OSCAR 2.2.1

클러스터 설치 가이드

Cluster installation Guide for OSCAR ver. 2.2.1

홍태영, 홍정우, 조혜영 편저

KiSTi 한국과학기술정보연구원
Korea Institute of Science and Technology Information
www.kisti.re.kr

OSCAR 2.2.1
클러스터 설치 가이드

KiSTi www.kisti.re.kr 한국과학기술정보연구원 Korea Institute of Science and Technology Information

OSCAR 2.2.1 클러스터 설치 가이드

Cluster installation Guide for OSCAR ver. 2.2.1

이 책은 영문 original version의 저자와의 협의 하에 출판되었으며, 이 책의 무단 복제 및 무단 전제를 지양해 주시기 바랍니다.

Published by Supercomputing Center, KISTI
Printed in Korea

ISBN 89-5884-068-4 93560

이 책에 대한 의견이나 조언을 주시고자 할 때 그리고 오탈자나 버그 등을 발견했을 경우, 언제든지 다음의 저자의 메일주소로 연락하시기 바랍니다.

tyhong@kisti.re.kr
jwhong@kisti.re.kr
chohy@kisti.re.kr

머리말

클러스터 컴퓨터는 우수한 가격대비 성능과 시스템의 유연한 구성과 확장성 등의 장점을 갖고 있으며 이와 더불어, 리눅스 및 병렬 프로그램 등 오픈소스 기반의 소프트웨어의 표준화 및 안정화가 이루어지면서, 고성능 컴퓨팅 시장을 급속도로 대체해나가고 있다. 2000년 6월 기준으로 세계 top500 슈퍼컴퓨터 리스트에서 2% 정도의 분포를 보이던 클러스터 컴퓨터의 수는 2004년 11월 현재 그 수가 약 60 %에 육박할 정도로 폭발적으로 증가하고 있으며 점차 대형화 및 고급화되어 가고 있으며, 범용 고성능 컴퓨팅 플랫폼으로 각광받고 있다. 하지만 그럼에도 불구하고 클러스터 컴퓨팅 설치 및 관리도구는 상용 클러스터 소프트웨어에 비하여 아직 초보적인 단계를 벗어나고 있지 못하며, 몇몇 공개 소프트웨어 기반의 클러스터 설치도구들만이 중소형 클러스터 시스템 환경에서 폭넓게 이용되고 있다. 이에 이 책에서는 그 중에서 가장 안정적이며, 사용자 환경이 편리하게 되어 있는 클러스터 설치도구인 OSCAR 2.2.1를 소개하고자 한다. OSCAR는 중소형 클러스터 컴퓨터를 주요 타겟으로 만들어졌으며, 편리한 사용자 환경 및 툴을 제공함으로서, 클러스터링 경험이 없는 시스템 관리자라도 쉽게 클러스터를 설치 운영할 수 있도록 하고 있다. 따라서 대학 혹은 연구소의 연구실 등에서 소형 클러스터를 자체 제작하여 운영하고자 하는 경우 최적의 솔루션 중 하나로 선택할 수 있는 도구이다. 이 책은 리눅스를 관리 및 운영해본 경험이 있는 사람을 대상으로 하고 있으며, 따라서 리눅스 운영체계 및 사용방법에 대해서는 별도의 설명을 생략하고 있다. 아무쪼록 이 책이 클러스터를 좀 더 쉽게 구축하여 사용하기를 희망하는 모든 사람에게 조금이나마 도움이 되었으면 한다.

홍태영, 홍정우, 조혜영
슈퍼컴퓨팅센터,
한국과학기술정보연구원

목차

1. 소개 1
1.1 용어 설 1
1.2 지원 리눅스 배포판 3
1.3 최소 시스템 요구사 3
1.4 테스트 베드 사양 4
1.5 문서 개요 5

2 리눅스 배포판별 주의사항 7
2.1 일반적인 주의사항 7
2.2 RedHat 7.2 주의 사항 7
2.3 Mandrake 8.2 주의 사항 8
2.4 Mandrake 9.0 주의 사항 9
2.5 IA64 시스템 기타 시스템 관련 주의 사항 10

3 System Installation Suite(SIS) 개요 11

4 세부적인 클러스터 설치 과정 13
4.1 Server Installation and Configuration 13
4.2 (선택사항) 추가적인 oscar packages 다운로드 14
4.3 OSCAR installer 실행 16
4.4 (선택사항) "Select packages to install"(step 1) 17
4.5 "Configure selected OSCAR Packages"(step 2) 18
4.6 "Install OSCAR server packages"(step 3) 20
4.7 "Build OSCAR client Image"(step 4) 21
4.8 "Define OSCAR Clients"(step 5) 24
4.9 "Setup Networking"(step 6) 26
4.10 (참고 사항) Client Installations 28
4.11 "Complete the Cluster Setup"(step 7) 28
4.12 "Test Cluster Setup"(step 8) 29
4.13 Congratulations! 30
4.14 Adding and Deleting client nodes 30
4.15 Starting over 31

5 클라이언트 시스템 업데이트 및 업그레이드 33

6 각 패키지별 설치 관련 주의 사항 37
6.1 C3 37
6.2 disable-services 38

6.3 HDF5 39
6.4 kernel_picker 41
6.5 loghost 41
6.6 networking 42
6.7 ntpconfig 42
6.8 ODA 43
6.9 OPIUM & SSH 43
6.10 PBS & Maui Scheduler 44
6.11 pfilter 47
6.12 switcher 48

7. 참고문헌 51

1 소개

OSCAR는 HPC용 클러스터 관련 분야에 종사하는 사람들의 비공식 모임인 OCG(Open Cluster Group)의 오픈 소스 프로젝트로서 2001년 2월 첫 번째 버전이 출시되었으며, 가장 최신 버전으로는 2003년 9월 16일 출시된 2.3.2 버전이 있다. OCG의 멤버로는 IBM, Indiana Univ., Intel, NCSA(National Center for Supercomputing Applications), ORNL(Oak Ridge National Laboratory) MSC.Software등이 참여 하고 있다.
OSCAR(Open Source Cluster Application Resource) 소프트웨어 패키지는 cluster를 설치하는데 필요한 복잡한 작업을 단순화시키는데 목적을 두고 있다. OSCAR cluster는 HPC(High Performance Computing)에 사용하도록 만들었지만, 다양한 종류의 cluster용(cluster-enabled) application에 사용할 수도 있다. OSCAR는 HPC를 겨냥해서 만들어 졌으므로, MPICH, LAM/MPI, PVM, PBS등 여러 HPC 관련 package들이 default로 설치된다. 이 문서는 시스템관리자에게 단계별 installation guide와 설치과정 중에 발생하는 것들에 대해 자세한 설명을 제공한다. 마지막으로 이 문서는 OSCAR version 2.2.1을 대상으로 만들어졌음을 주지하기 바란다.

1.1 용어 설명

Cluster: 소프트웨어와 하드웨어를 사용하여, 하나의 machine처럼 작동하도록 설계된 일단의 컴퓨터 그룹을 의미한다.

Node: cluster내의 개개의 computer를 의미한다. install할 OSCAR cluster의 경우 두 종류의 node가 존재한다. 하나는 서버노드(head node)로 클라이언트 노드의 요청에 대해 필요한 서비스를 제공한다. client node는 계산 전용이며, 일반적으로는 계산노드(compute node)라고도 부른다. OSCAR의 cluster는 1개의 서버와 1개 이상의 client로 이루어지며, 모든 client는 homogeneous(i.e., identical) hardware를 가지고 있어야 한다. 사실상 OSCAR 내의 package들은 heterogeneous(즉, non-homogeneous) cluster를 지원하지만, 현재 그 내용은 문서화 되어 있지 않다.

Image: 하나의 directory내에 root file system hierarchy를 완전히 갖추고 있으며, client에게 언제든지 설치 및 update를 통해 제공될 수 있다. 단 image 안의 file system에서 proc, nfs, devpts, iso9660은 제외된다. Image는 oscar install시 서버의 /var/lib/sysimager/images/에 만들어져서, client node의 local disk에 rsync를 이용하여 복사된다.

Client definition: 서버가 가지고 있어야 하는 client의 기본정보로, 여기에는 ip address, hostname, default gateway 등의 network 정보와 image 이름이 포함된다.

Packages: OSCAR의 Package들은 기본적으로 모두 RPM이거나 RPM의 collection이며, configuration 및 installation 관련 파일을 포함하는 경우도 있다(아래 참조).
client에 설치할 package는 root/oscar-2.2.1/oscarsamples/RedHat-7.3-i386.
rpmlist를 바탕으로 고를 수 있다. OSCAR의 패키지들은 크게 다음의 세 종류로 나눌 수 있다.

• Core packages: OSCAR의 공식 배포판에 포함되어 있으며, OSCAR installation 과정을 진행하기 위해 필요한 핵심 package(sis, switcher, c3, oda)이다.

• Included packages: OSCAR의 공식 배포판에 포함된 packages(autoupdate, hdf5, loghost, ntpconfig, pvm, kernel_picker, mpich, pbs, disable-services, lam, networking, opium, pfilter)이다. 이들은 보통 OSCAR의 developer가 만들거나 혹은 패 키지화하며, release되기 전에 상당한 수준의 공식 test 과정을 거친다.

• Third party packages: OSCAR의 공식 배포판에 포함되지 않은 package들로, OSCAR tree(/tftpboot/rpm)에 add-on되어 OSCAR installation 과정 중에 함께 설치 될 수 있 다.

1.2 지원 리눅스 배포판

OSCAR의 설치를 성공적으로 하기위해서는 아래의 표에서 "Fully Supported"로 표시된 distribution을 선택하기 바란다.

Distribution and Release	*Status*
RedHat 7.1 *RedHat 7.2* *RedHat 7.3*	*Experimentally supported* *Fully supported* *Fully supported*
Mandrake 8.2 *Mandrake 9.0*	*Fully supported* *Experimentally supported*

(표1)

1.3 최소 시스템 요구사항

OSCAR server

• i586이상의 cpu

• TCP/IP stack을 지원하는 network interface card 1개 이상 (OSCAR 서버가 public network과 cluster nodes를 연결하는 router 역할을 할 경우 2개가 기본적으로 필요하다)

• Linux 설치 후, OSCAR 설치 전에 /var에 2G, / 가 최소 2G 이상의 여분의 공간이 있어야 한다.(서버 Installation 과정 참조)

• 위의 표에 "Fully supported"가 표시된 Linux 배포판

OSCAR client nodes

• i586이상의 cpu (SMP 지원)

• 각 node당 2GB 이상의 disk (OSCAR를 default로 설치시 최소 600M 이상의 공간 필요)

• TCP/IP stack을 지원하는 network interface card (PXE 지원 NIC을 추천한다.)

• 위의 서버와 같은 version의 Linux 배포판
• 모든 client는 같은 architecture를 가지고 있어야 한다(e.g., ia32 vs ia64)
• PXE 지원 BIOS와 PXE NIC을 가지고 있다면 Floppy drive가 없어도 된다. 그렇지 않은 경우 FDD가 있어야 한다.

(주의사항) OSCAR 및 RedHat 7.3은 기본적으로 다음의 Gigabit Ethernet Card를 지원한다. 하지만 이 문서를 위해 직접 test를 해보지는 않았다.
Broadcom BCM5700 PCI-X 1000BASE-T
intel EtherExpress/1000 Gigabit

1.4 테스트베드 사양

서버

RedHat 7.3 "Workstation"
Pentium II 450 MHz
128M RAM
HDD IBM-DJNA-351520 15.2GB
FDD 3,5" 1.44 MB
SAMSUNG CD-ROM
100Mb Ethernet

4 client nodes

Pentium III 450 MHz SMP
256M RAM
QUANTUM FIREBALLP LM20.5 19.5GB
FDD 3,5" 1.44 MB
100Mb Ethernet

1.5 문서 개요

High-Performance cluster를 구성하는 데 있어서 고려해야 할 복잡한 특성들을 감안하여, 경험이 풍부한 관리자일지라도 이 문서를 어느 한 부분도 skip하지 않고, 끝까지 읽어본 후에 install과정을 진행하기 바란다. 또 이전에 cluster를 구축해 보지 않은 관리자일지라도, OSCAR를 통해 무리 없이 cluster 설치 과정을 진행할 수 있다는 것을 인지하기 바란다. 이 문서의 나머지 부분은 다음으로 구성되어 있다. Section 2 “Release Notes”에서는 각 리눅스 배포판에서 OSCAR 2.2.1을 설치하기위해 사전에 필요한 작업들을 설명한다. Section 3에서는 OSCAR cluster 설치의 핵심 tool인 SIS(System Installation Suite)에 대해 간단한 설명을 한다. Section 4에서는 cluster installation과정에 대해, “지금 설치 과정은 어떤 파일을 update하고 있다”와 “지금 설치 과정은 어떤 파일의 "xyz" 문자열과 "abc"를 교체하고 있다” 의 중간정도 레벨에서 자세히 설명할 것이다. Section 5에서는 OSCAR를 사용하여 install한 cluster를 update/upgrade하는 방법을 간단히 설명한다. 마지막으로 Section 6에서

는 OSCAR에 의해 설치된 package들에 대해 시스템 관리 시 알아야 할 사항들 위주로 설명한다.

2 주의사항

2.1 일반적인 주의사항

• OSCAR는 기존에 설치된 Linux 위에 설치 될 수 있지만, 가장 좋은 방법은 위의 (표1)에 열거된 Linux 배포판중에서 하나를 설치한 후, update없이 바로 OSCAR를 설치하는 것이다. 만일 update를 먼저 한 후, OSCAR를 설치하면 rpm requirement conflict가 발생할 수 있다. Redhat의 경우 모든 update는 OSCAR install이 끝난 후에 하는 것이 좋다. Mandrake의 경우 OSCAR를 install하기 전에 security update를 먼저 해야 한다. http://www.mandrakesecure.net/en/advisories/를 참고하기 바란다.

• 다음의 가벼운 경고 메시지가 OSCAR installation 과정 중에 자주 나타날 것이다.

rsync_stub_dir: no such variable at ....

이러한 경고는 무시해도 상관없다.

• OSCAR installer(i.e., /root/oscar-2.2.1/install_cluster)는 서버 node에 MySQL이 존재하지 않으면, 이를 자동으로 설치 할 것이다. OSCAR는 MySQL database의 root 패스워드가 없어야 설치 될 수 있다.

• OSCAR installer는 설치시 사용자의 실수에 대해 경고 메시지를 거의 대부분 보내지 않는다. 따라서 사용자가 잘못된 과정을 실행하거나, 잘못된 입력을 할 때에는 여러 가지 문제들이 일어날 수 있다. 따라서 이 문서에 제공된 절차를 잘 숙지하고 실행하기를 권고한다.

2.2 RedHat 7.2 주의 사항

• OSCAR installer GUI가 시작할 때, RedHat 7.2에 설치된 Perl은 여러 변수들에 대해 "will not remain shared"와 같은 경고 메시지를 보내지만, 무시해도 상관없다.

2.3 Mandrake 8.2 주의 사항

• OSCAR installer를 실행하기 전에, libexpat1_95-devel RPM을 설치해야 한다.(urpmi 명령어를 실행하기 위해서는 Mandrake 8.2 CD가 필요하다.)

urpmi libexpa1_95-devel

이 RPM은 OSCAR의 초기 installation과정에서 설치되는 perl-XML-Simple과 perl-XML-Parser Perl modules을 위해 필요하다. 또 다른 방법으로 OSCAR를 설치하기 전에 perl-XML-
Simple과 perl-XML-Parser를 위한 Mandrake RPM을 설치하는 방법이 있다.

urpmi perl-XML-Simple perl-XML-Parser

• OSCAR installer GUI(즉, OSCAR installation wizard)는 xterm을 사용한다. 따라서 xterm이 설치되지 않았다면, xterm-
165-3mdk.i586.rpm을 설치해 주어야 한다. Mandrake CD를 넣은 후 다음을 실행하면 된다.

```
# urpmi xterm
```

• Client Image를 만들 때, 두 가지의 경고 메시지가 보일 것이다. 즉, lam-oscar와 mpich-oscar RPM을 설치할 때, libtcl.so.0이 존재하지 않는다는 error 메시지를 볼 수 있을 것이다. 하지만 이것은 무시해도 괜찮다.

• third party package인 Ganglia를 설치하고자 한다면, OSCAR installer를 실행하기 전에, RPM version 1.0.8-2mdk인 libpng를 설치해야 한다. 이 RPM은 Ganglia 배포판의 다음의 위치에서 찾을 수 있다:

package/ganglia/extras/libpng-1.0.8-2mdk.i586.rpm

• root의 default shell configuration file들은 /etc/profile.d/의 startup script에서 무엇이 추가되건 간에 PATH 환경변수를 고정시켜 버린다. 이것은 /opt에 설치된 여러 OSCAR component에 악영향을 준다. 따라서 OSCAR installer는 다음의 shell configuration file에서 이 환경변수에 대한 설정을 주석처리 한다.

/root/.bashrc
/root/.cshrc
/root/.tcshrc

2.4 Mandrake 9.0 주의 사항

다음의 과정은 Mandrake 9.0에서 OSCAR installer를 실행하기 전에 해야 한다. Mandrake 9.0은 (표1)에서 볼 수 있는 바와 같이, OSCAR에서 “experimentally supported”되는 배포판이며, 가벼운 test만 이루어졌음을 유의하기 바란다.

• CPAN을 이용하여, 다음의 perl module을 설치 한다:

```
# perl -MCPAN -e shell
 install SQL::Statement
```

• python 2 compatibility RPM을 설치한다. 이것은 OSCAR 배포판의 packages/c3/RPMS 에서 찾을 수 있다.

```
# cd oscar-2.1.1/packages/c3/RPMS
# rpm -Uvh python2-compat-1.0-1.noarch.rpm
```

2.5 IA64 및 기타 시스템 주의 사항

• OSCAR auto-install diskette은 Itanium 시스템에서 제대로 동작하지 않을 것이다. 하지만, network booting은 대부분 제대로 동작할 것이다.

• Itanium 1, 2 시스템 중 일부는 Itanium용 RedHat Linux 7.2 배포판을 설치했을때, OSCAR 설치가 가능하다. booting과 network access를 위해 standard RedHat Itanium kernel를 사용할 수 있는 Itanium 시스템은 OSCAR가 제대로 동작할 것이다.

• Itanium 2 시스템은 Linux 공식 배포판에서 제공되는 것보다 새로운 kernel을 사용해야 제대로 동작하는 (network와 disk drive를 위한) 새로운 하드웨어 인터페이스를 갖고 있는 경우가 종종 있다. Vendor들은 일반적으로 이 새로운 하드웨어 인터페이스에 대한 드라이버를 디스크 혹은 on-line을 통해 제공할 것이다. 사용자들은 때때로 스스로 커널을 컴파일해서, OSCAR에서 제공되는 kernel_picker를 사용해서 Image에 올려야 하며, 이것은 종종 "do it yourself" 종류의 일일 것이다. 따라서 경험이 부족한 사용자에게는 추천할 만한 일이 못된다. 보다 많은 정보를 위해서 OSCAR 사용자의 mailing list를 사용할 것을 권한다.

3 System Installation Suite(SIS) 개요

SIS(Sytem Installation Suite): SIS는 OSCAR에 사용하는 cluster installation의 핵심 tool로서 System Installer, System Configurator, 그리고 SystemImager로 구성되어 있으며, IBM Linux Technology Center와 SystemImager team이 공동으로 개발했다. SIS는 다음과 같은 이유로 OSCAR의 cluster installation tool로서 채택했다.

- SIS는 개발 환경에서 잘 작동하며, 우수한 3rd party open source이다.
- SIS를 사용하면 client node에 사전에 리눅스를 설치할 필요가 없다.
- SIS는 클러스터내의 각 노드에 대한 설치 및 설정 정보를 포함하는 database를 관리 유지한다.
- SIS는 소프트웨어 설치의 표준으로 RPM을 사용한다.
- SIS는 heterogeneous hardware 및 software를 지원한다.(이 기능은 아직 OSCAR내에 서는 사용되지 않는다)

앞서 용어설명에서 나온 Image와 client definition은 SIS에서 정의된 개념이며, SIS의 특징을 다음과 같다.

- 많은 다양한 image들이 image server에 존재할 수 있다.
- Image를 기존 machine의 파일시스템으로부터 capture할 수 있다.
- Image는 직접 Image server에서 package set으로부터 제작할 수 있다.
- Image를 다른 machine에 propagate하기 위해 rsync를 사용하며, 따라서 차후 file level의 update 또한 용이하다.

Section 5에서 SIS를 이용한 Image update방법을 간단히 설명한다.

4 세부적인 클러스터 설치 과정

4.1 Server Installation and Configuration

(참고사항) 이후의 모든 설명은 Redhat 7.3을 기준으로 설명한다.
Redhat 7.3을 "Workstation" class로 설치한다. 단 설치시 "software development tools" package를 추가해준다.

4.1.1 디스크 공간 할당

/tftpboot/rpm: 여기에 최소 2GB의 여분의 공간의 있어야 한다. 이것은 "/"와 같은 드라이브에 있어야 하며, 여기에 Redhat CD의 모든 rpm을 저장할 것이다.

/var/lib/systemimager :여기에 최소 2GB의 여분이 있어야 한다. 클라이언트의 image file들이 여기에 위치한다.

4.1.2 소스 다운로드

oscar-2.2.1.tar.gz를 http://oscar.sourceforge.net에서 /root에 다운로드한 후 압축을 푼다. 일반적인 install은 작업은 기본적으로 /root/oscar-2.2.1/에서 하도록 권장하고 있다.

```
[root@cluster root]# tar xvzf oscar-2.2.1.tar.gz
[root@cluster root]# cd oscar-2.2.1
```

소스 패키지는 다음과 같이 구별된다.
regular: 모든 installation material을 포함한다.
extra crispy: regular + SRPMS, 패키지 이름이 oscar-including -srpms-2.2.1.tar.gz같이 알기 쉽게 주어져 있다. secret sauce: SRPM만을 포함한다. 패키지 이름은 oscar-only-srpms-2.2.1.tar.gz과 같이 주어져 있다.

4.1.3 클러스터 구축을 위한 서버의 기본 설정

- hostname을 localhost가 아닌 다른 이름으로 준다.
- cluster내의 private network용으로 이더넷 카드를 하나 설정한다. 즉 적당한 private ip address와 netmask를 설정하고, 부팅시 activate 되도록 하며, interface control을 "none"으로 세팅한후 activate 시킨다.
- public network을 위한 이더넷 카드는 필수 사항은 아니나, 가지고 있다면 미리 세팅해 둘 것을 권장한다.

4.1.4 RedHat CD의 RPM file 복사

/tftpboot/rpm을 만든 후, Redhat 배포판 CD 1, 2, 3의 RPMS을 /tftpboot/rpm 에 복사한다. 각각의 Redhat CD에 대해, mount한 후 아래의 과정을 반복한다.

```
# cp /mnt/cdrom/RedHat/RPMS/*.rpm /tftpboot/rpm
```

4.2 (선택사항) 추가적인 oscar packages 다운로드

OPD(OSCAR Package Downloader)를 사용하여 추가로 필요한 OSCAR packages를 다운로드할 수 있으며, 이때 다운로드한 패키지들을 OSCAR installation hierarchy 즉, /root/oscar-2.2.1/packages/ 내에 저장한다.

OPD를 사용하고자 할 때, 필요한 Perl modules를 요구할 수 있으며, 이는 CPAN을 이용해서 다운로드하거나 아니면, OSCAR Installer를 실행시켜서(다음 section 참조) GUI Window가 보일 때 닫으면 자동으로 다운로드가 진행된다.

OSCAR의 package의 RPM들 중 일부는 OSCAR용으로 패치되어 있으며, 예를 들면 mpich의 경우 mpich-oscar-1.2.4-7.i586.rpm 식으로 이름을 통해 구별이 가능하다. 이러한 naming은 OS에 의한 auto-update를 막는데 목적이 있다.

(주의사항) 일반적으로 OPD를 통해 OSCAR 패키지 형태로 다운로드 받은 RPM은 설치시 아무런 문제가 없다. 예를 들면 monitoring 도구인 Ganglia를 OPD로부터 다운로드(ganglia-oscar-2.1-1.tar.gz) 한 후 압축을 풀어 /root/oscar-2.1.1/packages/에 넣으면 SIS가 정상적으로 인식하여 서버와 클라이언트 노드에 설치시킨다. 그 외의 일반 RPM은 이러한 설치를 보장하지 못한다. 참고로 OSCAR-specific package의 하나의 예로 pbs package를 보면 다음의 구성으로 이루어져 있다.

```
[root@cluster packages]# ls -R pbs
pbs:
config.xml  doc  RPMS  scripts  testing

pbs/doc:
install.tex  license.tex  user.tex

pbs/RPMS:
maui-oscar-3.0.7p8-5.i586.rpm
openpbs-oscar-gui-2.3.16-7.i386.rpm
openpbs-oscar-2.3.16-7.i386.rpm
openpbs-oscar-mom-2.3.16-7.i386.rpm
openpbs-oscar-client-2.3.16-7.i386.rpm
openpbs-oscar-server-2.3.16-7.i386.rpm

pbs/scripts:
post_clients  post_install  update_maui_config
update_mom_config
```

```
pbs/testing:
pbs_script.shell   test_root   test_user
```

4.3 OSCAR installer 실행

다음과 같이 OSCAR installer를 실행시킨다. 뒤의 <device>에는 여러분이 설정한 private 네트웍의 이더넷 어댑터(e.g., eth1)를 적어준다.

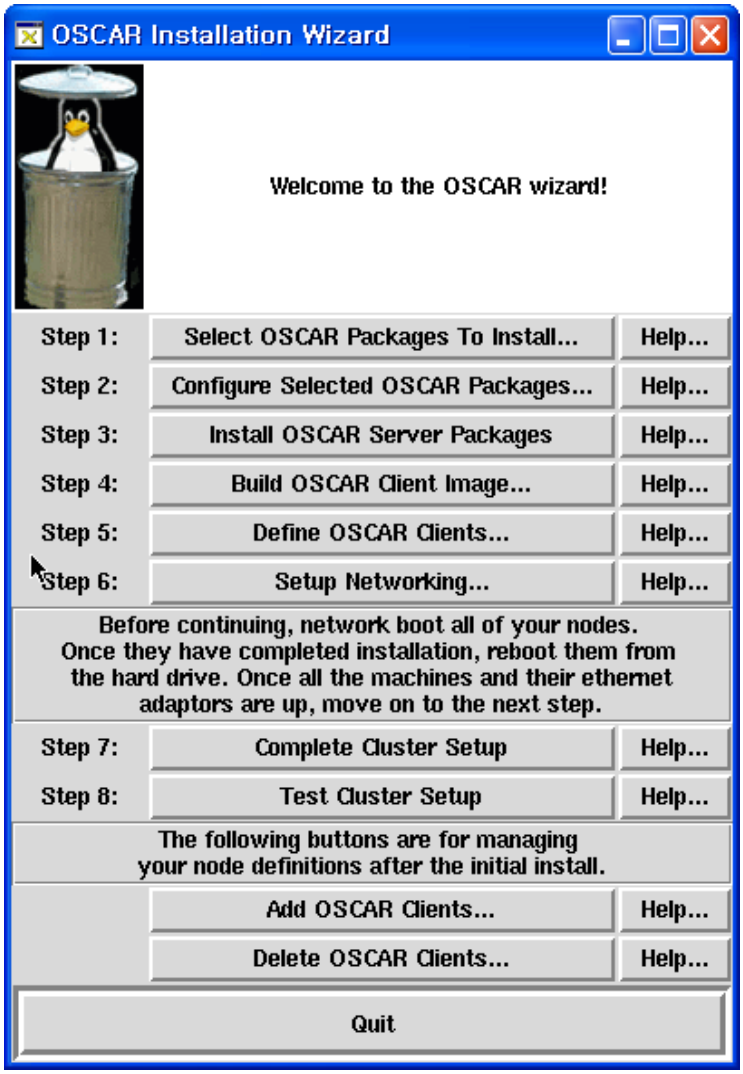
```
cd /root/oscar-2.2.1
./install_cluster <device>
```

이 script는 다음의 과정을 수행한다.

```
1. prerequisite packages를 설치한다. (OPD를 위한 perl
module등)
2. OSCAR package directory의 RPM을  /tftpboot/rpm 에
복사한다.
3. OSCAR server-core RPMS(sis, switcher, c3, oda)을
설치한다.
4. system startup ( /etc/rc.d/init.d) scripts를 update한다.
5. 연관된 services를 restart시킨다.
```

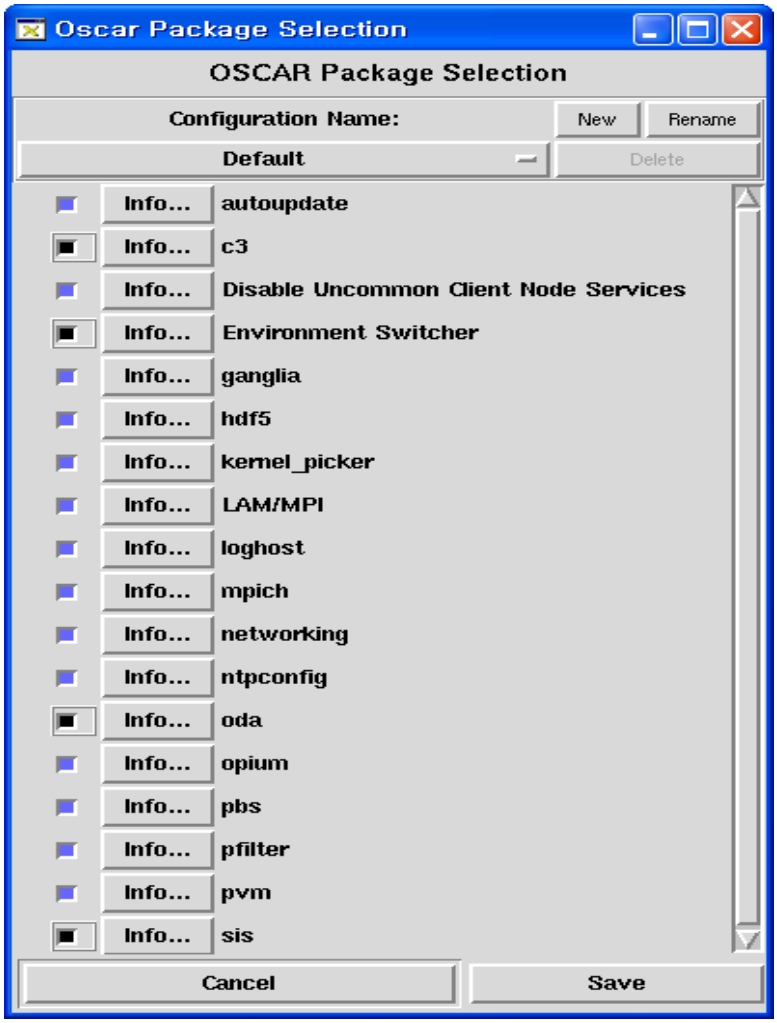
console상에서 위의 과정들의 진행상황이 보여지며, 나중에 oscarinstall.log를 통해 확인해 볼 수 있다.

위의 과정이 성공적으로 끝나면 다음과 같은 OSCAR INSTALLATION WIZARD GUI가 나타난다. 이제부터는 대부분의 install과정이 이 Window상에서 이루어진다.

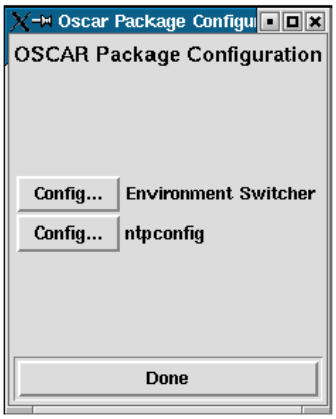


4.4 (선택사항) "Select packages to install"(step 1)

이 과정을 생략하면 모든 OSCAR용 package, 즉 /root/oscar-2.2.1 /packages/내의 모든 package가 설치된다.



4.5 "Configure selected OSCAR Packages"(step 2)



Environment Switcher: 이 패널을 통해 cluster의 LAM/MPI 와 MPICH 중 default MPI를 결정할 수 있다. 이것은 나중에 switcher script를 사용하여 user level과 system level에서 각각 바꿀 수 있다.

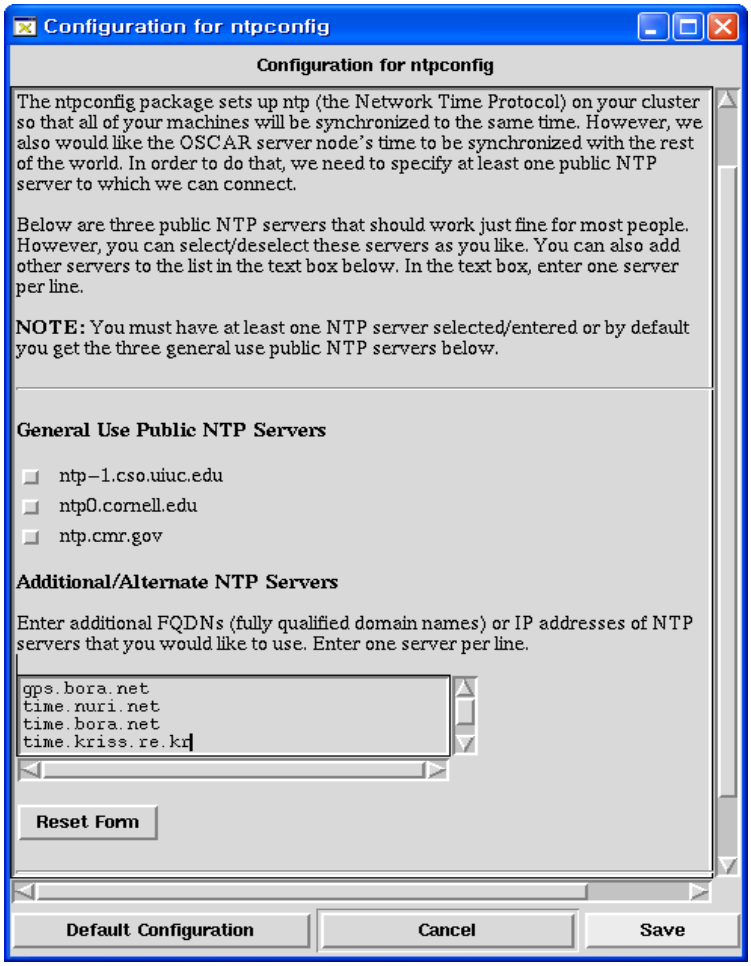
ntpconfig: time sync을 위한 public NTP 서버를 결정한다. 이는 나중에 /etc/ntp/step-tickers 와 /etc/ntp.conf를 통해 수정할 수 있다.

이번 test에서는 다음의 NTP 서버를 사용했다.

```
server gps.bora.net
server time.nuri.net
server time.bora.net
server time.kriss.re.kr
```

(주의사항) 중요한 것은 클러스터 내부에서 time이 sync되는 것이다. 따라서 서버노드를 time sever로 사용할 수도 있다. 이를 위해서는 다음과 같이 설정하면 된다.

```
server 127.127.1.1
fudge  127.127.1.1 stratum 10
```



4.6 "Install OSCAR server packages"(step 3)

이 과정에서는 scripts/install_server가 실행되며 다음의 일들이 진행 된다

- 서버에 included package를 install한다.
- 다음 파일을 수정하고, rsyncd, dhcpd, portmap의 run-level을 수정하며 nfs, ssh 서비스를 시작 시킨다

/etc/hosts

/etc/profile

/etc/exports

/root/.bashrc

이 과정이 끝나면 성공 혹은 실패를 알리는 pop-up 윈도우가 나타난다.

4.7 "Build OSCAR client Image"(step 4)

만일 기존에 설치된 RedHat Linux를 그대로 사용하고자 할 경우, 서버의 다음 설정을 확인한 후 이 작업을 시작하는 것이 좋다.

- /etc/ssh/sshd.config 에서 PermitRootLogin이 "yes"이어야 한다.
- /etc/hosts.allow, /etc/hosts.deny에서 모든 private subnet의 traffic을 허용하도록 setting 해야 한다.
- firewall software가 private subnet의 traffic을 제한하고 있는지 확인한다.

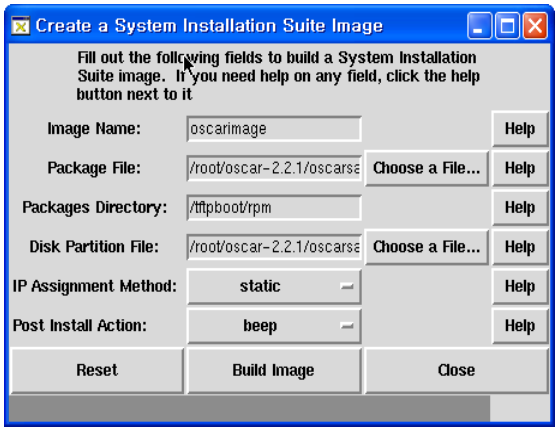


Image Name: Image의 이름을 결정한다. 초기값을 그대로 사용하면 /var/lib/systemimager/images/oscarimage 디렉토리가 만들어 지고 여기에 image가 위치하게 된다.

Package File: sample file(/root/oscar-2.2.1/oscarsamples/RedHat-7.3-i386.rpmlist)을 편집하여 install할 rpm을 결정할 수 있다.

Package Directory: RedHat CD로부터 복사한 rpm files이 있는 directory를 지정한다.

Disk Partition File: sample file(/root/oscar-2.2.1/oscarsamples/samples.disk.ide)를 이용하여 client nodes의 partitioning을 수정할 수 있다. sample file의 내용은 아래와 같다.

```
<partition>  < size in MB>  <type>     <mount point> <options>
/dev/hda1          24         ext2          /boot       defaults
/dev/hda5         128         swap
/dev/hda6           *          ext2          /           defaults
nfs_oscar:/home     -          nfs           /home       rw
```

size column의 * 표시는 partition이 남은 전체의 disk를 차지하도록 허용한다. nfs에서 사용할 서버의 hostname이 위에서 nfs_oscar로 주어져 있는데, 이는 /etc/hosts 파일에 alias 설정이 되어 있으므로 default 값을 그대로 사용해도 무방하다.

IP Assignment Method: 특수한 상황이 아니면, 당연히 static을 선택한다.

Post Install Action: 각각의 client node에서 install이 완료된 후, client node들이 취할 행동을 다음의 halt, reboot, beep 중에서 결정한다. “beep"을 권장한다. "reboot"로 설정하면 문제가 발생할 소지가 있다. 즉 client node들을 network boot을 통해 설치하고 나서, 자동으로 reboot하도록 설정하면, 리부팅한 후 다시 자동으로 network boot를 통해 동일한 설치 과정을 반복할 소지가 높다. 따라서 “beep" 이나 ”halt"로 설정하여 수동으로 reboot시킨 후,

BIOS에서 boot order를 바꾸어 주는 것이 좋다. beep으로 설정하면 수동으로 reboot하라는 메시지가 나오면서 beep 소리를 주기적으로 낸다.

(참고사항) customized kernel를 사용하고자 한다면, "Install OSCAR server packages"(step 3)와 "Build OSCAR client image"(step 4)사이에 앞서 설명한 kernel picker를 사용하여 바꿀 수 있다.(7.4 kernel_picker 패키지 설명 참조)

위의 세팅을 완료한 후, "Build Image" 버튼을 누르면 시스템 사양과 선택한 package 수에 따라 약 5-15 분 정도의 시간이 소요된다.

(참고사항) 위의 GUI에서 이루어지는 procedure는 SIS 명령어를 사용하여 shell prompt에서 다음을 수행하는 것과 동일한 결과를 준다.

```
# mksiimage -Add --name oscarimage \
--filename  \
/root/oscar-2.2.1/oscarsamples/RedHat-7.3-i386.rpmlist  \
--location /tftpboot/rpm/
```

위 명령어는 packages로부터 client image를 만든다.

```
#mksidisk -A --name oscarimage --type ide --file \
/root/oscar-2.2.1/oscarsamples/sample.disk.ide
```

위 명령어는 위에서 만든 image를 사용할 client의 partition scheme를 결정한다.

```
#mkautoinstallscript -image oscarimage \
-ip-assignment static -post-install
```

위 명령어는 client가 위에서 만든 image를 install할 때 필요한 master install script를 만든다.

4.8 "Define OSCAR Clients"(step 5)

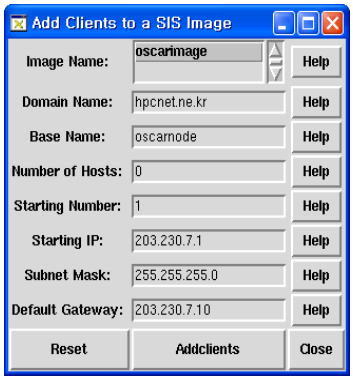


Image Name: 사용할 client image를 결정한다.
Domain Name: cluster의 private network을 위한 domain name을 설정한다.
Base Name: client의 hostname의 처음부분을 결정한다(아래 예 참조)
Number of Hosts: client의 수를 설정한다.
Starting Number: 첫 번째 client의 이름을 결정하기 위해 Base name에 덧붙일 숫자를 결정한다. 이후 두 번째 이후의 클라이언트의 hostname에는 이 Starting number가 하나씩 증가한 것이 붙게 된다.(예 참조)

```
(예)
아래와 같이 Base name과 Starting number, Domain name을
주었을 때,
(Base_name)(Starting_number).(Domain_name)
    node              1                kisti.re.kr

   -> node1.kisti.re.kr: 첫 번째 clinet
      node2.kisti.re.kr: 두 번째 client
      .....
```

Starting IP: 첫 번째 client의 IP address를 결정한다. 이후 두 번째 이후의 client는 IP address중 host id가 하나씩 증가하게 된다.
Subnet Mask: IP netmask를 설정한다.
Default gateway : 모든 client를 위한 default route를 설정한다.

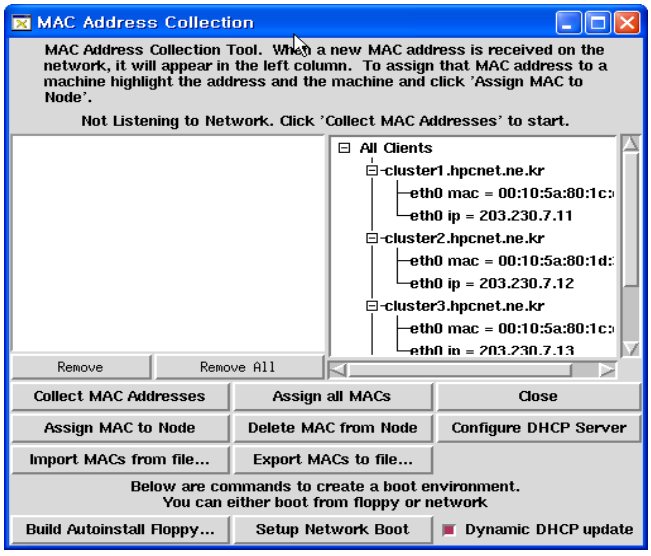
위의 setting이 끝나면 Addclients 버튼을 눌러 작업을 완료한다.

(주의사항) 254 client(즉, C class) 이상의 노드를 설정하고자 한다면 , 위의 셋업 과정을 여러 번 반복해야 하며, IP address와 subnet mask의 설정을 주의해서 해주어야 한다.

(참고사항) 위의 GUI에서 이루어지는 procedure는 shell에서 다음의 SIS명령어를 사용하는 것과 같은 결과를 준다. 이 과정을 통해 각 클라이언트 노드에 대한 설치 스크립트 즉, <nodename.sh>가 /var/lib/systemimager/scripts/에 만들어지며, 서버의 /etc/hosts파일도 위의 client정보를 포함하도록 업데이트된다.

```
# mksirange -b cluster -i 203.230.7.11 -s 11 \
-c 4 -g 203.230.7.10 -m 255.255.255.0 -p 3 \
-d hpcnet.ne.kr -n oscarimage
```

4.9 "Setup Networking"(step 6)



이 과정에서 Client의 MAC address와 (바로 전 단계에 설정한) network setting을 match 시키면, 자동으로 모든 client에 partitioning 및 서버에서 image를 copy 하여 설치하는 과정을 수행하며, "Build OSCAR client Image"(step 4)에서 Post Installation Action에서 정해준대로 client는 행동하게 된다. 모든 노드에 대한 설치작업이 끝나고, 정상적으로 리부팅이 되면, 서버노드에서 다음 단계(step 7. complete the cluster setup)를 진행하면 된다.

(주의사항) PXE(Preboot eXecution Environment) booting을 지원하는 NIC은 특별한 준비사항이 없으며, PXE를 지원하지 않는 NIC을 가지고 있을 때는 network booting을 위해서 각각의 client마다 3.5" FDD를 가지고 있어야 한다. 만일 floppy로 network booting을 하고자 한다면 "Build Autoinstall Floppy"를 사용하여 SystemImager boot floppy를 만든 후, 이것을 install하고자 하는 client의 FDD에 넣으면 PXE booting처럼 network booting을 할 수 있게 된다.

"Collect MAC Address" 버튼을 누른 상태에서 install하고자 하는 client를 켜면, 이 client의 MAC address가 왼쪽의 window에 나타난다. 이때 하나의 노드를 오른쪽 window에서 선택한 후 "Assign MAC to Node"를 이용하여 왼쪽의 Mac address를 연결시키면, 이 client에 install이 시작된다. 이 과정은 해당 client의 console를 통해 확인가능하며, RARP 요청부터, file copy까지 모든 정보가 출력된다.

(주의사항) 만일 PXE를 이용한 network booting을 사용한다면 client의 설치 과정이 끝난 후 rebooting시 BIOS에서 boot order를 local hard drive가 먼저 오도록 바꾸어 주어야 한다. floppy를 이용한 network booting을 사용한다면, client에 install이 끝난 후, rebooting 시, 반드시 floppy disk를 제거해 주어야 한다. 그렇지 않으면, 위의 install과정을 계속해서 반복하여 수행하게 된다.

(참고사항) 위의 Mac address와 IP address의 mapping 과정은 노드수가 많을 때는 번거로운 작업이 될 수 있다. 만일 모든 노드의 Mac address를 알고 있다면, "Import MACs from file"를 사용하여 MAC과 노드의 IP address와의 mapping을 좀더 쉽게 할 수 있다. 또 다른 방법은 /etc/dhcpd.conf 파일에 직접 모든 노드의 IP address등의 네트웍 관련 정보 및 Mac address를 직접 입력한 후, dhcpd 서비스를 다시 시작하는 방법이 있다.(tftp 서비스가 xinetd내에서 실행중인지도 확인해야 한다)

(참고사항) Etherboot 및 기타 다른 network boot에 대해서는 test를 하지 못했으며, 공식적으로 OSCAR 에서는 위의 PXE와 SystemImager boot floppy(즉, "Build Autoinstall Floppy"을 통해 만든 floppy) 만을 지원한다.

(참고사항) 위의 MAC address mapping은 서버의 terminal에서 mkdhcpserver 혹은 mkdhcpstatic 명령어를 사용하여 작업할 수 있으며 /etc/dhcpd.conf에서 확인 할 수 있다.

(참고사항) 노드하나에 설치가 끝나는데 대략 15분 정도의 시간이 소요 되었다.

4.10 (참고 사항) Client Installations

다음은 Client intallation 과정으로, PXE를 이용한 network boot를 할 때이다. SystmeImager boot floppy를 사용할 때에는 floppy로부터 install kernel이 먼저 load된 후 DHCP request를 한다는 점만이 다르다.

0. BOOTP/DHCP를 사용하여 서버로부터 IP address를 얻는다.
1. TFTP를 이용하여 install kernel를 load한다.
2. /var/lib/systemimager/scripts/에서 이름이 <nodename>.sh인 auto-install script를 찾으며, 이 script는 client에 대해 다음의 일을 수행한다.
3. <imagedir>/etc/systemimager/partionschemes을 참조하여 disk partitioning을 한다.
4. 새로 만들어진 파티션을 /a로 마운트한다.
5. /a 에 chroot를 한후, rsync를 사용하여 image에 있는 모든 file을 복사 해온다.
6. systemconfigurator를 불러와 client의 hardware(NIC, Storage device)를 check 및 setup 하며, 여러 가지 configuration(time zone, initrd, network scripts, boot loader, modules.conf등)을 복사해온 image file에 대해 수행하여 수정한다.
7. /a를 unmount한다.

4.11 "Complete the Cluster Setup"(step 7)

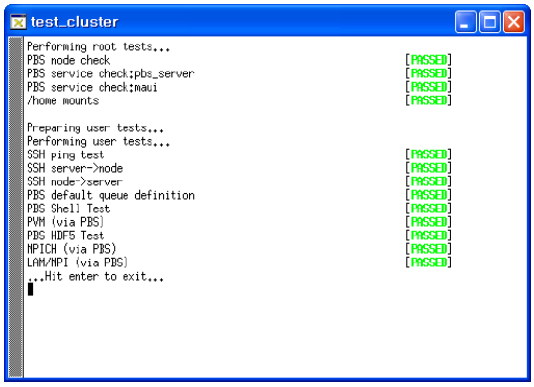
(주의사항) 이 과정을 진행하기 전에 모든 client 노드가 install를 마지고 정상적으로 reboot를 완료한 상태인지 확인한다.

"Complete the Cluster Setup"버튼을 누르면, OSCAR software package들에 들어 있는 post-install script를 수행하고, 아래와 같이 syslogd, pbs queue등을 비롯한 서버와 client의 configuration를 완료하고 필요한 서비스를 다시 시작시킨다.

• makewhatis, slocate, kudzu 서비스를 클라이언트 노드에서 중지시킨다.
• C3에서 사용할 nodes information을 설정한다.
• MPICH의 machines.LINUX를 설정한다.
• opium을 통해 user 정보를 주기적으로 sync하도록 설정한다.
• pbs의 nodes 정보와 기본적인 queue 정보를 설정한다.
• syslogd를 통해 클라이언트 노드의 log 정보를 서버에서 모을 수 있도록 설정한다.
• pfilter를 통해 각 service 및 port에 대한 클러스터 보안 설정을 한다.

4.12 "Test Cluster Setup"(step 8)

OSCAR에서 제공하는 간단한 test suite가 중요한 cluster components(OpenSSH, PBS, MPI, PVM, etc.)들이 제대로 동작하는지 여부를 test한다. test중 일부라도 error가 발생하면, 설치과정 상에 문제가 있음을 나타낸다. 다음은 test 예이다.



test 실패시, 갑자기 위의 test window가 사라질 수도 있다. 이런 상황이 발생하면, testing/test_cluster 스크립트를 shell에서 실행하여 문제를 분석해야 한다.

4.13 Congratulations!

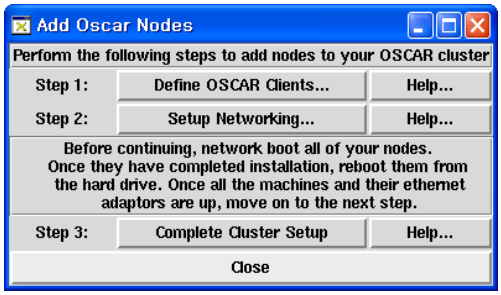
이제 OSCAR의 cluster setup이 완료되었다.

4.14 Adding and Deleting client nodes

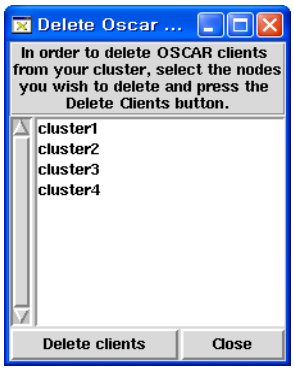
Add OSCAR Clients: OSCAR installation wizard를 아래와 같이 다시 시작한다.

/root/oscar-2.2.1/install_cluster <device>

"Add OSCAR Clients" 버튼을 눌러서, Define OSCAR Clients(위의 step 5와 동일)에서 number of hosts, starting number, starting IP 등을 결정한후, Setup Networking(위의 step 6과 동일)과정을 거치면 새로운 client를 추가 할 수 있다.



Delete OSCAR Clients: “Delete OSCAR Clients” 버튼을 눌러서, 지우고자 하는 노드를 선택한 후, “Delete clients” 버튼을 이용하여 지울 수 있다.



(주의사항) “Add/Delete Oscar Clients" 기능은 현재의 OSCAR 2.2.1 version에서 완벽히 test되지 못했다. 특히 새로운 client를 추가할때, 만일 SystemImager boot floppy를 사용하여 network boot를 하고자 한다면, 반드시 새로운 boot floppy를 만들어서 사용할 것을 권한다.

(주의사항) OSCAR installation wizard는 한 session에서 노드를 지운 후, 다시 지운 노드를 추가하는 것을 지원하지 않는다. 지운 노드를 다시 추가하고자 한다면 반드시 OSCAR installation wizard를 다시 시작한 후 해야 한다.

4.15 Starting over

만일 OSCAR installation 과정을 다시 시작하고 싶다면, 먼저 /root/oscar-2.2.1/scripts/에 위치한 start_over 스크립트를 실행시킬 것을 권한다. 이것은 서버가 OSCAR가 재 설치과정을 하기에 적당한 상태가 되도록 해준다. 또한 재설치하기 전에 완전히 logout을 한후 다시 login 하여 설치 작업을 수행할 것을 권장한다. 이는 예전 install의해 변경된 환경변수 등으로부터 영향 받지 않게 하기위해서이다.

(참고사항) start_over 스크립트는 uninstaller가 아니다. 예로, RedHat 7.x series는 서버 이상의 class로 설치할 때, 기본적으로 LAM/MPI RPM을 설치하지만, OSCAR는 installation 과정 중에 이러한 RedHat-default RPM을 제거한 후, OSCAR용 LAM/MPI RPM을 설치한다. 이 상태에서 start_over 스크립트를 실행하면, OSCAR용 LAM/MPI RPM을 제거하지만, RedHat-default RPM을 재 설치하지는 않는다.

5 클라이언트 시스템 업데이트/업그레이드

클라이언트 노드 전체에 새로운 hardware driver를 설치해야 하거나, 혹은 새로운 compiler를 install하고자 할 때, 혹은 kernel patch/upgrade를 해야 할 때 OSCAR에서는 sis의 SystemImager를 사용하여 Image를 update 함으로서 쉽게 위의 작업을 할 수 있다. Image를 update하는 방법에는 크게 두 가지가 있다.

(방법 1)
서버에서 chroot로 mirror directory (/var/lib/systemimager/images/<image_name>)에 들어가서, 마치 현재 사용 중인 machine인 것처럼, rpm 및 기타 필요한 compile 및 설치 작업을 이 image에 대하여 할 수 있다.

(방법 2)
하나의 client를 택해, 직접 console 상에서 필요한 작업을 한 후, 아래의 예에서 보듯이 getimage명령어를 사용하여 서버에 위치한 Image를 update할 수 있다.
Image를 update하기 위한 절차는 다음과 같다.

client(e.g., cluster1)에서 적절한 작업을 한 후 다음의 명령을 내린다. 이 명령은 rsyncd 대몬을 시작시킨다.

```
[root@cluster1 root]# prepareclient
```

서버(e.g., cluster)에서 다음의 명령어를 내려 Image update를 시작한다.

```
[root@cluster root]# getimage -golden-client \
cluster1 -image <image_name>
```

여기에서 <image_name>은 새로운 Image로 만들고 싶으면, 원하는 새로운 이름을 주면 된다. 그러면 client의 root file system 전체가 새로운 Image로 만들어져 복사된다.

만일 기존에 만들어진 Image를 update 하고자 한다면 <image_name>에 예전의 Image 이름을 적어주면 된다. 새로 수정된 file에 대해서만 rsync를 사용하여 copy가 이루어지므로 시간이 훨씬 단축된다. 새로 만든 image를 SIS database에 추가하기 위해서는 다음의 명령어를 사용하면 된다.

```
[root@cluster root]# mksiimage --Add  --nobuild \
 --name New_Image_Name
```

만일 새로 만든 이 image를 이용하여 각 계산노드를 재설치하고자 한다면 client definition을 다음과 같이 재정의해야 한다.

```
[root@cluster root]# mksimachine -U --all \
--image New_Image_Name
```

이 명령은 각 계산노드의 install script도 update시킨다.

만일 하나의 계산노드에 대해서만 재설치를 하고자 할 때는 --all 옵션 대신 --name Node_Name을 사용하면 된다.

(공통)
위의 (방법1), (방법2), 두 가지 방법에 의해 새로이 update 된 Image를 모든 client에 대해 적용하기위해서는 각 client(e.g., cluster2)에서 다음의 명령어를 수행하면 된다. -directory 옵션을 사용하여 정해진 디렉토리에 대해서만 업데이트 할 수도 있다.

```
[root@cluster2 root]# updateclient -server cluster \
-image <image_name>
```

위의 과정 역시 수정된 file에 대해서만 copy가 이루어지므로, 시간이 오래 걸리지 않는다. 단 update시 /etc/systemimager/updateclient.local.exclude 에 명시된 파일들은 update에서 제외된다. 여기에는 /home/*, /etc/HOSTNAME, /etc/sysconfig/network-scripts/ifcfg-*등이 포함된다. ssh를 이용하는 간단한 script를 작성하여 위의 명령어를 모든 client에 대해 실행시킬 수 있다. 혹은 c3 package내에 들어 있는 cexec(s) command를 사용하여 모든 클라이언트 노드 (e.g, cluster1, 2, 3, 4)에 대한 update를 one-liner 명령어로 할 수 있다.

```
[root@cluster root]#cexecs :1-4 updateclient -server \
cluster -image <image_name>
```

혹은 더 간단히 다음의 c3 명령어로 할 수도 있다. 이 명령어는 위의 updateclient 명령어를 클러스터내의 모든 클라이언트에 대해 실행한다.

```
[root@cluster root]#cpushimage <image_name>
```

c3의 명령어에 대한 매뉴얼은 다음의 web page에서 찾을 수 있다.
http://www.csm.ornl.gov/torc/C3/C3documentation.shtml

SIS의 다양하고 풍부한 기능에 대해서는 다음 web page를 참조하기 바란다.
http://www.sisuite.org/

(주의사항) 만일 각 계산노드를 처음부터 재설치하고자 한다면 서버노드에서 다음 대몬이 실행중인지 확인한 후, 각 계산노드가 PXE를 이용한 네트웍 부팅을 하도록 해야 한다.

- systemimager
- xineted(tftp)
- dhcpd

6 각 패키지별 설치 관련 주의사항

6.1 C3

ORNL(Oak Ridge National Laboratory)에서 개발한 C3는 PC cluster의 시스템 관리를 돕기 위한 user interface로 개발되었다. C3는 command line interface이며, 또한 다른 프로그램 내에서도 호출하여 실행 시킬 수 있다. 현재의 OSCAR package에는 3.1.2-1 버전이 들어 있으며, multi-thread로 동작하며, 큰 규모의 cluster에 대해서도 scalability를 갖도록 설계되었다. c3의 명령어는 일반적으로 서버노드에서 실행하며, 특정한 노드를 지정하지 않을 경우(위에서 :1-4 와 같이) 모든 클라이언트 노드들에 대한 명령(single-system illusion)으로 실행된다.

cexec - c3의 가장 일반적인 utility로서 모든 클라이언트 노드에 대하여 어떤 종류의 셸 명령어도 실행할 수 있다.
cget - 모든 클라이언트 노드들로부터 동일한 파일 혹은 디렉토리를 서버노드로 복사할 때 사용한다. 파일이름 뒤에 노드이름이 붙어서 구별이 편리하다.
ckill - 모든 클라이언트 노드에서 실행중인 특정한 process를 죽이고자 할 때 실행된다.
cpush - 파일 혹은 디렉토리를 모든 클라이언트 노드에 복사할 때 사용한다.
cpushimage - 앞의 예에서 보듯이 client에 image를 update할 때 사용된다.
crm - 모든 클라이언트 노드에 존재하는 파일 혹은 디렉토리를 지우고자 할 때 사용한다.
cshutdown - 모든 클라이언트 노드를 shutdown, 혹은 reboot 시킬때 사용한다.

다음은 간단한 예이다.

이 패키지에 대한 자세한 내용은 다음의 web-site를 참조하기 바란다. http://www.csm.ornl.gov/torc/C3/

6.2 disable-services

disable-services OSCAR 패키지는 client에 대해 다음의 service를 disable 시킨다.

• incoming mail service: sendmail, exim, postfix 대몬들을 disable 시킨다. 이것은 클라이언트 노드로 들어오는 메일을 차단시키나, 클라이언트 노드로부터 밖으로 나가는(outgoing) 메일에 대해서는 차단을 하지 않는다. 대부분 메일은 즉각적으로 보내지 나 그렇지 않을 경우, mail service queue에 보내진 후, crontab의 cron.hourly에 의해 매 시간 queue된 mail을 재전송한다.

• Kudzu: 매 부팅시마다, Kudzu service는 새로운 하드웨어를 검색하는데 많은 시간을 소비한다. 개개의 노드의 부팅시간을 단축하고자 이것을 disable 시킨다.

• slocate: slocate service는 모든 local file system에 대해 주기적으로(Distribution에 따라 매일, 혹은 매주) top-level의 find 명령어를 실행하여 모든 file name을 인덱싱 한다. 이것은 locate 명령어를 사용한 빠른 lookup을 가능하게 하지만, top-level의 find 명령어는 상당한 system resource를 소모하므로, disable 시킨다.

• makewhatis: makewhatis 명령어는 crontab에 의해 실행되며, Linux Distribution에 따라 매일 혹은 매주, 매뉴얼 페이지 인덱스를 갱신한다. slocate와 마찬가지로 이 명령어도 상당한 system resource를 소모하므로, disable 시킨다.

위의 서비스들은 uninstall 시킨 것이 아니라, 단지 disable 시켰음을 주의하기 바란다. 관리자는 언제든 위의 서비스를 enable시킬 수 있다.

6.3 HDF5

HDF5는 data format specification과 지원 library로 이루어진 Hieracrchical Data Format product이다. HDF5는 scientific data format에 널리 이용될 수 있는 다음의 특징을 갖는다.

• reserch, development 그리고 production 환경에서 크고, 변형된(varied) data들을 조작하기위해, 다양한 datatype과 속성(attribute)을 사용하여 user-defined hierarchical grouping structure를 지원한다.

• MPI I/O를 사용하는 parallel computing platform에서 실행된다.

• compound, user-defined datatype을 포함하여, 모든 encompassing datatype을 지원한다.

• 매우 큰 파일(한계치를 테스트해보지는 않았지만, 이론적으로 1 terabyte 이상)을 지원한다.

HDF5의 문서는 /opt/hdf5-oscar-1.4.4-post2/doc/hdf5/index.html와 web-site http://hdf.ncsa.uiuc.edu/HDF5/doc/ 에서 찾을 수 있다.

OSCAR는 HDF5 library를 /opt/hdf5-oscar-1.4.4-post2/doc/hdf5/lib/에 설치하며, 현재의 OSCAR 버전에는 serial operation을 위한 library만이 들어 있다. parallel operation을 위한 library는 추후 OSCAR 버전에 추가될 예정이다.

HDF5를 parallel mode로 컴파일하여 install할 때에는 다음의 두 가지 주의해야 할 사항이 있다.

• gcc-2.96 컴파일러에는 HDF5에 영향을 주는 버그가 있다. 제대로 동작하게 하기 위해

서는, HDF5와 MPI 모두 2.96이 아닌 gcc로 컴파일해야 한다. HDF5 group은 이를 위해 2.95.3 버전의 gcc를 추천한다.

- kernel 2.4이상의 리눅스에서 MPI와 함께 HDF5를 parallel mode로 컴파일할때, 반드시 2GB이상의 파일크기를 지원하도록 설정해야 한다. 다음은 MPICH의 install시 configuration command-line option으로 이를 준 예 이다.

-cflags="-D_LAGEFILE_SOURCE -D_LAGEFILE64_SOURCE \
-D_FILE_OFFSET_BITS=64"

6.4 kernel_picker

kernel_picker는 OSCAR image에 default로 설치된 kernel대신 다른 kernel을 사용하고자 할 때, 쓸 수 있는 Perl script이다. OSCAR installation Wizard의 Select packages to install"(step 1)와 "Configure selected OSCAR Packages"(step 2)의 과정 중간에 다른 콘솔을 열고 kernel_picker command-line 명령어를 사용할 수 있다. 아무런 옵션 없이 이 명령어를 사용하면 interactive mode에서 동작하며, 사용할 image 이름부터, kernel의 전체 경로 등을 차례대로 입력하면 된다. kernel_picker 프로그램은 모든 OSCAR image의 위치가 /var/lib/systemimager/images내에 있다고 가정하며, default OSCAR image의 이름은 oscarimage 이다.

kernel_picker는 /opt/kernel_picker/bin 디렉토리에 설치된다. HTML, Postscript, PDF, plain text와 manpage format등의 문서가 제공된다. manpage를 보려면 다음을 참조하기 바란다.

```
$ man /opt/kernel_picker/man/man1/kernel_picker.1
```

6.5 loghost

loghost는 각 계산노드의 syslog가 서버의 log file에 저장되도록 각 노드의 /etc/syslog.conf와 서버노드의 /etc/sysconfig/syslog를 편집하고 syslogd와 klogd daemon를 다시 시작한다.

6.6 networking

Network package는 OSCAR 서버를 caching nameserver로 동작하도록 설치된다.
/etc/name.conf에서 configuration을 확인할 수 있다.

6.7 ntpconfig

기본적으로 ntpd 대몬은 run level 2 ~ 5에서 부팅시 시작하도록 설정된다. 어떤 이유에서건 uninstall 없이 NTP를 disable 시키려면, 다음의 명령어를 실행한다.

```
# /sbin/service ntpd stop
```

위의 명령어는 단지 현재에 실행중인 ntpd daemon을 stop시키는 역할만하며, 부팅시에 NTP의 시작을 막기 위해서는 다음의 명령어를 실행한다.

```
# /sbin/chkconfig --level 2345 ntpd off
```

NTP의 restart 명령어와 부팅시의 start-up을 위한 명령어는 아래와 같다.

```
# /sbin/service ntpd start
# /sbin/chkconfig  --level 2345 ntpd on
```

NTP에 대한 자세한 정보는 http://www.ntp.org/ 에 있는 Documentation을 참조하기 바란다.

(참고사항) 위에서 세팅한 NTP 서버에 제대로 접속할 수 있는지는 다음의 명령어를 이용하여 확인 할 수 있다.

```
# ntpq -p
```

6.8 ODA

ODA(OSCAR Database)는 OSCAR 개발의 일부로서 만든 package로서 Mysql을 사용하는 Perl script로 작성되어 있으며, OSCAR cluster를 위한 database를 제공한다. oda package는 OSCAR cluster와 OSCAR에서 사용하는 software package에 사용되는 다양한 정보를 저장하거나 retrieve하는 데 사용된다. 자세한 내용은 다음의 매뉴얼을 참조하기 바란다.

6.9 OPIUM & SSH

```
$ man /usr/share/man/man1/oda.1.gz
```

OPIUM(OSCAR Password Installer and User Management)은 사용자(user)와 암호(password)를 관리하며, 사용자 계정 파일을 일정한 주기(default는 15분)에 따라 각 노드에 복사한다.
/opt/opium/etc/sync_users.conf의 파일 내용을 변경하여 다른 파일에 대해서도 이 기능을 사용할 수 있다.
OPIUM package는 또한 SSH key setup을 관리한다. 이것은 SSH의 user key를 generate하는 start-up 스크립트(/etc/profile.d/ssh-oscar.sh)를 /etc/profile.d에 위치시킴으로서 가능해 진다. 이 스크립트는 또한 user들이 password 입력없이, 안전하게 전체의 클러스터 내에서 자유로이 오고가고(traverse) 할 수 있도록 authorized_keys를 만든다. 이때에 만들어지는 파일들은 user의 home 디렉토리에 만들어지므로, 반드시 home 디렉토리는 각각의 계산 노드에 mount 되어야 한다. 또한 /etc/profile.d의 스크립트는 OSCAR 2.2.1에서는 계산 노드에는 존재하지 않으므로, user는 반드시 head 노드(지금의 설치환경에서의 서버 노드)에 로그인을 한 번 이상은 해야 한다. SSH1 과 SSH2 protocol 모두를 지원한다.

(참고사항) OSCAR는 기본적으로 cluster내에서 telnet과 rsh, ftp 및 기존의 legacy networking을 사용하지 않는다. 대신 보다 안전한 ssh, scp, 그리고 sftp를 사용하며, 위에

서 보여진 바와 같이 모든 사용자를 위한 authentication 및 key management를 자동으로 처리한다.

6.10 PBS & Maui Scheduler

OSCAR는 open source 버전의 PBS, OpenPBS를 job launcher와 batch queueing system로 사용한다. 상업용 version(PBSPro)의 PBS도 존재한다. OpenPBS는 기본적인 FIFO scheduler를 사용하나, OSCAR에서는 보다 견고한 open source scheduler인 Maui를 대신 사용한다. OSCAR 2.2.1에서는 OpenPBS version 2.3p16과 Maui 3.0.7p8을 사용한다. Maui는 추후 OSCAR 버전에서 계속 update될 것이다. 기본적인 PBS의 기능은 OSCAR test suite("Test Cluster Setup"(step 8))에서 테스트되며, 또한 OSCAR의 다른 소프트웨어를 테스트할 때도, job를 launch시키는 역할을 PBS가 하게 된다. OSCAR는 PBS의 테스트가 통과되면, PBS와 Maui를 실행시킨다.

만일 OSCAR cluster의 사용자들이 예전에 PBS를 사용해 보지 못했다면, 관리자는 사용자에게 PBS에 적응하는 데 필요한 정보(OSCAR에서 제공하는 sample PBS script, User Manual)를 제공해야 할 것이다. 일단 step 8에서 PBS의 테스트를 실행하면 /home/oscartst에서 sample PBS script를 볼 수 있을 것이다.

- PBS설정

기본적으로 PBS는 queue나 cluster-specific한 parameter에 대한 설정 없이 설치된다. OSCAR는 PBS를 SIS의 database에 있는 정보를 토대로 PBS를 세팅한다. "Complete Cluster Setup"(step 7)의 과정이 OSCAR Wizard에서 실행되면, PBS 패키지내의 post_install(/root/oscar-2.2.1/packages/pbs/scripts/)스크립트가 실행되며, 이때 기존의 PBS configuration이 존재한다면, 이 configuration을 덮어 쓰지 않는 범위 내에서, 정의 되어 있지 않은 PBS의 parameter를 설정한다. 그러나 만일 이 경우에도 모든 parameter를 OSCAR의 default값으로 사용하고자 한다면, post_install 스크립트를 "--default" 옵션과 함께 command prompt내에서 실행시키면 된다.
qmgr은 queue와 PBS server parameter를 설정하는 데 사용할 수 있으며, OSCAR PBS의 post_install 스크립트역시 이것을 사용하고 있다. man page를 이용할 수도 있으나, PBS의 admin guide를 읽는 것이 이것을 익히는 가장 빠른 방법이다. 이 Open PBS의 admin guide는 OpenPBS의 홈페이지에서 계정을 등록한 후 사용 인증을 받으면 다운로드 할 수 있다.

- PBS Resource

관리자는 각 node에 "property"을 설정할 수 있으며, PBS에서는 이 properties를 "resources"와 같은 의미로 사용한다. 사용자가 qsub 명령어를 사용하여 job을 submit할 때, 사용하고자 하는 특정한 nodes를 이 resources를 이용하여 지정할 수 있다. 각 node에 대한 "property"는 OSCAR의 경우, /var/spool/pbs/server_priv/nodes 파일에서 지정할 수 있으며, 그 내용을 변경했을 경우에는 반드시 PBS server를 restart시켜야 변경 내용이 적용된다. 이것은 또한 qmgr명령어를 사용하여 할 수도 있다.

- FAQueue

PBS queue에 대한 가장 일반적인 오해는, 그것이 특정한 nodes의 그룹과 결합되어 있다고 생각하는 것이다. 이것은 (node의 property를 사용하고 있는 queue를 만들었을 때를 제외하고는) 일반적으로 사실이 아니다. 만일 4개의 node를 사용하는 queue가 정의되어 있다면, 이것은 cluster내의 어떤 특정한 node와도 결합되어 있지 않다. job을 submit할 때 주의할 또 다른 점은 cluster의 nodes, ppn(procs per node), walltime등에 대한 (max, min)값 안에 들어가는 범위 내에서 job이 실행되도록 PBS job script를 작성하는 것이다.
만일 특정한 node를 사용하고자 한다면 사용하고자 하는 node에 대한 resource attribute, 즉 property를 job script내에 명시해 주어야 한다.

만일 PBS server에 대한 모든 설정 정보를 보고자 한다면 다음의 명령어를 사용하면 된다.

```
# qmgr -c "print server"
```

qmgr interface는 새로운 queue와 그 parameter를 정의하는 데 사용할 수 있다. 또한 OSCAR에서 설치한 default queue인 "workq"의 parameter도 변경할 수 있다. 예를 들면, workq의 설정 정보를 보고자 한다면, 다음의 명령어를 실행하면 된다.

```
# qmgr -c "list queue workq"
```

출력된 값들 중에서, 특정한 값을 변경하고자 한다면 다음을 사용하면 된다.

```
# qmgr -c "set queue workq PARAMETER = VALUE "
```

여기에서 PARAMETER는 위의 "list queue workq" 명령어에 의해 출력된 parameter중 하나이며, VALUE는 그 parameter에 주고자 하는 적절한 값이다. "print server"와 "list queue" 명령어를 통해 변경 내용을 확인할 수 있다.

만일 post_install 명령어를 "--default" 옵션과 함께 사용한다면, 관리자가 변경시킨 내용을 읽어 버리게 된다. 또한 OSCAR의 default wallclock의 limit은 10,000 hours이므로, 적절한 값으로 변경시켜야 한다.

아래는 PBS와 관련된 사이트들이다.
OpenPBS: http://www.openpbs.org/
PBSPro: http://www.pbspro.com/
Maui Scheduler: http://www.supercluster.org/

6.11 pfilter

pfilter는 perl로 작성된 firewall compiler(/usr/sbin/pfilter)이며, Linux kernel의 packet filtering 기능을 컨트롤하는데 사용된다. pfilter는 시스템 관리자가 복잡한 iptables 세팅을 직접 하지 않고도 (/etc/pfilter.conf, configuration file에서) high-level의 directive만을 사

용하여, 똑같은 결과(/usr/sbin/pfilter.cmds 참조)를 얻도록 만들어져 있다.

OSCAR가 설치될 때, 기존의 pfilter 서버 configuration을 새로운 configuration과 결합시키며, 또한 client machine의 새로운 pfilter configuration file을 만든다. 그 결과로 만들어진 configuration은 다음의 일을 한다.

- 기존의 서버에 의해 허용된 모든 네트웍 연결은 그대로 main OSCAR 서버 노드에서 허용된다.
- OSCAR 서버 노드와 모든 클라이언트 OSCAR node는 ssh 로그인을 모든 곳에 대해 허용한다. (만일 모든 클라이언트 노드가 real IP를 사용하도록 설치할 경우, 외부에서 직접 클라이언트 노드에 access할 수 있다. 이것이 보안상 바람직하지 않다고 판단하면, 각 클라이언트 노드의 pfilter 설정파일을 수정해 주어야한다. 자세한 내용은 man pfilter를 참고하기 바란다)
- OSCAR 서버 노드는 http service를 모든 곳에 대해 허용한다.
- 외부로부터의, 그 외의 모든 네트웍 연결은 차단된다.
- syslog는 bad network packet에 대한 로그를 기록하지 않는다.
- cluster내의 노드간의 모든 종류의 network 연결은 허용된다.

만일 OSCAR 서버가 두 개 이상의 네트웍 카드를 가지고 있다면, pfilter는 그중에서 public network card를 결정한 후, packet forwarding를 실행시키며, 가능하다면 다른 나머지 network card에 대한 network address translation(NAT Howto 참조)을 수행하도록 한다.

pfilter는 default가 turn-on 상태이며, 이것은 클러스터를 위한 "good defense in depth"로 생각할 수 있다. 어떤 이유에서건, pfilter를 disable시키고자 한다면, 다음의 명령어를 실행하면 된다.

```
# chkconfig --level=2345 pfilter off
```

pfilter를 stop시키기 위해서는 다음의 명령어를 사용하면 된다.

```
#service pfilter start
```

6.12 switcher

OSCAR는 system-level default MPI (implementation)을 선택하는 것과 사용자가 system-default를 override하고 자신이 원하는 MPI를 선택하는 것에 있어 일반적인 mechanism을 제공한다. 이것은 즉 여러 종류의 MPI가 OSCAR cluster에 install되도록 하며(e.g., LAM/MPI와 MPICH), 또한 사용자에게 MPI implementation에 대한 명확한 선택권을 준다.

- setting the system-level default

system-level MPI를 선택하는 데 있어 다음의 두 가지 방법이 있다.

1. OSCAR installation중에, "Configure selected OSCAR Packages" (step 2)에서 "Environment switcher"에서 MPICH 와 LAM/MPI 중에서 선택할 수 있다.

2. root로서 다음 명령을 실행한다.

```
# switcher mpi --list
```

이용 가능한 MPI implementation이 출력된다. 이 중에서 하나를 system-level default로 선택하기 위해서는 다음의 명령을 내린다.

```
# switcher mpi = MPI_NAME --system
```

여기에서 MPI_NAME은 위의 "--list" 명령어로 출력되는 MPI 중 하나이어야 한다.
"mpi = MPI_NAME"에서 '=' 양 옆에 공간이 있어야 제대로 명령이 실행된다.

(참고사항) "switcher mpi --show"명령어를 통해 default MPI를 확인할 수 있다.

(주의사항) switcher를 통해 선택한 System-level default MPI는 일정한 주기(default는 15분 단위로 crond이 user information과 함께 sync시킴)로 각 node에 전파된다. 따라서 swither로 선택한 이후에, configuration information이 자동으로 다음에 "push"될 때까지 기다리거나 혹은 /opt/opium/bin/sync_users 명령어를 prompt상에서 실행 시켜야 한다.

(주의사항) switcher 명령어를 이용한 default MPI implementation의 교체는 차후의 모든 shell environment(PATH, MANPATH)에 적용된다. 그러나 이미 실행중인 shell에 대해서는 영향을 미치지 않는다.

- setting the user-level default

user-level default setting 방법은 기본적으로 system-level default setting과 같다. 단 "--system" argument를 사용할 필요는 없다. 다음은 그 예이다.

```
$ switcher mpi = lam-6.5.9
```

MPI 이름에 "none"을 사용하면, 어떠한 module도 올라오지 않게 된다. 이것은 특정한 software만을 load하여 사용하는 것을 원하지 않는 사용자들이 이용할 수 있는 방법이다.

```
$ switcher mpi = none
```

user-level default를 지우고자 한다면 (따라서, system-level default로 되돌아가고자 한

```
$  switcher mpi --rm-attr default
```

다면) “default” attribute를 사용하면 된다.

이외의 자세한 내용은 switcher man page를 참조하길 바란다.

7. 참고문헌

OSCAR – http://oscar.sourceforge.net/
NTP – http://www.ntp.org/
C3 – http://www.csm.ornl.gov/torc/C3/
LAM/MPI – http://www.lam-mpi.org/
Maui PBS Scheduler – http://supercluster.org/maui/
MPICH – http://www-unix.mcs.anl.gov/mpi/mpich/
OpenSSH – http://www.openssh.com/
OpenSSL – http://www.openssl.org/
PBS – http://www.openpbs.org/
PVM – http://www.csm.ornl.gov/pvm/
System Installation Suite – http://www.sisuite.org/
LUI – http://oss.software.ibm.com/developerworks/projects/lui/
HDF5 – http://hdf.ncsa.uiuc.edu/HDF5/