

一种用于网络计算的可定制启动协议*

周悦芝⁺, 张尧学, 王 勇

(清华大学 计算机科学与技术系, 北京 100084)

A Customizable Boot Protocol for Network Computing

ZHOU Yue-Zhi⁺, ZHANG Yao-Xue, WANG Yong

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

+Corresponding author: Phn: 86-10-62788076, Fax: 86-10-62782118, E-mail: zyz00@mails.tsinghua.edu.cn

<http://www.cs.tsinghua.edu.cn>

Received 2002-01-23; Accepted 2002-04-11

Zhou YZ, Zhang YX, Wang Y. A customizable boot protocol for network computing. *Journal of Software*, 2003,14(3):538-546.

Abstract: With the development of computer and network technologies and emerging of diverse types of terminal, the ability of network computing to tailor applications to the capabilities of heterogeneous client devices will realize its full potential at last. Most of the existing terminals first start OS locally and then load network-computing software. However, this booting way can't satisfy the diversity of terminals and the increasing of complexity of network-computing software. A reliable, secure, effective and customizable remote boot protocol named as NCBP (network-based client boot protocol) is presented in this paper, which is hoped to increase the flexibility of existing network computing terminals' booting process and thus increase the flexibility and availability of network computing. NCBP uses extended DHCP (dynamic host configuration protocol) to acquire local identifier, then loads a batch script language environment of MBatch (MenuBatch) by using secure APTP (active program transfer protocol). When executing the loaded MBatch script, NCBP allows users to select which OS to be loaded and then load the corresponding OS image in a customizable way. NCBP can be used for remote boot of network computers, PCs, and digital appliances.

Key words: network computing; remote boot; digital appliance; network computer

摘 要: 随着计算机网络技术的发展和各种终端设备的出现,网络计算技术能够根据终端类型定制应用程序的特点将全面实现其潜力.然而,现有的终端设备先从本地启动操作系统,再启动网络计算软件的单一的启动方式已不能满足终端设备的多样性和不断复杂的网络计算软件的需求.为了提高终端设备启动方式的灵活性和可用性,提出了一种可靠、安全、高效并可定制的远程启动协议 NCBP(network-based client boot protocol).NCBP 通过扩展 DHCP(dynamic host configuration protocol)协议获得本机标识,再利用安全的 APTP(active program

* Supported by the National High Technology Development 863 Program of China under Grant No.863-306-ZD05-01 (国家 863 高科技发展计划); the National Grand Fundamental Research 973 Program of China under Grant No.G1998030409 (国家重点基础研究发展规划(973))

第一作者简介: 周悦芝(1975—),男,湖南衡阳人,博士生,主要研究领域为网络计算,家庭网络.

transfer protocol)协议加载 MBatch(MenuBatch)脚本执行环境.通过对 MBatch 脚本的解释执行,并根据用户的选择从服务器上加载相应的操作系统内核,从而实现终端设备的可定制远程启动.NCBP 可用于网络环境下的网络计算机、PC 机以及数字家电设备的远程启动.

关键词: 网络计算;远程启动;数字家电;网络计算机

中图分类号: TP393 文献标识码: A

计算技术在过去的 40 多年里得到了很大的发展,经历了从大型主机(main frame)到客户/服务器(client/server),再到 B/S(browser/server)这 3 层结构网络计算的演变^[1].

在网络计算中,造价低的客户终端按需从集中服务器上下载并执行程序,如网络计算机;或客户终端请求在服务器上执行应用程序并得到结果,如 Windows 终端.与传统的计算模式相比,网络计算具有简单、安全、支持移动计算、总体造价低、可用于电子商务等特点,并且能够根据终端的不同种类提供可定制的计算服务^[2].随着计算能力的不断增强以及可嵌入在芯片中微处理器的出现,瘦计算机(thin computer)将得到极大的发展和应用.

目前,大部分网络终端设备是在加电后从本地启动操作系统,然后再启动网络计算软件,如 JVM、中间件、GUI、WWW 浏览器等^[3-5].虽然这种方式在一定程度上满足了终端设备发展多样性的需求,并在服务级别上可根据终端类型对应用程序进行剪裁,但它的灵活性不能适应终端设备数量增多以及网络计算软件复杂度增加及其不断升级的要求.基于网络的远程启动协议为提高终端设备的灵活性和可用性提供了一种新思路.然而,现有的远程启动协议(比如 RPL 和 PXE)都是专门为 PC 设计的.它们在实用性、可靠性和可定制性方面都难以满足目前各种网络终端设备启动的需要.

本文针对这一问题,提出了一种可靠、安全、高效并可定制的新型远程启动协议:NCBP(network-based client boot protocol).NCBP 首先利用扩展的 DHCP(dynamic host configuration protocol)协议获得本机的标识.此后利用安全的 APTP(active program transfer protocol)协议下载并执行 PreOS 镜像.在 PreOS 环境下,NCBP 通过解释执行 MBatch(MenuBatch)脚本,按需从服务器上下载并执行操作系统内核镜像,从而可为用户提供可定制的启动服务.因此,NCBP 克服了目前网络终端启动方式在灵活性方面的不足.尤其在家庭网络^[6]环境下,NCBP 为解决各种数字家电设备的启动和应用程序的按需加载问题提供了一条有效的解决途径.

本文第 1 节介绍 NCBP 协议过程及其与之相关的技术问题.第 2 节给出 NCBP 的一个原型系统,并对其性能进行了分析.在第 3 节中,我们把 NCBP 与目前用于 PC 远程启动的方法进行了比较.最后是本文的结论.

1 NCBP 启动协议

NCBP 是基于客户/服务器模型的,分为客户端和服务端两部分.客户端是指需要进行远程启动的网络设备,而服务器是指提供远程启动服务的计算机.启动过程主要就是客户端根据协议与服务器进行交互,按需下载并执行程序的过程.在服务器端,NCBP 提供由扩展 DHCP 服务和 APTP 服务组成的启动服务.启动服务协议的控制流程由 DHCP 服务和 APTP 服务及其相关的代码(主要是 MBatch 脚本)共同实现.

在客户端,由驻留在 ROM 中的 ICode(initial code)代码提供 NCBP 协议客户端的功能.ICode 代码主要由 4 部分组成:网卡驱动以及相关协议;扩展的 DHCP 客户端代码;APTP 客户端代码;PreOS 加载器.ICode 代码在机器加电后获得机器的控制权,并按照 NCBP 协议过程下载并执行程序.

NCBP 客户端利用 DHCP 协议来获得本机的网络标识.为了防止网络上多个 DHCP 服务器对 NCBP 的影响,NCBP 对 DHCP 进行了扩展,增加了防止干扰的鉴别选项.同时,也增加了对多播(multicast)支持的扩展选项.在局域网的环境下,使用多播技术能够提高系统的并发能力和负载能力.为了保证安全性,NCBP 采用较安全和可靠的 APTP 协议来传输代码文件,而不是采取传统的基于 TFTP 的方法^[7].与扩展 DHCP 相适应,APTP 也支持多播.

与普通的远程启动过程不同,NCBP 还增加了一个脚本执行环境 PreOS 来与用户进行交互,从而实现对各操作系统定制加载的支持.

1.1 NCBP协议过程

计算机加电启动后,将执行 POST 过程.在 POST 过程的最后一步,操作系统启动中断将找出已存储在 CMOS 中的 IPL(initial program load)设备,并执行 IPL 设备提供的加载代码以加载操作系统代码^[8,9].在远程启动过程中,一般使用 PnP 可选 ROM 作为 IPL 设备.在原型系统中,NCBP 也是使用在系统中增加一个可选 ROM 的办法.在机器加电自检结束后,系统控制权将交给可选 ROM 中的 ICode 代码.

ICode 代码启动以后,就开始执行 NCBP 协议,该协议过程如图 1 所示,各步的含义说明如下:

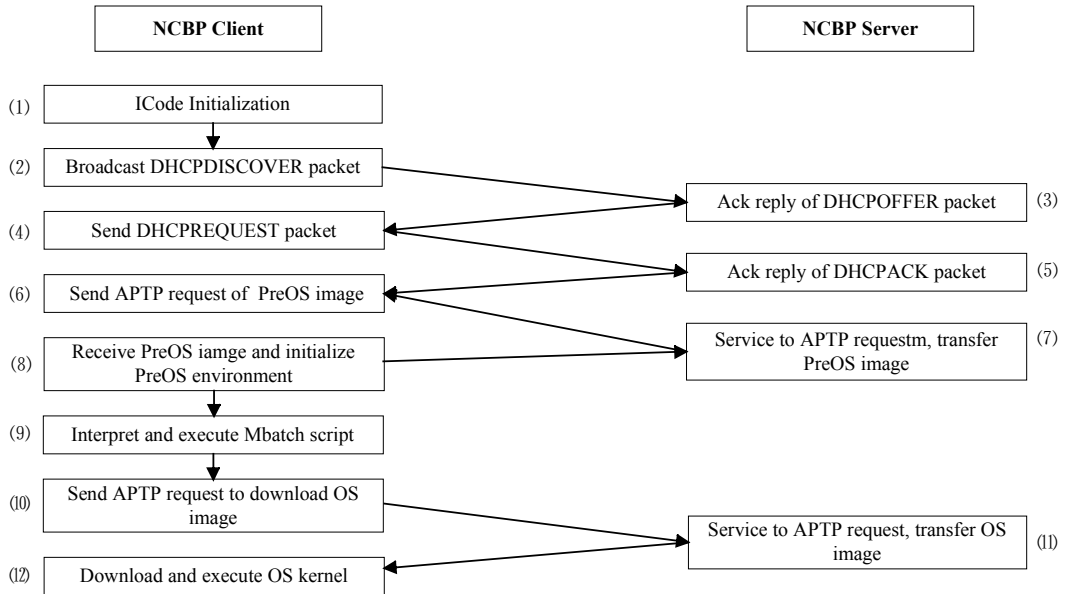


Fig.1 The protocol process of NCBP

图 1 NCBP 协议过程

(1) 客户端本地初始化,该过程包括检查 ICode 代码的合法性和完整性;初始化本地运行环境和网络环境等;

(2) 客户端驱动网卡工作,并广播 DHCPDISCOVER 消息分组,请求本机的 IP 地址、网关地址、启动服务器地址、PreOS 镜像等标识;

(3) 服务器接收到 DHCPDISCOVER 消息分组后,检查是否是 DHCP 扩展协议的消息分组(关于 NCBP 对 DHCP 的扩展将在第 1.2 节详细讨论),如果是,则向客户端发送应答的 DHCP OFFER 消息分组;

(4) 客户端收到 DHCP OFFER 消息分组后,验证该消息分组是否是 DHCP 扩展协议的消息分组,如果该消息分组是 DHCP 扩展协议的消息分组,则记录有关 DHCP 协议参数、服务器地址、PreOS 镜像等内容,同时向服务器发送 DHCPREQUEST 消息分组,并等待服务器的同意应答;

(5) 服务器收到 DHCPREQUEST 消息分组后,向客户端发送 DHCPACK 消息分组作为应答;

(6) 客户端收到 DHCPACK 消息分组后,记录相关的内容;同时根据返回的内容向服务器发出请求下载 PreOS 镜像的传输请求;

(7) 服务器接收到客户端程序传输服务的请求后,利用 APTP 协议(APTP 协议参见第 1.3 节)的单播或多播方式向客户端传送 PreOS 镜像(PreOS 镜像及下面提及的操作系统镜像格式参见第 1.5 节);

(8) 客户端接收完 PreOS 镜像,同时启动执行 PreOS 镜像,初始化 PreOS 环境;

(9) 客户端 PreOS 环境解释执行 MBatch 脚本,在此阶段,用户可使用输入设备选择需要加载的操作系统;

(10) 客户端根据用户选择的操作系统内核镜像的程序名称、路径或摘要,向服务器发出请求下载操作系统内核镜像的程序传输请求;

(11) 服务器接收到客户端发出的程序传输服务请求之后,利用 APTP 协议的单播或多播方式向客户端传送操作系统的内核镜像;

(12) 客户端接收操作系统内核镜像并将其存储在本地存储器中,客户端再加载执行相应的操作系统内核镜像,至此,操作系统加载完毕。

1.2 NCBP对DHCP的扩展定义

NCBP 协议利用 IETF RFC2131 DHCP^[10]协议为客户端提供网络标识.为了提供对多播的支持并保证 NCBP 协议的可靠性,NCBP 对 DHCP 的扩展选项^[11]进行了扩充和修订。

DHCP 消息分组的格式如图 2 所示。

op (1)	htype (1)	hlen (1)	hops (1)
xid (4)			
secs (2)		flags (2)	
ciaddr (4)			
yiaddr (4)			
siaddr (4)			
giaddr (4)			
chaddr (16)			
sname (64)			
file (128)			
options (variable)			

Fig.2 The format of DHCP message

图 2 DHCP 消息分组

DHCP 主要有两种类型的消息,它是由 op 域来标志的,一种是 BOOTREQUEST 请求消息分组,一种是 BOOTREPLY 应答消息分组.对其他域的解释,因篇幅限制,不再加以说明.NCBP 方法利用 DHCP 协议来配置客户端的主机参数,并利用 file 域传递需要下载的 PreOS 镜像。

当网络上存在多个 DHCP 服务器的时候,根据 DHCP 协议的规定,客户端将随机选择一个 DHCP 服务器为之服务.在 NCBP 中,这种情况将会引起混乱.为此,NCBP 对 DHCP 的可选参数域扩展了一些定义.通过识别 60 号和 43 号标签的值,NCBP 可避免同一网络上存在多个 DHCP 服务时引起混乱的情况,并有效防止从另外的 DHCP 服务器上下载代码的行为,增强了健壮性和安全性.同时,为了支持 APTP 协议的多播要求,NCBP 也扩展定义了其他一些选项.NCBP 对 DHCP 的扩展定义见表 1。

Table 1 The extended definition of DHCP options in NCBP

表 1 NCBP 对 DHCP 选项的扩展定义

Tag name	Tag #	Length	Description
Vendor class identifier	60	18	This option indicates that the DHCP packet is an extended DHCP packet. It must be "NCBP-EDHCPSEClient".
Vendor specific information	43	8	Vendor information, defined with "03:05:08:00:00:00:ff:ff". With tag 60 and 43 it can make NCBP only reply to an extended DHCP packet.
Message type	53	1	Packet type,1=DHCPDISCOVER,2=DHCPOFFER,3=DHCPREQUEST,4=DHCPDECLINE,5=DHCPACK,6=DHCPNAK,7=DHCPRELEASE,8=DHCPINFORM
PreOS parameters	165	Indefinite	Parameters should be passed to the PreOS environment.
APTP multicast address	138	4	The multicast IP address when APTP server transfers files.
APTP client port	139	2	The UDP port that should be listened by APTP clients when using multicast method.
APTP server port	140	2	The UDP port that should be listened by APTP servers when using multicast method.
APTP waiting time	141	2	Number of seconds an APTP client must wait before restarting an APTP request.
APTP listening delay	142	2	Number of seconds delayed when an APTP server receives an APTP request.
File digest	143	Indefinite	The next file name of digest of program needed to transfer. The digest is produced by MD5. This can guarantee the correct transfer of file and protect the programs from maliciously tampering.

1.3 APTP协议

NCBP 协议利用 APTP 协议从服务器上下载代码.APTP 协议是一个简单、安全的程序传输协议.APTP 协议简单,便于实现,它能完全放置在一个 ROM 中.APTP 不提供像 FTP 协议那样的目录和程序浏览服务.APTP 采用基于 MD5^[12]单向哈希函数作摘要的方法来命名程序,能够保证程序被正确地传送到客户端,并能防止程序在传输过程中被恶意篡改.APTP 协议不支持写操作,避免对服务器实施写操作而产生的安全问题.APTP 在 UDP 协议的上层运行,采用超时重传技术来保证可靠的数据传输.APTP 没有连接的建立过程,是一个无状态协议.APTP 支持多播方式,能够以多播方式传送数据,从而大大减少并发启动时对网络带宽的冲击.

APTP 采用固定大小的数据块进行传输.目前采用的数据块大小是 512 字节.

APTP 协议的报文格式如图 3 所示.

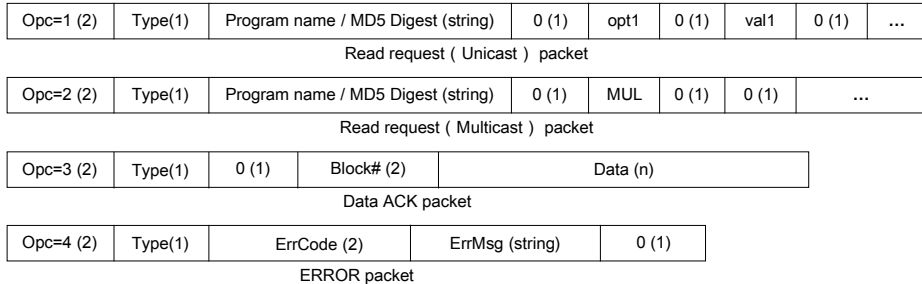


Fig.3 The format of APTP packets

图 3 APTP 协议报文格式

APTP 共有 3 种协议报文,分别为“读请求”(read request,简称 RRQ)、“数据应答”(DATA acknowledgement,简称 DATA ACK)、“错误”(error,简称 ERROR).分别说明如下:

“读请求”报文:Opc 表示报文的种类.为了方便起见,读请求报文的 Opc 代码有两个:1 和 2.它们分别表示普通读和多播读.图 3 中括号中的数字为该域的长度(以字节为单位).Type 为 0 时,表示使用程序名称来标识需要传送的程序,为 1 时表示使用 MD5 摘要来标识需要传输的程序.opt1 和 val1 分别表示第 1 个选项及其取值,依次类推.当选项为‘MUL’时,表示请求多播传送数据报文.

“数据应答”报文:其中块号从 1 开始连续编号.每块的大小为 512KB.

“错误”报文:当传输过程发生错误时,服务器将向客户端发送错误报文.客户端收到错误报文之后,立即终止传输过程,并向用户报告.

在 NCBP 协议中,如果在 DHCPPOFFER 和 DHCPACK 消息分组的选项中包含了多播参数,并且客户端也愿意使用多播方式来接收数据报文,则客户端可以接收多播的 APTP 数据报文.APTP 客户端以多播方式来接收数据报文的过程分为 3 个阶段:多播监听、多播打开和多播关闭.监听过程中,客户端通过监听 APTP 多播端口来收集有效的 APTP 数据报文.如果能收集到所有需要的数据报文,则不进入打开阶段.当在监听阶段没有收集到数据报文时,则进入打开阶段,使用单播(unicast)方式从服务器请求数据.如果又收到多播数据,则又可转入监听阶段.

1.4 MBatch语言

MBatch 是一个脚本解释执行语言.NCBP 通过对脚本的解释,可以和用户进行交互,从而利用 APTP 协议下载执行不同的操作系统镜像.

MBatch 是基于命令行的.命令后面是命令的参数,其格式为

Command parameter1,parameter2,...

MBatch 语言的特点是:

- 命令以行为单位进行解析,行与行之间用换行或回车隔离;
- 行的最大长度为 255 个字符;

- 关键字或变量大小写不敏感;
- 双引号(")为字符串的分隔符;
- \$开始的字符串表示变量;
- 转义符的使用:\b 后退;\n 换行;\r 回车;\tTab;\nnn 用值为八进制数 nnn 的 ASCII 码代替;\X 表示不在上述的字符中的字符 X.

在 MBatch 语言中,主要的表达式就是字符串表达式和数值表达式.其中数值表达式只能做整数的+,-,*,/,等运算.而且各个运算符之间没有优先级的区别,用括号来表示运算的顺序,例如:((4*6)+8).

在 MBatch 语言中,最主要的命令有延迟命令、监控命令、控制命令、输入输出命令等等.其中最主要的命令就是内核镜像加载命令:LoadImage "kernel image name or digest".其中内核镜像的命名规则如下:

文件名为字符串.它们必须用双引号引起来,文件名是大小写敏感的,可用转义符来表示特殊的字符.

如果文件名不是以/开始,则默认为从/aptpboot 目录下寻找.如果文件名中出现冒号,则冒号前的部分将被解释为计算机名.如果用摘要命名镜像,则必须以//开始.后面的摘要部分使用 Base64 编码^[13]来进行传输.服务器根据摘要决定传输哪个镜像程序.

1.5 NCBP对内核镜像格式的规定

为了加载不同操作系统的内核镜像,必须对内核镜像的格式加以规定.使用统一的内核镜像的格式,加载器就能透明地加载各种操作系统的内核.

NCBP 中的内核镜像格式如图 4 所示,有关的含义说明如下:

Image Identification:内核镜像格式的标识,目前的取值为 05h,0ah,0ah,05h,作为内核镜像的标识.

Location Address:内核镜像在内存中的装载地址.

Execute Address:该地址是内核镜像的首次执行的地址,当内核镜像加载程序成功地装载到 Location Address 之后,跳转到 Execute Address 处执行,从而释放系统的控制权.

Image Length:内核镜像的长度,该长度是指内核镜像解压前的长度.

Memory Length:内核镜像解压后所占用的内存长度.

Data Segment:镜像执行后的真正数据区.

op (1)	htype (1)	hlen (1)	hops (1)
xid (4)			
secs (2)		flags (2)	
ciaddr (4)			
yiaddr (4)			
siaddr (4)			
giaddr (4)			
chaddr (16)			
sname (64)			
file (128)			
options (variable)			

Fig.4 The format of kernel image

图 4 内核镜像格式

2 实现

2.1 原型系统

NCBP 原型系统以 X86 体系结构作为原型系统的硬件平台,以 Windows 98,Windows 2000 和 Linux 作为实施的对象.NCBP 原型系统已经在清华大学研制开发的网络终端——SOPCA 中得到了实现和应用.图 5 是实现 NCBP 原型系统的示意图.网络终端首先与启动服务器交互进行远程启动,然后再从应用服务器上下载应用程序

序,并在本地执行.

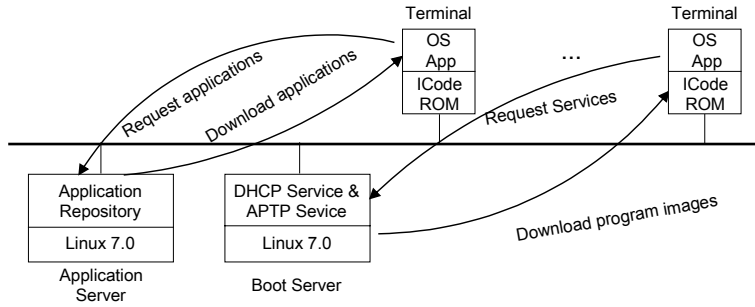


Fig.5 The prototype of NCBP

图 5 NCBP 原型系统示意图

其中 SOPCA 网络终端是基于 X86 体系结构的,其配置为 Cyrix Media GX,64M RAM,AMD AM79C901A 1M/10M Ethernet PHY 网卡.在每个 SOPCA 终端的主板上都安装了一个 ROM.该 ROM 中烧制了 ICode 代码.启动服务器的配置为 Pentium III-866,128M RAM,100M 以太网卡.应用服务器的配置为 Pentium III-866,256M RAM,100M 以太网卡.整个系统由 D-Link DES-1024 交换机连接成局域网.

启动服务器上面启动了 DHCP 服务和 APTP 服务.DHCP 服务的配置文件示例如下(为节省篇幅,该示例将几行用分号隔开写在一行):

```
{
#NCBP-specific options
option dhcp-class-identifier "NCBP-EDHCPSEClient"; option vendor-encapsulated-options 03:05:08:00:00:ff:ff;
#global options
option subnet-mask 255.255.255.0; default-lease-time -1;
# NCBP APTP epecific options
option 138 255.255.255.0; option 139 1005; option 140 1006; option 141 8;
#subnet specific options
subnet 192.168.0.0 netmask 255.255.255.0 {
    option routers 192.168.0.2; option dynamic-bootp 192.168.0.100 192.168.0.180;
    filename "aptboot/Preos.image";
}
}
```

其中#号后面的部分是注释.Preos.image 是 PreOS 镜像名称,它被执行后形成 MBatch 的脚本解释环境.在 PreOS 镜像中还包括一个 MBatch 脚本.

下面是一个用上下箭头选择的菜单脚本,两个菜单分别启动 Windows 98 和 Linux 系统(为节省篇幅,该脚本将几行用分号隔开写在一行).

```
Set MenuNum=2; Set MenuItem={"Starting Windows 98","Starting Linux"};
Set CurrentItem=1;
:menu;
Set exitmenu=" "; Print "Boot selection";
Set i=1;
:loop;
Set j=(i-1 ); Print "$MenuItem"{$j}; Print "\r\n"; Set I=(I+1);
If $I<=$MenuNum goto loop
GetKey Key;
```

```

If "$Key"=="UpKey" if $CurrentItem>1 Set Current=($CurrentItem-1); # Up key
If "$Key"=="DnKey" if $CurrentItem<$MenuNum Set Current=($CurrentItem+1); #Down key
if "$Key"=="r" Set ExitMenu="OK"; #Enter key;
if "$ExitMenu"!="OK" goto menu; if $CurrentItem===1 goto a1; if $CurrentItem===2 goto a2;
:a1;
LoadImage "192.168.0.72:/tftpboot/WINSSOP.SYS";
:a2;
LoadImage "192.168.0.72:/tftpboot/Linux.image";

```

2.2 性能测试

由于网络终端启动的过程很复杂,难于进行程序模拟或仿真.因此为了测试 NCBP 的并发和时间性能,必须进行实际的实验.为此,我们搭建了由 50 台 SOPCA 终端、一台应用服务器、一台启动服务器和交换机组成的实验环境(实验环境的配置情况参见第 2.1 节).我们测试了 NCBP 在单播和多播两种方式之下的启动时间.为了对比,我们还测试了单机本地启动以及基于 RPL 方法的启动时间.所测得的结果如图 6 所示.图 6 中的值为 10 次测试的平均值.

从图 6 可以看出,使用网络启动比单机的本地启动要慢一些.这是显然的,因为网络启动需要进行网络的协议交互过程.与 RPL 方法比较,NCBP 方法速度稍慢.一方面是由于安全方面的开销,另一方面是由于 NCBP 对脚本的执行需要一定的时间.从图 6 中还可以看出,在数量较多的情况下,使用多播方式具有较好的效果.

要进一步提高 NCBP 的性能,就必须优化安全算法以及脚本执行的环境和速度.

3 相关工作

Novell 公司是较早提出远程启动技术的公司之一.它提出了一种用于远程启动的 RPL(remote program load)协议^[14].RPL 只能用于 Novell 网络环境,而 Novell 网络的建立和维护命令非常多,很复杂.因此,RPL 在取得良好的时间性能的同时,在通用性和可靠性方面还存在着不足.同时,RPL 只提供对单一操作系统的支持,因此当操作系统升级后,RPL 芯片也需要升级,难于满足用户定制启动过程的要求.为了解决这些问题,Intel 公司在 1998 年提出了一种称为 PXE(preboot execution environment)的远程启动技术^[15].PXE 规范提出了一种新的基于 TCP/IP 协议簇的远程启动方法.PXE 建立在业已成为工业标准的 Internet 技术规范之上,因此它的稳定性比较好.但 PXE 对多操作系统的启动支持也很弱,它没有提出多操作系统启动的方法.在安全性上,它采用 TFTP 来传输文件,安全性较差.

上面两种方法都是针对 PC 的启动提出来的,不太适用于网络计算的环境.而本文提出的网络环境下终端设备远程启动的新方法,安全、可靠,速度较快,并且能支持对多操作系统的启动,是一种较好的远程启动方法.表 2 总结了 NCBP 与传统的 RPL 以及 PXE 远程启动方法的比较情况.

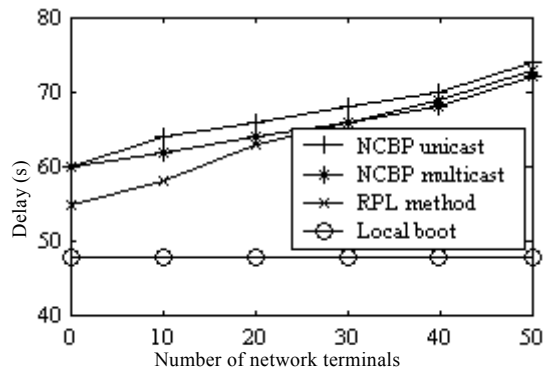


Fig.6 The performance testing of NCBP

图 6 NCBP 性能测试

Table 2 The comparison of NCBP, RPL and PXE**表 2** NCBP, RPL 和 PXE 的比较

Methods	RPL	PXE	NCBP
Security	Low	Low	High
Reliability	Low	High	Very high
Generality	Low	High	High
Response time	Short	Short	Short
Multicast support	No	Yes	Yes
Complexity	Difficult	Simply	Simply
User interface	Complex	Easy	Easy
Efficiency	High	High	High
Boot time	Short	Short	A little long
TCP/IP support	No	Yes	Yes
Customization support	No	No	Yes

4 结 论

本文针对目前网络终端在启动方法上灵活性和可用性的不足,提出了一种可靠、安全、高效并可定制的远程启动协议 NCBP. NCBP 通过对 DHCP 进行扩展定义,并利用 APTP 下载程序,保证了可靠性和安全性. APTP 协议的简单性和无状态性,保证了网络终端在启动过程中可自动恢复,保证了该协议方法的健壮性. 通过对 MBatch 脚本的解释执行,实现了对多操作系统定制加载的支持. 该方法可用于网络环境下的网络计算机、PC 机以及数字家电设备的启动.

NCBP 方法已经在清华大学研制开发的网络终端 SOPCA 中得以实现和应用. 目前我们正在做进一步提高其性能的工作.

References:

- [1] Noguchi S. Network computing technology and information society toward 21st century. In: Makoto T, ed. Proceedings of the 7th International Conference on Parallel and Distributed Systems. Los Alamitos, CA: Institute of Electrical and Electronic Engineers, Inc., 2000. 3~9.
- [2] Revett M, Boyd I, Stephens C. Network computing: a tutorial review. IEE Electronics & Communication Engineering Journal, 2001, 13(1):5~15.
- [3] Redford J, Taunton M. Acorn's technology for network computing. In: Bob W, ed. Proceedings of the IEEE COMPCON'97. Los Alamitos, CA: IEEE Computer Society Press, 1997. 124~129.
- [4] Richardson T, Stafford-Fraser Q, Wood KR, Hopper A. Virtual network computing. IEEE Internet Computing, 1998,2(1):33~38.
- [5] Hasedawa A, Nakajima T. A user interface system for home appliance with virtual network computing. In: Makoto T, ed. Proceedings of the 21st International Conference on Distributed Computing Systems Workshop. Los Alamitos, CA: Institute of Electrical and Electronic Engineers, Inc., 2001. 229~234.
- [6] Home Wireless Networks, Inc. Home networking. <http://www.iec.org/online/tutorials/>.
- [7] Sollins K. The TFTP protocol (Revision 2). RFC 1350, 1992.
- [8] Compaq Computer Corporation, Phoenix Technologies Ltd., Intel Corporation. BIOS boot specification. Version 1.01, 1996. <http://www.phoenix.com/techs/specs.html>.
- [9] Compaq Computer Corporation, Phoenix Technologies Ltd., Intel Corporation. Plug and play BIOS specification. Version 1.0A, 1994. <http://www.phoenix.com/techs/specs.html>.
- [10] Droms R. Dynamic host configuration protocol. RFC 2131, 1997.
- [11] Alexander S, Droms R. DHCP options and BOOTP vendor extensions. RFC 2132, 1997.
- [12] Rivest R. The MD5 message-digest algorithm. RFC 1321, 1992.
- [13] Borenstein N, Freed N. MIME (multipurpose internet mail extensions) part one: mechanisms for specifying and describing the format of Internet message bodies. RFC 1521, 1993.
- [14] Novell Inc. NetWare 4 feature guide. 1998. <http://www.novell.com/documentation/lg/nw42/pdfdoc/newftenu.pdf>, December.
- [15] Preboot Execution Environment (PXE) Specification. Version 2.2, 1998. <http://developer.intel.com/ial/wfm/wfmspecs.htm>.