

Adaptec Storage Manager for Supermicro AOC-2020SA RAID controller

While playing around with a Supermicro 6014P-T server I wanted to install a (Linux) CLI to manage the AOC-2020SA RAID controller.

Bad enough that Adaptec Storage Manager (ASM) included [with the Supermicro driver disk](#) does not include a CLI.

Let's not argue about whom had the great idea to provide a GUI-only java-based management application to be used on 19" 1U rackmount servers.

Of course I'd never run such a server headless or without X11 installed so why would I ever need a CLI!?

At least they make it up by supplying a newer version of ASM which includes a CLI.

It's only a shame that it's hidden away on their ftp server within the driver disk of another disk controller, making it somewhat an adventure to find it eventually...

Look for the [AOC-LPZCR1/AOC-SOZCR1 driver](#) to get the x86 or x86_64 linux version of ASM.

Install ASM CLI

As I only need the ASM CLI without the java GUI stuff et all, the following will describe how to do it.

This command will download the x86 package and extract the "arconf" binary.

When asked to "rename ./usr/StorMan/arconf" simply provide the new location where it is to be stored, e.g. /usr/sbin/arconf.

Replace the "ftp://url/to/rpmfile" stanza by the actual url.

```
# lynx -dump ftp://url/to/rpmfile | rpm2cpio - | cpio -ivmur '*arconf'  
rename ./usr/StorMan/arconf -> /usr/sbin/arconf
```

Afterwards the extracted binary needs to be assigned execute permissions:

```
# chmod 700 /usr/sbin/arconf
```

Now since the binary is dynamically linked a check for eventually missing libraries is required:

```
# ldd /usr/sbin/arconf  
linux-gate.so.1 => (0xffffe000)  
libpthread.so.0 => /lib/libpthread.so.0 (0xf7fdd000)  
libdl.so.2 => /lib/libdl.so.2 (0xf7fd9000)  
libstdc++-libc6.2-2.so.3 => not found  
libm.so.6 => /lib/libm.so.6 (0xf7f6f000)  
libc.so.6 => /lib/libc.so.6 (0x00afb000)  
/lib/ld-linux.so.2 (0x00ade000)  
libgcc_s.so.1 => /lib/libgcc_s.so.1 (0xf7f62000)
```

More recent distros, RHEL 5 in may case, usually require compatibility libraries to be installed.

Depending on your distro YMMV.

On RHEL, and other rpm-based distros with YUM support, missing libraries may be located like this:

```
# yum whatprovides libstdc++-libc6.2-2.so.3  
Loading "rhnpugin" plugin  
Loading "installonlyn" plugin  
This system is not registered with RHN.  
RHN support will be disabled.  
Setting up repositories
```

Reading repository metadata in from local files

compat-libstdc++-296.i386 2.96-138

rhel-base

Matched from:

libstdc++-libc6.2-2.so.3

Finally the package named in yum is to be installed accordingly:

```
yum install compat-libstdc++-296.i386
```

So after getting around these hurdles the CLI is ready to go.