quick on bge0 proto tcp from 192.168.8.128/25 to 192.168.8.200 port = 7007 flags S keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to 192.168.8.200 port = 7008 flags S keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to 192.168.8.200 port = 7009 flags S keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to any port = 7009 keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to 192.168.8.200 port = 7010 flags S keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to 192.168.8.200 port = 7011 flags S keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to 192.168.8.200 port = 7012 flags S keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to 192.168.8.200 port = 7013 flags S keep state
pass in quick on bge0 proto tcp from 192.168.8.128/25 to any port = 7013 keep state
pass in quick on bge0 proto udp from 192.168.8.128/25 to any port = 40000 >> 42000 keep state
# NFS Services - use sshd rpcinfo -p
# rpcbind
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 111 flags S keep state
pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.0/24 port = 111 keep state
# krb5
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 4045 flags S keep state
pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.0/24 port = 4045 keep state
# nfs
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 2049 flags S keep state
pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.0/24 port = 2049 keep state
# nfs miscellaneous ports
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 32768 < 33000 flags S keep state
pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.0/24 port = 32768 < 33000 keep state
# SunRPC - Linux X-server
pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200 port = 6000 flags S keep state
# Block everything else
block in log on bge0 all

Once the rules are defined then check the ipf SMF service is automatically loaded on any re-start. The OpenIndiana release is not configured to reload the ipf rules by default and the SMF service may need to be customised. The /network/ipfilter:default rules for the following properties should be defined as
follows:

```
firewall_config_default/policy astring custom
firewall_config_default/custom_policy_file astring /etc/ipf/ipf.conf
```

View the properties of the service:

```
root@hal:~# svcprop /network/ipfilter:default
....
firewall_config_default/policy astring custom
firewall_config_default/custom_policy_file astring /etc/ipf/ipf.conf
....
```

If the properties are not defined with these settings then modify them as follows and refresh the service:

```
root@hal:~# svccfg -s /network/ipfilter:default
svc:/network/ipfilter:default > listprop firewall_config_default
firewall_config_default/apply_to astring com.sun, fw_configuration
firewall_config_default/exceptions astring
....
svc:/network/ipfilter:default > setprop firewall_config_default/policy = astring: custom
svc:/network/ipfilter:default > setprop firewall_config_default/custom_policy_file = \
    astring: /etc/ipf/ipf.conf
svc:/network/ipfilter:default > listprop firewall_config_default
svc:/network/ipfilter:default > quit
root@hal:~# svcadm refresh /network/ipfilter:default
```

With the SMF service modified then start the service:

```
hal# svcs -a | grep ipf
disabled 13:31:28 svc:/network/ipfilter:default
hal# svcadm enable network/ipfilter:default
```

Check the status of the service.

```
www# svcs -xv network/ipfilter:default
svc:/network/ipfilter:default (IP Filter)
State: online since Sat Sep  1 13:45:27 2012
    See: man -M /usr/share/man -s 5 ipfilter
    See: /var/svc/log/network-ipfilter:default.log
Impact: None.
```

Ideally run something like `zenmap` from another location to check connectivity of the host. If the rules need to be fixed then edit `ipf.conf` and re-start the service or manually reload the rules from the command line as follows:

```
root@hal:~# ipf -Fa -f /etc/ipf/ipf.conf
```

The rules that are currently loaded may be verified as follows:

```
root@hal:~# ipfstat -h -i
0 pass in quick on lo0 all
0 pass in quick on bge0 proto tcp from 192.168.8.0/24 to 192.168.8.200/32 port = domain keep state
0 pass in quick on bge0 proto udp from 192.168.8.0/24 to 192.168.8.200/32 port = domain keep state
....
```

The firewall may be interactively monitored as follows, use Ctrl-C to break out:

```
root@hal:~# ipmon -a
16/02/2014 10:05:32.910434 STATE:NEW 192.168.8.200,5353 -> 224.0.0.251,5353 PR udp
16/02/2014 10:06:52.907592 STATE:NEW 192.168.8.200,123 -> 149.255.102.233,123 PR udp
16/02/2014 10:05:32.910434 STATE:NEW 192.168.8.200,123 -> 224.0.1.1,123 PR udp
16/02/2014 10:05:32.910434 STATE:NEW 192.168.8.200,123 -> 224.0.1.1,123 PR udp
```

```
Ctrl-C
```
6 UPS Protection

For protection against power outage then a APC 620inet UPS has been used which provides a serial communication interface to the server and is used in conjunction with apcupsd. The server does not include a serial interface by default and a MOXA CP-102EL-DB9M 2-port RS-232 low profile PCI Express serial board has been used. A Keyspan USB-serial adapter was tried but did not play well with the UPS and resulted in a lot of intermittent communication disconnections. The MOXA card proved to be much more reliable (albeit expensive).

6.1 Installing MOXA serial card

Power off the system and install the MOXA serial card, power on and download the Solaris 10 Moxa CP-102E/EL drivers from www.moxa.com.

```
hal# unzip driv_solaris10_smart_i386_v1.0_build_10081617.zip
Archive: driv_solaris10_smart_i386_v1.0_build_10081617.zip
   inflating: version.txt
   inflating: readme.txt
   inflating: driv_solaris10_smart_i386_v1.0_build_10081617.pkg
hal# pkgadd -d driv_solaris10_smart_i386_v1.0_build_10081617.pkg
```

The following packages are available:

```
1 MxSIBoard MOXA Smartio/Industio Multiport Serial Board Driver
   (x86/x64) v1.0 (Build 10081617)
```

Select package(s) you wish to process (or 'all' to process all packages). (default: all) [?,??,q]: all

```
Processing package instance <MxSIBoard> from <tmp/drv_solaris10_smart_i386_v1.0_build_10081617.pkg>
MOXA Smartio/Industio Multiport Serial Board Driver(x86/x64) v1.0 (Build 10081617)
Moxa Inc.

The selected base directory </usr/lib/MxSIBoard> must exist before installation is attempted.

Do you want this directory created now [y,n,?,q] y
```

Do you want to continue with the installation of <MxSIBoard> [y,n,?] y

```
Installing MOXA Smartio/Industio Multiport Serial Board Driver as <MxSIBoard>

## Installing part 1 of 1.
/usr/lib/MxSIBoard/README
/usr/lib/MxSIBoard/amd64/mxsiBOARD
```

## Executing postinstall script.

The following MOXA Smartio/Industio board(s) have found and installed.

MOXA Smartio/Industio CP-102EL Series (ttyMUE0-ttyMUE1)

Done.

Installation of <MxSIBoard> was successful.

Test that the serial board is working correctly

```bash
hal# prtconf -v | grep MOXA
value='MOXA Smartio/Industio CP-102EL Series (ttyMUE0-ttyMUE1)'

hal# stty -a < /dev/ttyMUE0
speed 9600 baud; rows 0; columns 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>;
eol2 = <undef>; swtc = <undef>; start = ^Q; stop = ^S; susp = ^Z;
d susp = ^?; rprnt = ^R; w erase = ^W; lnext = ^V; flush = ^O;
-parn b -parodd cs8 -hupcl -cstopb cread -clocal -crtscts
-lgbrk brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl ixon
-ixoff -iucv -ixany imaxbel opost -olcuc -ocrnl onlcr -onocr -onlret
nofill -ofdel n10 cr0 tab3 bs0 vt0 ff0 isig ican on iexten echo echoe
ch echo = echo1 = noflsh -xcase -tostop = echopr = echoctl echoke
```

### 6.2 Installing apcupsd

**apcupsd** is an excellent open source daemon for controlling APC UPSes which works wonderfully well. Download the latest version of APCUPSD from [http://www.apcupsd.org/](http://www.apcupsd.org/) and unpack it.

```bash
tar xzvf Downloads/apcupsd-3.14.10.tar.gz
```

Once unpacked then build it locally

```bash
cd apcupsd-3.14.10/
./configure --enable-usb --with-upstype=usb --with-upscale=usb --prefix=/usr --sbindir=/sbin --with-log-dir=/var/log/apcupsd
make
```

Once built then install as root

```bash
bob@hal:~/apcupsd-3.14.10$ sudo make install
src
src/lib
src/drivers
src/drivers/apcsmart
```
6.2 Installing apcupsd

src/drivers/dumb
src/drivers/net
src/drivers/pcnet
src/drivers/usb
src/drivers/usb/generic
src/drivers/snmpLite
src/libusbhid
COPY apcupsd => /sbin/apcupsd
COPY apctest => /sbin/apctest
COPY apcaccess => /sbin/apcaccess
COPY smtp => /sbin/smtp

platforms
COPY apcupsd.conf => /etc/opt/apcupsd/apcupsd.conf
COPY changeme => /etc/opt/apcupsd/changeme
COPY commfailure => /etc/opt/apcupsd/commfailure
COPY commok => /etc/opt/apcupsd/commok
COPY offbattery => /etc/opt/apcupsd/offbattery
COPY onbattery => /etc/opt/apcupsd/onbattery

 platforms/sun

------------------------------
Sun distribution installation
------------------------------
COPY apcupsd => /etc/init.d/apcupsd
LN //etc/rc0.d/K21apcupsd -> ../init.d/apcupsd
LN //etc/rc1.d/S89apcupsd -> ../init.d/apcupsd
LN //etc/rc2.d/S89apcupsd -> ../init.d/apcupsd

------------------------------
apcupsd script installation for Solaris Solaris complete.
You should now edit /etc/opt/apcupsd/apcupsd.conf to correspond
to your setup then start the apcupsd daemon with:

/etc/init.d/apcupsd start

Thereafter when you reboot, it will be stopped and started
automatically.

------------------------------
Configuring ugen driver to match APC UPSes...

Driver (ugen) is already installed.

NOTE:
"(usbif51d,class3) already in use" and
"Driver (ugen) is already installed"
errors may be safely ignored.

------------------------------
Driver configured. You must PERFORM A RECONFIGURE
BOOT "reboot -- -r" before running Apcupsd.
------------------------------
COPY apccontrol => /etc/opt/apcupsd/apccontrol
COPY apcupsd.8 => /usr/share/man/man8/apcupsd.8
COPY apcaccess.8 => /usr/share/man/man8/apcaccess.8
COPY apctest.8 => /usr/share/man/man8/apctest.8
COPY apccontrol.8 => /usr/share/man/man8/apccontrol.8
COPY apcupsd.conf.5 => /usr/share/man/man5/apcupsd.conf.5

bob@hal:~/apcupsd-3.14.10$
6.3 apcupsd logging

If you require logs then make the logging directory `/var/log/apcupsd`, the logging directory location was specified as part of the build configuration.

```bash
mkdir -p /var/log/apcupsd
```

The `apcupsd.conf` file defines the location of the event file with variable `EVENTSFILE` which may be defined as `/var/log/apcupsd/apcupsd.events`

6.4 apcupsd configuration

Configure `apcupsd` and edit `/etc/opt/apcupsd/apcupsd.conf`. With serial communication with the MOXA card connected to the APC u620inet device then the configuration entries are:

```bash
UPSTYPE apcsmart
DEVICE /dev/ttyMUE0
```

Edit `/sbin/rc0` and add the following at the bottom of the script. This kills power to the UPS.

```bash
# see if this is a powerfail situation
if [ -f /etc/powerfail ]; then
  echo
  echo "APCUPSD will power off the UPS"
  echo
  /etc/opt/apcupsd/apccontrol killpower
  echo
  echo "Please ensure that the UPS has powered off before rebooting"
  echo
  echo "Otherwise, the UPS may cut the power during the reboot !!!"
  echo
  exit 0
fi
```

`apcupsd` may now be started with the command line:

```
hal# /etc/init.d/apcupsd start
```

Check the logs `/var/log/apcupsd.log` and then run through the power down checks.

6.5 apcupsd starting and stopping

`apcupsd` may be started and stopped with the following command line:

```
hal# /etc/init.d/apcupsd start
Stopping apcupsd power management ... Failed.
hal# /etc/init.d/apcupsd start
Starting apcupsd power management ... Done.
```

6.6 apcupsd USB configuration

This system outlined here is running with serial communication, however it was exercised with a USB APC UPS which was lying around (the UPS was a little too big for the low power consumption of the HP N40L and was swapped out for a smaller UPS).

On a HP N40L when the USB is not fully initialised then `apcupsd` fails and the following appears in the log:

```
www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
```
This seemed to be a problem with the boot up and the USB sub-system was not completely initialised before the apcupsd daemon was started. This may be fixed by editing the /etc/init.d/apcupsd script and to crudely add a 20s delay in the start up sequence. The boot-up time is not considered critical as the system will run 24/7 and are expecting an uptime measured in months.

```bash
hal# vi /etc/init.d/apcupsd

case "\$1" in
  start)
    rm -f \${POWERFAILDIR}/powerfail
    echo "Starting apcupsd power management waiting for USB ..\c"
    sleep 20
    echo "Starting apcupsd power management ..\c"
    \${SBINDIR}/apcupsd || return=" Failed."
    touch \${LOCKDIR}/apcupsd
    echo "\$return"
  ;;
```

## 7 ZFS File System

At this point in the system build we have been running from the system disk and a basic system is running with UPS protection. The system is provisioning basic networking name resolution services including DNS, mDNS and DHCP, the system is firewalled. Printing services are available. The next step is to add the data storage disks to the system which are used for high capacity data storage.

In our system then 2x3TB disks are to be set up as a single mirrored file system (RAID-0), in addition we have an additional 250GB disk that was shipped with the system we do not really need this disk but it has been left in the system and may be used for more volatile data that we do not mind loosing. Were more disks to be present then we could consider a different RAID configuration.

First find the disks in the system:

```bash
hal# cfgadm -al sata
sata0 /0:: dsk/ c3t0d0 disk connected configured ok
sata0 /1:: dsk/ c3t1d0 disk connected configured ok
sata0 /2:: dsk/ c3t2d0 disk connected configured ok
```

to see what the disks are then run format to list them and then quit out of the utility:

```bash
hal# sudo format
Searching for disks... done

c3t0d0: configured with capacity of 2794.52GB

c3t1d0: configured with capacity of 2794.52GB

AVAILABLE DISK SELECTIONS:
 0. c3t0d0 <ATA-KDC WD30EZRX-00M-0A80-2.73TB>
     /pci0,0/pci103c,1609@01/disk@0,0
 1. c3t1d0 <ATA-KDC WD30EZRX-00M-0A80-2.73TB>
     /pci0,0/pci103c,1609@01/disk@1,0
 2. c3t2d0 <ATA-VB0250EAEVER-HPG7 cyl 30399 alt 2 hd 255 sec 63>
     /pci0,0/pci103c,1609@01/disk@2,0
```

www.jasspa.com

oi_setup v1.25 2015/08/16 11:39:06
The two 3TB disks c3t0d0 and c3t1d0 will form our disk mirror. Create a new zfs pool with a name of your choice, in this case I used tank01 for want of a better name.

```
hal# zpool create tank01 mirror c3t0d0 c3t1d0
```

Create some directories with some folders in the pool, depending on the use then some of the file systems are assigned a specific mount point in the file system.

```
hal# zfs create tank01/udata
hal# zfs set mountpoint=/tv tank01/udata
hal# zfs create tank01/mail
hal# zfs create tank01/aux
hal# zfs set mountpoint=/aux tank01/aux
hal# zfs create tank01/cvs
hal# zfs create tank01/www
```

Later in zone www we use tank01/mail as the /home directory and tank01/www as the /www directory for Apache web services. These file systems are not mounted at any special location in the global zone. The spare 250GB disk may be formatted and mounted with ZFS but will not be mirrored.

## 8 Setting up WAN server

In this section we consider setting up a separate virtual server called “www” or “www.mydomain.co.uk” which will provide all of the WAN facing services. This is partitioned from the rest of the system. “www” will provide services such as Mail, HTTP web services including Calendar services, Address book, WebDAV and HTTP(S).

As a recap then currently in the system we have a single root file system on a SSD disk(s) and 2x3TB HDD and 1x250GB supplied with the system which are mounted. Up to now then everything has been installed and configured on the SSD drive.

### 8.1 Zone Preparation

The www zone will host the WAN facing services with host name www.mydomain.co.uk. First create a new file system for zones in the root pool and mount it at the root /zones i.e.

```
hal# zfs create rpool/zones
hal# zfs set mountpoint=/zones rpool/zones
hal# zfs create rpool/zones/www
```

Change the permissions so that only root has access.

```
hal# chmod go-rwx /zones/www
```

The www zone will be created as a physical zone with a VNIC, this seems to play better with the manually created static IP address (created earlier). Before the zone is created then a VNIC is created for that zone.

### 8.1.1 Creating a VNIC

From the global zone then create a virtual network, the state of the current network may be interrogated as follows:
Create the new virtual network for our www zone and connect it to our physical connection.

```
hal# dladm create -vnic -l bge0 vnic0
hal# dladm show-link
```

The virtual network now exists in the system, no further configuration is required in the Global zone, the interface will be configured from within the zone to which it is attached.

8.2 Zone Creation

Create the zone www as root

```
zonecfg -z www
zonecfg:www> create
zonecfg:www> set zonepath=/zones/www
zonecfg:www> set autoboot=true
zonecfg:www> set ip-type=exclusive
zonecfg:www> add net
zonecfg:www:net> set physical=vnic0
zonecfg:www:net> end
zonecfg:www> add fs
zonecfg:www:fs> set dir=/home
zonecfg:www:fs> set special=/tank01/mail
zonecfg:www:fs> set type=lofs
zonecfg:www:fs> end
zonecfg:www> add fs
zonecfg:www:fs> set dir=/www
zonecfg:www:fs> set special=/tank01/www
zonecfg:www:fs> set type=lofs
zonecfg:www:fs> end
zonecfg:www> info
zonenname: www
zonepath: /zones/www
brand: ipkg
autoboot: true
bootargs: 
pool: 
limitpriv: 
scheduling-class: 
ip-type: exclusive
hostid: 
fs-allowed: 
fs:
```
Go and make a cup of tea, the command will take some time as the zone is created (depending on the speed of the system). Then Verify what has been done:

```
hal# zonecfg -z www info
zonename: www
zonepath: /zones/www
brand: ipkg
```

See the current zone state

```
hal# zoneadm list -vc
   ID  NAME STATUS PATH BRAND IP
 0  global   running / ipkg shared
 1   www   running /zones/www ipkg excl
hal#
```

Now try to boot the zone

```
hal# zoneadm -z www boot
```

Assuming it successfully boots then login to the zone and initialise the zone using the on-screen prompts, this is the same as installing a new system.

```
hal# zlogin -C www
```

To subsequently exit the zone `www` from the console then exit the zone then exit the console and use . to close the connection i.e.:

```
www# exit
www# ~.
[Connection to zone 'www' pts/6 closed]
hal#
```

### 8.3 Zone Static IP

The zone is now connected to the VNIC `vnic0` and needs to be configured with an IP address, we will be assigning IP address 192.168.8.201.

```
www# ipadm create-addr -T static -a 192.168.8.201 vnic0/v4address
```

www.jasspa.com
Look at the status

```
www# ipadm show-if vnic0
IFNAME    STATE    CURRENT   PERSISTENT
vnic0      ok        bm--------46 -46
```

```
www# ipadm show-addr
ADDROBJ       TYPE    STATE   ADDR
lo0/v4        static   ok      127.0.0.1/8
vnic0/v4address static   ok      192.168.8.201/24
lo0/v6        static   ok      ::1/128
vnic1/v4      static   disabled  192.168.8.201/24
```

Enable the interface

```
www# ifconfig vnic0 inet 192.168.8.201 up
```

As with the static networking configuration in the globalzone then add the default route and set up the network.

```
www# route -p add default 192.168.8.1
add net default: gateway 192.168.8.1
add persistent net default: gateway 192.168.8.1
```

```
www# netstat -r
Routing Table: IPv4
---------------------------------------- ------ ---- ----------
Destination Gateway Flags Ref Use Interface
-------------------- -------------------- ----- ----- ----------
default 192.168.8.1 UG 11 3222256
localhost localhost UH 2 19524 lo0
192.168.8.0 www U 12 14166533 vnic0

Routing Table: IPv6
---------------------------------------- ------ ---- ------- ----
Destination  /Mask Gateway Flags Ref Use If
-------------------- --------------------------- ----- --- ------- ----
localhost localhost UH 2 6748 lo0
```

Now that the physical network has been setup then configure the routing information. The DNS server will be our global zone (hal). Check the network files /etc/nsswitch.conf which should include DNS entries and optionally the mDNS entries:

```
...
ipnodes: files dns mdns
hosts: files dns mdns
...
```

Create or check the file /etc/defaultdomain

```
www# vi /etc/defaultdomain
mydomain.co.uk
```

Execute the domainname command to set the domain as follows:

```
www# domainname 'cat /etc/defaultdomain`
```

Set up the /etc/resolv.conf file, the first is the name of the domain (i.e. mydomain.co.uk) and then we use the global zone hostname to resolve addresses 192.168.8.200:

```
www# Localhost
domain mydomain.co.uk
nameserver 192.168.8.200
```

Enable the DNS client and mDNS services if required:
root@www:~# svcadm enable network/dns/client:default
root@www:~# svcadm enable network/dns/multicast:default
root@www:~# svcscs -a | grep dns

Test that names are resolving correctly though DNS and mDNS.

9 Server Certificate

We are using a static IP, valid DNS domain name with SSL services and require the system to respond legitimately to any client so a legitimate SSL certificate is required rather than a self signed root certificate. The trust authority used was Trustico www.trustico.co.uk QuickSSL Premium RN, valid for 36 months, with a single named server with Common Name CN=www.mydomain.co.uk.

The certificate location should be managed making it easier to utilise in the different components. A directory at the root level called /CA3yr has been created especially for the certificates, this could have been created in /etc and is a personal preference. The directory and all contained files should be readable by root only, there should be no write access.

The certificates from the trust authority are named according to their content as follows:

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ca3yr_cert.crt</td>
<td>mydomain Certificate.</td>
</tr>
<tr>
<td>ca3yr_cert.txt</td>
<td>Text of mydomain Certificate.</td>
</tr>
<tr>
<td>ca3yr_cert.key</td>
<td>Private RSA key for the certificate.</td>
</tr>
<tr>
<td>ca3yr_introot.crt</td>
<td>Intermediate root certificate(s).</td>
</tr>
<tr>
<td>ca3yr_introot.txt</td>
<td>Text of Intermediate root certificate.</td>
</tr>
<tr>
<td>ca3yr_ca-bundle.crt</td>
<td>Intermediate chain + mydomain excluding root certificate.</td>
</tr>
<tr>
<td>ca3yr_cert-chain.crt</td>
<td>mydomain + intermediate chain + root certificate chain.</td>
</tr>
<tr>
<td>GeoTrust_Global_CA.cer</td>
<td>Root certificate</td>
</tr>
</tbody>
</table>

Table 5: Certificate Naming Convention

The files ca3yr_ca-bundle.crt and ca3yr_cert-chain.crt are created as follows:

`www# cat ca3yr_introot.crt ca3yr_cert.crt > ca3yr_ca-bundle.crt`

`www# cat ca3yr_cert.crt ca3yr_introot.crt GeoTrust_Global_CA.cer > ca3yr_cert-chain.crt`

The private certificate may be viewed using openssl (the .txt version of the files):

`openssl x509 -in ca3yr_cert.crt -text -noout`

Ensure that ALL files are owned by root, not writable and only readable by root.

`www# cd /`
`www# chown -R root:root /CA3yr`
`www# chmod -R a-w /CA3yr`
`www# chmod -R go-r /CA3yr`
10 E-Mail Service

OpenIndiana installs sendmail by default for our server then we are going to use postfix as the mailer daemon so sendmail will have be removed. Two instances of postfix are required to be running as we need a mail relay to send mail to our ISP in addition to a SSL protected SMTPS server allowing mail to be sent through the server. dovecot is used to provide IMAP services for the mail clients. fetchmail is used to collect mail from our ISP through procmail to the mail daemon.

On the www server then we will set up some user accounts specifically for E-Mail these are disconnected from regular user accounts on the server and only used for E-Mail, this means that E-Mail accounts may be provided independently of other services. We previously created a directory in the global zone called /tank01/mail which has been imported into the zone as /home this will be the file system area that is used to hold the user mailboxes. There are lots of different methods that could be used for E-Mail, for this configuration as there are so few users we have not needed to use any sort of network information service (NIS, LDAP etc.). How this is set up will be determined by the individual site requirements.

10.1 Mail packages

Get the new packages that we are going to use for the mail subsystem. The default MTA sendmail will be replaced by postfix. The “Spec Extra Repositories” need to be declared to the package manager in order to install some of these packages.

```
www# pkg install postfix
www# pkg install fetchmail
www# pkg install dovecot
www# pkg install procmail
```

10.2 Creating user accounts

Before setting up the mail server then create at least one user account which can be used for testing the mail server configuration. For this server configuration then we are using a user_id and group_id base of 1000 and assigning them manually.

```
www# groupadd -g 1008 bob
www# useradd -d /home/bob -c "Bob Fullname" -G bob -m -u 1008 bob
www# groupadd -g 1009 alice
www# useradd -d /home/alice -c "Alice Fullname" -G alice -m -u 1009 alice
```

If you make a mistake then the user account may be deleted with:

```
www# userdel -r bob
www# groupdel bob
```

The mail accounts require a password at some stage which may be assigned as follows:

```
www# passwd bob
newpassword
newpassword
```

For each user account then prepare the directory for mail. The /Procmail directory is used for logging. If you do not require this then disable the LOGFILE option in the .procmailrc below. Create the appropriate directories and default .procmailrc for each user.

```
www# mkdir -p /home/bob/Procmail
www# mkdir -p /home/bob/mail
www# cat << EOF >> /home/bob/.procmailrc
# It is essential that you set SHELL to a Bourne-type shell if
```

www.jasspa.com

oi_setup v1.25 2015/08/16 11:39:06
# external commands are run from your procmailrc, for example if
# you use rc.spamassassin, rc.quarantine, or other advanced recipes.
# Setting SHELL should not be needed for the simple sorting recipes in
# this step-by-step section, but to be safe and to future proof your
# procmailrc, set it anyway! Details are in Check Your `$SHELL` and `$PATH`.
SHELL=/bin/sh

# Directory for storing procmail configuration and log files
# You can name this variable anything you like, for example
# PROCMAILDIR, or don’t set it (but then don’t refer to it!)
PMDIR=$HOME/Procmail

# LOGFILE should be specified ASAP so everything below it is logged
# Put `##` before the next line if you want no logging (not recommended)
LOGFILE=$PMDIR/procmail.log

# To insert a blank line between each message’s log entry in $LOGFILE,
# uncomment the next two lines (this is helpful for debugging)
## LOG="
## "
##
# Set `VERBOSE` to yes when debugging; `VERBOSE` default is no
## `VERBOSE=yes`

# Replace `$HOME/Mags` with the directory where your personal (non-system-spool)
# mailboxes reside. Mailboxes in maildir format or served by Courier IMAP are
# often in `$HOME/Maildir`. Mailboxes served by UW IMAP are sometimes in `$HOME`,
# sometimes in `$HOME/mail` and sometimes elsewhere. MAILDIR default is the value
# of `$HOME` Make sure that MAILDIR exists and that it is a directory!
MAILDIR=$HOME/mail

# The default mail drop
DEFAULT=$MAILDIR/Inbox

#### End Variables Section; Begin Processing Section ####

# Include standard templates
#
#INCLUDERC=$PMDIR/rc.testing
#INCLUDERC=$PMDIR/rc.subscriptions
# Messages that fall through all your procmail recipes are delivered
# to your default INBOX. To find out yours, run `procmail -v`

#### End Processing Section ####

EOF

## 10.3 Setting up Dovecot

Firstly we set up `dovecot` which will provide the IMAPS service for user mailboxes located on port 993.

Create the logging directory for dovecot

```bash
www# mkdir -p /var/log/dovecot
```

Create the dovecot config file. For the `postfix` SMTPS then a SASL authentication service will be used, this is configured within the configuration file using a socket on port 12345 (use whatever port you want).

```bash
www# edit /etc/dovecot/dovecot.conf
```
# Protocols we want to be serving: imap imaps pop3 pop3s
protcols = imap

# Log file to use for error messages, instead of sending them to syslog.
log_path = /var/log/dovecot/syslog

# Log file to use for informational and debug messages.
info_log_path = /var/log/dovecot/infolog

# PEM encoded X.509 SSL/TLS certificate and private key. They are opened before
# dropping root privileges, so keep the key file unreadable by anyone but
# root.
ssl_cert = </CA3yr/ca3yr_cert-chain.crt
ssl_key = </CA3yr/ca3yr_cert.key

# How often to regenerate the SSL parameters file.
# The value is in hours, 0 disables regeneration entirely.
ssl_parameters_regenerate = 168

# SSL ciphers to use
ssl_cipher_list = ALL:!LOW:!SSLv2:!EXP:!aNULL

# Show protocol level SSL errors.
verbose_ssl = no

# Greeting message for clients.
login_greeting = Why are you here?

## Mailbox locations and namespaces

## Location for users' mailboxes. This is the same as the old default_mail_env
# setting. The default is empty, which means that Dovecot tries to find the
# mailboxes automatically. This won’t work if the user doesn’t have any mail
# yet, so you should explicitly tell Dovecot the full location.
#
# If you’re using mbox, giving a path to the INBOX file (eg. /var/mail/%u)
# isn’t enough. You’ll also need to tell Dovecot where the other mailboxes are
# kept. This is called the "root mail directory", and it must be the first
# path given in the mail_location setting.
#
# There are a few special variables you can use, eg.:
#
# %u - username
# %n - user part in user@domain, same as %u if there’s no domain
# %d - domain part in user@domain, empty if there’s no domain
# %h - home directory
#
# See doc/wiki/Variables.txt for full list.
mall_location = mbox:~/mail/INBOX =~/mail/Inbox

# Valid UID range for users, defaults to 500 and above. This is mostly
# to make sure that users can’t log in as daemons or other system users.
first_valid_uid = 1000
last_valid_uid = 1009

# Valid GID range for users, defaults to non-root/wheel.
first_valid_gid = 1000
last_valid_gid = 1009
## IMAP specific settings

```bash
protocol imap {
    imap_client_workarounds = delay-newmail tb-extra-mailbox-sep
}
```

## Authentication processes

```
auth_username_chars = abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ01234567890._-@

auth_verbose = no

auth_debug = no

passdb {
    driver = pam
}

userdb {
    driver = passwd
}

auth_mechanisms = login plain
```

```
imap support.

service imap-login {
    inet_listener imaps {
        port = 993
        ssl = yes
    }
}
```

```
Add Postfix SASL support

service auth {
    # auth_socket_path points to this userdb socket by default. It’s typically
    # used by dovecot-lda, doveadm, possibly imap process, etc. Users that have
    # full permissions to this socket are able to get a list of all usernames and
    # get the results of everyone’s userdb lookups.

    # The default 0666 mode allows anyone to connect to the socket, but the
    # userdb lookups will succeed only if the userdb returns an "uid" field that
    # matches the caller process’s UID. Also if caller’s uid or gid matches the
    # socket’s uid or gid the lookup succeeds. Anything else causes a failure.

    # To give the caller full permissions to lookup all users, set the mode to
    # something else than 0666 and Dovecot lets the kernel enforce the
    # permissions (e.g. 0777 allows everyone full permissions).

    unix_listener auth-userdb {
    }

    inet_listener {
```
With the recent security advisory then it is recommended that SSLv3 is disabled and a 2048 DH exchanged is required. The security may be increased further with the following configuration:

```bash
ssl_cipher_list = -SSLv3:EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH:
    ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:
    DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:
    ECDHE-RSA-AES256-SHA: ECDHE-RSA-AES128-SHA: DHE-RSA-AES256-SHA256:
    ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:_AES256-GCM-SHA384:
    AES128-GCM-SHA256:AES256-SHA256:AES256-SHA:
    !EXP:!3DES:!aNULL:!eNULL
ssl_prefer_server_ciphers = yes
# Set the Diffie Hellman parameter length to 2048 for OSX
ssl_dh_parameters_length = 2048
```

10.3.1 Starting the service

Start the IMAPS service.

```bash
www# svcs -a | grep dovecot
disabled 11:45:04 svc:/site/dovecot:default
www# svcadm enable /site/dovecot:default
```

Once running then it should be possible to connect to the IMAPS server from a mail client using SSL at www.mydomain.co.uk:993 using the username and password.

10.3.2 Log management

Manage the dovecot logging files.

```bash
www# logadm -w /var/log/dovecot/infolog -C 24 -c -p 1m -t \$file=\%Y-%m-%d' -z 0
www# logadm -w /var/log/dovecot/syslog -C 24 -c -p 1m -t \$file=\%Y-%m-%d' -z 0
```

10.4 Removing Sendmail

First remove the existing `sendmail` installation:

```bash
www# svcs -vx sendmail
# Disable sendmail
www# svcadm disable svc:/network/smtp:sendmail
# Remove the package
www# pkg uninstall sendmail
```

10.5 Postfix local Mailer

With `sendmail` removed then `postfix` provides the necessary files that are used by the rest of the system to interface with mail, first restore those commands and set up any mail aliases:

```bash
www# cd /usr/lib
www# ln -s sendmail.postfix sendmail
```
www# cd /usr/bin
www# ln -s newaliases.postfix newaliases
www# cp /etc/postfix/aliases /etc/mail/aliases
www# vi /etc/mail/aliases
www# newaliases

The exact location of the *aliases* file may be determined through procmail as follows:

www# postconf alias_maps
cd /etc/mail/
postconf alias_maps
alias_maps = dbm:/etc/mail/aliases

Whenever the *aliases* file is modified then always run *newaliases*.

The file /etc/postfix/aliases file will contain something like the following. You should map the root mail onto the user who will process this mail:

```
# Person who should get root’s mail. Don’t receive mail as root!
root: jon

# Basic system aliases -- these MUST be present
MAILER-DAEMON: postmaster
postmaster: root

# General redirections for pseudo accounts
bin: root
daemon: root
named: root
nobody: root
uucp: root
www: root
ftp-bugs: root
postfix: root

# Put your local aliases here.

# Well-known aliases
manager: root
dumper: root
operator: root
abuse: postmaster

# trap decode to catch security attacks
decode: root
```

Edit the postfix configuration file /etc/postfix/main.cf this instance of *postfix* provides the port 25 mail services which are used for our local relay which sends external mail to our ISP.

www# vi /etc/postfix/main.cf
View the postfix configuration with `postconf -n`. The postfix configuration file will be something like the following:

```
www# postconf -n

inet_interfaces = all
inet_protocols = ipv4
mail_owner = postfix
mailbox_command = /usr/bin/procmail -a "$_EXTENSION" \
                 DEFAULT=$HOME/mail/Inbox MAILDIR=$HOME/mail
mailbox_size_limit = 0
mailq_path = /usr/bin/mailq.postfix
manpage_directory = /usr/share/man
message_size_limit = 15728640
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
mydomain = mydomain.co.uk
myhostname = www.mydomain.co.uk
mynetworks = 192.168.8.0/24, 127.0.0.0/8
myorigin = $mydomain
newaliases_path = /usr/bin/newaliases.postfix
queue_directory = /var/spool/postfix
readme_directory = /usr/share/doc/SFEpostfix/readme
relay_domains = relayhost = mailhost.myisp.co.uk
sample_directory = /etc/postfix
setgid_group = postfix
smtpd_banner = $myhostname ESMTP $mail_name ($mail_version)
unknown_local_recipient_reject_code = 550
```

Check the master configuration file `/etc/postfix/master.cf` which should include a SMTP server.

```
www# vi /etc/postfix/master.cf

This will include something like:

```
....
# ==========================================================================
# service type private unpriv chroot wakeup maxproc command + args
# (yes) (yes) (yes) (never) (100)
# ==========================================================================
smtp inet n - n - - smtpd
....
```

Start the mail service

```
www# svc -a |grep postfix
disabled 20:22:42 svc:/site/postfix:default

www# svcadm enable svc:/site/postfix:default
www# svc -vx postfix
svc:/site/postfix:default (Postfix Mailserver)
State: online since 19 August 2012 13:07:54 BST
   See: man -M /usr/share/man -s 1 postfix
   See: /var/svc/log/site-postfix:default.log
Impact: None.
```

The mailer should now be running, confirm its operation by sending some mail to root with the `mail` command and then reading it.
10.6 Global Zone Mailer

Repeat the process to swap sendmail for postfix in the global zone. Configure the mail server to relay through www by configuring /etc/postfix/main.cf as follows:

```
www# postconf -n
alias_database = dbm:/etc/mail/aliases
alias_maps = dbm:/etc/mail/aliases
command_directory = /usr/sbin
config_directory = /etc/postfix
daemon_directory = /usr/lib/postfix
data_directory = /var/lib/postfix
debug_peer_level = 2
html_directory = /usr/share/doc/SFEpostfix/html
inet_protocols = ipv4
local_recipient_maps =
mail_owner = postfix
mailq_path = /usr/bin/mailq.postfix
manpage_directory = /usr/share/man
mydomain = mydomain.co.uk
mynetworks = 127.0.0.0/8
myorigin = $mydomain
newaliases_path = /usr/bin/newaliases.postfix
queue_directory = /var/spool/postfix
readme_directory = /usr/share/doc/SFEpostfix/readme
relay_domains =
relayhost = www.mydomain.co.uk
sample_directory = /etc/postfix
sendmail_path = /usr/sbin/sendmail.postfix
setgid_group = postfix
unknown_local_recipient_reject_code = 550
```

Check and start the service as previously described and then check that mail is delivered to the relayhost as required.

10.7 Postfix SMTPS Mailer

A second instance of the postfix mailer has to be created in order to provide a SMTPS service. In this server then we are configuring SMTP over SSL on port 465 in addition to the submission port 587.

10.7.1 Creating a new postfix-smtps service

Create a new manifest file for our postfix-smtps service. Use the existing postfix manifest file in /var/svc-/manifest/site/postfix.xml.

```
www# cp /var/svc/manifest/site/postfix.xml /tmp/postfix-smtps.xml
www# vi /tmp/postfix-smtps.xml
```

Edit the file and carefully change instances of “postfix” to “postfix-smtps” as follows:

```
<?xml version="1.0"?>
<!--
# CDDL HEADER START
# The contents of this file are subject to the terms of the
# Common Development and Distribution License (the "License").
# You may not use this file except in compliance with the License.
# -->
```

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06  58
# You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
# or http://www.opensolaris.org/os/licensing.
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at usr/src/OPENSOLARIS.LICENSE.
# If applicable, add the following below this CDDL HEADER, with the
# fields enclosed by brackets "[]" replaced with your own identifying
# information: Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END
#
-->
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<!--
Copyright 2007 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

#ident "@(#) postfix.xml 0.1 20090417"

NOTE: This service manifest is not editable; its contents will
be overwritten by package or patch operations, including
operating system upgrade. Make customizations in a different
file.
-->

<service_bundle type='manifest' name='SFEpostfix:postfix-smtps'>

<service name='site/postfix-smtps' type='service' version='1'>

<create_default_instance enabled='false' />

<single_instance />

<dependency name='net-loopback' grouping='require_any'
            restart_on='none' type='service'>
  <service_fmri value='svc:/network/loopback'/>
</dependency>

<!--
<dependency name='net-service' grouping='require_all'
             restart_on='none' type='service'>
  <service_fmri value='svc:/network/service'/>
</dependency>

<dependency name='net-physical' grouping='require_all'
             restart_on='none' type='service'>
  <service_fmri value='svc:/network/physical'/>
</dependency>

<dependency name='filesystem-local' grouping='require_all'
             restart_on='none' type='service'>
  <service_fmri value='svc:/system/filesystem/local'/>
</dependency>

<dependency name='name-services' grouping='require_all'
             restart_on='refresh' type='service'>
</dependency>

</service>

</service_bundle>
<service_fmri value='svc:/milestone/name-services' />
</dependency>

<!--
<dependency name='identity' grouping='optional_all'
            restart_on='refresh' type='service'>
  <service_fmri value='svc:/system/identity:domain' />
</dependency>
-->

<dependency name='system-log' grouping='optional_all'
            restart_on='none' type='service'>
  <service_fmri value='svc:/system/system-log' />
</dependency>

<dependency name='autofs' grouping='optional_all'
            restart_on='none' type='service'>
  <service_fmri value='svc:/system/filesystem/autofs' />
</dependency>

<exec_method type='method' name='start'
            exec='/usr/sbin/postfix -c /etc/postfix-smtp
            timeout_seconds='180' />

<exec_method type='method' name='stop'
            exec='/usr/sbin/postfix -c /etc/postfix-smtp
            timeout_seconds='60' />

<exec_method type='method' name='restart'
            exec='/usr/sbin/postfix -c /etc/postfix-smtp
            timeout_seconds='60' />

<stability value='Unstable' />

<property_group name='general' type='framework'>
  <propval name='action_authorization' type='astring'
           value='solaris.smf.manage.sendmail' />
</property_group>

<template>
  <common_name>
    <loctext xml:lang='C'>Postfix Mailserver</loctext>
  </common_name>

  <documentation>
    <manpage title='postfix' section='1'
             manpath='/usr/share/man' />
  </documentation>
</template>
Verify and import the manifest into the system.

```
www# svccfg validate /tmp/postfix-smtps.xml
www# svccfg import /tmp/postfix-smtps.xml
www# svc -xv postfix-smtps
```

```
svc:/site/postfix-smtps/default (Postfix Mailserver)
  State: disabled since 2 January 2014 10:04:51 GMT
  Reason: Disabled by an administrator.
  See: man -M /usr/share/man -s 1 postfix
  See: /var/svc/log/site-postfix-smtps/default.log
Impact: This service is not running.
```

### 10.7.2 Creating postfix-smtps configuration files

Create a new instance of postfix-smtps by cloning the existing postfix /etc configuration.

```
www# cd /etc
www# ls -lad postfix
  drwxr-xr-x 2 root sys 26 Sep 1 16:22 postfix
www# cp -rp postfix postfix-smtps
www# mkdir /var/spool/postfix-smtps
www# ls -lad /var/spool/postfix-smtps
  drwxr-xr-x 16 postfix bin 16 Aug 15 20:35 /var/spool/postfix-smtps
www# chown postfix :bin /var/spool/postfix-smtps
www# mkdir /var/lib/postfix-smtps
www# chown postfix :root /var/lib/postfix-smtps
```

### 10.7.3 Setting up SASL authentication

Running a SMTPS then we require users to authenticate with the server before submitting mail. This user authentication is provided by the SASL component from `dovecot`. Create the SASL directory and configuration file for the SMTPS authentication.

```
www# mkdir /etc/postfix-smtps/sasl
www# cat << EOF >> /etc/postfix-smtps/sasl/smtpd.conf
heredoc > pwcheck_method : auxprop
heredoc > mech_list : LOGIN PLAIN
heredoc > EOF
www# more /etc/postfix-smtps/sasl/smtpd.conf
pwcheck_method: auxprop
mech_list: LOGIN PLAIN
EOF
```

```
Make the directory `/etc/sasl` and symbolically link the SMTP file.

```
www# mkdir -p /etc/sasl
www# cd /etc/sasl
www# ln -s /etc/postfix-smtps/sasl/smtpd.conf smtpd.conf
```

### 10.7.4 Postfix Configuration

With the SASL configuration and certificates set up then we are ready to configure `postfix`. Edit file `/etc/postfix-smtps/master.cf` and change the `smtp` line to `465`. Additionally, comment out the `submission`
line for port 587.

```bash
# Modify the smtp service to 465
#smtpd inet n - n - - smtpd
465 inet n - n - - smtpd
  -o syslog_name=postfix/smtps
  -o smtpd_tls_wrappermode=yes
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_client_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
......
# Submission - Port 587
submission inet n - n - - smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_client_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
```

Edit the file `/etc/postfix-smtps/main.cf` and change the message queue and data directories in addition to local site information. The aliases are re-used for both mail daemons. The file is configured to run SSL with our site certificates and authenticate the user. The additional configuration items include the following:

```bash
queue_directory = /var/spool/postfix-smtps
data_directory = /var/lib/postfix-smtps
smtpd_recipient_restrictions = permit_mynetworks, permit_sasl_authenticated, reject_unauth_destination
smtpd_saslauth_enable = yes
smtpd_sasl_path = inet:127.0.0.1:12345
smtpd_sasl_type = dovecot
smtpd_tls_auth_only = yes
smtpd_tls_cert_file = /CA3yr/ca3yr_cert-chain.crt
smtpd_tls_key_file = /CA3yr/ca3yr_cert.key
smtpd_tls_mandatory_protocols = !SSLv2
smtpd_tls_security_level = encrypt
tls_random_source = /dev/urandom
```

With the recent security advisory then it is recommended that SSLv3 is disabled and a 2048 DH exchanged is required. The security may be increased further with the following configuration:

```bash
smtpd_tls_mandatory_protocols = !SSLv2, !SSLv3
smtpd_tls_exclude_ciphers = aNULL, eNULL, EXPORT, DES, RC4, MD5, PSK, aECDH, EDH-DSS-DES-CBC3-SHA, EDH-RSA-DES-CBC3-SHA, KRB5-DES, CBC3-SHA
```

# Increase the DH key exchange size.

```bash
smtpd_tls_dh1024_param_file = /etc/postfix-smtps/dh1024.pem
smtpd_tls_dh512_param_file = /etc/postfix-smtps/dh512.pem
```

The `dh1024.pem` file may be built with `openssl` as follows:

```bash
openssl dhparam -out dh1024.pem 2048
openssl dhparam -out dh512.pem 512
```

Verify the configuration. Note that because we are using user authenticated connections then we do not validate the domain of any mail received. This allows us to relay messages that are not for our domain; useful when running mobile accounts and users are sending with domains other than our own. Quite whether messages from another domain is able to relay though your ISP is a different matter and depends on your ISP. There is the possibility of relaying messages yourself however some mail servers do not accept a relay from a DSL address. My preference is to use the ISP mail relay but one does need to choose ones ISP carefully.
OpenIndiana Small System Server Build
10.8 fetchmail

inet_protocols = ipv4
mail_owner = postfix
mailbox_command = /usr/bin/procmail -a " $EXTENSION DEFAULT=DEFAULT=$HOME@mail/Inbox MAILDIR=$HOME/mail
mailbox_size_limit = 0
mailq_path = /usr/bin/mailq.postfix
manpage_directory = /usr/share/man
message_size_limit = 0
maildestination = $myhostname, localhost.$mydomain, localhost, $mydomain
mydomain = www.mydomain.co.uk
myhostname = www.mydomain.co.uk
mynetworks = 192.168.8.0/24, 127.0.0.0/8
myorigin = $mydomain
newaliases_path = /usr/bin/newaliases.postfix
queue_directory = /var/spool/postfix-smtp
readme_directory = /usr/share/doc/SFEpostfix/readme
relay_domains =
relayhost = mailhost.zen.co.uk
sample_directory = /etc/postfix
sendmail_path = /usr/sbin/sendmail.postfix
setgid_group = postfix
smtpd_banner = $myhostname ESMTP $mail_name
smtpd_recipient_restrictions = permit_mynetworks, permit_sasl_authenticated, reject_unauth_destination
smtpd_sasl_auth_enable = yes
smtpd_sasl_path = inet:127.0.0.1:12345
smtpd_sasl_type = dovecot
smtpd_tls_auth_only = yes
smtpd_tls_cert_file = /CA3yr/ca3yr_cert-chain.crt
smtpd_tls_key_file = /CA3yr/ca3yr_cert.key
smtpd_tls_mandatory_protocols = !SSLv2
smtpd_tls_security_level = encrypt
tls_random_source = /dev/urandom
unknown_local_recipient_reject_code = 550

Check the postfix settings

www# postfix -c /etc/postfix-smtps check

10.7.5 Starting the service
dovecot should be configured before starting the service because of the SASL dependency, refer to the previous section and start dovecot first.

www# svcadm enable postfix-smtps
www# svc -xv postfix-smtps
svc:/site/postfix-smtps:default (Postfix Mailserver)
State: online since Sat Sep 1 17:55:12 2012
See: man -M /usr/share/man -s 1 postfix
See: /var/svc/log/site-postfix-smtps:default.log
Impact: None.

10.7.6 Postfix version number
To find the version number of the Postfix installation:

hal% /usr/sbin/postconf -d mail_version
mail_version = 2.9.5
hal%

10.8 fetchmail
In order to collect mail from the ISP and other mail providers that is not delivered via SMTP (port 25) then fetchmail is used to poll the mailbox and collect mail. fetchmail collects the E-mail messages and delivers them to the postfix MTA via procmail.

10.8.1 Creating a new fetchmail service
Create a new manifest file for our fetchmail service which may be created in /tmp/fetchmail.xml.
<?xml version="1.0"?>
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type='manifest' name='OIfetchmail:fetchmail'>
  <service name='site/fetchmail' type='service' version='1'>
    <create_default_instance enabled='false' />
    <single_instance />
    <dependency name='loopback' grouping='require_all'
      restart_on='error' type='service'>
      <service_fmri value='svc:/network/loopback:default'/>
    </dependency>
    <dependency name='physical' grouping='optional_all'
      restart_on='error' type='service'>
      <service_fmri value='svc:/network/physical:default'/>
    </dependency>
    <dependency name='autofs' grouping='optional_all'
      restart_on='none' type='service'>
      <service_fmri value='svc:/system/filesystem/autofs'/>
    </dependency>
    <exec_method name='start' type='method'
      exec='/usr/bin/fetchmail -f /etc/fetchmailrc'
      timeout_seconds='60'>
      <method_context>
        <method_credential user='root' group='other' />
      </method_context>
    </exec_method>
    <exec_method name='stop' type='method'
      exec=':kill' timeout_seconds='60'>
      <method_context>
        <method_credential user='root' group='other' />
      </method_context>
    </exec_method>
  </service>
</service_bundle>

Verify and import the manifest into the system:

root@www:/tmp# svccfg validate fetchmail.xml
root@www:/tmp# svccfg import fetchmail.xml
root@www:/tmp# svcs -xv fetchmail
svc:/site/fetchmail/default (Fetchmail from a server)
  State: disabled since Tue Aug 21 18:03:14 2012
  Reason: Disabled by an administrator.
  See: http://illumos.org/msg/SMF-8000-05
  See: man -M /usr/share/man -s 1 fetchmail
Impact: This service is not running.
10.8.2 Creating fetchmail configuration files

Create the configuration files for fetchmail, the file defines the logging file locations and rules to pull E-mail from the ISP or other mail provider.

```
www% cat << EOF > /etc/fetchmailrc
# Set the background poll mode in seconds. (every 15 mins)
set daemon 900
#
# Give the name of the last-resort mail recipient
#
set postmaster "postmaster"
#
# Error logging location
set logfile /var/adm/fetchmail.log
#set syslog
#
# Set the id file
set idfile /var/adm/.fetchids
#
#
poll myIsp.co.uk protocol pop3:
  uidl
  user "bob@myIsp.co.uk" password "bob-password" is "bob" here;
  no keep limit 15728500 limitflush
  user "alice@myIsp.co.uk" password "alice-password" is "alice" here;
  no keep limit 15728500 limitflush
#
EOF
```

The file contains private password information so ensure that it is not readable by anybody else.

```
www% chmod go-rw /etc/fetchmailrc
```

The example fetchmailrc file provided limits the size of messages that may be collected to 15MB, if files exceed this length then they are deleted from the server and not delivered.

10.8.3 Starting the service

fetchmail is now set up and can be started.

```
www% svcadm enable fetchmail
www% svc -xv fetchmail
svc:/site/fetchmail:default (Fetchmail from a server)
  State: online since Tue Aug 21 18:04:49 2012
  See: man -M /usr/share/man -s 1 fetchmail
  See: /var/svc/log/site-fetchmail:default.log
  Impact: None.
www% ps -eaf | grep fetchmail
  root 1910 14147  0 18:05:22 pts/2 0:00 grep fetchmail
  root 1834 13554  0 18:04:49 ?    0:00 /usr/bin/fetchmail -f /etc/fetchmailrc
```

10.8.4 Managing logs

Fetchmail can generate some long logs and it is best if the log is rolled. Edit /etc/logadm.conf and add the following lines to the end of the file.

```
#
# Fetchmail log
```
# /var/adm/fetchmail.log -C 4 -p 7d -N -c

10.8.5 TODO

There are still a number of issues to be resolved with this configuration of fetchmail that need some further consideration.

- Change fetchmail service so that it does not run as root. Explore possibility of creating a new fetchmail user.
- Deal with large mail messages cleanly. Possibly remove the limit and always download the message and let postfix deal with the bounce.

11 Web Services

In this chapter we configure the Web Services, this includes a web server (HTTP, HTTPS and WebDAV) using apache with support for PHP and MySQL. CalDAV and CardDAV services are provided by apache using DAViCal and postgres SQL.

11.1 Web Server packages

Get the packages that we are going to use for the Web subsystem.

```bash
www# pkg install apache-22
www# pkg install apache-22/documentation
```

Download DAViCal from www.davical.org; both DAViCal and awl are required.

11.2 Creating the file system

When we created the www zone then we added the file system /www which will be used as the directory container for all of the web services. As this file system is imported from the global zone then it enables the file system to be interrogated without logging into the www zone.

Create the additional directories required:

```bash
www# mkdir -p /www/log
www# mkdir -p /www/var
www# mkdir -p /www/etc/DAVLockDB
www# mkdir -p /www/htdocs
www# mkdir -p /www/webDAV/share
www# mkdir -p /www/DAViCal
```

11.3 Apache

Configure apache, edit the configuration file /etc/apache2/2.2/httpd.conf

```bash
# Disable the output of Apache version information in any response.
ServerTokens Prod

....

# Change this to Listen on specific IP addresses as shown below to
```

www.jasspa.com

oi_setup v1.25 2015/08/16 11:39:06
# prevent Apache from glomming onto all bound IP addresses.
# Listen 192.168.8.201:80
...

# ServerAdmin: Your address, where problems with the server should be
e-mailed. This address appears on some server-generated pages, such
as error documents. e.g. admin@your-domain.com
ServerAdmin admin@mydomain.co.uk

# ServerName gives the name and port that the server uses to identify itself.
# This can often be determined automatically, but we recommend you specify
# it explicitly to prevent problems during startup.
# ServerName www.mydomain.co.uk

# DocumentRoot: The directory out of which you will serve your
documents. By default, all requests are taken from this directory, but
symbolic links and aliases may be used to point to other locations.
# DocumentRoot /www/htdocs

# Each directory to which Apache has access can be configured with respect
to which services and features are allowed and/or disabled in that
directory (and its subdirectories).
# First, we configure the "default" to be a very restrictive set of
# features.
#<Directory />
  Options FollowSymLinks
  AllowOverride None
  Order deny,allow
  Deny from all
</Directory>

# Note that from this point forward you must specifically allow
particular features to be enabled - so if something’s not working as
you might expect, make sure that you have specifically enabled it
below.
#
#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "/www/htdocs">
  # Possible values for the Options directive are "None", "All",
  # or any combination of:
  #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
  # Note that "MultiViews" must be named *explicitly* --- "Options All"
  # doesn’t give it to you.
  #
  # The Options directive is both complicated and important. Please see
  # http://httpd.apache.org/docs/2.2/mod/core.html#options
  # for more information.
Create the file `/www/htdocs/index.html` with a simple test page to enable the service to be confirmed and start the service.

```
www# svcs -a | grep apache
disabled 12:12:51 svc:/network/http:apache22
www# svcadm enable apache22
...
www# svcadm disable apache22
www# svcadm restart apache22
www# svcs -xv apache22
```

From a browser connect to the service to confirm operation.

Note: For OS X then you may need to clear the DNS cache. To clear the OS cache:

```
sudo killall -HUP mDNSResponder
```

To disable Safari DNS prefetching:

```
defaults write com.apple.safari WebKitDNSPrefetchingEnabled -boolean false
```
11.3.1 PHP support

PHP is an apache module, load the appropriate PHP packages, if MySQL and/or Postgres SQL are to be used then load the PHP connector(s) as well.

www# pkg install apache-php5
www# pkg install php-52/documentation
www# pkg install php-mysql
www# pkg install php-pgsql

Edit the PHP configuration /etc/php/5.2/php.ini and configure the logs files to be sent to our preferred location.

```
# Change the error log location.
error_log = /www/log/php5_exec.log
# Change the maximum post size, depending on your application.
post_max_size = 2K
```

Edit the apache configuration file /etc/apache2/2.2/httpd.conf and ensure that the PHP module is enabled. PHP include paths may also be defined.

```
<IfModule php5_module>
  <IfModule mime_module>
    AddType application/x-httpd-php .php
    AddType application/x-httpd-php-source .phps
  </IfModule>
  # Define the PHP5 configuration, where inc files are located etc.
  php_value include_path ".:/www/inc"
  php_value default_charset "utf-8"
</IfModule>
```

Create an appropriate test file and restart the apache service and confirm operation from a browser.

www# svcadm restart apache22
www# svcs -xv apache22

11.3.2 MySQL support

MySQL may be required by your HTTP server, the SQL database may be placed in the global zone and connect remotely to the database from any service on our www domain.

Install the MySQL package (root) and create a file system for the database, in this case then a MySQL database should be contained on the spinning disks.

```
hal# zfs create tank01/mysql
hal# chown mysql:mysql /tank01/mysql
hal# chmod 755 /tank01/mysql
hal# zfs set mountpoint=/var/mysql/5.1/data tank01/mysql
```

```
hal# svcs -a | grep mysql
disabled 20:38:58 svc:/application/database/mysql:version_51
hal# svcadm enable application/database/mysql:version_51
hal# svcadm -xv application/database/mysql:version_51
```

Change the root password to the database and create the permissions.

```
hal# mysql -u root -p Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 1 Server version: 5.1.37 Source distribution
Type ‘help’ or ‘\h’ for help. Type ‘\c’ to clear the current input statement.
```

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
mysql> UPDATE mysql.user SET password=PASSWORD("my-new-password") WHERE User='root';

mysql> create database mydb;

mysql> use mysql;

mysql> grant create,insert,select,update,delete,lock tables on mydb.* to dbadmin@192.168.8.201 identified by 'somepassword';

select * from user;

select Host, User, Password from user;

mysql> \q

Bye

The MySQL database may be accessed remotely from the Apache Webserver i.e. from PHP etc.

// mySQL database.
// Define the connections to the database.
$mysql_hostname = "192.168.8.200:3306";
$mysql_database_name = "mydb";
$mysql_username = "dbadmin";
$mysql_password = "somepassword";

11.3.3 HTTPS services

For an HTTPS service on port 443 then create a virtual host in the configuration file /etc/apache2/2.2/httpd.conf and add a new virtual host to the end of the configuration file.

Note that if we have disabled ipv6 then we use Listen 0.0.0.0:port in order to remove errors in the apache.log of the form (128)Network is unreachable: connect to listener on [::]:8081

......
#
# Note: The following must must be present to support
# starting without SSL on platforms with no /dev/random equivalent
# but a statically compiled-in mod_ssl.
#
<IfModule ssl_module>
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
</IfModule>
......

# SSL Server
#
Listen 0.0.0.0:443
NameVirtualHost 192.168.8.201:443

<VirtualHost 192.168.8.201:443>
  # General setup of the virtual host
  DocumentRoot "/www/htdocs"
  ServerName "www.mydomain.co.uk:443"
  ServerAdmin "admin@mydomain.co.uk"

  # Virtual server logging
  ErrorLog "/www/log/apache_error_443.log"
  TransferLog "/www/log/apache_access_443.log"

  # Turn on SSL for this port
SSLEngine on
SSLProtocol -all +SSLv3 +TLSv1
SSLCipherSuite HIGH:!MEDIUM:!SSLv2:!EXP:!ADH:!aNULL:!eNULL:!NULL
SSLOptions +StrictRequire

# Server Certificate
SSLCertificateFile "/CA3yr/ca3yr_cert.crt"
SSLCertificateKeyFile "/CA3yr/ca3yr_cert.key"
SSLCertificateChainFile "/CA3yr/ca3yr_ca-bundle.crt"

# SSL Protocol Adjustments:
BrowserMatch ".* MSIE .*"
   nokeepalive ssl-unclean-shutdown 
   downgrade=1.0 force-response=1.0

# Per-Server Logging:
CustomLog "/www/log/ssl_request_443.log"
   "%t %h %{ SSL_PROTOCOL }x %{ SSL_CIPHER }x "%r"
Alias /ssl "/www/somessldir"

# Define the values for the include path.
#<Directory "/www/somessldir">
   # Allow index translation.
   Options -Indexes FollowSymLinks -MultiViews -ExecCGI
   AllowOverride None
   # Enable the environment variables for our SSL environment
   <IfModule env_module>
      SetEnv SSL_BASEDIR /www/somessldir
   </IfModule>
   # Password access if required.
   #AuthType Basic
   #AuthName "Mydomain Secure Area"
   #AuthUserFile /www/etc/password
   #Require user bob
   #Satisfy All
   # On a directory access then run index.php
   DirectoryIndex index.php
   # Limits
   <Limit GET POST OPTIONS>
      Order allow,deny
      Allow from all
   </Limit>
   <LimitExcept GET POST OPTIONS>
      Order deny,allow
      Deny from all
   </LimitExcept>
</Directory>
</VirtualHost>

Create a test HTML file and restart the apache service and confirm operation from a browser.

```
www# svcadm restart apache22
www# svcs -xv apache22
```
11.3.4 WebDAV

The WebDAV service provides a remote Web file system which may be used for storage which may be used
buy the iOS iWorks applications, amongst others.

Create the file system space in our www filesystem and change the ownership to webservd:webservd.

```
www# mkdir /www/etc/DavLock
www# chmod a+rw /www/etc/DavLockDB
www# chmod a+rw /www/webDAV
www# chown -R webservd:webservd /www/webDAV
```

Create the password file. Use MD5 it is better then crypt which is the default. Obviously use a better
password than used here.

```
www# htpasswd -m -c -b /etc/apache2/2.2/dav.passwd bob "password"
www# htpasswd -m -b /etc/apache2/2.2/dav.passwd alice "password"
```

# Protect the password file.
```
www# chown root /etc/apache2/2.2/dav.passwd
www# chgrp webservd /etc/apache2/2.2/dav.passwd
www# chmod 640 /etc/apache2/2.2/dav.passwd
```

Groups can be useful with WebDAV, a groups file may be created as follows:

```
www% cat << EOF >> /etc/apache2/2.2/dav.groups
users: bob alice
admin: alice
readers: fred freda alice
EOF
```

# Protect the groups file.
```
www# chown root /etc/apache2/2.2/dav.groups
www# chgrp webservd /etc/apache2/2.2/dav.groups
www# chmod 640 /etc/apache2/2.2/dav.groups
```

Configure apache, edit the configuration file /etc/apache2/2.2/httpd.conf. In this case a WebDAV
service is created on port 8081 which is secured with SSL

```
# # Port 8081 - WebDAV with SSL
# #
Listen 0.0.0.0:8081
NameVirtualHost 192.168.8.201:8081
#
DAVLockDB /www/var/DavLockDB/DavLock
DAVMinTimeout 180
#
<VirtualHost 192.168.8.201:8081>
    # General setup of the virtual host
    DocumentRoot "/www/webDAV"
    ServerName "www.mydomain.co.uk:8081"
    ServerAdmin "admin@mydomain.co.uk"
    
    # Virtual server logging
    ErrorLog "/www/log/apache_error_8081.log"
    TransferLog "/www/log/apache_access_8081.log"
    
    # Turn on SSL for this port
    SSLEngine on
    SSLProtocol -all +SSLv3 +TLSv1
```

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06

72
SSLOptions +StrictRequire

# Server Certificate
SSLCertificateFile "/CA3yr/ca3yr_cert.crt"
SSLCertificateKeyFile "/CA3yr/ca3yr_cert.key"
SSLCertificateChainFile "/CA3yr/ca3yr_ca-bundle.crt"

# SSL Protocol Adjustments:
BrowserMatch ".* MSIE .*"
  nokeepalive ssl-unclean-shutdown \
  downgrade=1.0 force-response-1.0

# Per-Server Logging:
CustomLog "/www/log/ssl_request_8081.log" \ 
  "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x %r %b"

# For the root directory then only bob can write.
<Directory /www/webDAV>
  Options +Indexes
  IndexIgnore .. .
  IndexOptions ~IconsAreLinks NameWidth=* FancyIndexing \ 
    FoldersFirst SuppressLastModified
  IndexOrderDefault Ascending Name
  AddDescription "7-Zip archive" *.7z
  AddDescription "Log file" *.log
  AllowOverride None
  Order allow,deny
  Allow from all
  AuthType Basic
  AuthName "mydomain WebDAV Server"
  AuthUserFile /etc/apache2/2.2/dav.passwd
  Require valid-user
  
  DAV on
</Directory>

# Add the following if any user is allowed to see the root
directory.
#<Location />
#  Order Allow,Deny
#  Allow from all
#  Options +Indexes
#  IndexIgnore ..
#  IndexOptions ~IconsAreLinks NameWidth=* FancyIndexing
#  SuppressLastModified FoldersFirst
#  IndexOrderDefault Ascending Name
#  Require valid-user
#</Location>

# Users WebDAV - Valid for group of users.
# Alias /users "/*/www/webDAV/users"
<Directory /www/webDAV/users>
  DAV On
  Order Allow,Deny
  Allow from all
  AuthType Basic
  AuthName "mydomain WebDAV Server"
  AuthUserFile /etc/apache2/2.2/dav.passwd
</Directory>
AuthGroupFile /etc/apache2/2.2/dav.groups
Require group users
</Directory>

# We want to access this WebDAV directory using an Internet browser.
<Location /users>
Options +Indexes
IndexIgnore "."
IndexOptions -IconsAreLinks NameWidth=* \ 
   FancyIndexing FoldersFirst
# SuppressLastModified
IndexOrderDefault Ascending Name
AddDescription "7-Zip archive" *.7z
AddDescription "Log file" *.log
Require group users
</Location>

#
# Bob WebDAV - Only valid for a single user
#
Alias /bob="/www/webDAV/bob"
<Location /bob>
Options +Indexes
IndexIgnore "."
IndexOptions -IconsAreLinks NameWidth=* \ 
   FancyIndexing FoldersFirst
# SuppressLastModified
IndexOrderDefault Ascending Name
AddDescription "7-Zip archive" *.7z
AddDescription "Log file" *.log
Require user bob
</Location>

#
# Restricted WebDAV - Restrict area to mainly readers with a writer
#
Alias /sigen="/www/webDAV/restricted"
<Location /sigen>
Options +Indexes
IndexIgnore "."
IndexOptions -IconsAreLinks NameWidth=* \ 
   FancyIndexing FoldersFirst
# SuppressLastModified
IndexOrderDefault Ascending Name
AddDescription "7-Zip archive" *.7z
AddDescription "Log file" *.log
Require group readers
</Location>
# We want to access this WebDAV directory using an Internet browser.
# Alice is allowed to upload, everybody else is read only.

<Location /sigen >
Options +Indexes
IndexIgnore ".*"
IndexOptions -IconsAreLinks NameWidth=* FancyIndexing FoldersFirst
#SuppressLastModified
IndexOrderDefault Ascending Name
AddDescription "7-Zip archive" *.7z
AddDescription "Log file" *.log
<Limit GET OPTIONS PROPFIND>
  Require group readers
</Limit>
<LimitExcept GET OPTIONS PROPFIND>
  Require user alice
</LimitExcept>
</Location>

# upload WebDAV - Generic upload area for any valid user.

Alias /upload "/www/webDAV/upload"

<Directory /www/webDAV/upload>
DAV On
Order Allow,Deny
Allow from all
AuthType Basic
AuthName "mydomain_WebDAV_Server"
AuthUserFile /etc/apache2/2.2/dav.passwd
AuthGroupFile /etc/apache2/2.2/dav.groups
Require valid-user
</Directory>

# We want to access this WebDAV directory using an Internet browser.

<Location /upload >
Options +Indexes
IndexIgnore ".*"
IndexOptions -IconsAreLinks NameWidth=* FancyIndexing FoldersFirst
AddDescription "7-Zip archive" *.7z
AddDescription "Log file" *.log
IndexOrderDefault Ascending Name
Require valid-user
</Location>
</VirtualHost>

Restart the apache service and confirm operation from a browser or WebDAV client.

www# svcadm restart apache22
www# svcs -xv apache22

11.3.5 Log management

Manage the Apache logging, in this case we keep all of the logs just incase there are issues, they are maintained in dated files.

www# logadm -w /www/log/apache_access.log -C 24 -c -p 1m -t "$file-%Y-%m-%d" -z 0
www# logadm -w /www/log/apache_access_4443.log -C 24 -c -p 1m -t "$file-%Y-%m-%d" -z 0
www# logadm -w /www/log/apache_access_8443.log -C 24 -c -p 1m -t "$file-%Y-%m-%d" -z 0
www# logadm -w /www/log/apache_access_8008.log -C 24 -c -p 1m -t "$file-%Y-%m-%d" -z 0
www# logadm -w /www/log/apache_access_8081.log -C 24 -c -p 1m -t "$file-%Y-%m-%d" -z 0

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
12 Calendar and Address Book Services

The Calendar and Address Book services use DAViCal, which runs on top of Apache with PHP v5 and uses the postgres SQL database for event data storage.

Apache2 and PHP5 should be installed on www with the following PHP5 additional package modules php5-pgsql, php5-pdopgsql, php5-pdo, php5-ico and php5-gettext.

12.1 Getting DAViCal and installing

Download DAViCal from http://debian.mcmillan.net.nz; both awl and davical are required.

Create a directory for DAViCal, in our case then we use the /www directory on the www zone which is also accessible from the Global zone as /tank01/www.

hal# mkdir -p /tank01/www/DAViCal
hal# gtar zxvf awl-0.46.tar.gz -C /tank01/www/DAViCal
hal# gtar zxvf davical-1.0.2.tar.gz -C /tank01/www/DAViCal

# Create a symbolic link to un-version
hal# cd /tank01/www/DAViCal
hal# ln -s awl-0.46 awl
hal# ln -s davical-1.0.2 davical

12.2 Setting up Postgres

Before configuring Apache for DAViCal then set up the postgres SQL server. For our server then this is performed in the Global zone rather than www where it will be used, the database will be accessed remotely.

hal# pkg install database/postgres-84 service/database/postgres-84 / database/postgres-84/documentation postgres-common

We want to put the postgres data on the mirrored data disk so create a new zfs file system for postgres.

hal# zfs create tank01/postgres
hal# chown postgres:postgres /tank01/postgres
hal# chmod 755 /tank01/postgres
hal# mv /var/postgres/8.4 /tank01/postgres
hal# zfs set mountpoint=/var/postgres tank01/postgres

Edit the /etc/passwd file and change the home directory to "/home/postgres"

postgres:x:90:90:PostgreSQL Reserved UID:/export/home/postgres:/usr/bin/pfkh

Make the home directory:

hal# mkdir /export/home/postgres
hal# chown postgres.postgres /export/home/postgres
Create a `.profile` file and set up the environment for the postgres user. Create the file `/export/home/-postgres/.profile` containing the following commands:

```
PATH=/usr/postgres/8.4/bin:$PATH
PGDATA=/var/postgres/8.4/data
export PATH PGDATA
```

Ensure the file ownership is correct:

```
hal# chown postgres.postgres /export/home/postgres/.profile
```

Start the `postgres` service:

```
hal# svcs -a|grep postg
disabled 17:25:35 svc:/application/database/postgresql_84:default_64bit
disabled 17:25:35 svc:/application/database/postgresql_84:default_32bit
#hal svcadm enable postgresql_84:default_32bit
```

Initialise the `postgres` database (note I am not that familiar with `postgres` so there may be better ways of doing this).

```
su - postgres
OpenIndiana (powered by illumos) SunOS 5.11 oi_151a5 June 2012
postgres@hal:-$ psql
psql (8.4.4)
Type "help" for help.

postgres=#
postgres=# \\li

<table>
<thead>
<tr>
<th>Name</th>
<th>Owner</th>
<th>Encoding</th>
<th>Collation</th>
<th>Ctype</th>
<th>Access privileges</th>
</tr>
</thead>
<tbody>
<tr>
<td>postgres</td>
<td>postgres</td>
<td>UTF8</td>
<td>en_GB.UTF-8</td>
<td>en_GB.UTF-8</td>
<td>-c/postgres</td>
</tr>
<tr>
<td>template0</td>
<td>postgres</td>
<td>UTF8</td>
<td>en_GB.UTF-8</td>
<td>en_GB.UTF-8</td>
<td>-c/postgres</td>
</tr>
<tr>
<td>template1</td>
<td>postgres</td>
<td>UTF8</td>
<td>en_GB.UTF-8</td>
<td>en_GB.UTF-8</td>
<td>-c/postgres</td>
</tr>
</tbody>
</table>

(3 rows)

postgres=# \\q
```

Reset the password of the `postgres` user. The default superuser, called 'postgres', does not have a password by default. We need to add one:

```
$ sudo su postgres -c psql template1
template1=# ALTER USER postgres with PASSWORD 'password';
template1=# \\
```

Where 'password' is your password. After this we need to modify the password of the postgres UNIX user:

```
$ sudo passwd -d postgres
$ sudo su postgres -c passwd
```

You will be asked for a new password, enter the same password with the one you specified in the ALTER USER statement above. If the `ipf` firewall is running then open port 5432 to allow remote access to the server.

The default PostgreSQL installation in Solaris requires that a PostgreSQL user must also be a unix user, this makes it difficult to create a new PostgreSQL user. To allow a PostgreSQL user to be different from a UNIX user then the `pg_hba.conf` file needs to be altered.

Edit the file `/etc/postgresql/8.4/pg_hba.conf` and allow access for the localhost and our remote server 192.1.8.201 (www) only:

```
```
12.3 Initialising the DA ViCal Database

We are not finished yet, the database has been set up but we still need to initialise the DA ViCal database. To do this then we also need to install the Perl and Perl database connector:

```
hal# pkg install library/perl-5/database
hal# pkg install database/postgres-84/language-bindings
hal# pkg install library/perl-5/postgres-db
```

Install Perl YAML:

```
hal# perl -MCPAN -e 'install +YAML'
```

We are now ready to create the DA ViCal database, this is a Bash shell script and IT IS REALLY IMPORTANT TO RUN THE FOLLOWING STEPS IN THE BASH SHELL. Run the DA ViCal database creation script as the `postgres` user from a bash shell.

When creating the PostgreSQL database then DA ViCal needs to operate using UTF-8, if your Locale is set to something different to UTF-8 then you need to make sure that the database is set to UTF-8 and matches a Sun Locale e.g. `en_GB.UTF-8`. After initialising the database then this can be changed by editing `postgresql.conf` before creating the `davical` database with `bash dba/create-database.sh` otherwise the creation fails.

```
hal# su - postgres
postgres@hal: ~/www/DAViCal/davical-1.0.2$ cd /tank01/www/DAViCal/davical
postgres@hal: ~/www/DAViCal/davical-1.0.2$ bash ./dba/create-database.sh
```

If things go wrong then you will need to find out what went wrong and drop the database so that the operation can be fixed and restarted. The following steps may be used to undo the database creation:

```
postgres@hal: ~/www/DAViCal/davical-1.0.2$ psql
Password:
psql (8.4.4) Type "help" for help.

postgres=# DROP DATABASE davical;
postgres=# \\q
postgres@hal:
```

When the database succeeds then the administrator password will be displayed.
12.4 Importing an existing DAViCal Database

If you have an old DAViCal database then it can be imported into the new database as follows:

```
# Dump the old calendar
oldsystem# pg_dump -Fc davical > /tmp/davical.pgdump

# Restore the dump to the new system
newsystem# pg_restore -Fc -d davical /tmp/davical.pgdump
```

12.5 Remote Server postgres preparation

The `postgres` database has been installed in the Global zone and in this configuration then it will be accessed remotely from our `www` virtual machine. Prepare the environment on `www`

Install the postgres package

```
www# pkg install pkg://openindiana.org/database/postgres-83
```

Test the remote connection

```
www# psql -h hal -U postgres -d test
```

12.6 Davical Configuration

From our `www` zone then davical itself needs to be configured via the file `/www/DAViCal/davical/-config/config.php`. This is a PHP script which is used by DAViCal. Assuming that the SSL port is external facing then can disable administrator access on port 8443 with `$c->restrict_admin_port = '8008';`. If Apache2 is running in a zone and the SQL server is on another host or zone then change the connection to access the remote database with `$c->pg_connect[] = 'hostaddr=192.168.1.y port=5432 dbname=davical user=davical_app';`. Set the Locale to the same value as the PostgreSQL database with `$c->default_locale = "en_GB.UTF-8";

The file `config.php` will something like:
<?php

// Naming information
$c -> domainname = "www.mydomain.co.uk";
$c -> sysabbr = 'www';
$c -> admin_email = 'admin@mydomain.co.uk';
$c -> system_name = "CalDAV_Server";
// Set the locale that we are using.
$c -> default_locale = "en_GB.UTF-8";
// Database connection
$c -> pg_connect[] = 'hostaddr=192.168.8.200_port=5432_dbname=davical_user=davical_app';
// Restrict administration access to port 8008
$c -> restrict_admin_port = '8008';
?>

Refer to the DA ViCal WIKI for further information on authenticating, in this configuration then the admin interface of DA ViCal is used to create users with access rights.

12.7 Apache Configuration

With DA ViCal and Postgres configured then Apache may be configured to provide calendar and address book services on port 8443 and administrator access on port 8008. Edit the Apache configuration file (on www) and add a virtual host for each DA ViCal service by editing /etc/apache2/2.2/httpd.conf.

# # Port 8008 - CalDav port without SSL
# # DA ViCal - CalDav port without SSL
# Used for local hosts and administration access
# Listen 0.0.0.0:8008
NameVirtualHost 192.168.8.201:8008
#
<VirtualHost 192.168.8.201:8008>
    # General setup of the virtual host
    DocumentRoot "/www/DAViCal/davical/htdocs"
    Alias /images/ "/www/DAViCal/davical/htdocs/images/
    ServerName www.mydomain.co.uk:8008
    ServerAdmin admin@mydomain.co.uk
    # Virtual server logging
    ErrorLog "/www/log/apache_error_8008.log"
    TransferLog "/www/log/apache_access_8008.log"

    # Define the directory access
    <Directory "/www/DAViCal/davical/htdocs/>
    AllowOverride None
    Order allow,deny
    Allow from all

    # Default directory index.
    DirectoryIndex index.php
    </Directory>

    # Define the PHP5 configuration.
    php_value include_path "/www/DAViCal/davical/inc:/www/DAViCal/awl/inc"
    php_value magic_quotes_gpc 0
    php_value register_globals 0
    php_value error_reporting "E_ALL&~ E_NOTICE"
    php_value default_charset "en_GB.UTF-8"
</VirtualHost>
# Get rid of caldav.php in the path
RewriteEngine On
# Not if it’s the root URL.
RewriteCond %{REQUEST_URI} !^/$
# Not if it explicitly specifies a .php program, stylesheet or image
RewriteCond %{REQUEST_URI} !\.(php|css|js|png|gif|jpg)
# For iPhone
RewriteRule ^\.well-known/(.*)$ /caldav.php/.well-known$1 [NC,L]
# Everything else gets rewritten to /caldav.php/...
RewriteRule ^(.*)$ /caldav.php/$1 [NC,L]
</VirtualHost>

#
# Port 8443 - CardDav port with SSL
#
# DAViCal - CardDav port with SSL
# Used for internal and external access
#
Listen 0.0.0.0:8443
NameVirtualHost 192.168.8.201:8443
#
<VirtualHost 192.168.8.201:8443>
# General setup of the virtual host
DocumentRoot "/www/DAViCal/davical/htdocs"
ServerName www.mydomain.co.uk:8443
ServerAdmin admin@mydomain.co.uk
# Virtual server logging
ErrorLog "/www/log/apache_error_8443.log"
TransferLog "/www/log/apache_access_8443.log"

# Turn on SSL for this port
SSLEngine on
SSLProtocol -all +SSLv3 +TLSv1
SSLOptions +StrictRequire

# Server Certificate
SSLCertificateFile "/CA3yr/ca3yr_cert.crt"
SSLCertificateKeyFile "/CA3yr/ca3yr_cert.key"
SSLCertificateChainFile "/CA3yr/ca3yr_ca-bundle.crt"

# SSL Protocol Adjustments:
BrowserMatch ".* MSIE .*"
   nokeepalive ssl-unclean-shutdown
   downgrade=1.0 force-response=1.0

# Per-Server Logging:
CustomLog "/www/log/ssl_request_8443.log"
   *%t%h%l [%{SSL_PROTOCOL}x%{SSL_CIPHER}x]"%r"%b"

# Define the directory access
<Directory /www/DAViCal/davical/htdocs/>
   Dav off
   AllowOverride None
   Order allow,deny
   Allow from all
</Directory>

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
# Allow trailing names
AcceptPathInfo On

# Define the PHP5 configuration.
php_value include_path /www/DAViCal/davical/inc:/www/DAViCal/awl/inc
php_value magic_quotes_gpc 0
php_value register_globals 0
php_value error_reporting "E_ALL & ~ E_NOTICE"
php_value default_charset "utf-8"

# For CardDAV then we do not use caldav.php in the URL.
# Redirect everything in this instance (Option a)
# Activate RewriteEngine
RewriteEngine On

# Not if it’s the root URL.
RewriteCond %{REQUEST_URI} !^/$
# Not if it explicitly specifies a .php program, stylesheet or image
RewriteCond %{REQUEST_URI} !\.(php|css|js|png|gif|jpg)
# Everything else gets rewritten to /caldav.php/...
RewriteRule ^(.*)$ /caldav.php/$1 [NC,L]

</VirtualHost>

Restart the apache service client.

www# svcadm restart apache22
www# svcs -xv apache22

12.8 DAViCal User Configuration

With DAViCal running then connect to the administrator port 8008 and add users to the system (if required). The password for access was created automatically when the DAViCal database was set up and a note of this password should have been made during the installation process.
13 CVS

Setting up CVS server for a legacy source control system. In this section we set up the CVS services but do not address how to set up CVS as we are restoring a previously configured CVS repository. In our configuration then this is performed in the Global zone (not www).

Install the CVS package and create a directory to hold the repository.

```
hal# zfs list
hal# zfs create tank01/cvs
```

Create a CVS user and group

```
hal# groupadd cvs
hal# useradd cvs
```

Optionally, edit `/etc/passwd` and `/etc/group` and change the UID/GID to 91 (or some other UID/GID used by your organisation).

```
hal# mkdir /tank01/cvs/cvsroot
hal# chown -R cvs:cvs /tank01/cvs
hal# mkdir /export/cvs
hal# zfs set mountpoint=/export/cvs tank01/cvs/cvsroot
```

Create the services entry `tmp/cvspserver-tcp.xml`

```xml
<?xml version='1.0'?>
<!DOCTYPE service_bundle SYSTEM '/usr/share/lib/xml/dtd/service_bundle.dtd.1'>
<!--
Service manifest for the cvspserver service.
-->
<service_bundle type='manifest' name='cvspserver'>
  <service
    name='network/cvspserver/tcp'
    type='service'
    version='1'>
    <create_default_instance enabled='true'/>
    <restarter>
      <service_fmri value='svc:/network/inetd:default' />
    </restarter>
    <!--
    Set a timeout of 0 to signify to inetd that we do not want to
    timeout this service, since the forked process is the one that
    does the services work. This is the case for most/all legacy
    inetd services; for services written to take advantage of SMF
    capabilities, the start method should fork off a process to
    handle the request and return a success code.
    -->
    <exec_method
      type='method'
      name='inetd_start'
      exec="/usr/bin/cvs -f --allow-root=/export/cvs/cvsroot_pserver"
      timeout_seconds='0'>
      <method_context>
        <method_credential user='root' group='root' />
      </method_context>
    </exec_method>
  </service>
</service_bundle>
```
OpenIndiana Small System Server Build
13.1 User Configuration

Use inetd built-in kill support to disable services.

```xml
<exec_method
type='method'
name='inetd_disable'
exec=':kill'
timeout_seconds='0'>
</exec_method>

<property_group name='inetd' type='framework'>
  <propval name='name' type='astring' value='cvspserver' />
  <propval name='endpoint_type' type='astring' value='stream' />
  <propval name='proto' type='astring' value='tcp' />
  <propval name='wait' type='boolean' value='false' />
  <propval name='isrpc' type='boolean' value='false' />
</property_group>

<stability value='External' />

<template>
  <common_name>
    <loctext xml:lang='C'>cvspserver</loctext>
  </common_name>
</template>
</service>
</service_bundle>
```

Import the service entry:

```
hal# svccfg import /tmp/cvspserver-tcp.xml
hal# svcadm disable svc:/network/cvspserver/tcp:default
hal# svcadm enable svc:/network/cvspserver/tcp:default
ha1# svc -xv cvspserver/tcp
  State: online since 1 September 2012 12:28:06 BST
  Impact: None.
```

Initialise the CVS root directory in /export/cvs/cvsroot by setting up a new CVS repository or copying in an existing CVS repository.

### 13.1 User Configuration

As a user then update the shell login script `.profile` or other shell equivalent (i.e. `.zshenv`, `.bashrc`, etc.).

```
# Set up CVSROOT
CVSROOT=:pserver:username@hal.mydomain.co.uk:/export/cvs/cvsroot
export CVSROOT
```
14 Backup

Automated backups are most important in order to preserve the most critical data. The system configuration described here includes a SCSI Tape Drive which is the most efficient and cheapest method of preserving business critical data. 31 tapes are used each labelled with a day number and they are rotated around. One could use more tapes to snapshot each month and archive them to a secure fire safe.

The tapes are not large enough to store all of the data on the system but are sufficient to store all of the critical business files excluding anything that is automatically generated.

There are backup applications such as Amanda which could provide backup solutions. There are other methods where cheap large capacity removable HDDs could be used by exporting a ZFS snapshot.

In our case we use a simple shell script running on a daily cron job to backup the system to tape. Additionally the mail, calendar and other database’s of the system are snapshot and saved to a 7-zip archive. A number of backup and temporary directories are used.

/tank01/tmp - Temporary working area.
/tank01/backup - Archive backup area.
/tank01/backup/db1..3 - Database backup areas.
/tank01/backup/mail - Mail backup area.
/tank01/backup/calendar - Calendar backup area

The backup files are controlled by a shell script the files used are:

/etc/backup - The main script file
/etc/backup.include - A list of directories to include in the backup
/etc/backup.exclude - A list of directories to exclude from the backup
/etc/backup.exclude.sh - A list of commands to find files to exclude

Create the working directories and ensure that they are not readable by others.

```
hal# mkdir -p /tank01/tmp
hal# mkdir -p /tank01/backup
hal# chown root:root /tank01/backup
hal# chmod o-wrx /tank01/backup
```

Create the shall script which is used for the backup. Place the script in file /etc/backup. The file should only be readable by root as it will contain some database passwords.

```
hal# touch /etc/backup
hal# touch /etc/backup.include
hal# touch /etc/backup.exclude
hal# touch /etc/backup.exclude.sh
hal# chown root:root /etc/backup /etc/backup.include /etc/backup.exclude /etc/backup.exclude.sh
hal# chmod u+x /etc/backup /etc/backup.include /etc/backup.exclude
hal# chmod go -xwr /etc/backup /etc/backup.include /etc/backup.exclude
```

Edit the script file /etc/backup and add the script. Within the script then the databases and mail are snapshot and backed up onto the local disk in a compressed 7-zip archive which is AES encrypted (noted that this takes a significant amount of time if the archive is large). The archives are written to tape with other information. Finally a mail message is sent to root reporting on the state of the backup and ZFS file system.

```
#!/bin/sh
# set -x
# Dump the database to the system
BACKUP_DIR="/tank01/backup"
MYSQLDUMP="/usr/mysql/bin/mysqldump"
PGDUMP="/usr/postgres/8.4/bin/pg_dump"
GTAR="/usr/bin/gtar"
```
/usr/sbin/tar

/TAR="/usr/bin/7z","m0=lzma","mx=9","mfb=64","ms=on","password=mhe"

/GZIP="gzip"

/DISKTMPDIR="/tank01/tmp"

# DB1_DIR=${BACKUP_DIR}/db1
# DB2_DIR=${BACKUP_DIR}/db2
# DB3_DIR=${BACKUP_DIR}/db3
# MAIL_DIR=${BACKUP_DIR}/mail
# CAL_DIR=${BACKUP_DIR}/calendar

/DATENAME='date +%Y%m%d_%H%M%S'

# Dump the DB1 Database
mkdir -p ${DB1_DIR}
${MYSQLDUMP} -hlocalhost -udb1 -ppassword --opt DB1 > ${DISKTMPDIR}/db1_${DATENAME}.sql
${Z7} ${DISKTMPDIR}/db1_${DATENAME}.7z ${DISKTMPDIR}/db1_${DATENAME}.sql
chmod go -rw ${DISKTMPDIR}/db1_${DATENAME}.7z
mv ${DISKTMPDIR}/db1_${DATENAME}.7z ${DB1_DIR}/db1_${DATENAME}.7z
rm -f ${DISKTMPDIR}/db1_${DATENAME}.7z ${DISKTMPDIR}/db1_${DATENAME}.sql

# Dump the DB2 Database
mkdir -p ${DB2_DIR}
${MYSQLDUMP} -hlocalhost -udb2 -ppassword --opt db2 > ${DISKTMPDIR}/db2_${DATENAME}.sql
${Z7} ${DISKTMPDIR}/db2_${DATENAME}.7z ${DISKTMPDIR}/db2_${DATENAME}.sql
chmod go -rw ${DISKTMPDIR}/db2_${DATENAME}.7z
mv ${DISKTMPDIR}/db2_${DATENAME}.7z ${DB2_DIR}/db2_${DATENAME}.7z
rm -f ${DISKTMPDIR}/db2_${DATENAME}.7z ${DISKTMPDIR}/db2_${DATENAME}.sql

# Dump the DB3 Database
mkdir -p ${DB3_DIR}
${MYSQLDUMP} -hlocalhost -udb3 -ppassword --opt db3 > ${DISKTMPDIR}/db3_${DATENAME}.sql
${Z7} ${DISKTMPDIR}/db3_${DATENAME}.7z ${DISKTMPDIR}/db3_${DATENAME}.sql
chmod go -rw ${DISKTMPDIR}/db3_${DATENAME}.7z
mv ${DISKTMPDIR}/db3_${DATENAME}.7z ${DB3_DIR}/db3_${DATENAME}.7z
rm -f ${DISKTMPDIR}/db3_${DATENAME}.7z ${DISKTMPDIR}/db3_${DATENAME}.sql

# Dump the calendar
mkdir -p ${CAL_DIR}
${PGDUMP} -Fc davical -U davical_app > ${DISKTMPDIR}/davical_${DATENAME}.pgdump
${Z7} ${DISKTMPDIR}/davical_${DATENAME}.7z ${DISKTMPDIR}/davical_${DATENAME}.pgdump
chmod go -rw ${DISKTMPDIR}/davical_${DATENAME}.7z
mv ${DISKTMPDIR}/davical_${DATENAME}.7z ${CAL_DIR}/davical_${DATENAME}.7z
rm -f ${DISKTMPDIR}/davical_${DATENAME}.7z ${DISKTMPDIR}/davical_${DATENAME}.pgdump

# Backup the mail
# cd /
${GTAR} -cvf ${DISKTMPDIR}/mail_${DATENAME}.tar ./tank01/mail
${Z7} ${DISKTMPDIR}/mail_${DATENAME}.tar.7z ${DISKTMPDIR}/mail_${DATENAME}.tar
chmod go -rw ${DISKTMPDIR}/mail_${DATENAME}.tar.7z
mv ${DISKTMPDIR}/mail_${DATENAME}.tar.7z ${MAIL_DIR}/mail_${DATENAME}.7z
rm -f ${DISKTMPDIR}/mail_${DATENAME}.tar.7z ${DISKTMPDIR}/mail_${DATENAME}.tar

# Do the backup
# cd /

www.jasspa.com

oi_setup v1.25 2015/08/16 11:39:06
EXCFILES="/etc/backup.exclude"
DEXFILES="/tmp/backup.dynamic.exclude"
DSAFILES="/tmp/backup.dynamic.include"

# Find the inclusion list
rm -f ${DSAFILES}

Echo .${DB1_DIR}/db1_${DATENAME}.7z > ${DSAFILES}
Echo .${DB2_DIR}/db2_${DATENAME}.7z >> ${DSAFILES}
Echo .${DB3_DIR}/db3_${DATENAME}.7z >> ${DSAFILES}
Echo .${CAL_DIR}/davical_${DATENAME}.7z >> ${DSAFILES}
Echo .${MAIL_DIR}/mail_${DATENAME}.7z >> ${DSAFILES}
Cat ${SAVFILES} >> ${DSAFILES}

# Find the exclusion list
Cat ${EXCFILES} > ${DEXFILES}

Sh /etc/backup.exclude.sh >> ${DEXFILES}

# Backup to tape.
${TAR} cfvDEX /dev/rmt/0c ${DEXFILES} -I ${DSAFILES} \
1>/tmp/backup.spool 2>/tmp/backup.log

# We send an email message to root to notify that the backup has completed.
SYSADMIN=root
BACKUP_MAIL="/usr/bin/mailx"

HOSTNAME='hostname'
MSG="$HOSTNAME Backup completed"

Echo "Subject: $MSG"
Echo "$MSG"
Echo "Archive Disk Backup"
Echo "-------------------"
Echo .${DB1_DIR}/db1_${DATENAME}.7z
Echo .${DB2_DIR}/db2_${DATENAME}.7z
Echo .${DB3_DIR}/db3_${DATENAME}.7z
Echo .${CAL_DIR}/davical_${DATENAME}.7z
Echo .${MAIL_DIR}/mail_${DATENAME}.7z
Echo "Disk usage"
Echo """"""""
Df -k
Echo "Disk system status"
Echo """"""""
/sbin/zpool status
Echo "Backup Spool"
Echo """"""""
Head /tmp/backup.spool
Echo """"
Tail /tmp/backup.spool
Echo "Backup Log"
Echo """"
Cat /tmp/backup.log
)

| $BACKUP_MAIL -s "$MSG" $SYSADMIN
The file /etc/backup.include defines the directories to be included in the backup. Minimally, because space may be limited, then we need to backup the certificates, /etc directories with our system configuration, databases, mail and source control system. This is just sufficient to re-build the system and restore the existing functionality reasonably quickly. Ideally we would like to back-up everything.

Note “......” means there may be other files and is not part of the syntax:

```
./tank01/www
./etc
./zones/www/root/CA3yr
./zones/www/root/etc
./export/cvs
......
./export/home/bob
./export/home/alice
```

The file /etc/backup.exclude explicitly defines directories to exclude from the backup. Note “......” means there may be other files and is not part of the syntax:

```
./etc/svc/volatile
./etc/sysevent
./export/home/bob/.adabas
./export/home/bob/.adobe
./export/home/bob/.cache
......
./export/home/bob/.Trash
./export/home/bob/.updatemanager
......
./export/home/bob/tmp
./export/home/bob/working
......
./var/opt
./var/run
./var/sadm
./var/tmp
......
./tank01/root/www/etc/svc/volatile
./tank01/root/www/etc/sysevent
./tank01/root/www/var/opt
./tank01/root/www/var/run
./tank01/root/www/var/sadm
./tank01/root/www/var/tmp
./tank01/www/log
./tank01/www/DAViCal/awl-0.46
./tank01/www/DAViCal/awl
......
./etc/gconf
./etc/sane.d
./etc/security
./etc/ConsoleKit
./etc/net-snmp
./etc/X11
./etc/brltty
./etc/fonts
./etc/certs
```
The file `/etc/backup.exclude.sh` contains commands to dynamically construct an exclude list. Note “......” means there may be other files and is not part of the syntax:

```bash
#!/bin/sh
FIND=find
#
# Files to exclude
#
cd /
${FIND} ./export/home/bob -depth -name "core" -print
${FIND} ./export/home/bob -depth -name "*.mp3" -print
${FIND} ./export/home/bob -depth -name "*.log" -print
${FIND} ./export/home/bob -depth -name "*.o" -print
${FIND} ./export/home/bob -depth -name "*.a" -print
${FIND} ./export/home/bob -depth -name "*##" -print
${FIND} ./export/home/bob -depth -name "*~" -print
${FIND} ./export/home/bob -depth -name "*.##" -print
${FIND} ./export/home/bob -depth -name "*.iso" -print
${FIND} ./export/home/bob -depth -name "*.zip" -print
${FIND} ./export/home/bob -depth -name "*.7z" -print
${FIND} ./export/home/bob -depth -name "*.gz" -print
.....
#
${FIND} ./export/home/alice -depth -name "core" -print
.....
```

Edit crontab as root to schedule the backup job.

```bash
sudo su
hal# EXPORT EDITOR=me
hal# crontab -e
```

This runs up the editor `me`. Edit the cron job to run the script `/etc/backup` everyday at 3am. Add the following lines.

```bash
# Schedule the backup at 3am every day
0 3 * * * /etc/backup
```

The cron job will run everyday and send an E-Mail message as follows:

```plaintext
Subject: hal Backup completed
hal Backup completed
Archive Disk Backup
---------------------
```

www.jasspa.com
OpenIndiana Small System Server Build
14 Backup

/tank01/backups/db1/db1_20140301_030000.7z
/tank01/backups/db2/db2_20140301_030000.7z
/tank01/backups/calendar/davical_20140301_030000.7z
/tank01/backups/mail/mail_20140301_030000.7z

Disk usage
----------

Filesystem  kbytes  used  avail  capacity  Mounted on
rpool/ROOT/openindiana-151a7 102703104 9347731 82600490 11% /
/devices 0 0 0 0% /devices
/dev 0 0 0 0% /dev
ctrfs 0 0 0 0% /system/contract
proc 0 0 0 0% /proc
mnttab 0 0 0 0% /etc/mnttab
swap 3825928 428 3825500 1% /etc/svc/volatile
objsfs 0 0 0 0% /system/object
sharefs 0 0 0 0% /etc/dfs/sharetab
/usr/lib/libc/libc_hwcap2.so.1 91948221 9347731 82600490 11% /lib/libc.so.1
fd 0 0 0 0% /dev/fd
swap 3836316 10816 3825500 1% /tmp
swap 3825648 148 3825500 1% /var/run
tank01/aux 2873622528 682065064 1567710456 31% /aux
rpool/export 102703104 33 82600490 1% /export
tank01/cvs 2873622528 3508868 1567710456 1% /export/cvs
rpool/export/home 102703104 37 82600490 1% /export/home
tank01/export/home/bob 2873622528 95413540 1567710456 6% /export/home/bob
rpool 102703104 47 82600490 1% /rpool
tank01 2873622528 232 1567710456 1% /tank01
tank01/backups 2873622528 93235912 1567710456 6% /tank01/backups
tank01/export 2873622528 152 1567710456 1% /tank01/export
tank01/export/home 2873622528 152 1567710456 1% /tank01/export/home
tank01/homes 2873622528 9437752 1567710456 1% /tank01/homes
tank01/mail 2873622528 2498572 1567710456 1% /tank01/mail
tank01/opt 2873622528 21281648 1567710456 2% /tank01/opt
tank01/public 2873622528 183441576 1567710456 11% /tank01/public
tank01/share 2873622528 144 1567710456 1% /tank01/share
tank01/www 2873622528 1582704 1567710456 1% /tank01/www
tank01/tv 2873622528 129965272 1567710456 8% /tv
tank01/mysql 2873622528 2484 1567710456 1% /var/mysql/5.1/data
tank01/postgres 2873622528 69844 1567710456 1% /var/postgres
rpool/zones 102703104 32 82600490 1% /zones
rpool/zones/www 102703104 33 82600490 1% /zones/www
rpool/zones/www/ROOT/zbe 102703104 1091627 82600490 2% /zones/www/ROOT
/export/home/bob 1663123996 95413540 1567710456 6% /home/bob

Disk system status
--------------------

pool: rpool
state: DEGRADED
status: One or more devices are faulted in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
degraded state.
action: Replace the faulted device, or use ‘zpool clear’ to mark the device
        repaired.
scan: resilvered 14.9G in 0h4m with 0 errors on Sat May 25 16:59:45 2013
config:

NAME   STATE    READ  WRITE  CKSUM
rpool   DEGRADED 0     0      0
mirror-0 DEGRADED 0     0      0
c5d1a0  ONLINE   0     0      0

www.jasspa.com
oi_setup v1.25 2015/08/16 11:39:06
c5d0s0 FAULTED 5 284 0 too many errors

errors: No known data errors

pool: tank01
state: ONLINE
status: Some supported features are not enabled on the pool. The pool can still be used, but some features are unavailable.
action: Enable all features using ‘zpool upgrade’. Once this is done, the pool may no longer be accessible by software that does not support the features. See zpool-features(5) for details.
scan: scrub canceled on Sat Mar 1 11:57:40 2014
config:

<table>
<thead>
<tr>
<th>NAME</th>
<th>STATE</th>
<th>READ</th>
<th>WRITE</th>
<th>CKSUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>tank01</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mirror-0</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c3t0d0</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c3t1d0</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

errors: No known data errors

Backup Spool
-------------------

a ./tank01/backup/db1/db1_20140301_030000.7z 16 tape blocks
a ./tank01/backup/db2/db2_20140301_030000.7z 195 tape blocks
a ./tank01/backup/db3/db3_20140301_030000.7z 11 tape blocks
a ./tank01/backup/calendar/davical_20140301_030000.7z 928 tape blocks
a ./tank01/backup/mail/mail_20140301_030000.7z 2511817 tape blocks
....
a ./export/home/bob/..... excluded
a ./export/home/bob/.thunderbird excluded
a ./export/home/bob/somefile 23 tape blocks
a ./export/home/alice/ 0 tape blocks

Backup Log
-----------
tar: ./etc/dev/.devname_lookup_door is not a file. Not dumped
tar: ./etc/dev/.devfsadm_synch_door is not a file. Not dumped

15 JASSPA MicroEmacs

JASSPA MicroEmacs (me) is my default editor and needs to be installed and set-up. The pre-built zero install image is used.

```
hal% wget http://www.jasspa.com/development/me-standalone/
   jasspa-SunOS5.10-1386-20091212.gz
hal% gunzip -c jasspa-me-SunOS5.10-1386-20091212.gz > me
hal% chmod a+x me
hal% chmod a-w me
hal% sudo cp me /usr/bin
```

Install spelling dictionaries and icons for the desktop, these are installed globally in /usr/share/jasspa/-spelling.

```
hal% cd /tmp
hal% wget http://www.jasspa.com/spelling/ls_engb.tar.gz
hal% wget http://www.jasspa.com/spelling/ls_enus.tar.gz
hal% wget http://www.jasspa.com/release_20060909/meicons-extra.tar.gz
hal% sudo mkdir -p /usr/share/jasspa/spelling
hal% cd /usr/share/jasspa/spelling
```

www.jasspa.com
hal% sudo tar xzvf /tmp/ls_enqb.tar.gz
hal% sudo tar xzvf /tmp/ls_enus.tar.gz
hal% cd /usr/share/jasspa
hal% sudo tar xzvf /tmp/meicons-extra.tar.gz

Set up MicroEmacs as user and root, my preferred settings are:

hal% me
M-x user-set-up
Start-Up: Edit = OFF
Start-Up: Keyboard = British
Start-Up: Language = British
Platform Fonts:Font Name = -*-clean-medium-r-***-*--***-130-*--***-**
Platform Fonts: Fence Display = Always draw & jump on close
Platform Fonts: Scroll Bars = Wide with splitter
Platform Fonts: Color Scheme = Lumina

16 TeXLive

TeXLive is the \LaTeX Documentation System which typeset this document. To install then download the latest release as an ISO image from http://www.tug.org/texlive/

To mount an ISO image under Solaris

hal# lofiadm -a /export/home/bob/Downloads/texlive2012.iso

and list the ISO images

hal# lofiadm

then mount the ISO image

hal# mount -F hsfs -o ro /dev/lofi /mnt

Check to ensure that Solaris understands the image

hal# df -k /mnt
Filesystem kbytes used avail capacity Mounted on
/dev/lofi/1 512418 512418 0 100% /mnt

list the image

# ls /mnt

Then install TeXLive

hal# cd /mnt
hal#.install-tl

From the menu then change the install location to /opt/texlive/2012 using the d option and then proceed to install.

pre-generating all format files (fmtutil-sys --all), be patient... done
running package-specific postactions
finished with package-specific postactions

See
/opt/texlive/2012/index.html
for links to documentation. The TeX Live web site contains updates and corrections: http://tug.org/texlive.

TeX Live is a joint project of the TeX user groups around the world;
Please consider supporting it by joining the group best for you. The list of user groups is on the web at http://tug.org/usergroups.html.

Add /opt/texlive/2012/texmf/doc/man to MANPATH, if not dynamically determined. Add /opt/texlive/2012/texmf/doc/info to INFOPATH.

Most importantly, add /opt/texlive/2012/bin/i386-solaris to your PATH for current and future sessions.

Welcome to TeX Live!
Logfile: /opt/texlive/2012/install-tl.log

As a final step, unmount and detach the ISO image.

halt# cd /
halt# umount /mnt
halt# lofiadm -d /dev/lofi/1

16.1 TexLive User Setup

The user environment should then be edited to include TeXLive, in this case zsh is being used and the following is added to .zshrc in the user home directory

```
.....
# Include TeXLive 2012
#
if [ -d /opt/texlive/2012 ]; then
  PATH=$PATH:/opt/texlive/2012/bin/${PLATFORM}-solaris
  MANPATH=$MANPATH:/opt/texlive/2012/texmf/doc/man
  INFOPATH=$INFOPATH:/opt/texlive/2012/texmf/doc/info
fi
.....
# Export to the world
export PATH
export MANPATH
export INFOPATH
```

17 Client Device Configuration

This section defines the configuration of the client devices that use the services provided by the service. Some values are specific to the network and relate to the configurations used in previous sections of this document.

17.1 Static IP Addresses

When using static IP addresses in a network instead of DHCP then the following configuration is required:

IP Address: 192.168.8.x where 2<=x<=127
Netmask: 255.255.255.0
Gateway: 192.168.8.1
DNS: 192.168.8.200

Use the LAN DNS in preference to your ISP supplied DNS addresses.
17.1.1 OSX Lion DNS server priority

You do not need this fix if the DHCP server includes only the LAN DNS server.
In OS X v10.6 and later the search order is dynamic, this can cause problems with local DNS over-rides being resolved from the WAN rather than LAN which results in local server names becoming unresolved. To solve the problem then one could provide one DNS of the local server only.

My preferred solution is described here:
http://reviews.cnet.com/8301-13727_7-10471471-263.html
which is reproduced here:
To search DNS servers in a strict order in Mac OS X v10.6.3 or later. Making this change will result in DNS servers being tried in the specified search order for all queries, even if a server is not responsive. This may affect performance and reliability.

Log in as an administrator and back up the mDNSResponder.plist file. To do this open a terminal (in /Applications/Utilities) and execute the following command on a single line:
sudo mv /System/Library/LaunchDaemons/com.apple.mDNSResponder.plist /System/Library/com.apple.mDNSResponder.plist_previous_LaunchDaemon

NOTE: That Apple’s article says to use mv, but you should use cp in this command. Alternatively, just go to the mentioned folder via the Finder and copy the file to an alternate location.)

Close the Terminal and open the com.apple.mDNSResponder.plist file in a text editor, the file is located in /System/Library/LaunchDaemons/. Locate the following key in the file:

<key>EnableTransactions</key>
<true/>
</dict>

Between the last <true/> and </dict>, add the following lines:

<key>StrictUnicastOrdering</key>
<true/>

Save the file, open a Terminal and then restart mDNSResponder using the following two commands:
sudo launchctl unload /System/Library/LaunchDaemons/com.apple.mDNSResponder.plist
sudo launchctl load /System/Library/LaunchDaemons/com.apple.mDNSResponder.plist

17.1.2 OSX Lion DNS Search Domains

In Lion then the normal DNS search domain does not work as one might expect and short DNS names do not work. This may be fixed by reference to
which is reproduced here:

Make a backup of /System/Library/LaunchDaemons/com.apple.mDNSResponder.plist from the command line:
sudo cp /System/Library/LaunchDaemons/com.apple.mDNSResponder.plist /System/Library/LaunchDaemons/com.apple.mDNSResponder.plist.original

Edit com.apple.mDNSResponder.plist - it is a plain text file, so use whatever text editor you have handy. Do not forget to use sudo.
sudo vim /System/Library/LaunchDaemons/com.apple.mDNSResponder.plist

Add <string>-AlwaysAppendSearchDomains</string> after line 16
17.2 Mail Server

The mail server is available on the intranet (local) and internet (global)
SMTP: www.mydomain.co.uk:465 - SSL with plain authentication user/password
SMTP: www.mydomain.co.uk:587 - STARTLS with plain authentication user/password
IMAP: www.mydomain.co.uk:993 - SSL with plain authentication user/password

Use SSL with password authentication for both services. The outgoing password is the same as the incoming.
This is a relatively standard mail server configuration and is supported with most mailers such as Thunderbird, Outlook, OS-X, iOS etc. On Android **K-9 Mail** may be used as a mail client.

17.3 Calendar

A CalDAV Calendar client connects to the server as follows:
https://www.mydomain.co.uk:8443/user/calendar
Where *user* is the username, a password is required. Multiple calendars may be subscribed to if access has been granted, including calendar sharing. On Apple devices (OS-X, iOS) then the pathname `/user/-calendar` is not required and the system will locate this based on the user name.

17.4 Addressbook

A CalDAV Calendar client connects to the server as follows:
https://www.mydomain.co.uk:8443/user/addressbook
Where *user* is the username, a password is required. On Apple devices (OS-X, iOS) then the pathname `/user/addressbook` is not required and the system will locate this based on the user name.
On Android **CardDAV Sync Free** may be used to download your address book to the phone.

17.5 WebDAV

A WebDAV client connects to the server as follows:
https://www.mydomain.co.uk:8081/dir
Where `/dir` is optionally specified if the WebDAV server is configured to restrict users to specific locations on the server. A username and password is required.
The above syntax is supported natively on OS-X (Finder=>Go=>Connect to Server).
For Microsoft Windows the application **BitKinex** available from WWW does a good job for uploading content.
On iOS then WebDAV is supported natively in **Pages**, **Numbers** and **Keynote**. The **WebDAV Navigator** app is a free client application which can be useful.
The WebDAV service may be opened with a regular web browser using the aforementioned URL for reading and content download.

17.6 WebServer

Web services may be available with/without SSL, depending on the configuration
http://www.mydomain.co.uk - Without SSL
https://www.mydomain.co.uk - With SSL and possibly password authentication

17.7 DAViCal Administrator

Administrator access for DAViCal from machine hal only using a web browser.
https://www.mydomain.co.uk:8008

17.8 Printing

Printing may be performed via IPP, the DNS and mDNS configuration should allow the printers to be automatically located on OSX and iOS using Bonjour and Airprint.
For Microsoft Windows devices then hal.mydomain.co.uk:631 may be used for printing, the correct printer drivers should be installed.
For iOS Airprint then there are a couple of issues as follows:

- Disable duplex when printing otherwise nothing happens (iOS issue)
- Pictures are not scaled to a single sheet. I think this is a CUPS issue and the default in later versions of CUPS is to scale an image to fit the page.
- For iOS 7 then mDNS must be enabled, in addition then the DNS/mDNS definition must minimally include URF=DM3. The iOS 6 constraints also apply.
- For iOS 6 then the DNS definition pdl=... must include image/urf which must also be handled in the CUPS configuration (later releases of CUPS include this by default).

17.9 CUPs Print Server Administration

Administration of the CUPs Server from a web browser:
http://hal.mydomain.co.uk:613 - from the LAN
http://localhost:613 - from the global zone
Visibility will depend on the server configuration.

17.10 Samba

Samba file system
smb://hal.mydomain.co.uk/user - LAN
smb://hal.local/user - When mDNS is enabled
The above syntax is supported on OS-X (Finder=>Go=>Connect to Server).
From iOS then the FileBrowser(FB) app may be used to connect to a SMB share.
Windows natively supports SMB.
17.11 Samba Administration (SWAT)

Administration of Samba from a web browser:
http://hal.mydomain.co.uk:901 - from the LAN
http://localhost:901 - from the global zone
Visibility will depend on the server configuration.

17.12 SSH

If SSH has been enabled on the network then the server may be accessed as follows:

OS-X: ssh -X -Y -l user hal.local - Using mDNS
OS-X: ssh -X -Y -l user hal.mydomain.co.uk - Using DNS
*NIX: ssh -X -l user hal.local - Using mDNS
*NIX: ssh -X -l user hal - If not using mDNS
*NIX: ssh -X -l user hal.mydomain.co.uk - Using DNS

On iOS then the iSSH app may be used to connect and login to the server.
The SSH service has to be enabled on the server.

18 Conclusion

Some 18 months from commencing this project then I can say I have very few regrets over the selection of
HP Microserver hardware and OpenIndiana operating system environment.
OpenIndiana has been extremely solid. Writing this then the up time is 260 days, the last time the system
was re-booted was to replace a failed system disk.

```
hal% uptime
2:14pm up 260 day(s), 4:06, 1 user, load average: 0.05, 0.04, 0.04
```

The original system was installed with OpenIndiana oi_151a5 which was upgraded to oi_151a7 when the
system disk was replaced.
Setting up the system from scratch was quite time consuming and took some 2 weeks part-time to get the
system running with all of the services required with a few mistakes along the way and a lot of web searching.
Installation is by no means a point and click operation but the time spent on correctly setting up the system
is time saved later as the administration has been virtually zero and only forced through disk failures.
The advantages of using ZFS are huge, witnessed firsthand by the disk failures that have occurred. It is
difficult to comprehend the reliability of any system if there is any danger that a single bit in an executable
binary or data file becomes corrupted which will ultimately affect the running system and cause a mystifying
crash or to exhibit strange behaviour. ZFS protects the system and easily allows the failed storage to be
removed and replaced immediately (provided of course one watches out for a fault).
For the period then the savings in electricity consumption moving from a SunBlade 2500 to HP Microserver
have completely paid for the system as shown in Figure 8 (graph produced on iOS by Meter Readings).
Figure 8: Power consumption for the period

Power saving when replacing SunStale 2500 with HP Microserver