COMMUNICATING WITH SAM

Active Directory: Enabling a True Network Operating System

Sam Johnston

Question:

Our company is planning to migrate to Exchange 2000 for messaging, and as such we are evaluating whether to migrate from Windows NT 4.0 to Windows 2000. We have heard that the NT 4.0 to Windows 2000 migration path can be complex and painful. Given that NT 4.0 has been quite stable in our environment, we are hesitant to make major changes unless there are significant benefits to be gained. Can you comment on whether our perceptions are consistent with your experience, and what we can do to lessen the pain associated with the migration. Will the benefits outweigh the costs, or should we stay with the tried and true?

Answer:

If you ask enough people about their implementation experiences, there is not a technology in the world that won't generate horror stories associated with integration. Windows 2000 and Active Directory, the key element associated with Windows 2000, is no different. Although some technologies may be more difficult than most to integrate, generally speaking the grief and misery associated with most technologies is usually self-inflicted. The most common cause of grief is lack of planning resulting in poorly conceived plans for deployment. Windows 2000, given the pervasive impact it has as a network-wide operating system, is no different. Consequently, following best practices for the deployment of Active Directory should eliminate most if not all of the grief you fear.

The Active Directory component of Win2000 is the primary reason why you would want to migrate from NT 4.0. Active Directory (AD) is essentially a directory service, which is a network service that identifies all resources on a network (a directory) and makes them accessible to users and applications (the service). As the centralized control point that manages and brokers all relationships between various network resources, AD is integral to network operations and is the catalyst that has transformed NT 4.0 from essentially just a server platform to become a full Network Operating System (NOS) in the Win2000 iteration.

The key benefit of an AD is that it creates a single point of administration for all published resources (files, peripheral devices, host connections, databases, Web access, users, services and other objects), and enables users to more easily find these resources as they can search for resources based on a single attribute in situations where little is known by the user. Security can also be improved via AD by protecting resources from being access by unauthorized users.

From an IT management perspective, AD simplifies network administration and set-up by the use of peer domain controllers only. The directory itself is organized into sections, which permits easy scaling to a large number of objects. The Win2000 directory services have adopted Internet standards for name and space to enable a homogenous approach for managing heterogeneous applications.

Active Directory uses DNS for naming and complies to open standards through the ability to communicate with an application using LDAP or HTTP. Compliance with several Standard Name Formats permits users to use the format they know best.

Hopefully these benefits have sold you on the merits of migrating to Active Directory, but if they haven't, then you may not like the fact that Exchange 2000 requires AD, thus in your case you have no choice. Exchange 2000 requires that you deploy Active Directory and Win2000, and this in our experience is the major source of deployment horrors. The common error is to plan AD as an element of Exchange 2000, rather than treating them as two separate projects with their rightful focus to ensure success.

Planning an Active Directory can look quite simple, but is in reality quite complex, with the implementation involving three elements or hierarchy layers:

1. Planning a namespace
2. Planning Organizational Units (OUs)
3. Planning a Site

For the balance of the article, we'll concentrate on some of the best practices you can use to ensure a successful migration to Win2000 and AD.

Planning a Namespace

A namespace needs to be a fully qualified domain name, just like the DNS. In planning the namespace, make sure you take into account the business structure and operation of your organization, as well as the geography, growth, network and the required access to network resources.

Several steps and decision points are involved in designing a namespace. First, you need to plan the DNS or external namespace, which includes the domain hierarchy, global catalog, trust relationships and replication models.

Next you will need to consider the internal versus external namespace design.

The namespace is the top-level active directory domain name for the organization. You will have two primary choices, and can either implement the same namespace for both, or a namespace separate from the established registered external DNS namespace. Each has advantages and disadvantages.

Same Internal and External Namespace Design Approach

The advantages of this approach are:

- The tree name is consistent for both the private and public network.
- Users have the same logon name to public and private network access.

The disadvantages of this approach are:

- The configuration is more complex since proxy clients must be configured to know the difference between internal and external resources.
- Security becomes more complex, and administrative care must be taken to ensure private resources are not published to Internet.
- There is duplication of effort in managing resources.
- Despite the namespace being the same, users will get a different view of internal and external resources.

Separate Internal and External Namespace Design Approach

If you take this approach, it is recommended that both domain names be registered with Internet DNS. For this approach to work, two zones are established. One zone resolves internal requests, and a second zone resolves external transactions. The advantages of this approach are:

- There is a clearer difference between internal and external resources, which assists in security management.
- There is no duplication of effort
- Configuration of proxy clients is simpler since exclusion lists are simpler

The disadvantages of this approach are:

- From a user perspective, logon names are different between internal and external connections
- You will need multiple names registered with DNS

In making a decision, you need to review your organization's business approach to areas such as security to put relative weight on each advantage and disadvantage. In general, good namespace architecture generally has the following characteristics:

- The namespace is representative of the structure of organization
- The namespace provides administrative granularity required to manage enterprise wide global network
- The namespace should be scalable and extensible

From a practical perspective, there are two key considerations in defining namespace architecture. First, the organization's WAN network design is critical since replication with the domain will rely on the network. Additionally, the reality is that organizational structure changes constantly and your design must be flexible enough to support changes. In general terms, your goal must be a namespace that is scalable, adaptable to change, and can easily distinguish between internal and external resources while concurrently protecting the company's data. The namespace should represent the structure of the organization and simultaneously provide granularity required to manage the enterprise-wide global network. A common structure to provide this is a three-layer model deploying a root domain, first layer domain and second-layer domain. The root domain is a basic concept related to the DNS namespace. The objective of the first layer domain is to create domain names that are not likely to change even in the event of an internal corporate reorganization of resources. The simplest way to provide a first layer domain is to name domains based on continental or geographical boundaries. The second layer domains should be countries only and branch off the corresponding first layer domains. This means that child level domains can be created below these domains.

Planning Organizational Units (OUs)

The next step in deploying Active Directory is the planning of your Organizational Units. It is strongly recommended that you use the same naming convention when creating an OU within a domain, as this will enable future promotion of a specific OU to a domain with minimal user impact. Similar to the namespace, the logic of the OU convention should reflect the details of the organization's business structure. Your objective in creating an OU is to delegate control over smaller groups of resources. Additional granularity is gained via top-level OUs that can contain lower level OUs. Creating granularity in this manner does not necessarily generate administrative complexity as OUs inherit rights from either the parent OUs or the domain unless it is specifically disabled. In designing your Organization Units, use the following guidelines:

- Create OUs to delegate administration
- Create OUs to apply security policies
- Create OUs to provide or restrict visibility of published resources from certain users
- Create OUs that are relatively static.
- But avoid allocating too many child objects to any particular OU

In designing the structure of your OUs, it important to determine what concept will be used to define OUs, and once this is done you need to ensure that standards are followed across the organization. The obvious decision is to simply mirror your business hierarchy. Based on our experience, the following are common categories that can be used to classify your OU hierarchy.

- Administration or object based
- Geographical based
- Business function based
- Department based
- Project based

A helpful hint is to review your VLANs while defining your OUs. The VLANs may have created categories that can be replicated for the OUs, or conversely, your OU design may drive the need to redefine your VLANs, as both are a logical means to grouping resources within your network.

Planning a Site

Until this point in your design, the focus will be logical considerations via the namespace and Organization Units. However, attention to the physical design is also critical to a successful implementation of a Windows 2000 server network supporting Active Directory services. The physical design layer is demarcated at the site layer.

A site is a combination of one or more IP subnets. Often a site has a natural boundary such as the LAN or perhaps a VLAN configuration. In the case of a high bandwidth WAN, a site may expand across physical locations, and it may look like one LAN.

The key reason site definition is important is that the AD replication engine allows an administrator to differentiate between replication that takes place over a LAN and replication that takes place over a WAN. How your sites are setup will impact Windows 2000 in two ways.

First, the workstation logon will have impact due to the fact that when a user logs on the active directory services, enabled clients will try to find a domain controller in the same site as the user's computer and subsequent logon.

Secondly, directory replication will also vary your deployment, as the schedule and path for replication of a domain's directory can be configured differently for inter-site replication. Keep in mind that in AD, the sites are not part of the namespace and the site structure is kept in a separate part of the directory.

Your network design will have a large impact on site design. When planning how to group subnets into sites, make sure you consider the connection speed of the network and available resources. To economize on bandwidth needs, you should configure replication to occur at times that will not impact network performance. We have provided some overall design guidance based on best practices and a green field deployment.

The reality is that most environments have an existing NT 4.0 architecture that needs to be considered. Although the design principles discussed generally apply in all situations, it needs to be noted that your existing architecture will have an impact on some of the decisions you will make. Specifically, from a best practice perspective, you have two choices once the Forest Root has been created, as you can either create a new Regional Domain, or upgrade the in place Regional Domain.

The right decision will be driven by the unique attributes of your network, and mastering the design and planning process at this juncture will define the success or failure of the project.

While it is not practical to cover all the variables that may drive a particular approach, it is possible to define the process you need to follow. We will briefly cover the key best practice tactics and project phases recommended by Microsoft that will ensure success of the project.

Envisioning

The first phase of the project is envisioning, or definition of a Mission Statement. It is crucial that you avoid simply jumping into execution, and instead define your needs. In this phase, make sure you review your current and future authentication needs, including assessing two-factor authentication for enhanced security, and a single user sign-on for access to all network resources.

You should also do an initial Risk Assessment of the various options available to you for the migration to and creation of a W2K AD environment, including options available for providing directory services that will have bearing on final design.

Planning and Design

At this phase, you will need to perform a detailed review of the options available for W2K migration. Specifically, this is where you will define whether you will do an in place upgrade of your existing WIN NT environment or are better to install a new AD environment and migrate devices, services and users to the new AD.

You will also need to review critical issues and problem areas, including verifying if there are applications that are not compatible with Windows 2000. You should also evaluate existing sys

tems such as applications and network resources such as DNS and WINS. You need to determine if your name resolution will be adequate for AD. It is key to note that In order for AD to function properly it relies on DNS, and consequently your DNS must be stable. Don't forget to evaluate existing support infrastructure, including response times, SLAs needed by the business and the technical skill of your staff. Active Directory can be as simple or complicated as your business requires, however, even the simplest AD design requires skilled staff in order to ensure it functions properly and remains stable

Development

Resist jumping right to deployment, and avoid under-estimating the complexity of the project. Rather, create a development environment. Start by creating an installation planning checklist, then apply design principles documented in the planning and design phase in a development environment. The key is that the theory needs to be put to the test. At the end of this phase, update your business goals based on the outcome of development environment, and return to the design phase if the development phase does not provide validation.

Piloting and Deployment

A pilot period is recommended as a stress test. Make sure your document clear pilot objectives and timelines. Ensure that you provide feedback channels for pilot users and deployment personnel so that issues are not ignored when there is still time

to modify your plans. Monitor activities closely and conduct a post-pilot review with key personnel before moving to the final steps of deployment.

You should break this phase into smaller segments, including a preparation for installation period during which no actual implementation will be performed, but rather detailed planning of all events will take place so there is no room for interpretation by deployment staff.

For actual deployment, start with the implementation of the network services necessary to support Active Directory, including TCP and DNS if they are not already in place. Next, implement W2K on the domain controllers, subsequently moving to implementation of W2K on desktops.

This component is optional, as you can install an AD client on any Windows client that is Win9X and higher. Remember, monitor and stabilize the network by department, as AD is a complicated network directory service that also impacts how users go about their business. Make sure you have the appropriate end user support resources in place so that your project does not fail due to end user frustration from change.

In the end, if you do your homework, there is no reason for you to experience the horrors you fear, and the upside of having a true network operating system via Active Directory will pay large dividends. T G

Sam Johnston *is a partner and Chief Technology Officer of Intesys Network Communications Ltd., providing value-added networking and e-commerce solutions to the iSeries community. He can be reached at (416) 438-0002 or via e-mail at sjohnston@intesys-ncl.com.* *Any TUG member wishing to submit a question to Sam can forward their typewritten material to the TUG office, or to Intesys. The deadline for our next issue is Friday June 14th, 2002.*