Migrating Proprietary Software-Based Services to Open Platforms

A White Paper
April 2003

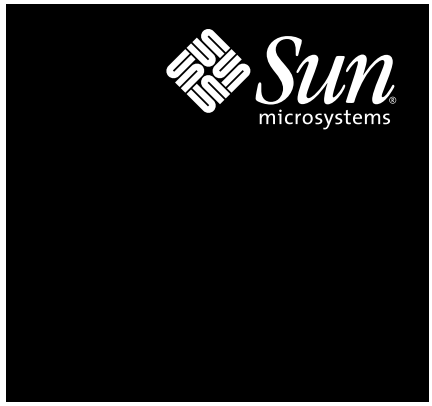


Table of Contents

Chapter 1

Introduction

In today's challenging business climate, IT organizations everywhere are compelled to find ways to lower costs and increase the efficiency of their operations. With Microsoft's new "Software Assurance" program being the only avenue available for companies to upgrade their Microsoft server products, some organizations must first upgrade to current versions in order to qualify for the program. Once in the program, companies must continue to pay up-front for upgrades that provide software features they may or may not need. Indeed, as yahoo.com reports (March 24, 2003), a recent Sunbelt and Yankee group survey of 1,000 technology managers finds that 60 percent are paying more for their software under LIcensing 6.0 than they were in the previous program. Costs such as these have brought software licensing fees into focus for many companies — but they aren't the only costs under scrutiny.

Security vulnerabilities cost companies administrator time to apply frequent patches — and the costs of not keeping up with security patches include lost revenue due to e-commerce sites being unavailable and the costly and time-consuming effort required to recover from damage done by viruses and worms. The problems with Microsoft Internet Information Service (IIS) Web server became so acute in 2001 that Gartner Group made the unprecedented recommendation for companies to immediately consider alternatives to IIS (CNET News, September 25, 2001). Running complex software like Microsoft Exchange often requires application-specific experts to be on staff, experts that command high salaries yet often lack the flexibility to handle other administrative chores across an IT organization.

All of these factors have lead IT organizations to consider migrating to open platforms as a way to reduce costs and increase the efficiency of their staff. When organizations use open platforms like Linux and the Solaris™ Operating System, they participate in a rich, vibrant

marketplace where vendors and open-source developers create products that are based on open standards, competing to produce the best implementation at the lowest cost. The result is that open-source tools like the Apache Web server can be used on platforms from a variety of vendors — for example, Apache runs on the Linux operating system, supported by the Sun LX50 server and the Sun Fire™ Blade Platform; it runs on the Solaris Operating System, supported by the Sun LX50 server as well as Sun's UltraSPARC® processor-based servers ranging from the low-cost, single-processor Sun Fire V100 server to the 106-processor Sun Fire 15K server. If for some reason organizations decide to use platforms from other sources, they have other vendors to choose from — one of the many reasons why Sun works so hard to stay ahead of the competition.

Using open-source products like Samba for identity management, file and printer sharing; sendmail and open-source IMAP and POP servers for e-mail services; and Apache for Web services lowers licensing costs to zero, and can reduce administrative costs as well. Most seasoned UNIX® system administrators are capable of managing the entire range of services hosted on a UNIX system, and can cross-over from Linux to the Solaris Operating System and even to systems from Sun's competitors, resulting in a more flexible and adaptable work force. Because of the degree to which open-source developers are motivated to produce solid software products, security issues tend to arise less frequently — indeed, this is one of the reasons that Gartner advised using Apache over IIS in September 2001.

The purpose of this white paper is to help IT organizations assess the amount of effort involved in migrating network-based services including identity, file/print sharing, e-mail, and Web hosting from proprietary Microsoft-based approaches to open platforms. The paper gives guidelines on how to assess the level of effort required, strategies for making the transition as smooth as possible, and step-by-step approaches for executing the transition. Although it is focused on migrating to zero licensing cost, open-source software like Samba, sendmail, and Apache, organizations more comfortable with commercial-grade software will find Sun's rich complement of Sun™ Open Net Environment (Sun™ ONE) products like Sun ONE Directory Server, Sun ONE Messaging Server, and Sun ONE Web Server referenced throughout the paper. As organizations make the choice to migrate to open-platform systems, the Sun Services organization is standing by with the knowledge and experience needed to help make the transition as easy as possible.

Chapter 2

Migrating User Identity Services to Open Platforms

User identity management is required to support user-specific functions including e-mail, file, and print services, and therefore user identity services is one of the first issues to consider when migrating to open platforms. Many IT organizations managing user identity with Microsoft Windows Domains are at a crossroads. With the push to move to Microsoft Active directory, they are rightfully assessing whether to risk further proprietary lock-in, licensing, and administration costs by re-architecting their networks to use Active Directory. The alternative is to take the opportunity to re-host on open platforms that can support and interoperate with Windows clients.

Samba has been in use since 1992, and is further described at www.samba.org.

Samba, one of the most stable and mature open-source products available, can be used to support user identity services for Windows, Linux, and UNIX environments. Samba open-source file and print services software can replace a Microsoft Windows Domain controller today, and in a future release will support Microsoft Active Directory-based clients. Samba is included with all commercial Linux distributions, and is included with every copy of Sun's Solaris™ Operating System. Migrating to Samba is compelling not only for the Linux/UNIX/Windows interoperability it supports, but also for the choice to run it on reliable entry-level X86-architecture systems like Sun's dual-processor LX50 server, and also on Sun's highly-scalable UltraSPARC processor-equipped servers, offering customers the ability to scale from 1 to 128 processors per server.

Migration Goals

This chapter discusses the issues involved with migrating user identity services from Microsoft Windows-based servers to open platforms like Linux and UNIX running open-source Samba

software. Its goal is to help IT managers asses the level of effort required to migrate to open platforms, expose the issues that must be considered, and to give suggestions on how to manage the transition.

Migrating user identity services is a logical first step in the process of migrating file, print, e-mail, and Web services to open platforms. The goal of this phase is to provide user identity services through a Samba-based directory so that:

- A heterogeneous network of Microsoft Windows, Linux, and UNIX systems can be supported
- The fact that identity services is supported on a non-Microsoft platform is transparent to Microsoft Windows client systems
- The new server can be phased in gradually, or all at once.

This chapter addresses the issue of how to replace a server acting as a Microsoft Windows Domain controller; the ability to replace an Active Directory server is currently being developed by the Samba team, and a future update will address how to accomplish this migration.

Know Your Environment

The first — and most important — step in any migration effort is to assess what services are supported on the server that is to be replaced, and how it is used in its particular environment. Knowing what services are running on an existing server minimizes the risk of any surprises during the migration process itself.

In a typical Microsoft Windows network, user identity services are supported by one or more Domain controllers that provide a network-accessible directory of both users and systems that are allowed to join a given domain. Microsoft Windows systems must first authenticate to join a domain, and once it is part of the domain, users defined as members of the domain may log into the system. Because of the additional complexity and increased licensing costs for deploying Microsoft Active Directory, many organizations have delayed the move. This chapter assumes that Windows Domains, not Active Directory is to be replaced.

Depending on the organization's size, Windows Domain services are sometimes hosted on servers that also provide network naming, file, and print services. Windows Domains is indeed a distinct service that can be migrated independently to an open platform, with other services migrated to the same or other open-platform server at any time in the future. Take note of the directory-related services that are supported, including the following:

- Windows Domains, not Active Directory
- Windows Internet Names Server (WINS), which is a TCP/IP-hosted version of Microsoft's NetBEUI name service. Most Windows Domain controllers also support WINS, and it's usually best to migrate both services at the same time.
- Dynamic Host Configuration Protocol (DHCP), which dynamically assigns Internet Protocol (IP) addresses to hosts at boot time. Some Windows Domain controllers are configured also to act as DHCP servers. DHCP is a standard service supported on Linux and UNIX systems, and can be configured independently of Windows Domains.
- Domain Name Services (DNS), which is the Internet-standard name-to-IP address mapping mechanism. DNS servers are standard components of both Linux and UNIX systems, and can be configured independently of Windows Domains.

When migrating services from a Windows Domain controller that also supports WINS, it is best to migrate both at the same time. If the goal is to immediately de-commission the Microsoft

Windows server, and it also provides DHCP and/or DNS services, these should be migrated as well. If immediate de-commissioning of the Windows server is not a goal, services such as DHCP and DNS can be migrated to Linux or UNIX servers at any time. Both DHCP and DNS are easy to configure in Linux and UNIX environments, and these configurations are not discussed in this paper.

Gathering Information

There are several pieces of information that need to be gathered in order to further qualify the services for which the server is being used. Sun recommends gathering information on each host and user in the domain and using it as a reference for this, and future migration efforts. The information that should be gathered is summarized in Table 2-1, and includes the following:

- *Host-Specific Information*
  Gather a list of names that includes all hosts that are part of the domain to be supported by open platforms. For each host, gather information on whether its addressing is static or dynamic, and whether DHCP configuration is used. If a host uses static IP addressing, then its WINS server address may have to be changed when and if WINS services are hosted on a new server. For those clients using DHCP, all WINS server address changes can be done on the DHCP server, with no changes required on the clients. Where DHCP servers are used, you may wish to configure DHCP services on the new Linux or UNIX server.
- *User-Specific Information*
  A list of all user names in the domain are needed. For each user, it's useful to know whether they have a local or roaming profile, and in the case of a roaming profile, where it points to. This information will be used when file sharing services are set up, and gathering this data now simplifies future migration work.
- *Server-Specific Information*
  Does the server to be replaced support any services beyond its function as a Windows Domain controller, WINS, DHCP, or DNS server? If the server also supports file and print sharing, for example, those services may also be migrated to the new server using information from future installments of this white paper.

Table 2-1: Having complete information about the server, its clients, and how they are configured and used helps when making migration decisions.

	Information to Gather	Remarks
Host Information	Host name	A list of all hosts in the domain
	Static or Dynamic IP Addressing?	If the host is configured using static IP addresses, you may need to set up a new WINS server address on that host
	DHCP configuration used?	You may need to set up a DHCP server on the new Linux/UNIX server

	Information to Gather	Remarks
User Information	User Name	A list of all user names in use for the domain
	Local Profile or Roaming Profile?	This information will be used when file sharing is configured. If a roaming profile is used, note where it is located
Server Information	List of services supported	If more than Windows Domains, WINS, DNS, and DHCP, then other services will have to be described as defined in future installments of this white paper

Deciding on the Migration Approach

There are basically two approaches to how to conduct the migration. One is to configure a new open platform server with its own host name and IP address, and point clients to them through the use of WINS or DHCP services. The other approach is to configure the new server with a temporary host name and IP address, turning off the Microsoft Windows server and rebooting the open platform server with the same host name and IP address as the server that is to be replaced.

The first approach allows the migration to proceed incrementally, testing services and migrating clients as desired. If, however, the information-gathering phase finds a significant number of hosts using static IP addressing (which entails a manually configured WINS server address) it may be easier to take the second approach. Bringing up a new server with the same IP address as the old server is completely transparent to users and should require no re-configuration of clients. The disadvantage of this approach is that it precludes experimenting with the new server until it is re-booted as the primary Domain controller.

Regardless of the approach taken for IP addressing, WINS services should be the last ones configured on the new server before the old one is de-commissioned.

Introducing Samba

For more than 10 years, Samba has been used to provide file and printer sharing services to Windows systems from non-Windows systems including Linux and UNIX. Samba is one of the most tried-and-true open-source software packages. Samba is so universal that it is provided with every commercial Linux distribution, and is included as part of the Bonus Software CD shipped with every copy of the Solaris Operating System. Samba uses NetBIOS over TCP/IP to deliver four major Server Message Block (SMB) services:

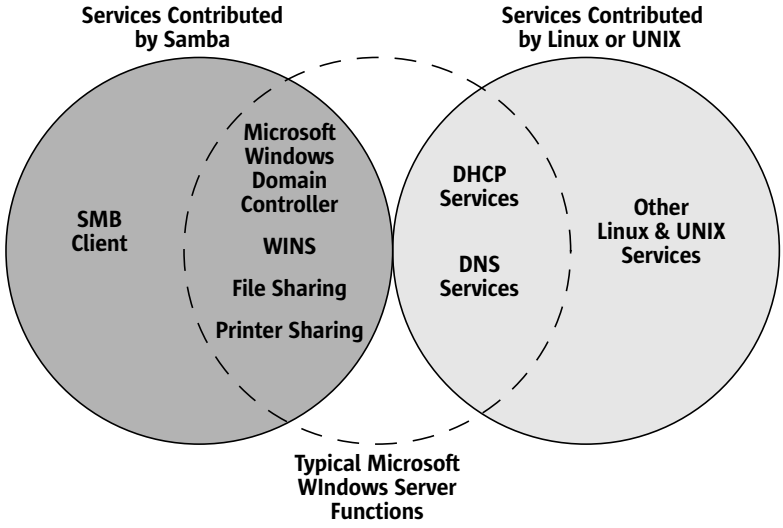
- File and print sharing are the most important services provided by Samba, and is the original purpose for which the software was written.
- Directory services, including authentication and authorization, can replace the functions of a Windows Domain controller, and this chapter focuses on this aspect of Samba.
- Name mapping from Windows names into IP addresses that is typically done by a WINS server.
- Service announcement, which allows Samba-based file servers to appear in a WIndows client's Network Neighborhood.

Wherever you get your copy of Samba, check with www.samba.org to be sure you have the latest version.

More Samba migration how-to information can be found at: samba.idealx.org. The unofficial Samba how-to site is: hr.uoregon.edu/davidrl/samba.html.

Between the services provided by Samba and those provided by Linux and UNIX operating systems, most of the services typically provided by a Microsoft Windows NT or 2000 server can be supported using open platforms (Figure 2-1). Samba of course provides file, print, WINS, and Windows Domain services. When DHCP or DNS are required, these services can also be hosted on the open platform using native operating system facilities. One interesting Samba facility that doesn't contribute to replacing Windows server functions is its SMB Client, which allows other Linux and UNIX systems to act as file-sharing clients to SMB servers including Samba and Microsoft Windows servers.

Figure 2-1: Between Samba and Linux/UNIX operating systems, most functions typically provided by Microsoft Windows servers can be hosted on open platforms.



Samba cannot currently act as a back-up domain controller, nor can it support Microsoft Windows-based back-up Domain controllers; Samba must be the definitive source for domain authentication. For organizations needing multiple domain controllers for greater availability, the open-source `rdist` utility can be used to keep an up-to-date duplicate of Samba's configuration files on a back-up server which can be brought in the event of a failure.

For information about swat and other graphical user interfaces for configuring Samba, please refer to: http://www.samba.org/samba/GUI.

Managing Samba services is simplified through the use of a Graphical User Interface (GUI) known as `swat`. This interface supports basic configuration of features including identity, file, and print services. It also supports viewing status of running Samba services.

Configuring the Open Platform Server

With intimate knowledge of the Microsoft Windows server and how it is used in the network environment, and some background information on configuring Samba, the process of configuring the open platform server to replace Windows Domain controller functions is straightforward.

The pwdump utility is described at: www.samba.org/samba/ftp/pwdump.

An enhanced version, pwdump2, is described at: razor.bindview.com/tools/desc/pwdump2_readme.html.

1. Get the list of Microsoft Windows userids/passwords and host names/passwords to incorporate into the Samba environment. Samba makes this process easy by providing a utility, `pwdump`, that can extract both userids and passwords into the Samba directory service. This is important because it enables the open platform server to authenticate users against their Windows passwords without requiring any clear-text transmission or storage of user passwords. For networks with large numbers of servers, or for retrieving Active Directory user names and passwords, consider the `pwdump2` utility.
2. Create a corresponding Linux or UNIX account for each user and host name. Samba uses the native user and group ID numbers on the server, and it maps between the Windows and open

platform identities based on user and host *names* rather than numbers. There are command-line interfaces for creating users and their home directories in both Linux and the Solaris Operating System.

3. Running the `pwdump` utility on the Windows server as an Administrator will list the encrypted password for each account name on the system. This resulting list can be used as the Samba server's `/etc/smbpasswd` file, providing a transparent way to move user passwords from one system to the other.
4. To configure the open platform server as the primary Domain controller:
   - Configure the Samba software with the domain name and turn on the Domain controller functionality. Do not activate the Samba server until you perform the next step
   - Demote the Windows Domain controller. With Windows 2000 this can be done via Microsoft's `dcpromo.exe` utility. If the Windows Domain controller is not providing any other critical services, it can be removed from the network at this point. If it is still providing services such as file/print sharing or Web services, you may wish to test your new Domain controller by temporarily disconnecting the old server.
   - Turn on the Samba service

For information on how to demote your Windows Domain controller, please refer to: support.microsoft.com/default.aspx?scid=kb;en-us;307304.

When this procedure is completed, the open platform server is running with two parallel directory services: Samba and its copy of Windows passwords; UNIX userid names and passwords stored in `/etc/password` and `/etc/shadow`. As you begin to allow users to access non-Samba functions (like e-mail) on open platform servers, the lack of consistent passwords between the two environments could present a problem. Fortunately, Pluggable Authentication Modules (PAM) can be used to overcome the password issue. PAM is an innovation in the Solaris Operating System that has been created for the Linux environment by the open-source Linux-PAM project.

Learn about Solaris Operating System PAM modules by visiting: www.sun.com/software/solaris/pam.

Learn more about Linux PAM modules at: www.kernel.org/pub/linux/libs/pam.

PAM modules can be used to implement a wide range of authentication mechanisms on open-platform systems ranging from simple user name/password mechanisms to more complex mechanisms using smart cards and biometric scanners. Administrators can determine which PAM modules are required for authentication through the configuration file `/etc/pam.conf`.

Samba software provides a PAM module that can be used to authenticate open platform users specified in `/etc/passwd` with their Windows passwords located in `/etc/smbpasswd`, so users can continue to use their existing Windows passwords regardless of the services they use (Figure 2-2). In addition, Samba includes mechanisms for changing passwords on the open platform server.

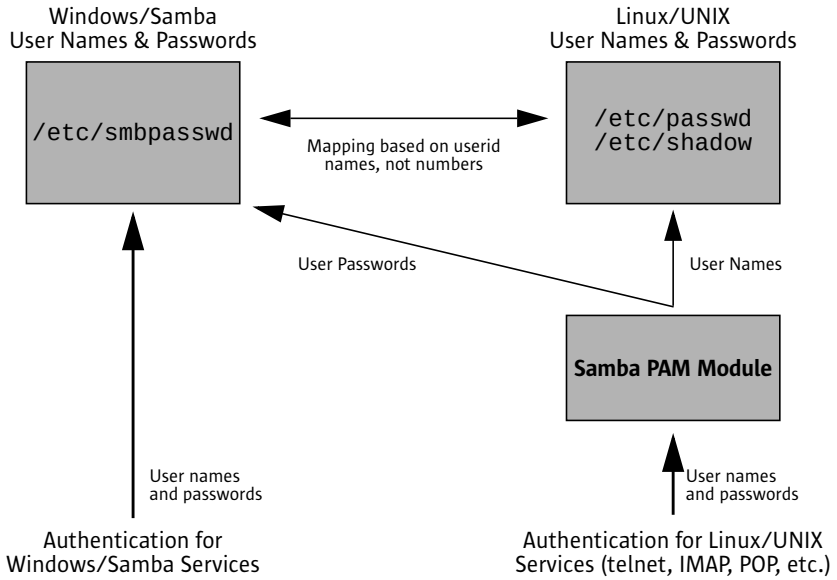


Figure 2-2: User authentication can be unified between Windows and open platform environments through the use of a PAM module provided with Samba software.

Summary

Migrating identity management services from Microsoft Windows-based servers to open platforms running Linux or the Solaris Operating System is the first step towards independence from per-user licensing fees and proprietary lock-in, while maintaining complete interoperability and transparency between environments.

Visit www.sun.com/software/product_family/middleware to learn more about Sun ONE Directory Server.

Completing this first migration step provides an open platform server that acts as a Microsoft Windows Primary Domain controller whose Windows and Linux/UNIX passwords remain and are unchanged regardless of the services being accessed, and through no effort on the user's part. This step opens the door to providing other user-specific services that include file and print sharing; e-mail servers with POP and IMAP services; and Web servers with user-specific features. As the benefits of using Linux and the Solaris Operating System are proven to your organization, and more Microsoft Windows systems are replaced with open platforms, true open standards-based user identity services can be deployed organization-wide. Building on the accomplishments of this first step, future organization-wide identity management can be implemented using Lightweight Directory Access Protocol (LDAP), for example using the Sun ONE Directory Server included with every copy of the Solaris Operating System.

Chapter 3

Migrating File and Printer Sharing to Open Platforms

One of the easiest and most compelling functions to migrate from Microsoft Windows-based servers to open platforms is file and printer sharing. Samba, one of the most stable and mature open-source software products available, was originally written to provide SMB-based file and print-sharing services from UNIX platforms to Microsoft Windows-based clients. With more than ten years of use providing file-sharing services, Samba has evolved into an ever more easy-to-configure, secure, and stable software product that runs on Linux and UNIX platforms from a wide range of vendors including Sun's Solaris Operating System.

When the maturity and stability of Samba is paired with flexible, scalable server technologies from Sun Microsystems, the combination becomes even more compelling. Small- and medium-sized businesses and workgroups can deploy rack-mountable systems from Sun including the Sun LX50 server with up to two Intel Pentium III processors, running either Linux or the Solaris Operating System. As business requirements grow, customers can vertically scale up to larger servers — like the UltraSPARC processor-based server product line from Sun that ranges from the single-processor Sun Fire™ V100 server to the 106-processor Sun Fire 15K system. All of Sun's UltraSPARC processor-based servers run the rock-solid Solaris Operating System, which enable vertical scaling without the need to re-architect services as underlying platforms change. File sharing requires flexible, scalable storage systems, and Sun's external storage options include the low-end, three-disk, 1U high Netra™ st D130 storage device; the 3U, 12-disk Sun StorEdge™ 3300 RAID Array (expandable to up to 36, 73 GB drives); and the Sun StorEdge 9900 series systems with storage capacity of up to 74.7 TB. Open standards-based storage systems deliver pay-as-you-grow

scalability; customers can begin with flexible rack-mount storage systems to which they can add more drives as needed, without having to re-host entire systems.

Flexible, scalable, servers, storage, and operating system software are only the beginning of the list of reasons why organizations choose to host file-sharing services on open platforms. Some of the many additional reasons include:

- Open platforms have no licensing fees, especially the per-user fees that hamper organizations using proprietary software from Microsoft. With open systems, there's no risk of running out of per-user licenses as business grows.
- Hosting file-sharing services on open platforms gives organizations more flexibility: by supporting NFS, SMB, and CIFS protocols, open platform-based file servers actively enable heterogeneous, cross-platform file sharing.
- User home directories are automatically positioned to their own home directories on the server, with no additional navigation required. This simultaneously enhances both user experience and network security.
- Cross-platform support is extended through the use of macros that can map files on the server differently depending on the client platform — simplifying the sometimes confusing web of workstation-specific drivers, applications, and support files that are often located on file servers.
- Software product installation can be simplified by using server-mounted CD images. Administrators can copy entire software product install CD contents to a file, mount that file as a file system, and share the file system through Samba, making Microsoft Windows-based clients operate as if they're accessing a CD drive over the network.
- File system performance and functionality can be tuned by using software and hardware-based RAID approaches. Customers can choose from a wide range of file system products that best meet their needs, including EXT2, EXT3, XFS, and ReiserFS formats supported by Linux; and native UFS, journaling, and third-party VERITAS VxFs systems supported by the Solaris Operating System. For business-critical data, remote replication products like Sun StorEdge Network Data Replicator (SNDR) can transparently mirror file system contents using standard TCP/IP networking technologies. Volume managers like Solaris Volume Manager software can help administrators add disk storage and grow file systems so that the only noticeable result is that existing volumes have more storage associated with them.
- Samba is known for its attention to security, and has a reputation for being among the first organizations to discover and report security flaws to Microsoft. Of course Samba software patches are quickly made available on *www.samba.org* by a large contingent of open-source software developers.
- Hosting file services on open platforms makes a rich set of command-line tools available to administrators. For example, the `find` command can be used to quickly find and delete files whose names end in `.tmp` and which are more than 1 month old.

For both Linux and the Solaris Operating System, ISO 9660 images contained in a file can be mounted and shared using the loop-back file system and the `lofiadm` command (used in the Solaris Operating System) followed by the `mount` command (used in both environments).

Migration Goals

This chapter discusses the issues to which organizations should pay attention when migrating their Microsoft Windows-based file and print server to open platforms like Linux and the Solaris Operating System. Its goal is to help IT managers asses the level of effort required to migrate to open platforms, expose the issues that must be considered, and to give suggestions on how to manage the transition.

Refer to "Migrating User Identity Services to Open Platforms" on page 3.

Migrating file/print sharing services should be done after considering the implications of user identity services that were discussed in the first chapter, although using Samba as a file server does not require all user identity services to be migrated to open platforms as a prerequisite.

Know Your Environment

One of the most important steps in any migration effort is to assess the way in which the existing server is being used so that when it is replaced, the exact same functions are supported on the new server. Making sure there are no gaps in service helps ensure a smooth and successful transition.

Setting up file-sharing services using Samba is one of the most straightforward of the migrations covered by this document. As a result, the most important aspects of the environment that need to be assessed are not the core file-sharing services, but some of the various ways that they are used.

Gathering Information

There are several pieces of information that can help the transition to the new server. Knowing how users access the server can help in the process of duplicating those services on the new system. Knowing how the server is used from its own perspective helps to determine whether all of the clients and their particular uses of the server are covered. And knowing the amount of storage used on the existing server helps to properly size the new server and its storage system layout. The information helpful in file and print sharing migration includes:

- *Roaming Profiles*

  Roaming profiles enable users in a Microsoft Windows domain to log onto any client system and have transparent access to the files in their home directory as if they were local. When user directory services were assessed as discussed in "Gathering Information" on page 5, notes were taken as to which users have roaming profiles set up. For each user, take note as to whether a roaming profile is established, and how it is mapped. Creating a table such as that illustrated in Table 3-1 helps to keep track of how each user accesses the server.

Table 3-1: Know what users have roaming profiles, and the locations to which they are mapped.

User	Roaming Profile?	Where it is Mapped?
User id in Windows domain	yes/no	Location on server

- *Server Shares*

  Now look at the server itself to see what shares are made available to clients, so they can be appropriately duplicated on the open platform. All shares should be duplicated except those ending in a dollar sign ($), with the exception of the special share `print$`. The `print$` share provides a location where platform-specific device drivers can be stored and automatically accessed by client systems.

  A table like that illustrated in Table 3-2 can be initialized using the output of the `net share` command, which lists all shares on the server. Fill in the rest of the table by assessing the amount of disk space currently used by each share. At the command line, use the `dir /s` command with the resource name as a parameter, or right-click on the resource in Windows Explorer and view properties.

Share Name	Resource	Disk Space	Remarks
Name of share	Resource Location, for example C:\office60	Current usage in gigabytes	Copy remarks field to replicate in the Samba environment

Table 3-2: Know how the existing server makes files available to users by listing all of the shares that it support.

- *Users, Groups, and Access Control Lists*
  Again on the server, evaluate what users are members of which groups so that group membership can be replicated on the new open platform server. The Microsoft Windows tool User Manager for Domains can be used to obtain this information. If the number of users is large, consider using a script to extract this information. Perl modules support scripts that can easily provide a list of users and group membership.

  A Perl module that can help with this process can be found at: search.cpan.org/author/GSAR/libwin32-0.191/NetAdmin/NetAdmin.pm.

  Most real-world use of Access Control Lists (ACLs) can be replicated using users and groups, and this migration step may provide a good opportunity to clean up the use of ACLs into more simple, easy-to-understand, and easy-to-maintain user/group structures. Remember that users can be members of multiple groups, which makes it easy to configure groups and group membership so that ACLs are unnecessary. ACLs can be replicated using Samba with both Linux and the Solaris Operating System, however this task goes beyond the scope of this document because it requires compiling additional Samba libraries that are often not included in the default distribution.

  A free utility to dump user, group, and ACL information can be found at: www.somarsoft.com.
- *Printer Shares*
  Assess what printer shares are made available from the existing server, and keep track of the share name, the resource location, and the remarks field.

Deciding on the Migration Approach

There are two key decisions to make regarding how to approach the actual migration. The first is whether to replace the existing server immediately or incrementally; the second is whether to use the existing Microsoft Windows domain controller or replace it as well.

Immediate or Incremental

One of the simplest ways to handle the actual migration is to replace the existing server all at once by bringing up the new, open platform-based server with the same IP address and host name as the former Microsoft Windows-based server. This approach has the advantage that no settings need to be changed on the client systems. It has the disadvantage that it is time consuming to fall back to the Microsoft Windows server, in particular because all files on the server will need to be re-copied to the open-platform server when the problem is resolved.

Using the incremental approach, the existing server can be replaced one file share at a time, gradually replacing the Microsoft Windows-based server. Because users and applications cannot be allowed to access or modify their files as they are moved from the old server to the new, organizations with large amounts of disk storage may prefer an incremental approach. The disadvantage of the incremental approach is that all clients accessing a particular file share must be modified to access the new server once files are moved.

Regardless of approach, setting up Microsoft Windows users with login scripts that cause file shares to be mapped to specific drives is one way to enforce consistency throughout the network environment.

Windows Domain Controller

If the migration described in the first chapter was followed and the new server acts as a Windows domain controller, then adding file-sharing services to the server is very straightforward. If the Windows domain controller function remains on a Microsoft Windows-based server, or is re-hosted to an open-platform server other than the one being configured for file sharing, some adjustments need to be made.

More details on PAM and how to configure it are contained in "Migrating User Identity Services to Open Platforms" on page 3.

If the file-sharing server is not the primary domain controller, user identities must be established in the Linux or Solaris Operating System. If the primary domain controller is hosted on an open platform, then open standards like Network Information Service (NIS) or Lightweight Directory Access Protocol (LDAP)-based can be used to extend user identities to the new server. If the primary domain controller remains on a Microsoft Windows-based server, Samba can be configured to access user and group names on the open-platform server and use the existing primary domain controller for password authentication. Samba does this through Pluggable Authentication Modules (PAM).

Configuring File Sharing

With knowledge of how the current file-sharing environment is configured, and decisions made on the migration strategy, several steps need to be taken to implement the strategy.

1. Set up the server with a temporary host name and IP address.
2. Set up user accounts in one of three ways:
   - Set up the primary domain controller on the new server as described in chapter 2.
   - Add user accounts manually with the `useradd` command and configure PAM to authenticate using a Microsoft Windows-based primary domain controller
   - Use NIS or LDAP to access userid and group information from another open-platform system
3. Configure groups and group membership by editing `/etc/group`.
4. Configure Samba using its graphical user interface called `swat`, or edit `/etc/smb.conf`:
   - Add share names that were found on the existing file server
   - Locate shares on the disk volumes that you've determined to have sufficient space (and room to grow)
   - Make sure that you have a backup strategy that covers the shared volumes
5. Turn on the Samba file-sharing service.
6. Copy files from the existing file server to the new open-platform file server. First quiesce applications and users so that no file-sharing activity is happening. As an alternative to manually copying files from the Microsoft Windows-based server to the Samba server, the `robocopy` utility can help copy files, correctly map long file names, and if used, maintain access-control lists.
7. Test this procedure with a few files first, and test critical applications to be sure that they can properly access files on the new server.
8. Change the host name and IP address to replace the existing file server if you have decided on that strategy and have no other services to migrate. For this case, roaming profiles will simply work without having to change locations on each client system. Otherwise, you will need to visit each client system and re-name all file shares to reference the new open-platform server.

For information about swat and other graphical user interfaces for configuring Samba, please refer to: http://www.samba.org/samba/GUI.

`Robocopy` is part of the Windows Resource Kit available from Microsoft.

Configuring Printer Sharing

Printer configuration is just as simple as file-sharing configuration. Once the printer is configured on the open-platform server, either as a local printer or one accessed over the network, Samba makes it available through a printer share to Microsoft Windows-based clients.

1. Use native operating system utilities to set up the printer. On most Linux systems, use the command `printconf -gui` for access to a graphical user interface, or use command-line tools. Other Linux distributions may have this command under a different name. In the Solaris Operating System, use the `admintool` command to set up the printer.
2. Configure the print share in Samba using `swat` or editing `/etc/smb.conf`.
3. Enable print drivers to be automatically installed onto client systems accessing the print share. Copy the appropriate printer drivers and capabilities files (ending in `.inf` and `.ppd`) to a known directory on the open-platform server. Then share that directory as `print$` from Samba.

Summary

Migrating file and printer sharing services from proprietary Microsoft Windows-based servers to open platforms running Linux or the Solaris Operating System is a significant step towards independence from per-user licensing fees and proprietary lock-in, while maintaining complete interoperability and transparency between environments. Once choices regarding user identity services are made, configuring Samba to provide SMB-based file and printer sharing services using open-platform servers is very straightforward. Once accomplished, the benefits include increased flexibility, scalability, security, reliability, and interoperability with other open-platform systems.

Chapter 4

Migrating E-Mail Services to Open Platforms

Maintaining a Microsoft Exchange e-mail server can be one of the more costly propositions that an IT department can face. The combination of server and per-client licensing fees drive up IT costs; software configuration is so complex that in-house Exchange experts are often needed; and regular maintenance is required to keep servers from crashing frequently, events that can grind business to a halt. More subtle is the proprietary lock-in that occurs when organizations use the non e-mail functions of Microsoft's Outlook client, tying functions like help desks and calendaring into the Exchange e-mail server. With a fixed client and server relationship established using proprietary, non-Internet standard protocols, it is sometimes difficult to leave Microsoft's lock-in behind.

Fortunately, there is a wealth of open-source software available that uses open standards including Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP), and Internet Message Access Protocol (IMAP) — in fact, because these standards are so ubiquitous, even the Microsoft Outlook client itself can be configured to operate using open standard, rather than Microsoft's proprietary Messaging Application Programming Interface (MAPI). Using open-source software and open standards for e-mail is attractive to IT organizations for several reasons:

- Reduced complexity through the use of a set of interoperable building blocks
- Lower administration costs because e-mail server experts are not necessary — any competent UNIX or Linux system administrator can configure and manage e-mail systems
- Higher reliability from service interruption, data loss, and intrusion

- Flexibility to integrate other open-source and commercial products into the mail system, including virus scanning and spam filtering
- Escape from proprietary lock-in, including the ability to use open-source and commercial products for secondary functions including calendaring and help-desk operations
- Lower or zero license fees; licensed Microsoft Outlook clients can still be used for e-mail access and personal calendaring and synchronization, or zero license fee, open-source client software can be used

Migration Goals

This chapter discusses the issues to which organizations should pay attention when migrating their Microsoft Exchange server functions to open platforms like Linux and the Solaris Operating System. Its goal is to help IT managers asses the level of effort required to migrate to open platforms, expose the issues that must be considered, and to give suggestions on how to manage the transition.

Migrating e-mail services to open platforms is different than other services like identity management and file/print sharing for a number of reasons. IT organizations can choose to replace the mail server, client software, or both. Whether existing Microsoft Outlook clients continue to be used or whether new software is deployed, each client system's preferences will need to be changed during the migration process to access the open-platform server using SMTP and IMAP or POP protocols, so an incremental, rather than an all-at-once migration is recommended.

This chapter provides direction for migrating mail *server* software from Microsoft Exchange to open platforms with a focus on e-mail hosting for a single domain. Alternatives are suggested for some of the other functions that might be managed by a Microsoft Exchange server, namely calendaring and Web-based e-mail.

Know Your Environment

One of the most important steps in any migration effort is to assess the way in which the existing server is being used so that when it is replaced, the same or similar functions are supported on the new server. Making sure there are no gaps in service helps ensure a smooth and successful transition. Because of the number of additional functions that can be integrated into a Microsoft Exchange server, it is especially important to take note of the additional services that must be replicated in the open-platform environment.

Gathering Information

Information needs to be gathered on how users access the server, what clients they use, how much storage they use, and what services are configured on the server.

- *Client Information*

  In order to know what services the open-platform server needs to provide, it's important to know the range of client software in use around the organization. For each user, find out what client software is used, and whether it can be configured to use either POP3, IMAP, or both protocols; keep a list of the clients and the protocols they support as illustrated in Table 4-1.

  IMAP is the preferred protocol for accessing e-mail on the server because it provides a list of e-mail messages to the client, enabling individual messages to be retrieved as requested, while retaining messages on the server. This provides better performance, leaves messages in a central location for protection via backup, and it enables users to access their mailboxes and

mail folders from multiple clients, making it easy to support telecommuters, traveling employees, and those needing Web-based access to their e-mail. In contrast, POP3 servers download all new messages to the client each time e-mail is accessed. If all clients are capable of communicating with the e-mail server via IMAP, then a POP3 server does not need to be supported.

This chapter assumes that all clients are capable of using SMTP for outgoing mail, which is almost surely the case with any e-mail client.

Table 4-1: Assess e-mail client software in use and whether it can be configured to use POP3, IMAP, and SMTP protocols.

Client Software	POP3 Capable?	IMAP Capable?	SMTP Capable?
Microsoft Outlook, Eudora, Mozilla, etc.	yes/no	yes/no	Almost surely yes

- *User Address Books*
  Assuming that most users have some form of address book for their commonly-used e-mail addresses, determine whether they are stored on the Exchange server or in the client. If they are stored on the server, some accommodation will need to be made to store the directories on the open-platform server. On Linux systems, packages like OpenLDAP can be used; in the Solaris Operating System, bundled Sun ONE Directory Server software provides LDAP functionality.

Two sources of information on LDAP servers include OpenLDAP (www.openldap.org) and Sun ONE Directory Server (www.sun.com/software/products/directory_srvr/home_directory.html)

- *Disk Space*
  Determine how much disk space is used for user mailboxes so that sufficient space can be configured on the open-platform server.
- *Server Aliases*
  Using the Microsoft Exchange administration GUI, determine what aliases are in use on the server so they can be duplicated on the open-platform server.
- *Additional Functions*
  Decide what functions the Microsoft Exchange server is performing in addition to e-mail, and which of them should be supported in the open-platform environment. Additional functions might include calendaring, integration with mainframe e-mail systems, Web mail interfaces, spam and virus scanning. In the open-platform environment, each of these functions can be replicated using different integration approaches. The next section outlines some of the replacement options available from both open-source and commercial sources.

Options for Additional Functions

This chapter focuses on how to migrate e-mail services from Microsoft Exchange to open platforms. Because organizations sometimes integrate many additional functions into Exchange, and there are so many open platform options for replacing these functions, it is useful to mention some of the options to consider.

Web-Based E-Mail

For more Web-based e-mail packages than are listed below, consult www.freshmeat.net and search for 'webmail.'

One of the most common functions that organizations need beyond standard SMTP, POP, and IMAP protocol support is Web-based e-mail that employees can access from home, from customer locations, and while travelling. Web-based e-mail packages typically access user mailboxes through an IMAP interface, and appear to the mail server as just another client. Some of the open-source Web-based e-mail packages available also support calendaring, and they are summarized in the next section.

Calendaring and Synchronization

Calendaring is the most common non e-mail function that is often integrated into a Microsoft Exchange server. It is important to determine whether calendaring needs to be supported at the client level or at the server level. If users create personal calendars using the Outlook client, but don't use shared calendars, then they can continue to use their Outlook client without any need to reproduce shared calendaring at the server level. If shared calendaring is used, then most open-platform alternatives use a Web-based shared calendar. Some, but not all, software packages are capable of synchronizing with Outlook clients.

A summary of both Web-based e-mail packages and calendaring packages is provided in Table 4-2. The open-source alternatives are plentiful, and there are powerful commercial products like Sun ONE Messaging Server and Sun ONE Calendar Server that can be used independently or together to provide integrated e-mail and calendaring services. In addition, some commercial packages are advertised as drop-in replacements for Microsoft Exchange servers that run on open platforms.

Table 4-2: Assess e-mail client software in use and whether it can be configured to use POP3, IMAP, and SMTP protocols.

Software Name and Reference	Status	Web Mail	Calendaring	Synchronization
Sun One Messaging Server Complete e-mail server, including SMTP, POP, and IMAP, with Web mail support *wwws.sun.com/software/products/messaging_srvr/home_messaging.html*	Commercial	Yes	No	No
Sun ONE Calendar Server Carrier-grade, Web-based calendar server with synchronization capabilities *wwws.sun.com/software/products/calendar_srvr/home_calendar.html*	Commercial	Yes	Yes	Yes
EMU Webmail www.emumail.com/solutions	Commercial	Yes	Yes	Yes
Joydesk *www.joydesk.com/product_business.html*	Commercial	Yes	Yes	Yes
Internet Messaging Program (IMP) Reputed to be one of the best packages available. Calendaring through Horde's Kronolith application. *horde.org*	Open Source	Yes	Through additional software	No
SquirrelMail Very popular, and has lots of plug-ins including calendaring. *www.squirrelmail.org*	Open Source	Yes	Through Plug-in	No
Open WebMail Site indicates that it has assistance for migration from Microsoft Exchange *www.openwebmail.org*	Open Source	Yes	Yes	No

Information about SpamAssassin can be found at www.spamassassin.org.

Spam Filtering

There are a variety of open-source spam filtering packages ranging from DNS-based black hole lists that can be configured with a single line in a sendmail configuration file, to SpamAssassin, which applies a sophisticated, rule-based rating system to every e-mail message that it scans, allowing users to determine the appropriate response, from storing the message in a separate mailbox to deleting the message automatically.

Information about Trend Micro's InterScan VirusWall product is available at www.trendmicro.com. SurfControl's Web site is www.surfcontrol.com

Virus Scanning

Trend Micro's Interscan VirusWall is a commercial virus scanning product that integrates with a variety of mail servers and is available for both Linux and the Solaris Operating System. Another commercial spam and virus-scanning tool is available from SurfControl.

Locate problem tracking software by searching at www.freshmeat.net.

Help Desk Functions

Some organizations use their Microsoft Exchange server to support help-desk and problem-tracking functions that are supported on open platforms by a variety of open-source products.

Find out how Sun can help you by referring to www.sun.com/service.

Forms and Legacy System Integration

In order to replicate Microsoft Exchange functions like forms and integration with legacy systems, custom software needs to be created or configured. For assistance in scoping this effort, consult with architects from the Sun Services organization.

Migration Approach

This chapter focuses on an approach for migrating e-mail functions from a Microsoft Exchanged server to an open platform using sendmail and IMAP server software.

Details on the open-source version of sendmail is available at www.sendmail.org, while the commercial version is described at www.sendmail.com.

The *de facto* message transfer agent for open platforms is sendmail, software that has been used to transfer mail between UNIX systems for more than 20 years. With both commercial and open-source versions available, sendmail is included with all Linux distributions and the Solaris Operating System.

Washington University's IMAP server is described at www.washington.edu/imap, and the Cyrus server is described at asg.web.cmu.edu/cyrus/imapd. A listing of IMAP servers and their features can be found at www.imap.org/products/longlist.htm#Server.

As the recommended mail access protocol is IMAP, this chapter will discuss implementing an open-platform e-mail system using IMAP server software. IMAP servers are provided in both Linux and on the Software Supplement CD included with the Solaris Operating System. Washington University's IMAP server is the most commonly-used, and requires almost no configuration. More complex and secure IMAP servers include the Cyrus IMAP server, which stores all mail in a private database, and allows e-mail accounts without userids on the same server. Regardless of the IMAP server that is adopted, the migration approach remains the same.

Per-User Migration Required

This migration process requires user involvement in two areas: first, each client system must have its settings changed to send outgoing mail via SMTP, and to read e-mail through IMAP on the new mail server. At this point, either Microsoft Outlook clients can be so configured, or different mail clients can be installed. Second, existing user mailboxes and folders need to be transferred to the new mail server, which requires a separate mailbox transfer process for each user.

Because of the per-user migration steps, Sun recommends an incremental migration process, where users are moved one at a time to the new server. This process involves setting up mail servers temporarily so that e-mail can be delivered to the proper server until the migration is complete and the Microsoft Exchange server can be eliminated.

Configuring the Mail Server

The recommended procedure for configuring the open-platform server is as follows:

1. Set up user accounts as described in chapter 2. Individual accounts can be established using the `useradd` command, or PAM, LDAP, or NIS can be used to access user identity services on other platforms.

More details on how to configure user identity is contained in "Migrating User Identity Services to Open Platforms" on page 3.

2. Configure sendmail for your particular environment. Despite the large number of options supported creating a sendmail configuration file is straightforward. The standard sendmail distribution contains template files (ending in `.mc`) with several macro invocation lines for basic, single-domain hosting for almost any platform imaginable. Copy the appropriate template file to one with a unique name, like `yourdomain.mc`, edit the file appropriately, and use the `make` command to generate a sendmail configuration file (ending in `.cf`) The configuration file is placed in `/etc/mail/sendmail.cf`. A template file edited to support the single domain 'yourdomain.com' looks like:

The README file in the cf directory of the sendmail distribution is very helpful and explains all configuration options.

```
divert(0)
VERSIONID('your-configuration-version')
OSTYPE('solaris8')dnl
DOMAIN('yourdomain.COM')dnl
MAILER('local')
MAILER('smtp')
```

3. Translate the system-level aliases that were gleaned from the Microsoft Exchange server into the format required by the `/etc/aliases` file.
4. Add an alias for each user to specifically deliver mail not to the open platform server, but to the Microsoft Exchange server.
5. Configure the Microsoft Exchange server to 'smarthost' e-mail to any unknown user to the open-platform server, and set up the open platform server to accept incoming e-mail from the Internet. Note that this sets up an environment where internal e-mail is transferred on the Microsoft Exchange server, but incoming and outgoing Internet e-mail goes first to the open-platform server. In the case of incoming e-mail, the aliases configured in the previous step cause mail to be delivered to the Microsoft Exchange server until the per-user migration is done. This configuration is illustrated in Figure 4-1,

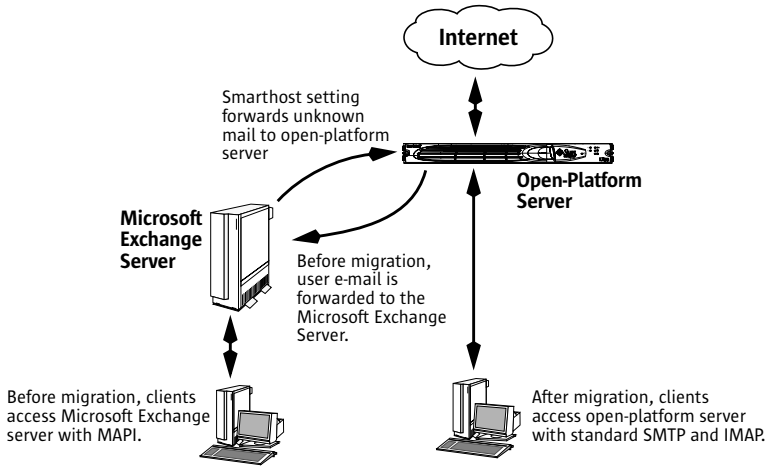


Figure 4-1: The open-platform server is set up as a smarthost before the per-user migration. Before the per-user migration is done, incoming e-mail is forwarded to the Microsoft Exchange server. After a user is migrated, e-mail is smarthosted to the open-platform server and delivered there. This configuration enables a graceful, incremental, migration.

6. Prepare for the per-user migration by creating a backup copy of the source file used for password authentication. This might be `/etc/smbpasswd` in the case of using Samba as a

Windows Domain controller; it might be `/etc/passwd` and `/etc/shadow` if Linux or Solaris Operating System authentication is used. These backup files will be used to restore user passwords after they are temporarily modified by the administrator.

7. The per-user migration can be performed for one user at a time, for a batch of users at a time, or for all users at a time, depending on how you wish to handle the sequence of steps. It is best to perform the per-user steps at a time when no intra-company mail is transferred on the Microsoft Exchange server. This can be accomplished by scheduling the migration on off-hours and/or prohibiting client access to the mail server.
   a. Remove the alias on the open-platform server to discontinue forwarding to the Microsoft Exchange server and allow local mail delivered on the open-platform server. Be sure to execute the `newaliases` command so that sendmail is updated.
   b. Change the user's password to something known on both the Microsoft Exchange and the open-platform server.
   c. Use an IMAP-to-IMAP transfer utility to access all user mailboxes on the Microsoft Exchange server and copy them to the open-platform server.
   d. Migrate user address books if necessary, following the advice given in "User Address Books" on page 19.
   e. Delete the user account on the Microsoft Exchange server. This results in all future internal e-mail being delivered to the open-platform server.
   f. Set the user password to its original state by restoring the encrypted password field in `/etc/smbpasswd` or in `/etc/shadow`.
   g. Change the user's mail client settings to use SMTP and IMAP for e-mail access on the open-platform server.
8. When all users are migrated, the Microsoft Exchange server can be decommissioned. There are a couple of important issues to note regarding this migration process:
   - Once any given user's mailboxes have been migrated and the account deleted, integrated utilities like calendaring are no longer available to that user. As a result, it is best to migrate users to open-platform versions of these facilities before the e-mail migration is done.
   - There is a window between steps 7a and 7e where if the user being migrated receives e-mail from an internal source it will be delivered on the Microsoft Exchange server and not transferred to the open-platform server; this underscores the importance of working with a quiescent network during the per-user migration steps.

The `imapxfer` command is part of the utilities package available at www.washington.edu/imap

Summary

Because of the degree to which Microsoft Exchange ties together non e-mail functions into its server software, migrating from Microsoft Exchange to open-platform servers is one of the more challenging migrations to orchestrate. Once accomplished, however, organizations can benefit from reduced complexity, lower licensing fees, reduced administration costs, flexibility to use a wealth of open-source tools, and higher reliability due to lower complexity and software written with more attention to security.

Chapter 5

Migrating Web Servers to Open Platforms

The single most compelling application to move from Microsoft Windows-based servers to open platforms is Web site hosting. Between the abysmal security record of Microsoft's Internet Information Service (IIS) Web server, the amount of effort that administrators must spend to keep the software's patch level up-to-date, and the high cost of licensing, IT organizations have a significant number of reasons to switch. Indeed, the problems with Microsoft's IIS became so critical with its susceptibility to both Code Red and Nimda viruses that in September 2001 Gartner Group made the unprecedented recommendation for organizations to immediately move away from IIS. Gartner Group suggested alternatives like iPlanet™ Web Server (now Sun ONE Web Server) and Apache, as reported by CNET News, September 25, 2001. From a business standpoint, using IIS is risky because growing a Web hosting infrastructure to meet a sudden, unexpected demand is both a time-consuming and costly endeavor due to the need to purchase additional licenses before deploying. Not so with open-source Web servers, where companies can grow their infrastructure with the speed of their business.

The Apache Web site is www.apache.org.

The Netcraft Web Server Survey is at www.netcraft.com/survey.

When searching for a Web server to replace Microsoft IIS, there is no more popular choice than the open-source Apache Web server. According to a January 2003 survey of more than 35 million Web sites by Netcraft, more than 66 percent of them were hosted by the Apache Web server. Apache's popularity is due to the fact that for years it has provided a robust, secure, easy-to-configure, and scalable platform for Web hosting, unencumbered by any licensing costs whatsoever. Having been developed and incrementally improved by a security-conscious open-source developer community, Apache has been the focus of a relatively few number of security

advisories. In addition to its basic Web server functions, Apache provides a platform for a wide range of open-source plug-ins for creating dynamic content, including the Java™ Servlet API, JavaServer Pages™ (JSP™) technology, and scripting languages including PHP and Perl.

Migration Goals

This chapter provides guidance on the issues that need attention when migrating from Microsoft's IIS Web server to open-source Apache. The focus is on first assessing how the Microsoft server environment is being used, and then using that information to determine how to deploy a similar service on open platforms like Linux and the Solaris Operating System.

One of the key technologies available to facilitate moving towards open-platform technologies is Sun ONE Active Server Pages, formerly Sun™ Chili!Soft ASP. This software, available on both Linux and Solaris Operating System platforms, provides a direct path for re-hosting existing Microsoft Active Server Pages-based content on servers including Sun's Intel Architecture-based LX50 server and Sun's entire UltraSPARC processor-based product line.

For more information on Sun ONE Active Server Pages, refer to www.sun.com/software/chilisoft.

Know Your Environment

One of the most important steps in any migration effort is to assess the way in which the existing server is being used so that when it is replaced, the same or similar functions are supported on the new server. This helps ensure a smooth and successful transition. Fortunately, in the realm of Web services, it's easy to re-host Web applications on an open-platform server and thoroughly test the re-hosted site before deploying it.

Gathering Information

In order to assess the difficulty of undertaking the migration, gather information on how the site is structured and how dynamic content is generated, if any. A taxonomy of likely Web site technologies using IIS is presented in Table 5-1, and is summarized in the following sections.

Table 5-1: Taxonomy of Web site technology discussed in this chapter.

Web Site Technology	Content Generation Technology	Data Source
Static Web Site	None (static)	None (static)
Dynamic Web Site	Uses open-source tools like PHP or Perl/CGI	None (static) Uses open-source data source
	Uses Active Server Pages Uses Active Server Pages with custom COM objects	None (static) Uses Microsoft SQL Server or Microsoft Access as data source Uses commercial or open-source database as data source

- *Static Web Sites*

  Static Web sites use no content-generation technology nor external data sources, making them the easiest of all to migrate. These sites consist of only static files with no dynamic content nor forms that cause the server to take an action based on user-supplied information.

- *Dynamic Sites using Open-Source Tools*
  These sites provide dynamic content or support transactions using open-source tools including Perl and the Common Gateway Interface (CGI), or page scripting languages like PHP. With dynamic sites, an external data source is often used. In the case of dynamic sites using open-source tools, the external data source is likely from the open-source family as well. For example, MySQL is a commonly-used relational database that might be used to store product images and customer orders.
- *Dynamic Sites using Active Server Pages*
  More common in the Microsoft Windows environment are dynamic sites that use Active Server Pages (ASP). ASP is a server-side scripting technology that allows access to objects following Microsoft's Common Object Model (COM). If a dynamic site is built using Active Server Pages, it's important to determine whether any custom-developed COM objects are used, as these will need to be re-created on the open platform.

  Dynamic sites often use an external data source, which is likely to be either Microsoft Access or Microsoft SQL Server when ASP technology is used. It's still possible for a commercial database management system like Oracle to be used in conjunction with ASPs, or an open-source alternative like MySQL to be used, so it's important to determine exactly what technologies are in use. Every dynamic site using a database is unique, thus the details of how to migrate database content from a Microsoft database management system to an open-source alternative are not covered in this chapter.
- *Variations and Custom Configurations*
  There are many variations and custom configurations that might be used, including:
  - Virtual hosting might be used to host multiple Web sites on a single server
  - User directory mapping might be used to allow individual users to independently create their own sites accessed with URLs like *http://www.yourdomain.com/~username.*
  - Microsoft FrontPage server extensions might be supported to allow Microsoft's Web-authoring application to upload Web content through the server interface itself.
  - Sites might be secured using Secure Socket Layer (SSL) encryption.

Migration Approach

The basic migration approach is to configure the Apache Web server on the open platform system, migrate content, page-generation and transaction technology, followed by databases. Whether one or many sites are hosted per server, the migration can be accomplished by moving one site at a time, testing, and deploying.

More information on Sun ONE Web server is available at wwws.sun.com/software/products/web_srvr/home_web_srvr.html.

This chapter focuses on how to use the open-source Apache Web server as the core technology for serving Web pages to Internet users, with suggestions on what extensions to configure in order to effectively replace the functions provided by Microsoft IIS. Apache is not the only Web server that is easily extensible. The commercial Sun ONE Web Server can also be extended with the same technologies that can be used to complement Apache's capabilities.

Open Platform Configuration Strategies

The strategy that works best for migrating from Microsoft's IIS to an open-platform Web server like Apache depends on where the site to be migrated fits into the taxonomy described in Table 5-1, and what variations and custom configurations are used.

Static Sites

Static site content can be moved directly to the open-platform server with no changes. When using Apache on an open-platform system, its default file to serve in any requested directory is the file `index.html`. Because of the traditional three-character file name suffix used in Microsoft Windows, Apache defaults will have to be changed to serve the file `index.htm` if no file name is given. This can be accomplished by setting `DirectoryIndex` to `index.htm`.

Dynamic Sites using Open-Source Tools

Migrating dynamic sites that are written using open-source tools like Perl/CGI and PHP requires re-configuring the open-source tools on the new server, configuring the Apache extensions, and making sure that no platform-dependent code is included in the software to be migrated.

Dynamic Sites using Active Server Pages

Dynamic sites built using Microsoft Active Server Pages usually can be supported by extending Apache with the capabilities of Sun ONE Active Server Pages, available for both Linux and the Solaris Operating System. Sun ONE Active Server Pages provides full support of ASP software, including that generated from popular Web authoring and design tools like Macromedia's Dreamweaver and UltraDev.

If the site to be migrated uses custom-programmed COM objects, these can be reproduced using "Write Once, Run Anywhere™" Java™ technology and incorporated into the ASP environment using Sun ONE Active Server Pages COM-to-Java bridge capabilities. This feature enables organizations to create highly-portable, component-driven ASP applications that, unlike COM objects, can run in many different Web server environments, including Apache, Sun ONE Web Server, and Zeus. Another alternative is to program the dynamic content generation capabilities of the site directly using Java Servlet or JavaServer Pages technologies.

For help migrating to an open-platform, Java technology-based environment, consult with the experienced architects from the Sun Services organization.

Sun ONE Active Server Pages includes ADO and ODBC database drivers, enabling easy access to Oracle, DB2, Informix, Sybase, Microsoft SQL Server, Microsoft Access, MySQL, PostgreSQL, and dBASE databases for dynamic content generation. Given that the goal of migrating to open platforms is to transition away from the Microsoft environment, open-source databases like MySQL are sufficient for many small- to medium-sized Web hosting environments. Incorporating these databases requires modifying the ASP code to access data through different Data Source Names (DSNs), and migrating the data stored in the database.

Information on the MySQL database management system is available at www.mysql.com.

Variations and Custom Configurations

One of Apache's strength is the degree to which it supports modular extensions and custom configurations. Some of the more common variations and custom configurations that are easy to set up include:

- Virtual hosting through name-based and IP address-based virtual hosting mechanisms
- User directory mapping that allows individual users on a system to set up their own Web sites with URLS of the form: *www.yourdomain.com/~user*

Refer to www.apache.org/docs/vhosts for details

Refer to www.openssl.org for details

- Secure sites can be established using the Secure Socket Layer (SSL) implemented by open-source OpenSSL software. OpenSSL supports SSL v2 and v3 and Transport Layer Security v1 specifications. Virtually every commercial certificate authority supports Apache and OpenSSL with their certificate products, and some vendors will re-issue site certificates for the open platform software at no additional cost.

For details, refer to www.microsoft.com and search for "FrontPage server extensions."

- Microsoft FrontPage server extensions are compatible with the Apache Web server, and Microsoft supports versions for both Linux and the Solaris Operating System.

Summary

The high cost of licensing and maintaining Microsoft's Internet Information Service drives many IT organizations to consider the move to the open-source alternatives used by the majority of Web sites worldwide. Tools available from Sun like Sun ONE Active Server Pages makes it easy to take dynamic sites created using Microsoft-compatible tools and deploy them on reliable, secure, open platforms available from Sun Microsystems. The main issues to consider when making the transition is whether the existing ASP code uses custom COM objects, and whether migration to an open-source database management system should take place at the same time.

Once the transition is complete, IT organizations will have access to a huge number of open-platform tools to support the site today and well into the future — along with access to an eager workforce skilled in using such tools. The power of Java technology can be unleashed through techniques including Java Servlets, JavaServer Pages, and JavaBeans™ components that extend the capabilities of Sun ONE Active Server Pages. Perl can be used to program CGI scripts; PHP can be used for server-side scripting; user directories can be maintained using Sun ONE Directory Server or OpenLDAP software; databases of site content, product information, customer data, and sales orders can be stored using open-source MySQL.

Chapter 6

Conclusion

The high cost of licensing and maintaining servers running software from Microsoft, including Windows Domain, Microsoft Exchange, and IIS Web servers, drives many IT organizations to consider open-platform alternatives. Indeed, Samba is one of the most stable open-source applications available; sendmail has been in use and has been hardened for years longer than Microsoft Exchange; and the Apache Web server hosts the majority of Web sites worldwide. Migrating to high-quality, open-platform applications such as these not only gives IT organizations freedom from per-server and per-user licensing fees and proprietary lock-in — they offer a high degree of flexibility, scalability, security, reliability, and interoperability with existing Microsoft and other open-platform systems.

Visit www.sun.com/migration to learn more about how Sun can help with your migration efforts

It should come as no surprise that information on migrating to open platforms is provided by Sun Microsystems and customer migration efforts are supported by the Sun Services organization. Sun has been a champion of open platforms since its inception in 1982, when it shipped its first products based on the open-standard TCP/IP networking protocols. Since then, Sun has built its business around creating innovative new products based on open standards, and then competing to produce the best implementation. When migrating to open platforms, there is no better partner than Sun, whose support for Linux and UNIX platforms is exemplified in its support of both X86 and SPARC® architecture processors, and whose Sun Services organization is ready to assist you in taking your next step towards vendor independence.

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA **Phone** 1-800-555-9786 or 1-800-555-9SUN **Web** sun.com

We make the net work.

Sun Worldwide Sales Offices: Africa (North, West and Central) +33-13-067-4680, Argentina +5411-4317-5600, Australia +61-2-9844-5000, Austria +43-1-60563-0, Belgium +32-2-704-8000, Brazil +55-11-5187-2100, Canada +905-477-6745, Chile +56-2-3724500, Colombia +571-629-2323, Commonwealth of Independent States +7-502-935-8411, Czech Republic +420-2-3300-9311, Denmark +45 4556 5000, Egypt +202-570-9442, Estonia +372-6-308-900, Finland +358-9-525-561, France +33-134-03-00-00, Germany +49-89-46008-0, Greece +30-1-618-8111, Hungary +36-1-489-8900, Iceland +354-563-3010, India–Bangalore +91-80-2298989/2295454; New Delhi +91-11-6106000; Mumbai +91-22-697-8111, Ireland +353-1-8055-666, Israel +972-9-9710500, Italy +39-02-641511, Japan +81-3-5717-5000, Kazakhstan +7-3272-466774, Korea +822-2193-5114, Latvia +371-750-3700, Lithuania +370-729-8468, Luxembourg +352-49 11 33 1, Malaysia +603-21161888, Mexico +52-5-258-6100, The Netherlands +00-31-33-45-15-000, New Zealand–Auckland +64-9-976-6800; Wellington +64-4-462-0780, Norway +47 23 36 96 00, People's Republic of China–Beijing +86-10-6803-5588; Chengdu +86-28-619-9333; Guangzhou +86-20-8755-5900; Shanghai +86-21-6466-1228; Hong Kong +852-2202-6688, Poland +48-22-8747800, Portugal +351-21-4134000, Russia +7-502-935-8411, Singapore +65-6438-1888, Slovak Republic +421-2-4342-94-85, South Africa +27 11 256-6300, Spain +34-91-596-9900, Sweden +46-8-631-10-00, Switzerland–German 41-1-908-90-00; French 41-22-999-0444, Taiwan +886-2-8732-9933, Thailand +662-344-6888, Turkey +90-212-335-22-00, United Arab Emirates +9714-3366333, United Kingdom +44-1-276-20444, United States +1-800-555-9SUN or +1-650-960-1300, Venezuela +58-2-905-3800