Systems Integration

Making Many Protocols and Networks Interoperate

Nick Urbanik <nicku(at)nicku.org>

A computing department

The Problem

- There are so many Operating Systems:
  - Microsoft:
    - Windows 9x
    - Windows CE
    - Windows NT
    - Windows 2000
    - Windows XP
    - Windows 2003
  - Linux, from various vendors
  - Unix
    - Sun Solaris
    - AIX from IBM
    - HPUX from Hewlett Packard
  - Apple: OS X, and the previous MAC OS
  - Cisco: IOS, various others (e.g., for Catalyst switches, PIX firewall,...)

So Many Protocols

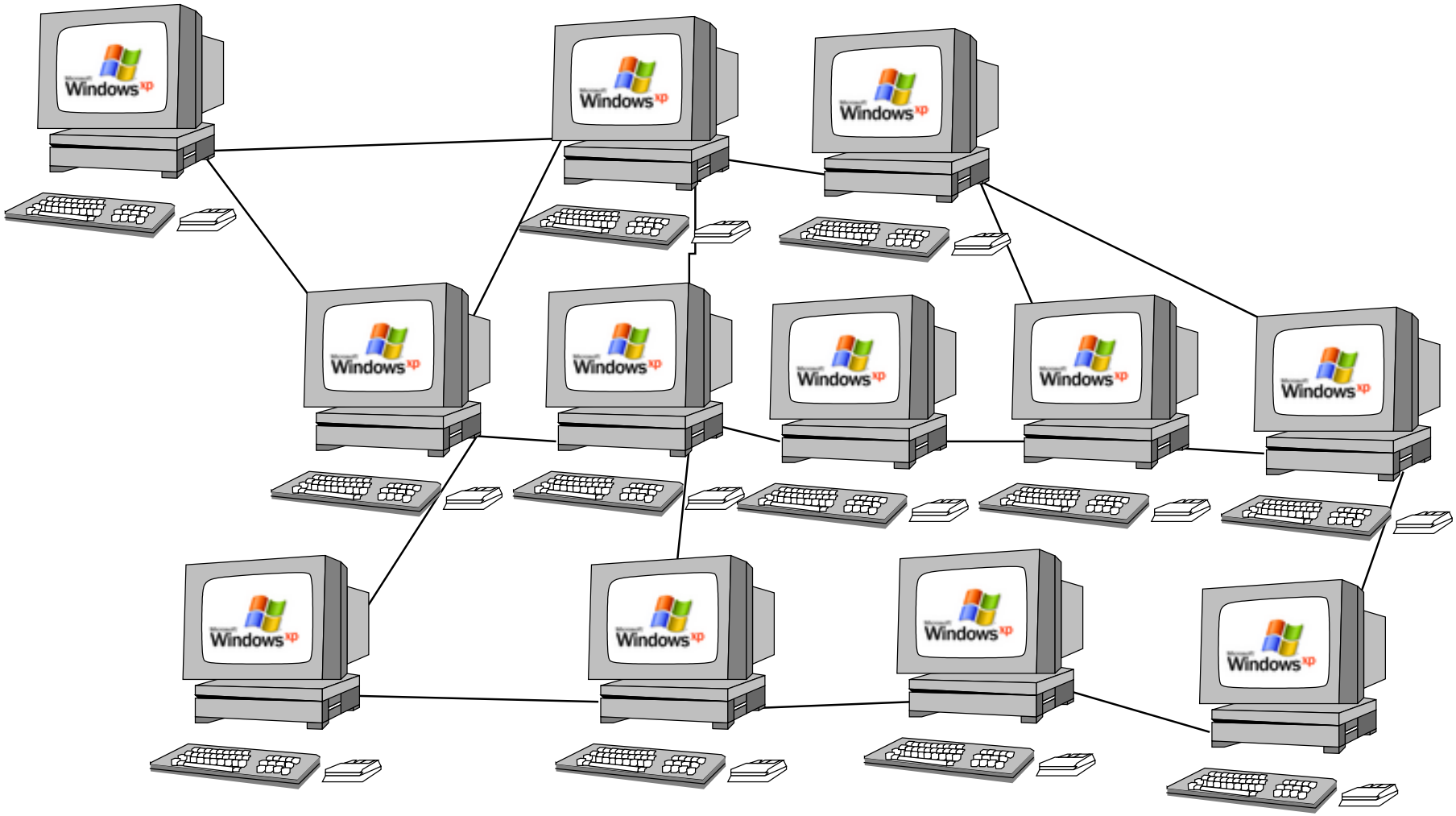
- Standard Protocols:
  - TCP/IP
  - SMTP
  - HTTP
  - FTP
  - SSH
  - LDAP
  - telnet
  - DHCP
  - DNS
  - PPP
  - XML
  - SNMP
  - SOAP
  - RADIUS
  - TLS
  - NTP
  - NNTP
  - POP3
  - IMAP
  - Kerberos
- Proprietary Protocols:
  - NetBIOS (Microsoft file sharing)
  - Active Directory
  - WINS
  - Novell Directory Services
  - Database services from many providers
  - Appletalk
  - Novell IPX

Many Different Hardware Platforms

- So many computing platforms:
  - IBM mainframes
  - handheld devices
  - RAID systems
  - Cluster systems
  - PCs
  - Notebooks
- So many different communication media:
  - CAT5 network cabling
  - Wireless LANs
  - Gigabit Ethernet
  - Optic fibre

Monoclulture

- Why not just buy from one supplier?



How to Make them Work Together?

- Free Software works hard to *include* as many protocols, file systems, vendor products, hardware platforms as possible
- Solutions are cross-platform
  - Java, Perl, Python, C, C++
  - Linux (runs on tiny handhelds to huge mainframes, almost everything between)
  - Samba for integration with Windows Networks
  - *Netatalk* for integration with *Appletalk* (for older Macintosh OSs)
  - Apache Web server runs on almost any platform
  - OpenLDAP for directory services

Prefer Open Protocols

- Use *open* and standard protocols as much as possible
- Avoid “locking in” to proprietary solutions *where a good open solution exists*

Samba

- Implements Microsoft's SMB protocol
- SMB = Symmetric Message Block, gave project its name
- achieved through reverse engineering Microsoft's proprietary protocols (no help from MS, but hindrance)
- good reputation for stability and performance outperforming MS servers in both respects
- Current production version supports use as a Wondows NT compatible server (file sharing, printing, support for network browsing)
- Runs on many platforms, including very powerful Solaris machines
  - Most powerful windows servers run Solaris, not Microsoft software!

Samba 2.2.x

- The release provided with current Linux systems
- Works as an NT 4 compatible PDC
- *Winbind* (part of samba) allows Linux and Unix machines to join a Windows Domain

Limitations of Samba 2.2.x

- Does not support Active Directory in the way that a Windows 2000 server does
- Samba 2.2 can neither be a Backup Domain Controller (BDC) nor use one
- User information stored on a Samba PDC is not as complete as that stored on a Windows PDC
- Samba obeys Linux group file access permissions on the PDC, but it does not tell the client machine about it properly. Group file permissions are hard to set from a client.
- Full support for ACLs (access control lists) depends on applying a patch to the Linux kernel and recompiling the kernel, or waiting till the Linux 2.6.x kernel is released

Samba Version 3 (alpha release)

- Currently used in some commercial systems, but documentation not complete
- See
  `http://us1.samba.org/samba/ftp/alpha/WHATSNEW.txt`
- Supports Active Directory: a Samba 3 server can join an ADS realm as a member server and authenticate users using LDAP/kerberos
- Supports migrating from a Windows NT 4 domain
- Supports trust relationships with Windows NT domain controllers

Using LDAP to Authenticate

- LDAP = Lightwieght Directory Protocol
- A network directory
- Can be used to store user accounts, group information, and information about network devices
- Any application can be made to authenticate against LDAP
- Samba can use LDAP to authenticate against
- Can build an infrastructure that uses LDAP to authenticate *everything*

A Case Study: ICT

- We use OpenLDAP to hold all user accounts (thousands), both full-time, part-time and staff
- All Linux systems authenticate against this directory
- Maintained only by me as a (very!) part time activity
- I did the programming in my spare time
- All home directories are on the same server

Next Step

- The next steps are:
  - Provide better hardware
    - We have an Adaptec clustering system with a dedicated shared RAID system
  - Will run Red Hat Advanced Server
  - Provide home directories via NFS (as currently do) and via samba
  - Provide support for old Macintosh clients via Netatalk
  - Provide a single sign-on for all services for all students and staff
- Time frame: by next academic year.