Possible Scenarios for Managing the Enterprise Administrator Group in Virginia Tech's Windows 2000 Domain Forest

Michael Johnson
Middleware Services
Virginia Tech
1/14/02

Abstract

The Enterprise Administrators (EA) group resides in the root domain of every Windows 2000 forest of domains. By default, accounts that are members of this group have nearly unfettered access to data throughout the enterprise. This introduces unique security concerns that did not exist in the Windows NT 4.0 domain model. This paper serves as a discussion of methods that may be undertaken to mitigate the authority of the EA group and to secure the accounts that are members of the EA group. Many of those methods are collected from a literature review and must be tested thoroughly before child domains are allowed to implement them. However, some suggested methods, including ensuring complete non-anonymity of EA accounts and using temporary accounts for EA actions that require secondary logons, such as child domain promotion, can and should be implemented immediately.

Introduction

Windows 2000 stores all user, group, and computer data as objects in its own directory service, Active Directory. Using a directory services instead of NT 4.0's Security Account Manager (SAM) database provides improved flexibility, scalability, redundancy, and security for Windows 2000 domains. Multi-master replication (all domain controllers have a write-able copy of the directory), Global Catalogs (stores a copy of all objects in the forest with a sub-set of attributes), controlling access to objects with Access Control Lists (ACLs), a hierarchical structure to facilitate inheritance of permissions, and an extendible schema (allowing the definition of new object attributes that can be utilized by in-house developed directory applications) are just a few of the advantages Windows 2000 has over NT 4.0.

The domain model in Windows 2000 has also changed significantly from NT 4.0; a partial reflection of the difference between storing accounts in a hierarchical directory service rather than in isolated databases. A Windows 2000 "forest" of domains consists of one "root" domain and one or more "child" domains. Domains within Windows 2000 forests provide boundaries for authority. Each domain has its own directory service over which a Domain Administrators group has, by default, sole authority. While being a member of a forest does imply some shared components between sibling domains (including a common schema, two-way transitive trusts with other domains in the forest, and a common Global Catalog), a domain administrator in one child domain has no default administrative access to other domains in the forest.

The autonomous authority for child domains is compromised by a built-in group in the root domain known as Enterprise Administrators (EA). This group was envisioned by Microsoft to be responsible for administrative tasks that affect the entire forest. Those tasks include adding new child domains into the forest and authorizing DHCP servers. A definitive list of tasks which, by default, only EA can perform is given in Appendix A. The EA, by default, is added to the Domain Administrators group in every child domain, and the Domain Administrators group is in turn added to the Administrators group of every member server and workstation in that child domain. This can cause obvious political problems in diverse organizations, such as a university, where various groups or departments may want to maintain absolute autonomy from a centralized IT organization that will most likely be running the root domain. Furthermore, EA's unrestricted access to all objects in the forest raises security concerns since the group is a single-point-of-failure. That is to say, compromising accounts placed in the Enterprise Administrators group compromises the entire forest, from the root domain controllers to an individual's workstation.

This paper acts as a summary of documentation provided by Microsoft and an investigation into various scenarios that Virginia Tech might use to increase security for the EA group. Current EA use at Virginia Tech will be articulated, an examination of Smart Card logon for EA accounts will be discussed, followed by an investigation into potential methods for delegating EA authority and blocking EA access to child domains.

Current EA Use

Currently, the Enterprise Administrators group in Virginia Tech's Windows 2000 Active Directory forest contains one user account with a complex password. Two individuals, the persons tasked with administrating the forest, know that password. This raises some obvious security concerns. For example, it is impossible to know from examining system logs which of those individuals performed any given task. Anonymous access to EA authority makes auditing the Enterprise Administrators difficult, if not impossible, to perform. Also, this structure does not facilitate organizational changes, since if one of the individuals leaves the group the EA password must be reset; there is no other way to restrict the forme0.1diviwee's forest-wide access.

One obvious potential solution to this problem is to add an account to the EA group for each of the persons charged with administration of the forest. While this addresses the concern for anonymous EA authority, the solution introduces multiple single-point-of-failures for the entire forest. If there are n number of accounts in the EA group, then a hacke0.has n accounts to attempt to crack, instead ofonly one. After one password has been compromised, the intruder can remove all other accounts from the EA group, thus gaining control of the entire forest.

However, if it becomes possible to secure the accounts in the Enterprise Administrators group such that hacking is extremely difficult to perform, EA anonymity can be removed without fear of increasing exposure to the forest. The following sections discuss how to "harden" accounts placed in the EA group and possible scenarios for mitigating exposure to the forest if EA accounts are compromised.

Microsoft Consulting Services characterizes how EA authority is exercised within Virginia Tech's Windows 2000 forest as "a great example of the way Microsoft intended the EA role to be exercised." EA credentials are most commonly used for child domain creation and disaster recovery in both the root domain and in child domains. EA authority has been used once for authorizing a DHCP server in the CEE (Department of Civil and Environmental Engineering) child domain. There are currently fourteen child domains and three instances of serious disaster recovery procedures being initialized by EA. Therefore, EA authority is used conservatively by Virginia Tech's root domain administrators.

Smart Card Logon

Interactive logon in Windows NT 4.0 took the form of password challenge/response. The user was prompted for a username and a password, and a central authority (the NT 4.0 domain controller, which contained a hashed version of the password) confirmed or denied that the user knew the password for the given username. At some point in the process, the username and password (perhaps encrypted) had to travel over the network from the client computer to the domain controller. Security of the transmission could be as weak as clear-text, or as strong as NTLM encryption. Clear-text is obviously undesirable, but even NTLM traffic is susceptible to brute-force decryption hacking.

Windows 2000 provides support for public-key interactive logon using an X.509 version 3 certificate stored on a smart card. Microsoft has integrated smart card logon with the Kerberos services that come with Windows 2000, ensuring that the user's password never travels across the network. At logon, instead of being prompted for a username/password, the user places her smart card in a reader and is then prompted for a Personal Identification Number (PIN) which authenticates her to the card. The user's certificate is extracted from the card and sent to a Kerberos Key Distribution Center (KDC). The KDC looks up the user object in Active Directory and verifies that the certificate was issued by a Certificate Authority that is trusted in the Active Directory forest. The KDC encrypts its response to the client with the certificate's public key, which guarantees that only a person in possession of the certificate's private key can decrypt the response. If the private key on the smart card can decrypt the response, then the user is in possession of a Kerberos Ticket Granting Ticket (TGT) and can use this to authenticate with network services. The user has authenticated to the domain without ever entering her password or sending it across the network.

Smart card logon provides some serious security benefit to the average user, but there are some tasks that must be performed by Administrators or Power Users which the scenario described above cannot address. For example, the following three items require a secondary authentication with a username, domain name, and password.

- Joining a computer to a domain
- Promoting a server to be a domain controller
- Configure a connection for remote access

The second item is one of the operations that require EA authority. Unfortunately, smart cards alone cannot solve the exposure problem for Enterprise Administrators.

There are also some technical obstacles to overcome if Virginia Tech wishes to use smart card-based authentication. The first problem is that a Certificate Authority (CA) must be in place to create certificates, and that authority must be advertised to Active Directory so that the KDC can verify the certificate's issuer. Currently, there is no CA in place at Virginia Tech that is advertised to Active Directory. CA software comes with Windows 2000 Server, so there is no expense in purchasing a CA or certificates from an outside vendor. However, to implement a decent Public Key Infrastructure (PKI), Microsoft recommends at least two servers, one to act as the root authority and the second to act as the subordinate.

Requiring accounts in the Enterprise Administrators group to use smart cards for authentication creates a strong dependency on the PKI. The PKI must be absolutely reliable, because if the PKI fails then authentication for accounts that require smart card logon is impossible.

Restricting EA Access

Given that the Enterprise Administrators group compromises the authority boundary between domains, potential customers of Virginia Tech's Active Directory forest may be interested in restricting EA access to their child domain. What follows is a summary of documentation provided by Microsoft describing steps that can be taken to remove EA access to a child domain. It is important to note that all of these steps should be tested in the w2k-pilot forest to ensure that they do not impair the normal operation of the child domain or the forest.

There are three main operations in removing the EA account.

- Remove the EA group from the Administrators group
- Alter permissions on objects in the domain to ensure that EA does not have access
- Revoke any EA rights to the Administrator account

Removing the Enterprise Administrators group from the Administrators group will block any member of the EA group from having administrative authority within that domain. However, EA will still have access to any object which explicitly lists them in the object's ACL. By default, EA has rights to each domain's Administrator account object, which permits an EA to change the password and use that account to circumvent the removal of the EA group.

Perform these tasks with the Active Directory Users and Computers MMC on the child domain controller.

- Remove the EA group from the Administrators group
- In Advanced view, remove the EA group from the ACL on the domain object and ensure that these new permissions are set to be inherited by all child objects. Additionally, check the ACL on every object and make sure that the EA group was not explicitly added to that ACL
- In Advanced view, remove EA from the ACL of the Administrator account

There are some consequences to consider before allowing child domains to restrict EA access. One of the primary ways that EA is used at Virginia Tech is disaster recovery. Having a centrally authority permits quick intervention when a child domain encounters technical problems that might be damaging to itself and the integrity of the entire forest. However, it is important to remember that removing the EA from a child domain is a reversible process; the domain administrator of the child domain could choose to restore EA access if he thought that it would be helpful in a trouble-shooting situation.

It's important to note that Microsoft has no guidelines for restricting EA access to child domains that contain Exchange servers. Suffice to say that such an operation would not be recommended as Exchange 2000 has an extremely complex and intricate relationship with Active Directory. Also, any other applications in child domains that might need EA-level access must be tested thoroughly in a pilot situation before EA is removed from a child domain.

Another consequence of restricting EA access is that many operations can only be performed by Enterprise Administrators, including authorizing DHCP servers and adding child domains. Removing EA access to child domains could make those operations impossible to perform within the child. The next section describes how to segment and redistribute some of the authority that by default is granted only to Enterprise Administrators.

Segmentation and Redistribution of EA Authority

Appendix A contains a list of all tasks which, by default, only members of the EA group can perform. In NT 4.0, there was no mechanism for dividing up Administrator privileges and delegating only some of the authority to different groups or users. In a sense, authority was an all-or-nothing deal; you were in the Administrators group, you were in the Account Managers group, or you were a user. Windows 2000 breaks away from this restrictive paradigm by storing its user, group, and computer objects in a directory service known as Active Directory. Within Active Directory, Access Control Lists (ACLs) control which users or groups have access to an object, as well as dictating what type of access they have. The type of access that can be given to a group or user is extremely fine-grained. Thus, by modifying ACLs on directory objects, it is possible to delegate authority, as well as defining the scope of that authority. Microsoft provides "Wizard" applications for implementing common Delegation of Authority scenarios, but segmenting and redistributing the sweeping authority given to the EA group is a complicated and intricate process which can only be performed by manually manipulating ACLs on Active Directory objects.

Delegating EA authority is a complicated task, and in many cases the steps necessary to segment and redistribute that authority can have unintended security consequences. Furthermore, the control of some tasks should be maintained by a central entity because their consistent and successful implementation is critical to the health of the forest. Such tasks include Schema management, root domain management, DNS management, and site link management. This section discusses three primary tasks that by default require EA authority, but delegation of which might be of interest to current and potential customers of Virginia Tech's Windows 2000 forest.

It is important to note that ALL of the techniques described below must be very thoroughly tested before they are used in a production environment. The steps described below are collected from a literature review and have not yet been tested in the w2k-pilot AD by this author.

Child Domain Pre-Creation

By default, only members of the Enterprise Administrators group can add new child domains to the forest. This is by far the most commonly used form of EA authority at Virginia Tech. While it is impossible to completely delegate this responsibility to another group, it is possible for the EA group to perform a child domain pre-creation process which would allow a user or group other than EA to perform the final promotion event on the child domain controllers.

In essence, granting the ability to promote child domain controllers to a group other than EA requires modifying ACLs on certain directory objects. The first step, then, is to create a group in the root domain to which we will delegate this authority. For the purposes of this discussion, we will name the group *Domain Creators*. Clearly we want members of this group to only have control over the objects that we allow them to create and deny them access to objects that belong to other child domains. To that end we add a built-in group to the ACL, *Creator Owner*, which allows us to transfer object permissions to another group when they create objects. For example, a user account is granted **create child object** privileges on a container and the *Creator Owner* has **full control** of that container and all child objects. When the user account creates an object within that container, the user is automatically made the **owner** of the object, which means that, with respect to that object, the user is considered a member of the *Creator Owner* group. Since the *Creator Owner* group has full control over that object, the user account is granted full control of the object, without gaining unwanted access to any other object within the container. This scenario will be used in many of the delegation procedures described in this section.

The Schema and Configuration naming contexts are shared across the forest, so the *Domain Creators* group must be able to read and replicate their contents in order to perform a successful promotion. The following permissions must be given to the *Domain Creators* group on both the Configuration (cn=Configuration, dc=w2k, dc=vt, dc=edu) and Schema (cn=Schema, dc=w2k, dc=vt, dc=edu) containers:

- Read
- Manage Replication Topology
- Replicating Directory Changes
- Replication Synchronization

In Active Directory, a "site" is defined as a collection of highly-connected IP subnets. If a forest has to span one or more physical locations, which internally have fast connection but between themselves have slower connections, then the typical practice is for the EA to create multiple sites, one for each location, and then manually schedule replication traffic between the locations so that AD replication does not swamp the already burdened WAN link. In essence, a site is a replication boundary. Between domains within the same site, Windows 2000 automatically establishes replication links between the domains. But between sites, Windows 2000 expects the administrators of the forest to schedule when and how replication should occur.

Why is this important for child domain creation? Whenever a new domain controller is added to the forest, regardless of whether it's added to a new domain or an existing domain, it must be assigned to a site before it can participate in replication. That is to say, an object for that domain controller must be added to the site container. Therefore, the following permissions need to be applied to the container for the site that will contain the new child domain.

- *Domain Creators* group needs **Read** and **Create child objects** for THIS OBJECT AND ALL CHILD OBJECTS.
- *Creator Owner* group needs **Full Control** for THIS OBJECT AND ALL CHILD OBJECTS.

Note: in order for the root domain administrator to maintain control of the site creation process, EA will need to create a new site where the child domain controller will reside. To register the new server in the proper site, the EA will also have to add the server's subnet and associate it with the new site (the site has no meaning without a subnet). Then the EA needs to grant the two permissions listed above on the new site.

After these tasks have been accomplished, the *Domain Creators* group has permission to replicate the Configuration and Schema information as well as create a new server in the appropriate site. Next, the EA needs to create a cross-reference object for the new domain and give the *Domain Creators* group full control of that object.

To create the cross-reference object for the child domain, use the command-line tool NTDSUTIL on one of the root domain controllers while logged on as an EA. Here are the steps required pre-create a child domain cross-reference for the hypothetical child domain ENG, where server1 will be the first server in that domain.

- At a command prompt > ntdsutil
- Type **Domain Managements**
- Type **Connections**
- Type **Connect to Server grok.w2k.vt.edu**
- Type **quit**
- Type **precreate dc=ENG,dc=w2k,dc=vt,dc=edu server1.eng.w2k.vt.edu**

After you've created the cross-reference, you must give the *Domain Creators* group permission to access the information. Locate the new cross-reference in the Partitions container within the Configuration-naming context. Add the *Domain Creators* group and give it **Full Control** of the object.

Windows 2000 uses Kerberos transitive trusts between the domains within the forest. The trust objects reside in the System container within the Domain naming context of the parent domain. For a child domain to be created successfully, a trust object must be created within the System container in the root domain. Modifying the following permissions:

- Add the *Domain Creators* group to the System container under the Domain naming context and give the group the rights to **Read** and **Create** child objects.
- Add the *Creator Owner* group and give it **Full Control** for THIS OBJECT AND ALL CHILD OBJECTS.

Now the local administrator can begin the promotion process by running DCPROMO on the soon-to-be child domain controller. After which, the child domain administrator can take actions to remove EA access to the new child domain.

DHCP Authorization

Windows 2000 requires DHCP servers to be registered in Active Directory before they are allowed to operate within the domain. This prevents rogue DHCP servers from granting leases to domain members. By default, only members of the Enterprise Administrators group are authorized to register DHCP servers, but that authority can be delegated to another group. If the root domain is in Native mode, then the group to which we transfer this authority should be a Universal group so that it can include accounts from child domains.

To allow the group *DHCP Administrators* to authorize DHCP servers, make these changes to the ACL on the NetServices container (cn=NetServices, cn=Services, cn=Configuration, dc=w2k, dc=vt, dc=edu).

- Give the *DHCP Administrators* group the rights **Read, Write,** and **Create Child Objects** for THIS OBJECT ONLY.
- Add the *Creator Owner* group and give it **Full Control** for THIS OBJECT ONLY.

RIS Authorization

Remote Installation Services (RIS) is a mechanism in Windows 2000 by which Administrators can automate the deployment and initial configuration of Windows 2000 Professional. RIS servers must be authorized in the same way that DHCP servers must be. The object who’s ACL that you must modify to delegate that authority to a group other than EA (call it *RIS* Administrators) are the same as for DHCP servers.

Non-Technical Solutions

Some of the concerns that present or future customers of Virginia Tech's Windows 2000 forest have with EA authority could be resolved by non-technical, political solutions. The basic goal of a political solution would be to engender trust and to make child domain administrators comfortable with the role of Enterprise Administrators. One potential means for doing so would be to create a committee of concerned individuals, those tasked with administrating the root domain plus child domain administrators, and dividing up the EA password amongst the members of the committee. This would prohibit any EA authority from being exercised without the cooperation and agreement of all members of the committee, and would elevate concerns from child domain administrators about rampant EA authority.

However, there are some drawbacks to such a plan. Currently there are fourteen child domains and two root domain administrators, so the committee would have 16 members if it was formed today. The size of the committee can be expected to grow as more child domains join the forest. That's a large number of individuals from whom a unanimous decision is required before any exercise of EA authority can be done. One of the tasks where EA authority is invaluable is disaster recovery for the forest. If something happens in the root domain or in a child domain, EA can act quickly to solve the problem. However, EA will be severely hamstrung and the ability to act quickly to solve a problem negated if sixteen people have to line up and enter their part of the password before disaster recovery can begin. Furthermore, what happens if one of the sixteen is unreachable? A possible solution to the large number of committee members is to restrict membership in someway. But what criteria could be set down? What would make one child domain administrator more important than another? Trying to limit the size of the committee would introduce a new political problem, instead of solving the original one.

A compromise solution would be to form a working group composed of root domain administrators, child domain administrators, account managers (IRM), and security officers. This group would be open to individuals thinking about joining their domain to the Virginia Tech Windows 2000 forest. The group would be a forum for disseminating information about how EA authority would be used, as well as discussing other topics that affect the forest, such as Group Policy or Central Services. Decisions about how and when to use EA authority could be discussed with the members of the group, and any loud criticism would be taken into consideration. Marc DeBonis has already begun the process of forming such a group.

Another potential method for reducing EA exposure, one that does not require developing new technologies, is to create temporary accounts for the purpose of child domain promotion. Since the ACLs on objects created during promotion list the EA group, not the individual account within the EA group that authorizes the promotion, then the temporary account can be safely removed from the EA group and disabled after promotion without impacting the functionality of the new child domain.

Conclusion

Enterprise Administrator authority is a concern to potential customers of Virginia Tech's Windows 2000 Active Directory forest because it compromises the concept of the domain as the boundary of authority. This paper has discussed both technical and non-technical methods for mitigating that concern. The proposed technical solutions need to be tested thoroughly in a pilot Windows 2000 forest before any production child domain is permitted to attempt those processes. There are potential situations, such as departments that are obligated by Non-Disclosure Agreements to enforce rigid control over who has access to sensitive data, where permitting child domains the ability to restrict EA authority is the only way that they'll join Virginia Tech's Windows 2000 forest. In these situations, the child domain administrator should be made aware of the consequences (e.g., inability for root administrators to assist directly in disaster recovery for the child domain), before being allowed to restrict EA access. It is important to remember that if the child domain administrators should decide to restore EA access to his domain, the process outlined in this paper is reversible (although that theory also needs to be tested in the w2k-pilot forest). Also, if the child domain should become a nuisance or detriment to the rest of the forest, the root domain administrators can protect the forest by deleting the offending child's metadata from Active Directory.

This paper also addressed security concerns that arise from Enterprise Administrators unrestricted access throughout the forest. The EA group is a single-point-of-failure in terms of security. If accounts within the EA group are compromised, the entire forest is compromised, from the root domain controllers to an individual's workstation. Therefore it is desirable to limit the exposure of EA accounts. One method for limiting EA exposure is to segment and redistribute some EA authority to different groups. The process for delegating authority to *Domain Creators*, *DHCP Administrators*, and *RIS Administrators* should be tested thoroughly; and if found to be reasonable, implemented in the production forest. By delegating EA authority to different groups, we effectively reduce the number of times that members of the EA group must authenticate, i.e., we have reduced the number of times that EA credentials traverse the network.

Windows 2000 has a built-in mechanism for smart card authentication that greatly reduces EA password exposure. However, smart card logon cannot be used for applications that require a secondary logon, such as the domain promotion process, although future releases of the OS may eliminate this requirement. But if the segmentation and redistribution of EA authority described above can be accomplished, then EA authority would only be necessary to add accounts to the *Domain Creators* group and then those accounts' credentials could be used for the secondary logon. A much less complicated solution would be to create temporary accounts in the EA group and use those for child domain promotion. After promotion, the temporary account can be removed from EA and disabled. This scenario would allow the standard EA accounts to utilize smart card logon technology.

Some changes to the EA structure can and should be made immediately. Anonymity of the current EA account should be removed and individual accounts with obvious accountability should be added into Enterprise Administrators. Tasks that do not require EA privileges, such as backups, should be given to other accounts or groups. The costs in time and money for implementing a Public Key Infrastructure, a prerequisite for smart card logons, should be investigated. The process described for segmenting and redistributing EA authority should be tested, as well as the process for restricting EA access to child domains.

Appendix A: Tasks that Require EA Authority

Description	Tool Used	Reason EA is required
Install Enterprise Certification Authority	Install Certificate Services using Add/Remove Programs	Creates CN=Public Key Services, CN=Services, CN=Configuration and objects in this subtree
Create new domain in forest	Active Directory Setup and Install Wizard (DCPROMO)	Creates crossRef objects in CN=Partitions, CN=Configuration
Manage Sites and Subnets	Active Directory Sites and Services snap-in	Creates and modifies objects in CN=Sites, CN=Configuration subtree
Install Certification Authority for a child domain	Install Certificate Services using Add/Remove Programs	Creates objects in CN=Public Key Services, CN=Services CN=Configuration subtree
Create Admission Control Service (ACS) policies	ACS snap-in	Creates subnet objects in CN=Subnets, CN=Sites, CN=Configuration Creates CN=ACS, CN=Subnets, CN=Sites, CN=Configuration and objects in this subtree
Install first Exchange 2000 server in forest	Exchange 2000 setup (see also /ForestPrep cmd line switch)	Creates objects in CN=DisplaySpecifiers, CN=Configuration subtree Creates CN=MS Exchange, CN=Services, CN=Configuration and objects in this subtree
Authorize a DHCP server	DHCP snap-in	Creates CN=DHCPRoot, CN=NetServices, CN=Services, CN=Configuration and objects in this subtree
Authorize a Remote Installation Server (RIS)	DHCP snap-in	Same as authorize DHCP
Set up printer location tracking	Active Directory Sites and Services snap-in	Set location attribute on subnet or site objects in CN=Sites, CN=Configuration subtree Set location attribute on computer object in any domain
Set up Simple Certificate Enrollment Protocol (SCEP) Add-on for Certificate Services (Reskit utility)	Run cepsetup.exe from Resource Kit on Enterprise CA server	Changes ACL on objects in CN=Public Key Services, CN=Services CN=Configuration subtree
Install Messaging Queuing (MSMQ) Routing Server	MSMQ setup, select Routing Server option	Creates CN=<Server>, CN=Servers, CN=<Site>, CN=Sites CN=Configuration and objects in that subtree
Run MSMQ Replication Service in MSMQ mixed mode (some servers running MSMQ 1.0 on NT4)	Service must run under EA account	Creates objects in every domain in the forest
Run MSMQ Upgrade wizard	Command line program that comes with MSMQ	(Data pending)
Configure MSMQ site links	Active Directory Sites and Services snap-in	Creates objects in CN=MSMQ Settings, CN=<Server>, CN=Servers, CN=<Site>, CN=Sites, CN=Configuration subtree

Appendix B: Sources

The Smart Card Deployment Cookbook, Microsoft.
http://www.microsoft.com/technet/treeview/default.asp?url=/techNet/security/default.asp

Smart Card Logon While Paper, Microsoft.
http://www.microsoft.com/technet/treeview/default.asp?url=/techNet/prodtechnol/windows2000serv/deploy/confeat/sclogon.asp

Eaton, Tony. “Site Visit Summary: Virginia Polytechnic Institute and State University”. Microsoft Consulting Services. September 4, 2001.