

## 〈報告〉

## gurb を利用したクライアント OS のサーバによる制御

西村 英俊\*・奥野 浩\*\*

## Selection of OS controlled by server with grub

Hidetoshi NISHIMURA\* and Hiroshi OKUNO\*\*

## 1. はじめに

現在、順天堂大学さくらキャンパス計算機実習室におけるコンピュータの利用は、OSとしてWindowsXPを使った作業に限定されている。一方、最近では、Windowsだけではなく、Linuxなどを利用してクラスター等を組み科学計算などに利用する例が見られる。この場合、電源投入時に、起動するOSを指定しなければならない。しかし、実習で利用する場合、利用しないOSを起動されると実習の妨げになる。このため、ここでは、サーバから、端末の起動するOSを制御することを考えた。また、これを実現することにより、Windowsによる実習だけではなく、Linuxによる実習も可能になる。

従来より、ブートローダ grub<sup>1)</sup>を使った起動OSの制御は、メニュー形式のものがよく知られている。この場合、起動OSは端末の利用者によって選択される。サーバによる制御としては、サーバからのOSのカーネルファイルのTFTP転送によるものがある<sup>2)</sup>。この場合には、比較的大きなファイルであるOSのカーネルが転送されるため、多数の端末を同時に起動するには、ネットワークに大きな負荷がかかる。そこで、ここで

は、カーネルファイルは端末のローカルのハードディスク内に用意し、サイズの小さいgrubの設定ファイルをサーバからTFTP転送することによって起動OSをコントロールする方法を開発した。

grubは、機能の高いブートローダとして知られている。デフォルト以外のOSを起動するには、一端コマンドモードに入り変更することができた。コマンドモードでは、起動パーティションを選んだり、別の設定ファイルに変更しさまざまなコントロールができるようになっている。ただし、この場合には、端末を利用するユーザの介在が必要になる。ここでは、grubのソースファイルに変更を加え、設定ファイルによって、別の設定ファイルに変更できるようにし、ユーザの操作なしで、サーバから起動OSを指定できるようにした。

## 1. grub ソースの変更の内容

grub ソースで、grub 上利用可能なコマンドを定義している。コマンドは利用可能な状況（コマンドライン、ヘルプ、設定ファイル）を定義されている。ここでは、設定ファイルを変更するコマンド「configfile」を本来コマンドラインとヘルプ中でしか利用できなかったものを設定ファイルでも利用できるようにする。

この処理によって、サーバに書き込まれた設定ファイルを読みに行くことによって制御できる。

この処理によって、端末上におかれた設定ファ

\* スポーツ健康科学部 マネージメント学科  
School of Health and Sports Sciences Juntendo University

\*\* 医学部  
School of Medicine Juntendo University

イルから、サーバ上の設定ファイルを読みに行くようにできる。

### 3. 実習室の実際の設定

#### 1) grub の作成とインストール

端末上で Linux を起動して操作する。

grub のソースファイルを <http://www.gnu.org/software/grub/> よりダウンロードする。

次の定義ブロックに変更を加える。

```
static struct builtin builtin_configfile =
{
    "configfile",
    configfile_func,
    BUILTIN_CMDLINE | BUILTIN_HELP_
LIST,
    "configfile FILE",
    "Load FILE as the configuration file."
};
```

5 行目を次のように変更する。

```
BUILTIN_CMDLINE | BUILTIN_MENU |
BUILTIN_HELP_LIST,
```

編集終了後、コンパイルし、インストールする。ただし、実習室の端末の場合、更にネットワークカードのパッチをあてる必要があった。

```
#patch -p1 < ../grub-0.97-tg3.patch
#./configure --enable-diskless --enable-tg3
#make
#make install
#/usr/local/sbin/grub-install hd0

2) 端末の設定
/boot/grub/grub.conf を次のように作成する。
default=0
timeout=0
hiddenmenu
ifconfig --address=端末の IP アドレス--server
=サーバの IP アドレス
configfile=(nd)/grub/サーバ上の設定ファイル
名
title Fedora Core (2.6.18-1.2798.fc6)
    root (hd0,1)
    kernel /vmlinuz-2.6.18-1.2798.fc6 ro
```

```
root=/dev/VolGroup00/LogVol00 rhgb quiet
initrd /initrd-2.6.18-1.2798.fc6.img
title WindowsXP
    rootnoverify (hd0,0)
    chainloader +1
```

サーバ上の設定ファイルを読み込みようになってるので、ここでは、timeout の行から、configfile の行までが意味を持つ。サーバ上の設定ファイル名は自由につけられるので、端末毎に設定を変えることも、すべての端末に同じ設定をすることもできる。

#### 3) サーバの設定

設定ファイルの転送には tftp を利用している。その設定として tftpd の inetd.conf への登録とディレクトリ/tftpboot を tftpd の root に指定した。

各端末用の grub の設定ファイルとして次のファイルを作成した。

```
/tftpboot/grub/サーバ上の設定ファイル
default=0
timeout=0
title Fedora Core (2.6.18-1.2798.fc6)
    root (hd0,1)
    kernel /vmlinuz-2.6.18-1.2798.fc6 ro root=
/dev/VolGroup00/LogVol00 rhgb quiet
    initrd /initrd-2.6.18-1.2798.fc6.img
title WindowsXP
    rootnoverify (hd0,0)
    chainloader +1
```

このファイルの default= の値によって、0 の場合 FedoraCore が、1 の場合 WindowsXP が起動するように制御できる。

### 4. 考 察

1) 端末の電源投入時にサーバと通信できない場合、端末の起動が不完全な状態で停止する。この場合にも自動的に回避ができることが望ましいが、セキュリティ上の問題はない。

2) 端末上の grub の設定ファイルの timeout = 0 としたため、grub のコマンドモードに入れないので、ユーザの不用意な操作による誤作動を防げた。

3) grub.confでは、Linuxのカーネルファイルを指定しているので、残念ながら今のところLinuxのカーネルが自動更新された場合、手動でサーバ上の設定ファイルを変更する必要がある。これに対応することが今後の課題のひとつとなる。

## 文 献

- 1) GNU.ORG <http://www.gnu.org/software/grub/>
- 2) Patrick J. LoPresti Using PXE+GRUB to boot Linux on a Thinkpad T20 <http://www.uwsg.iu.edu/hypermil/linux/net/0207.3/0030.html>

(平成19年10月12日 受付)  
(平成19年11月28日 受付)