Security Engineering Lessons Learned for Migrating Independent LANs to an Enterprise Environment

Robert L. Marchant
marchant@psu.edu
Pennsylvania State University
State College, PA 16803

Thomas Bonneau
thomas.bonneau@soteradefense.com
Sotera Defense Solutions
Herndon, VA 20171, USA

Abstract

Transition from small, independent LANs into larger enterprise managed infrastructures is becoming more prominent in academia, business and government. Consolidation of IT resources into larger, more disciplined, and more professionally managed environments has significant advantages however they do bring their own unique issues to solve in order to make the transition for the organizations involved easier. The topics covered under this paper are critical areas of concern organizations and their administrator staff needs to consider and resolve in order that transition and migration can be as painless as possible. Loosely using NIST SP 800.53 controls as a reference, the areas presented within this paper include access control mechanisms, patch management considerations, the need to address difference in hardware and software monitoring, baselines and licensing.

Keywords: LAN migration, data center consolidation, access control, patch management.

1. INTRODUCTION

Large organizations have migrated and are continuing to consolidate independent working group Local Area Networks (LANs) into more formalized hosted environments in hosting platforms ranging from simple migrations of existing LAN equipment into the enterprise network to multi-tenant virtual environments. The reason include, but certainly aren't limited to, economy of scale (e.g. sharing virtual resources, software licensing); reducing cost for space, cooling and power; sharing IT professional maintenance cost (e.g. systems administrations and help desk personnel); increased connectivity (e.g. between previously isolated LANs or to external web service hosting platforms); sharing of resources to handle surges of demand; disaster recovery and long term storage (e.g. archive).

Independent LANs are often created for ad hoc (and sometimes impromptu) purposes. The technical support hired or appointed to support this LAN are very close to the user community and understand how to prioritize the needs of the community (it is very common for a small group lab support LAN organization to anoint one of the researchers as admin who assigns userids, installs software when needed, and configures shared resources). The security controls for these environments are often understandably loose and the "bureaucracy" is

typically non-existent; after all, the focus for the support to an independent LAN is the users of the LAN. If the LAN is in place for a long duration, this researcher may even install and maintain anti-virus software, post software upgrades and patches, and check to make sure licenses are up-to-date. For many individuals responsible for standing up and maintaining these independent LANs, connecting to or becoming part of a larger enterprise might be their first exposure to enterprise discipline and to enterprise level security controls.

The initial planning for conducting a migration from an independent support environment to an enterprise environment most often focusses on the network pieces. Usually this discussion involves determining the order the pieces to be transitioned should be migrated but always involves determining what services need to be augmented when the LAN migrates.

The authors of this paper have much experience in assisting organizations in understanding the security implications of migrating to an enterprise environment. This paper presents a few lessons we have learned that, if addressed early in the migration, can ease the process for both the users of the smaller LAN and the enterprise personnel assigned to support the migration. Please note that within the paper we will discuss at a foundational level technical descriptions intended to remind the reader of what data are needed by the enterprise security engineers during a transition.

The NIST Special Publication 800.53 (National Institute of Standards and Technology [NIST], 2009), defines security controls that can be tailored to the needs of an organization. The document, to anyone but a security specialist, is tedious to read and even more tedious to implement. Fortunately, the administrators and technicians involved in the migrations are usually spared the pain of having to work the details of whatever standard the enterprise follows as this is typically the realm of the "security engineer". Although both authors are experience with NIST SP 800.53 (NIST, 2009) and all of our lessons learned relate directly to a subset of these controls, we will spare the reader the tedium of referencing the specific controls that relate to each of our lessons learned (both authors will accept e-mail questions from any adventurous reader who wishes more details on the controls). We will instead discuss our lessons learned in the three topic areas; the large topic of Access Control, the midsize topic of Patch Management, and a small discussion on Systems Monitoring, Licenses and Product Acquisition.

2. ACCESS CONTROL

Our first lesson learned is to never underestimate the complexity of coordinating identity and access control. Issues arising in the access control area almost always involve coordination of directories, authentication mechanisms, and certification authorities. Even in LANs with well administered directory control, differences in directory structures, authentication and certificate structures have to be mitigated. In this section we will briefly discuss directories, authentication services and certification systems as a way providing common ground and to illustrate all the areas where mitigation may be necessary.

Although often believed to be simply the method used to add users to a LANs domain, directory services real function is to manage information about a computer network's users and network resources, and allow network administrators to manage users' access to those network resources. A directory service is intended to interface to a directory that holds the information about named objects contained in the network. The directory service then provides the access to the data contained in one or more directory namespaces. Since directory services can be responsible for authenticating access to network resources, the directory service interface must also be responsible for ensuring secure authentication for any access to the system resources that manage the directory data.

Directory services are almost always a set of applications implemented around a specific standard such as X.500 ("Directory Service," n.d.) or LDAPv3 often provided by the operating system or database vendor. This arrangement often makes sense as a directory service is a shared information infrastructure intended to provide the namespace for the network (a namespace defines the names used to identify objects on the network) and to assist users and applications in locating, managing, administrating, and organizing common items and network resources, to include volumes, folders, files, printers, users information (e.g. ID, Access, location, phone number, picture,

etc.), groups, devices. For example, a directory may have a set of objects defined named user-ids, under user-ids may be other objects like: Surname, telephone number, company, nationality, clearance, access, and other identifying information. Administrators will set up the directory namespace using standards that are most convenient for the users they support.

Directories are usually accessed using client/server communications model. Applications read (and write) information with a call to a function or application programming interface (API). The API defines the interface for a particular programming language. The format and content of the messages exchanged between the client and the server must conform to this API and an agreed to message protocol. Obviously, LDAP provides the message protocol, and there are existing industry standards for LDAP APIs for C and Java.

Online services provided within an organization's domain can use one set of security infrastructures for authenticating and authorizing users and propagating their identity attributes (e.g. LDAP server or Windows Active Directory). Security and identity management in an enterprise environment where the entire domain is under a single authority is full of well-established technology and practice. Providing access from external web-based applications, web services, and web users (as is usually the case in an enterprise environment), creates the need to provide cross domain identity management and sharing. Differences in directory services technology, privacy and legal issues related to sharing identity information, differences in controls (and confidence among sharing organization in each other's security practices and controls) make coordinating a federated directory structure difficult (identity is federated when it is shareable across domain and platform boundaries).

As desirable as it is to share identity information, implementation is often difficult. Four technologies are most apparent at proposing solutions to this problem:

- Federated LDAP solutions: These solutions provide security applications coupled to an LDAP architecture (e.g. IBM, Sun, LINUX). Federated LDAP solutions tend to have both the advantage and the disadvantage of being tied to a specific vendor. It is usually easier for a migrating LAN to simply become a participant in the enterprise LDAP.
- Certificate based systems like Kerberos (http://web.mit.edu/kerberos/) and SESAME (http://www.cosic.esat.kuleuven.be/sesame/html/sesame_what.html).
- Public Key Cryptography (asymmetric key systems) such as public key infrastructure - PKI (Adams & Lloyd, 2003).
- XML based standards like the Security Assertion Markup Language- SAML (http://saml.xml.org/). These standards tend to be oriented towards more loosely couple computer to computer communications and tend to be more supportive to one of the three techniques above than as standalone solutions.

Regardless of which technology is used for federated identity management in the enterprise, some method of establishing and maintaining trust is essential to security of the connected systems. Kerberos is an example of an authentication service. Its purpose is to allow users and services to *authenticate* themselves to each other in a manner that is more than just providing a userid and password. In most authentication systems like Kerberos the password is a *shared secret*--something that the user and the service hold in common, and which only the server and the client know. To establishing identity in a Kerberos type system, the shared secret key is used as an encryption key; the user takes something freshly created, a timestamp for example, and encrypts it with the shared secret key. This is then sent on to the service, which decrypts it with the shared key, and recovers the timestamp. If the user used the wrong key, the timestamp won't decrypt properly, and the service can reject the user's authentication attempt.

In Kerberos, both the user and service implicitly trust an entity called the Kerberos authentication server (AS); the AS coordinates user access to all services in the system. Both the user and the service must have a shared secret key registered with the AS.

Kerberos often relies on conventional or symmetric cryptography, in which the keys used for encryption and decryption are the same. As a

result, the key must be kept secret and periodically updated. Such a requirement can be circumvented with the use of public-key cryptography, in which there are two separate keys, a public key and a private key. These two keys are asynchronous pairs: Whatever one key encrypts, the other decrypts. As their names suggest, the public key is intended to be known by anyone, whereas the private key is known only by the user.

Public-key cryptography can be integrated into the Kerberos. When the AS generates its response, encapsulating the session key, it encrypts it with a randomly generated key, which is in turn encrypted with the user's public key. The only key that can reverse this public-key encryption is the user's private key, which only he or she knows. The user thus obtains the random key, which is in turn used to decrypt the session key, and the rest of the authentication proceeds as before.

Even though the user and the AS don't have to share a long-term key, they do have to share some kind of association. Otherwise, the AS has no confidence that the public key the user is asking it to use belongs to any given identity. An impostor could easily generate a public and a private key that go together, and assert that they belong to you, and present them to the KDC to impersonate you. To prevent that, public keys have to be *certified*. Some *certification authority*, or CA, must digitally sign the public key. In essence, the CA encrypts the user's public key and identity with its *private* key, which binds the two together. Typically, the CA is someone that is trusted generally to do this very thing. Afterward, anyone can verify that the CA did indeed sign the user's public key and identity by decrypting it with the CA's *public* key. If the migrating LAN has an existing relationship with a CA, care must be taken to preserve this relationship or to carefully migrate to using the enterprise CS(s).

In reality, the CA doesn't encrypt the user's public key with its private key, for the same reasons that the KDC doesn't encrypt the session key with the user's public key. Nor does it encrypt it first with a random key, since the user's public key and identity don't have to be kept confidential. Instead, it passes the public key and identity through a special function called a one-way hash*. The hash (sometimes called a message digest) outputs a random-looking short sequence of bytes, and it's these bytes that are encrypted by the CA's private key. This establishes that only the CA could have bound the public key to the user's identity, since you can't just create any other message that also hashes to those same bytes (that's why the hash is called one-way).*

Public Key Infrastructures can be established to support more than service coordinating and authorizing. The use of PKI enables a secure exchange of digital signatures, encrypted documents, authentication and authorization, and other functions in open networks where many communication partners are involved.

PKI has four parts:

- Certificate Authority (CA)
- Registry Authority (RA) or Local Registry Authorities (LRA)
- Directory Service
- Time Stamping (as an additional service

The Certificate Authority (CA) is the entity responsible for issuing and administering digital certificates. The CA acts as the agent of trust in the PKI. A CA performs the following main functions:

- Issues users with keys/Packet Switching Exchanges (PSEs) (though sometimes users may generate their own key pair)
- Certifies users' public keys
- Publishes users' certificates
- Issues certificate revocation lists (CRLs)

The Registration Authority (RA) is responsible for recording and verifying all information the CA needs. In particular, the RA must check the user's identity to initiate issuing the certificate at the CA. This functionality is neither a network entity nor is it acting online. The RAs will be where users must go to apply for a certificate. Verification of the user identity will be done for example by checking the user's identity card. A RA has two main functions:

- Verify the identity and the statements of the claimant
- Issue and handle the certificate for the claimant

The directory service has two main functions:

- Publish certificates

- Publish a Certificate Revocation List or to make an online certificate available via the Online Certificate Status Protocol (OCSP)

Timestamping is a special service that can be used to confirm the receipt of digital documents at a specific point in time. The service is used for contracts or other important documents for which a receipt needs to be confirmed.

To migrate a LAN into an enterprise, early discussion must resolve how the LAN directory will be transitioned (or assimilated), how to interface with the enterprise's authentication service, what certification authorities are used and how will they be migrated, and how to provide any special access related services to the LAN (e.g. timestamp). If the LAN namespaces and authorities are non-compliant with enterprise standards, ensuring that the changes necessary to directories, authentication services, and certificate authorities are clearly understood and explained to the LAN users will reduce a lot of migration delay.

3. PATCH MANAGEMENT

In today's dangerous cyber world, posting patches to all software as fast as is practical is not just a good idea; it is essential (National Institute of Standards and Technology [NIST], 2005). Most administrators, even admins of the smallest of LANs, are diligent about posting updates and patches as soon as possible. Our second lesson learned is that most independent LAN administrators, especially small LANs, are not prepared for the rigorous process and the automated tools that enterprises use to post patches. Be prepared to patiently walk the LAN admins through the process; be prepared for comments like "well, we can't just post patches whenever we feel like it, our engineers sometimes have process that have been running for days and patching will cause it to crash".

Most enterprise patch managers approach patching with a disciplined process that usually includes evaluating, prioritizing, testing, implementing, and monitoring the patches. As updates are received on products ranging from operating systems to desktop applications, the enterprise process usually involves determining the necessity and priority of a patch distribution. Critical patches will be implemented immediately; others will be scheduled to take advantage of routine maintenance outages. Some application and some products patches will require testing before implementation and most enterprises have some type of test environment to conduct these test (most independent LANs don't). Implementation at the enterprise level is almost always via some automated tool like Microsoft's System Center Configuration Manager (http://www.microsoft.com/en-us/server-cloud/system-center/configuration-manager-2012.aspx) for Windows, one of the many open source or inexpensive commercial update tools for Linux, or vendor specific tools for network devices and database systems.

LAN administrators have to struggle with a couple of issues. First, their privileges will usually be more restricted than what they were used to having (enterprises typical limit "local" administrators to only the level of privilege they need). This often means the LAN administrator is no longer in control of things like what security settings are implemented and when patches are scheduled. Second, enterprises are concerned with maintaining a consistent, enterprise wide, environment. LAN administrators will no longer be in control of when a product or operating environment is upgraded. And finally, LAN administrators will have to be prepared to reassure their users that enterprise patch policy is not intrusive and will not adversely impact their productivity. Spending a little time explaining the enterprise patch management process will help the admin deal with these issues.

4. SYSTEMS MONITORING, LICENSES AND PRODUCT ACQUISTION

Enterprises monitor. Enterprises typically have operations centers that use automated tools to check systems status, collect and analyze logs, and track events. Independent LANs typically do not. Although implementing monitoring very seldom affects the migration of the LAN, it can cause some unexpected resistance if the LAN users feel their privacy is being violated.

Enterprises control licenses and product acquisitions for at least three reasons. First; the penalty for unlicensed products on an enterprise are very expensive and very embarrassing. Second; having enterprise licenses for products applies leverage on the vendor and often leads to much lower cost. Third; standardizing products reducing the maintenance load and

increases the efficiency of the patch management process.

Independent LANs however, are used to purchasing what they want, when they want it, often with little regard for registering products and keeping track of licenses. Our final lesson learned to share is that explaining the product acquisition and license maintenance process early, talking it out with the LAN admin will help considerably in diffusing this mostly emotional issue.

5. CONCLUSION

We have discussed some lessons we have learned as security engineers about supporting the migration of independent LANs into an enterprise environment. On the surface these lessons appear to have little to do with security, in reality they are all about security. Although we have spared the reader the details contained within SP 800.53, identity management, patch management, systems monitoring, audit reduction and analysis, change control and configuration management are all security controls and security issues.

We have discussed that meeting with and working with as early as possible; the administrator(s) of a migrating LAN can drastically reduce potential problems relating to directories, authentication and certificate management, patching, monitoring and acquisition. Early meetings can also reduce both the administrators' and the users' anxiety.

The authors have extensive experience in security (combined experience of over 40 years). We are often asked what "things to look out for" in transitioning systems. Each transition is, in reality, different. But almost all transitions can be (at least from the security perspective) simplified by using some form or framework to work with. The best framework is whatever framework the enterprise uses.

The lessons learned we have presented above all can be associated with security controls. The most important lesson we have learned though, is not specifically called out in a security framework. Enterprise security managers must accept risk. They expect risk to be identified and mitigated. They don't like rushed implementations and they don't like surprises. Meeting early, getting security issues addressed early, always reduces the risk that arise when transitions are "rushed", and reduces the delays that are a natural consequence of surprising security managers.

6. REFERENCES

Adams, Carlisle, & Lloyd, Steve (2003). *Understanding PKI: concepts, standards, and deployment considerations*. Addison-Wesley Professional.

Directory Service (n.d.). In Wikipedia. Retrieved July 01, 2012, from http://en.wikipedia.org/wiki/Directory Services/

National Institute of Standards and Technology (2005). *Creating a Patch and Vulnerability Management Program* (Publication No. SP 800.40 Version 2). Retrieved Jun 19, 2012, from NIST website: http://csrc.nist.gov/publications/PubsSPs.html

National Institute of Standards and Technology (2009). *Recommended Security Controls for Federal Information Systems and Organizations* (Publication No. SP 800.53 revision 3). Retrieved Jun 19, 2012, from NISTwebsite:http://csrc.nist.gov/publications/PubsSPs