LTSP - 리눅스 터미널 서버 프로젝트(Linux Terminal Server Project)

Jim McQuillan (jam@ltsp.org)

v1.14 1 November 1999

박성호 (tempter@fourthline.com)

리눅스는 디스크없는 워크스테이션을 만들어 내는데에 굉장한 환경을 제공해 준다. 디스크 없는 워크 스테이션을 보다 쉽게 구현해줄 도구들을 만들어 내기 위해 오픈소스에서 시작했다. 이 문서는 ltsp의 결과물들을 어떻게 구하고 어떻게 설치하는지에 대하여 설명하고 있다.

목 차

제 1 절 서론

이 프로젝트는 IBM AS/400과 유닉스 서버에 동시에 접속할 수 있는 터미널을 필요로 하는 고객의 문제를 해결하기 위하여 시작되었다. TCP/IP기능이 필요했고, 값이 비싸지 않아야 했으며 유지보수가 쉬워야 했다. 더해서 웹브라우징과 이메일 사용이 가능하다면 더욱 좋았다.

중고PC를 사용해서 소프트웨어들을 탑재할 수도 있었으나 비용이 너무 높았다. 특히 초기 구매비용과 유지보수비의 두 관점에서 보면 더욱 그러했다. 결국 리눅스를 탑재한 디스크없는 워크스테이션이 고객의 요구에 가장 잘 맞을 것이라고 결정을 내리게 되었다.

사실 여기에서 새롭게 만들어진 것은 아무것도 없다. 웹을 뒤져서 etherboot,netboot패키지를 찾아 내었으며 부트롬 소켓이 달린 값싼 ne2000호환 카드를 구했다. eprom라이터를 확보했으며 어떤 칩을 쓸 것인가를 결정했다. 또한 bootp,xdm,nfs-root등등 에 대하여 공부했고, 이 모든 것을 모아서 굴러다니던 486PC에 조합해 넣고 보니 상당히 잘 작동했다. 결국 한대의 서버와 11대의 워크스테이션을 설치해 주었고 고객은 대단히 만족했다.

고객은 22대의 워크스테이션을 더 주문했으며 앞으로는 더 늘릴 예정이라고 한다. (현재 67대의 워크스테이션이 가동중이다.)

몇개월후 실제 유지보수상의 문제가 발생하지 않는 것으로 밝혀지고 우리는 이 방법을 공유하기로 결정했다.

우리가 만든 패키지는 우리의 목적에 잘 부합했으며 당신의 문제도 해결해 줄 수 있길 바란다. 질문이나 의견이 있으면 메일링 리스트(*http://www.ltsp.org/mailinglists.html* `<http://www.ltsp.org/mailinglists.html>`)에 참가하길 바라며 아마도 누군가가 도와줄 것이다.

제 2 절 저작권

저자의 저작권은 원문 그대로 옮기겠습니다.

그리고 번역자의 저작권(?번역권?)은 별도로 주장하지 않겠으나 어딘가에 인용이 된다면 메일이나 한장 보내주는 최소한의 성의를 보여주시길... :-)

제 3 절 디스크 없는 워크 스테이션

워크 스테이션이 부팅될 때 아래와 같은 과정을 밟는다.

1. bootp서버로부터 ip주소를 받는다.
2. tftp서버로부터 커널 이미지를 다운 받는다.
3. nfs서버에서 루트 파일시스템을 마운트한다.
4. X서버를 메모리에 로드하고 실행한다.
5. xdm서버에 접속하여 사용자로 하여금 로긴을 할 수 있게 한다.

대부분의 경우 bootp, tftp, nfs, xdm서버는 한 장비에서 가동된다. 이 장비를 앞으로 단순히 '*Server*'라 부를 것이며 디스크 없는 워크 스테이션을 단순히 '*워크스테이션*'이라 칭하겠다.

워크스테이션이 부팅되고 사용자가 로긴한 후에는 사용자가 사용하는 어떤 프로그램도 사실은 서버에서 실행이 되고 단지 디스플레이만 워크스테이션에서 보여진다. 이는 X서버의 기본적인 기능이며 워크스테이션에서는 리눅스 커널, XFree86, Init 그리고 필요하다면 로컬 프린트를 위한 lpd정도만이 가동된다. 워크스테이션에서 실행되는 프로그램이 거의 없기 때문에 값싼 저사양의 PC만으로도 충분히 사용할 수 있다. 초기에 16m메모리의 486장비에서 시험을 해 봤는데 X 터미널로서 아주 훌륭하게 작동했다.

The Etherboot Home Page `<http://www.slug.org.au/etherboot/>`에서 구할수 있는 etherboot패키지를 사용했는데 거기서는 eprom에 구워 넣을 수 있는 이미지에 관해서 다루고 있었으며 이는 워크스테이션에서 커널을 다운 받을 수 있게 구성하는데 도움이 되었으나 그게 전부였다.

디스크없는 워크스테이션을 가동하기 위한 서버의 셋업과정을 다룬 몇몇 하우투가 있었으나 동시에 여러대의 워크스테이션을 사용하는 것에 대한 논의는 없었다. 문제는 워크스테이션이 가동중에 무엇인가를 디렉토리에 기록한다는 것이며 따라서 개개의 워크스테이션이 각자의 디렉토리를 마운트해서 사용해야 한다는 것이다. 만약 워크스테이션이 50대라면 서버에는 50개의 별도의 디렉토리가 있어야 한다는 것이다. 이는 관리상의 큰 문제가 아닐 수 없다.

우리는 여러대의 워크스테이션에서 공유할 수 있는 루트 파일 시스템을 셋업하는 방법을 만들어 냈다. 커널은 루트 파일 시스템을 읽기전용 모드로 마운트하게 되며 그 다음에 4m크기의 램디스크를 /tmp디렉토리에 마운트 한다. 커널과 X서버가 가동되면서 몇몇 파일을 쓰려고 시도할 때 그 파일들을 램디스크상에 놓고 적절한 위치에 링크를 걸어준다면 된다는 것이다.

더해서 bootp와 같이 작동될 구성파일과 프로그램들을 만들어 냄으로써 각각의 워크스테이션은 서로 다른 하드웨어를 가질 수 있게 되었다. 예를 들면 랜카드, 그래픽카드, 마우스의 종류등등을 기본 구성뿐만 아니라 개별 워크스테이션 구성까지 지정할 수 있다는 것이다. 워크스테이션에서 가동되는 프로그램이 없는 관계로 스왑공간은 구성하지 않았으나 필요하다면 커널에서 nfs를 이용하여 스왑 공간을 만들 수도 있다.

이 구성방법은 400Mhz Pentium II서버에 레드햇 6.0을 인스톨하여 70대의 워크 스테이션을 가동시키는데에 성공적으로 적용이 되었다. 각각의 워크 스테이션은 32m램에 P-166장비를 사용했으며 더 저사양의 PC를 사용할 수도 있었으나 구하기가 어려웠다.

앞으로 워크스테이션에서 보다 많은 프로그램을 가동시킬 예정이다. 많은 사용자가 윈도우 매니져가 넷스케이프같은 웹브라우저등을 로컬에서 실행 시키길 제안해 왔다.

제 4 절 작동원리

디스크없는 워크스테이션을 부팅하는 과정은 몇단계로 나누어 볼 수 있다. 각 단계를 이해하는 것은 발생할 수 있는 문제를 해결하는데 많은 도움이 된다 .

아래 내용은 워크스테이션의 부팅을 지원해 줄 수 있는 서버가 구성되어 있다고 가정한다.

1. 워크스테이션의 파워를 켜면 POST(power on self test)를 수행한 후에 랜카드의 부트롬에 들어있는 부트코드를 실행한다.
2. 부트코드는 랜카드가 있는지를 검색하고(역자주: 랜카드위에 있는 부트롬에 들어있는 부트코드가 랜카드의 존재여부를 검색한다는 것이 이상하게 들릴지 모르지만 사실 부트롬과 랜카드는 별개의 것이라는 사실을 생각한다면 이해가 갈 것이다.) 발견된다면 초기화 한다.
3. 부트코드는 bootp 브로드캐스트 패킷을 로컬 랜에 송출한다. 이 패킷에는 자신의 랜카드의 맥어드레스가 들어 있다.
4. **inetd** 프로세스는 워크스테이션이 보낸 패킷을 검출하게 되면 **bootpd**데몬을 호출하여 응답하도록 조치한다.
5. **bootpd** 프로세스는 자신의 구성화일을 읽어서 해당 맥어드레스를 가진 항목이 있는가를 확인하고 항목이 존재한다면 들어있는 정보를 가지고 응답 패킷을 구성하게 된다. 이 때 몇가지의 정보가 서버로부터 워크스테이션으로 가게 되면 그 목록은 다음과 같다.
   (a) 워크스테이션에 지정된 IP 어드레스 ('ip=')
   (b) 로컬 랜의 서브넷 마스크('sm=')
   (c) Bootfile이 있는 디렉토리 ('hd=')
   (d) 다운받을 커널 이미지의 이름 ('bf=')
6. 워크스테이션의 부트코드는 서버로 부터 받은 정보를 가지고 TCP/IP인터페이스를 설정한다.
7. 부트코드는 서버로부터 커널 이미지를 다운 받기 위하여 tftp요청을 보내게 된다.
8. 커널 다운로드가 끝나게 되면 부트코드는 커널의 시작 코드로 제어권을 넘겨주게 된다.
9. 커널이 작동이 시작되고 각종 주변기기들을 초기화 한다.
10. 커널은 자신의 TCP/IP인터페이스를 설정하기 위하여 필요한 모든 정보를 입수하려고 다시 bootp요청을 보낸다. 이 때 부트코드가 커널에게 정보를 넘겨주지 않는 다는 점을 유의하자. 커널은 반드시 스스로 정보를 다시 요구해야 한다.
11. 서버는 워크스테이션의 커널에게 필요한 정보를 다시 보내준다.
    (a) 워크스테이션에 지정된 ip어드레스 ('ip=')
    (b) 로컬 랜의 서브넷 마스크('sm=')
    (c) nfs를 이용하여 마운트 될 루트 디렉토리의 위치('rp=')
    (d) 게이트웨이 어드레스 ('gw=')
    (e) DNS 서버의 주소 ('ds=')
    (f) 워크스테이션의 호스트 이름 (이 내용은 bootptab의 맨 처음 항목에 있다.)

    위의 정보가 모두 전달이 된 후에 네트워크 인터페이스가 구성되고 활성화 된다.
12. NFS를 통하여 루트파일 시스템이 마운트 된다. 파일 시스템은 읽기 전용으로 마운트되며 이는 여러대의 워크스테이션이 하나의 디렉토리를 공유 할 수 있도록 하기 위한 것이며, 각각의 워크스테이션이 공유 디렉토리의 내용을 바꾸어 버리는 것을 바라지 않기 때문이다.
13. 이 시점에서 제어권은 커널로부터 '**init**' 프로세스로 넘겨지게 된다.
14. init은 **/etc/inittab** 파일을 읽고 환경을 설정하기 시작한다.
15. inittab파일의 맨 처음 요소를 **rc.local** 명령이다. 시스템이 '**sysinit**'상태에 있을때 실행된다.
16. rc.local스크립트는 기록되거나 변경될 필요가 있는 파일들을 담을 4m크기의 램디스크를 만들고 마운트한다.

17. 이 램디스크는 /tmp디렉토리에 마운트 된다. 실제로 기록되거나 변경될 필요가 있는 파일들은 모두 이 디렉토리에 위치하게 되며 원래의 위치에는 모두 링크로 존재하게 된다. 예를 들어 워크스테이션이 가동중에는 **/dev/tty0**의 권한을 변경하려는 시도가 있게 된다. 만일 이 디바이스파일이 /dev 디렉토리에 위치한다면 루트 파일시스템 전체의 권한이 읽기 전용이므로 아마도 권한은 읽기 전용일 것이다. 따라서 우리는 실재로 이 파일을 쓰기 가능한 /tmp밑의 /nodes디렉토리에 두고 그 링크만을 /dev 에 두게 되는 것이다.
18. **/proc** 파일 시스템을 마운트 한다.
19. **loopback** 네트워크 인터페이스를 구성한다.
20. **/tmp**밑에 시스템이 가동중에 필요한 몇몇 파일들을 기록하기 위한 디렉토리들을 만든다.
    (a) **/tmp/compiled**
    (b) **/tmp/var**
    (c) **/tmp/var/run**
    (d) **/tmp/var/log**
    (e) **/tmp/var/lock**
    (f) **/tmp/var/lock/subsys**

    디렉토리들이 생긴다.
21. **/tmp/XF86Config** 파일이 **/tftpboot/lts/ltsroot/etc/lts.conf** 구성파일의 항목들을 참조하여 만들어진다. 참조되는 항목은 마우스의 종류등의 기타 X의 설정에 관련이 있는 항목들이다.
22. **/tmp/startx** 스크립트 파일이 생성된다. 이 스크립트는 실행될 X서버의 종류, **xdm**이 실행될 서버의 주소등의 내용을 담게 된다. 이 정보 또한 **/tftpboot/lts/ltsroot/etc/lts.conf** 구성파일의 내용에 의거하여 생성된다.
23. **/tmp/syslog.conf** 파일이 생성된다. 이 파일에는 시스템에서 발생된 각종 로그정보가 네트워크상의 어느 호스트로 보내져야 하는지에 대한 정보가 들어있다. 이 정보는 역시 **lts.conf**에 들어있으며 다른 파일들과 마찬가지로 /**etc**디렉토리 밑에 링크로서 존재하게 된다.
24. **syslogd** 데몬이 방금 생성된 구성파일에 의거하여 실행된다.
25. 제어권이 다시 **init** 에게 돌아간다. Init 은 **initdefault** 항목을 참조하여 **runlevel** 을 결정한다.
26. 만약 runlevel이 **3** 이라면 콘솔에 기본 쉘이 실행된다. 문제해결을 시도할 경우에 도움이 될것이다.
27. 만약 runlevel이 **5** 라면 **/tmp/startx** 이 실행되며 이는 X를 가동시킬 것이다.
28. X가 시작되면 바로 XDMCP질의를 서버에게 보내게 되며 이는 로그인 박스를 화면에 표시해 줄것이다.
29. 사용자가 로그인 되면 X세션이 서버에 실행되며 이후에 실행되는 모든 X프로그램들은 서버에서 실행되고 단지 화면 출력이나 사용자 입력만이 워크스테이션을 통해 이루어지게 된다.

기본적으로 X터미널이 완성되었다.

제 5 절 서버 설치하기

레드햇 6.0을 서버로 운영한다는 기준에 맞추어 필요한 모든 RPM들을 만들어 놓았다.

모든 소프트웨어는 *LTSP download page* `<http://www.ltsp.org/download/index.html>`에서 찾을 수 있으며 ftp로는 *LTSP ftp server* `<ftp://ftp.ltsp.org/pub/download>`에서 받을 수 있다.

5.1 다운로드 가능한 패키지들

XFree86-xfs-3.3.3.1-52.i386.rpm

X 폰트 서버의 업그레이드이며 레드햇 6.0에는 필요하지만 6.1에는 불필요하다.

bootp-2.4.3-7.i386.rpm

Bootp 데몬

lts_core-1.0.rpm

LTS package의 중심. 루트파일시스템, 각종 구성파일, 워크스테이션을 위한 문서들을 담고 있다. 문서들은 /usr/doc/lts-1.0 디렉토리에 들어가며 거기에는 이 문서의 HTML, SGML, Postscript 그리고 Text형식이 준비되어 있다.

lts_kernel_ne2000-1.0.rpm

NE2000어댑터를 위한 드라이버가 포함되어 컴파일된 커널이다. 이 커널을 위한 컴파일 `.config` 화일은 `/tftpboot/lts/kernel_configs/config.ne2000`에 들어가 있다.

lts_kernel_rtl8139-1.0.rpm

RTL8139칩을 이용한 어댑터를 위한 드라이버가 포함되어 컴파일된 커널이다. 이 커널을 위한 컴파일 `.config` 화일은 `/tftpboot/lts/kernel_configs/config.rtl8139`에 들어가 있다. 이 칩을 사용한 어댑터의 예로서는 SMC-1211TX 10/100 등이 있다.

X Servers

다양한 X 서버들이 ltsp.org site 에 준비되어 있다.

이 모든 패키지는 *LTSP website* `<http://www.ltsp.org/download>`나 *LTSP ftp server* `<ftp://ftp.ltsp.org/pub/download>`에서 구할 수 있다.

5.2 IP주소 체계 계획

네트워크 상의 모든 장비들은 고유한 ip주소를 가져야 한다. 이 예 에서는 사용 보류된 C클래스 어드레스인 **192.168.0.0** 대역을 사용하겠다. 물론 원하는 어드레스로 바꾸어도 된다.

서버의 주소로 **192.168.0.254** 를 택하고 워크스테이션용으로 **192.168.0.1** 부터 하나씩 올라가며 사용하도록 하겠다. 이 주소체계는 하나의 서버에 253개의 워크스테이션을 가능하게 할 수 있다.

만일 더 많은 수의 워크 스테이션이 필요하다면 서로다른 C클래스 어드레스를 갖는 여러개의 서버를 구성하든지, 아니면 하나의 서버와 B클래스 어드레스를 사용하면 된다. 이 경우 65533개의 워크스테이션이 가능하게 된다. (굉장하게 들리죠??)

워크스테이션의 이름으로는 '**ws001**'에서 시작하여 하나씩 위로 올라가는 방법을 택했다.

5.3 XFS 업그레이드 Upgrade the xfs package

레드햇 6.0에 포함된 xfs패키지는 원격 워크스테이션을 지원하지 않는다. 지금은 업데이트된 xfs가 있으며 이는 레드햇 사이트나 LTSP에서 다운 받을 수 있다. `XFree86-xfs-3.3.3.1-52.i386.rpm` 이나 그 후의 버젼이 필요하다.

만일 레드햇 6.1을 사용한다면 xfs는 신경쓸 일이 없다. 이 버젼에서는 원격 워크스테이션을 위한 기능이 있다. 그러나 물론 구성파일과 X를 위한 XF86Config파일은 수정을 해주어야 한다.

업그레이드 하는 명령은 다음과 같다.

```
rpm -U XFree86-xfs-3.3.3.1-52.i386.rpm
```

업그레이드를 했다면 시작 스크립트를 수정해야 한다. 수정해야 할 파일은 /etc/rc.d/init.d/xfs이며 두줄을 바꿔주어야 한다. 바꿔야 할 내용은 레드햇 버전에 따라 다르다.

레드햇 6.0

daemon --check xfs su xfs -c \"xfs -port -1\" -s /bin/sh으로 시작하는 줄을 찾아야 한다. 22번째 줄근처와 41번째 줄근처에 있다. 다음과 같이 수정한다.

```
daemon --check xfs su xfs -c \"xfs -port 7100\" -s /bin/sh
```

Redhat 6.1

daemon xfs -droppriv -daemon -port -1으로 시작하는 줄을 찾아야 한다. 22번째 줄근처와 41번째 줄근처에 있다.

다음과 같이 수정한다.

```
daemon xfs -droppriv -daemon -port 7100
```

위의 수정내용은 폰트서버가 원격 워크스테이션에게도 서비스 해주도록 하는 것이다.

/etc/X11/XF86Config파일도 수정해야 한다.

다음과 같은 줄을 찾는다.

```
FontPath   "unix/:-1"
```

다음과 같이 수정한다.

```
FontPath   "tcp/localhost:7100"
```

5.4 bootpd 설치

레드햇에는 bootpd가 포함되어 있지 않다. 따라서 전 버전에서 찾아 보거나 레드햇에서 다운을 받거나 ftp.ltsp.org에서 다운 받을 수 있다. 인스톨하는 명령은 다음과 같다.

```
rpm -i bootp-2.4.3-7.i386.rpm
```

이 명령으로 설치가 되고 샘플 구성화일도 설치가 된다.

5.5 lts_core 패키지 설치

lts_core-1.0.rpm 파일을 **ftp.ltsp.org**에서 다운 받아서 다음의 명령으로 인스톨한다.

```
rpm -i lts_core-1.0.rpm
```

이 명령은 **/tftpboot/lts** 디렉토리를 만들어 주고 그 안에 루트 파일시스템으로 마운트 될 기본적인 디렉토리 구조를 만들어 준다. 또한 **/etc/bootptab** 와 **/etc/exports** 파일안에 필요한 항목들을 추가해 준다.

5.6 적합한 커널 선택

워크스테이션이 부팅될때 서버로부터 커널을 다운받아 메모리에 적재하고 실행하게 된다. 이 커널은 네트워크 부팅을 위하여 좀 특별하게 컴파일 되어야 하며 네트워크 어댑터 드라이브가 포함되어 있어야 한다. 이 드라이버를 모듈로 로드할수는 없다. 필요한 설정이 되어 있는 커널을 준비했으며 단지 부착된 네트워크 어댑터에 맞는 커널을 골라서 다운받아 설치하기만 하면 된다.

RPM들이 ltsp사이트에 있으며 예를 들어 **lts_kernel_ne2000-1.0.rpm**을 다운받아서 설치하는 명령은 다음과 같다.

```
rpm -i lts_kernel_ne2000-1.0.rpm
```

이 명령은 **vmlinuz.ne2000** 커널이미지를 **/tftpboot/lts** 디렉토리에 넣어준다. 다른 커널이 필요하다면 같은 과정으로 설치할 수 있다.

5.7 X서버 선택

가지고 있는 비디오카드에 맞는 X 서버를 골라야 한다. XF86_SVGA 서버가 대부분의 경우 잘 작동되지만 가지고 있는 카드의 칩셋에 맞는 서버를 골라서 설치한다면 보다 좋은 성능을 기대할 수 있게 된다.

해당 X서버를 레드햇에서 다운받아도 좋으며 ltsp사이트에서 다운 받는다면 X서버를 적절한 위치에 설치까지 해주는 rpm을 구할 수도 있다.

만약 SiS-620칩셋을 가진 비디오 카드를 사용한다면 3.3.5이후의 XF86_SVGA를 사용해야 할 것이다. 이전 버젼의 서버는 SiS칩셋을 제대로 지원하지 않는다. 다음과 같은 파일들이 LTSP의 사이트에 있다. **lts_xmach64-1.1.rpm** 과 **lts_xsvga-1.1.rpm**등을 다운 받아서 설치할 수 있다.

예를 들어 SVGA서버를 인스톨하는 명령은 다음과 같다.

```
rpm -i lts_xsvga-1.1.rpm
```

이 명령으로 XF86_SVGA 서버를 **/tftpboot/lts/ltsroot/usr/X11R6/bin**디렉토리에 설치할 수 있다.

5.8 구성화일 편집

lts_core를 설치함으로써 몇몇 설정항목을 자동으로 추가해 주기는 하지만 대부분의 경우 상황에 맞게 수정을 해주어야 할 것이다.

5.8.1 /etc/inetd.conf

디폴트로 bootp와 tftp에 관련된 항목이 주석처리 되어있다. 다음과 같은 줄을 찾아서 #마크를 지워주어야 한다.

```
#
#tftp   dgram   udp    wait   root   /usr/sbin/tcpd  in.tftpd
#bootps dgram   udp    wait   root   /usr/sbin/tcpd  bootpd
#
```

5.8.2 /etc/X11/xdm/Xservers

서버에서 X를 자동으로 실행 할것인가를 결정해야 한다. 만일 시스템을 부팅할 때 자동으로 그래피컬한 로그인 화면을 보길 원한다면 아래 내용을 신경쓸 일이 없다. 그러나 X가 자동을 시작되길 바라지 않는다면 아래의 줄을 주석처리해야 한다.

다음과 같은 줄을 찾는다.

```
:0 local /usr/X11R6/bin/X
```

만약 # 기호를 이용하여 주석처리를 해 준다면 xdm이 X를 가동하는 것을 막아준다. 이 경우 X를 시작하기 위해서는 startx 스크립트를 실행하면 된다.

5.8.3 /etc/X11/xdm/Xaccess

이 파일은 원격 워크스테이션이 xdm과 교신할수 있는지 여부를 결정해 준다. 레드햇 6.0에서는 기본적으로 허용되어 있으나 6.1에서는 디폴트로 막혀있다.

원격 접속을 가능하게 하려면 40번째 줄 근처에 있는 다음 줄에서 #기호를 지우면 된다.

```
# *                                      #any host can get a login window
```

5.8.4 /etc/inittab

서버에서 xdm을 실행해줄 필요가 있다. 이는 보통 /etc/inittab에서 제어되며 사용가능한 xdm서버는 여러가지가 제공된다. (xdm, gdm, kdm등)

xdm을 사용할 것을 권장한다.

/etc/inittab에서 다음줄을 찾는다.

```
x:5:respawn:/etc/X11/prefdm -nodaemon
```

6.0에서는 prefdm은 gdm에 링크되어 있으며, 6.1에서는 prefdm은 어떤 xdm을 이용할 것인가를 결정하는 스크립트 파일이다.

이 줄을 다음과 같이 수정한다.

```
x:5:respawn:/usr/bin/X11/xdm -nodaemon
```

보통 xdm은 런레벨 5에서 작동된다. 서버가 어떤 런 레벨에 있는지는 다음 항목을 보면 알 수 있다. /etc/inittab파일의 initdefault항목을 보았을 때, 다음과 같다면

```
id:3:initdefault:
```

아래와 같이 고쳐준다.

```
id:5:initdefault:
```

5.8.5 /etc/bootptab

lts-core패키지를 인스톨 하면 다음 항목을 /etc/bootptab파일에 추가해준다.

```
## LTS-begin #####################################################
.ltsp:\
  :ht=ethernet:\
  :ds=192.168.0.254:\
  :gw=192.168.0.254:\
  :lg=192.168.0.254:\
  :sm=255.255.255.0:\
  :hn:\
  :hd=/tftpboot/lts:\
  :rp=/tftpboot/lts/ltsroot:
#
# The following is an example of a line needed for a workstation
#
# ws001:tc=.ltsp:ha=AABBCCDDEEFF:bf=vmlinuz.ne2000:ip=192.168.0.1:
## LTS-end #######################################################
```

각각의 워크스테이션을 위한 항목을 추가해 주어야 한다. 예로 추가되어 있는 라인을 본보기 삼아 수정하면 된다.

ha=항목을 채워주기 위하여 네트워크 어댑터의 맥어드레스를 알아야 하며 **bf=**항목을 채워주기 위하여 필요한 커널의 이름을 알아야 한다. 또한 **ip=**항목에는 해당 워크스테이션에 지정된 ip주소를 넣어주어야 한다.

또한 준비된 것과 다른 네트워크 어드레스를 사용한다면 해당 항목들을 모두 수정해 주어야 한다.

5.8.6 /etc/hosts

bootptab파일에 해당 워크스테이션에 대한 정보를 등록해 주었다면 워크스테이션의 ip주소와 이름을 연결해 줄 수 있는 뭔가를 해주어야 한다. 즉, /etc/hosts파일에 등록을 하거나 DNS서버에 등록을 해주거나 등이다. NFS서버는 반드시 마운트를 요구하는 클라이언트의 이름을 ip주소로 풀어낼 수 있어야 하기 때문이다.

5.8.7 /etc/exports

자동으로 추가된 항목만으로 잘 작동할 수 있으나 만일 다른 주소의 네트워크 어드레스를 사용한다면 상황에 맞게 수정해 주어야 한다. 기본은 아래와 같다.

```
## LTS-begin ##
/tftpboot/lts/ltsroot    192.168.0.0/255.255.255.0(ro,no_root_squash)
## LTS-end ##
```

5.8.8 /etc/rc.d/init.d/syslog

이 스크립트는 시스템이 부팅될 때 syslogd데몬을 시작하는 파일이다. 이는 네트워크 상의 다른 장비가 이 서버의 데몬에게 로그 메시지를 보내어 기록할 수 있도록 하기 위하여 수정될 필요가 있다.

22번째 줄근처에 다음 내용을 아래와 같이 바꾸어 준다.

```
daemon syslogd -m 0

daemon syslogd -m 0 -r
```

5.8.9 /tftpboot/lts/ltsroot/etc/lts.conf

이는 각각의 워크스테이션에 대한 설정 파일이다. 설정가능한 대부분의 내용이 여기에 들어 있다.

각각의 부분은 개개의 워크스테이션을 나타내지만 **Default**섹션도 있다.

섹션들의 제목은 각각의 워크스테이션의 이름을 나타내거나 아니면 'Default'일 수가 있다. 모든 워크스테이션이 동일한 장비를 사용한다면 디폴트 섹션만으로 충분하다.

/tftpboot/lts/ltsroot/etc/lts.conf의 예는 다음과 같다.

```
[Default]
        XSERVER            = XF86_SVGA
        SERVER             = 192.168.0.254
        X_MOUSE_PROTOCOL   = "PS/2"
        X_MOUSE_DEVICE     = "/dev/psaux"
        X_MOUSE_RESOLUTION = 400
        X_MOUSE_BUTTONS    = 3

[ws001]
        XSERVER            = XF86_SVGA
```

```
        X_MOUSE_PROTOCOL   = "Microsoft"
        X_MOUSE_DEVICE     = "/dev/ttyS1"
        X_MOUSE_RESOLUTION = 50
        X_MOUSE_BUTTONS    = 3
        X_MOUSE_BAUD       = 1200

[ws002]
        XSERVER            = XF86_Mach64

[ws003]
        XSERVER            = XF86_SVGA
        X_COLOR_DEPTH      = 24
```

가능한 항목들은 다음과 같다.

SERVER

이 항목은 XDM과 SYSLOGD, 양쪽의 서버를 동시에 지칭한다. 만약 별도로 두 서버를 지정하지 않는다면 이 서버의 주소로 모든것이 통일된다.

XDM_SERVER

별도로 xdm서버를 지정하려면 사용한다. 만약 별도로 이 항목이 사용되지 않는다면 위의 항목에 지정된 대로 사용된다.

SYSLOG_HOST

별도로 syslogd서버를 지정하려면 사용한다. 만약 별도로 이 항목이 사용되지 않는다면 위의 항목에 지정된 대로 사용된다.

XSERVER

워크스테이션에 사용될 X서버의 종류를 지정한다. 가능한 어떤 서버도 해당위치에 설치되어 있다면 사용할 수 있다. 기본값은 **XF86_SVGA**이다.

X_MOUSE_PROTOCOL

XFree86 의 Pointer Protocol키워드는 모두 사용될 수 있다. 일반적으로는 "Microsoft" 이거나 "PS/2"이다. 기본값은 **"PS/2"**이다.

X_MOUSE_DEVICE

마우스가 연결된 장치명이다. 만약에 마우스가 시리얼 마우스라면 이 항목은 `/dev/ttyS0`같은 시리얼 포트중의 하나가 될 것이다. 마우스가 PS/2라면 `/dev/psaux`가 될것이다. 기본값은 `/dev/psaux`이다.

X_MOUSE_RESOLUTION

XF86Config 파일의 Resolution 값이다. 일반적으로 시리얼 마우스는 **50**정도이고 PS/2마우스는 **400**정도이다. 기본값은 **400**이다.

X_BUTTONS

이 항목은 마우스에 달려있는 버튼의 숫자를 알려준다. 보통 **2** 이거나 **3** 이다. 기본값은 **3**이다.

X_MOUSE_BAUD

시리얼 마우스의 경우 이 값는 baud rate이다. 기본값은 **1200**이다.

X_COLOR_DEPTH

이 항목은 색상을 결정한다. 가능한 값은 **8**, **15**, **16**, **24**, **32** 등이다. 8은 256색, 16은 65536색 등등이다. 모든 X서버가 모든 색상비트값을 지원하지는 않는다. 기본값은 **16**이다.

Comments

주석은 '#' 기호로 시작하여 줄 끝까지 이다.

5.9 데몬 재시작

모든 설정파일을 수정하거나 생성한 후 각각의 데몬들이 재시작되어야 한다. 각각의 데몬을 재 시작하거나 그냥 서버를 리부팅 해도 된다.

xfs

데몬을 세우기 위하여 아래의 명령을 사용한다.

```
/etc/rc.d/init.d/xfs  stop
```

그리고 나서 다시 시작하기 위하여 아래의 명령을 사용한다.

```
/etc/rc.d/init.d/xfs  start
```

어떤 이유에 의하여 xfs restart가 정상적으로 작동되지 않았다. 그래서 먼저 stop을 한 후, start를 실행한다. (역자 주: 정확한 확인을 하지는 않았으나 되는 경우도 있는 것으로 보임.)

NFS

```
exportfs -ra
```

bootpd

```
killall -q -HUP bootpd
```

syslogd

```
/etc/rc.d/init.d/syslog restart
```

inetd

```
killall -HUP inetd
```

제 6 절 워크스테이션 설치

워크스테이션은 네트워크 어댑터와 부트롬이 필요하다. 부트롬 라이터를 사용할 수 있다면 LTSP에서 롬 이미지를 다운 받아서 롬에 기록하면 된다. 그 이미지는 etherboot프로젝트의 산물이며 소스코드는 www.slug.org.au/etherboot에서 구할 수 있다.

부트롬이나 부트롬을 장착한 네트워크 어댑터, 또는 디스크 없는 워크스테이션 전체를 <shameless plug>DisklessWorkstations.com</shameless plug> 에서 구매할 수도 있다. (역자주 : 필요하신 분이 있다면 제가 구해 드릴수도 있습니다. 약간의 수고비와 함께...:-))

다른 대안은 부트롬 이미지와 함께 floppyload.bin파일을 같이 다운로드 받는 것이다. 그리고 두 파일을 플로피 디스켓의 부트섹터에 카피하면 된다. 그런후에 플로피로 부팅을 하면 랜카드에 부트롬이 장착된것과 동일하게 네트워크 어댑터를 초기화 하고 부트 과정을 진행하게 된다. 테스트할 목적이라면 아주 유용할 것이다.

카피하는 명령은 다음과 같다.

```
cat floppyload.bin ne.rom >/dev/fd0
```

이 명령은 두 파일을 플로피의 마스터 부트 블록에 기록해 줄 것이다.

제 7 절 Testing

모든 설정을 마친후 서버를 리부팅 한 다음, 워크스테이션의 파워를 넣는다. 처음에 ip주소등의 정보를 주고받는 화면이 나온 후, tftp전송이 이루어 지는 것을 볼 수 있을 것이다.

바로 커널이 부팅되기 시작하며 다시 한번 bootp를 이용하여 ip주소등을 받아오는 것을 볼 수 있을 것이다. 그 후에 X서버가 가동되며 xdm을 통한 그래피컬한 로그인 화면이 뜨게 된다.

도중에 어디에선가 멈춘다면 이 설명서를 다시한번 주의깊게 읽은 후 빠트린 부분을 체크하기 바란다.

그래도 잘 되지 않는 다면 `jam@ltsp.org`에게 메일을 보내주면 도와줄수 있도록 노력할 것이다. (역자 주 : 영어가 되지 않는 분은 제게 메일을 보내도 도와드리겠습니다. :-))

제 8 절 Troubleshooting

8.1 워크스테이션화면에 'Searching for server':

```
NE2000 base 0x0300, addr XX:XX:XX:XX:XX:XX
Searching for server (BOOTP)...
<sleep>
<sleep>
```

라고 나온다면 bootp서버를 찾지 못하는 경우이다. 아래를 점검한다.

1. Cabling - 카드나 허브에 링크 LED가 점등되어 있는지 본다.
2. Bootpd 데몬 - 서버의 **/etc/inetd.conf**에 제대로 설정되어 있는지 체크한다.
3. /etc/bootptab file - 화면에 나온 맥어드레스가 서버의 /etc/bootptab 파일에 잘 등록되어 있는지 확인한다

8.2 Problem with NFS mounting root filesystem, errno=13.

이 에러는 권한문제이다.

1. 워크스테이션의 이름과 ip주소가 매핑이 잘 되어 있는가를 확인한다. `/etc/hosts` 이나 DNS 테이블을 확인해야 한다.
2. `/etc/exports` 에 해당 항목이 잘 설정되어 있는가를 확인한다.

   ```
   /tftpboot/lts/ltsroot         192.168.0.0/255.255.255.0(ro,no_root_squash)
   ```

   IP주소는 현재의 로컬랜의 세팅과 잘 맞아야 한다.
3. exports파일을 수정한 후에는 반드시 `exportfs -ra` 명령으로 수정된 내용을 확인하거나 서버를 리부트 해야 한다.

8.3 fh_verify errors on server while invoking X on the workstation

다음의 메시지가 서버에도 나올 것이다.

```
fh_verify: dev/tty2 permission failure, acc=8, error=30
fh_verify: dev/tty0 permission failure, acc=8, error=30
```

이는 X 서버가 디바이스 파일의 권한을 바꾸려고 시도하기 때문이다. 루트 파일시스템이 읽기 전용으로 마운트 되기 때문에 벌어지는 일이다. 사실, X 서버가 이러한 일을 할 이유는 없다. 따라서 이러한 일을 하지 않도록 패치한 X 서버를 LTSP에서 다운받아서 설치하면 된다.

8.4 XDMCP fatal error: Manager unwilling Host unwilling

```
Fatal server error:
XDMCP fatal error: Manager unwilling Host unwilling

when reporting a problem related to a server crash, please send
the full server output, not just the last messages

INIT: Id "2" respawning too fast: disabled for 5 minutes
INIT: no more processes left in this runlevel
```

이 에러는 /etc/X11/xdm/Xaccess에 항목 수정이 빠져서 그런것이다. 이 파일은 원격 워크스테이션이 xdm에 접속할 수 있는가를 제어하는 것이며, 레드햇 6.0의 경우에는 아래의 줄을 추가해 줘야 하며 6.1의 경우에는 아래의 줄이 이미 있으므로 다만 #기호만 제거하면 된다.

```
# *                                         #any host can get a login window
```

그 후에는 물론 xdm을 아래의 명령으로 재 실행 해야 한다.

```
killall -HUP xdm
```

8.5 다른 모든 경우

그래도 워크스테이션이 제대로 작동하지 않는다면 LTSP의 뉴스그룹을 이용해 보길 권한다. 도와줄 사람들이 있을 것이며 나 또한 뉴스그룹의 내용을 계속 지켜보도록 노력하는 중이다.

제 9 절 Additional references

1. *Linux Terminal Server Project (LTSP) home page* <http://www.ltsp.org>

2. *Diskless-Nodes HOW-TO document for Linux* <http://metalab.unc.edu/Linux/HOWTO/Diskless-HOWTO.html>

3. *Etherboot Home Page* <http://www.slug.org.au/etherboot/>

4. *XFree86-Video-Timings-HOWTO* <http://metalab.unc.edu/Linux/HOWTO/XFree86-Video-Timings-HOWTO.html>

5. *DisklessWorkstations.com home page* <http://www.DisklessWorkstations.com>