Linux ADSM Mini−Howto

Table of Contents

Linux ADSM Mini−Howto

by Thomas König, `Thomas.Koenig@ciw.uni−karlsruhe.de`

v, 15 January 1997

This document describes how to install and use a client for the commercial ADSM backup system for Linux/i386.

1. Introduction

ADSM is a network−based backup system, sold by IBM, in use at many organizations. There are clients for a large variety of systems (different UNIX brands, Windows, Novell, Mac, Windows NT). Unfortunately, at the time of this writing, there is no native Linux version.

You will have to use the SCO binary, and install the iBCS2−emulator for running ADSM. This description is for ADSM v2r1.

At the time if this writing, I am only aware of a version which works with the i386 version of Linux.

2. Installing the iBCS module

The iBCS2 module is available from ftp://tsx−11.mit.edu/pub/linux/BETA/ibcs2. If you are running kernel version 1.2.13, get `ibcs−1.2−950721.tar.gz`, unpac it and apply the patches `ibcs−1.2−950808.patch1` and `ibcs−1.2−950828.patch2`. You can then type "`make`" and install the iBCS modlue with "`insmod`".

For a 2.0 kernel version, get `ibcs−2.0−960610.tar.gz`, unpack it in a suitable place, chdir into that directory, and apply the following patch:

```
--- iBCSemul/ipc.c.orig Wed Jan 15 21:32:15 1997
+++ iBCSemul/ipc.c      Wed Jan 15 21:32:31 1997
@@ -212,7 +212,7 @@
        switch (command) {
                case U_SEMCTL:
                        cmd = ibcs_sem_trans(arg3);
-                       arg4 = (union semun *)get_syscall_parameter (regs, 4);
+                       arg4 = (union semun *)(((unsigned long *) regs->esp) + (5));
                        is_p = (struct ibcs_semid_ds *)get_fs_long(arg4->buf);
 #ifdef IBCS_TRACE
                        if ((ibcs_trace & TRACE_API) || ibcs_func_p->trace)
```

Then, copy `CONFIG.i386` to `CONFIG`, and type `make`.

If you don't have them already, create the needed device files by executing

```
# cd /dev
# ln -s null XOR
# ln -s null X0R
# mknod socksys c 30 0
# mknod spx c 30 1
```

3. Installing the ADSM client

The SCO binary is supplied as three tar files, or disks. Change to the root directory, set your umask according to your policies, and unpack them from there (as root). In your Directory /tmp, you will find an installation script; execute that.

You will then have to hand−edit `/usr/adsm/dsm.sys` and `/usr/adsm/dsm.opt`. In `dsm.sys`, important lines to specify are:

Servername

The name of the server

TCPServeraddress

The fully qualified host name of the server

NODename

Your own hostname

In `dsm.opt`, you will have to specify

Server

As before

Followsymbolic

Wether or not to follow symbolic links (not a good idea, in general)

SUbdir

Wether to back up subdirectories (you usually want that)

domain

The file systems to back up

You will then have to create a SCO−compatible `/etc/mnttab` from your `/etc/fstab`. You can use the following Perl script, `fstab2mnttab`, for this.

```
#!/usr/bin/perl

$mnttab_struct = "a32 a32 I L";

open(MTAB, "/etc/mtab") || die "Cannot open /etc/mtab: $!\n";
open(MNTTAB, ">/etc/mnttab") || die "Cannot open /etc/mnttab: $!\n";

while(<MTAB>) {
    next if /pid/;
    chop;
    /^(\S*)\s(\S*)\s(\S*)\s.*$/;
    $device = $1;
    $mountpt = $2;
    $fstype = $3;
    if($fstype ne "nfs" && $fstype ne "proc") {
        $mnttab_rec =
            pack($mnttab_struct, $device, $mountpt, 0x9d2f, time());
        syswrite(MNTTAB, $mnttab_rec, 72);
        print "Made entry for: $device $mountpt $fstype\n";
    }
}

close(MNTTAB);
exit 0;
```

You do not need to install any shared libraries for these clients; everything is linked statically.

4. Running the client

There are two clients, `dsm`, which is an X11 interface, and `dsmc`, a command−line interface. Your computer centre will tell you how to run it. Some startup script at boot, for example

```
dsmc schedule −quiet 2>&1 >/dev/null &
```

will probably be required.

5. Known Problems

Unfortunately, SCO can only deal with hostnames no longer than eight characters. If your hostname is longer, or fully qualified, you may need to specify your hostname on the `NODename` line in `/usr/adsm/dsm.sys`.

If you use the DISPLAY variable, you will have to supply the fully qualified host name (i.e. `DISPLAY=host.full.do.main:0` instead of `DISPLAY=host:0`).