Allowing Linux to Authenticate to a Windows 2003 AD Domain

Prepared by

Thomas J. Munn,
CISSP
11-May-06

Table of Contents:

Introduction

Active Directory Integrating LINUX servers with group-based restrictions for logins using CENTOS

One of the main problems with UNIX/Windows environments is the lack of integration between the two platforms. Userids have to be created separately on each environment, passwords changed separately, etc. This doubles administrative work. This paper will explore using one of several different ways that you can active directory integrate your LINUX boxes to your windows AD forest. This document will give you integration between your linux boxes and your Windows AD forest. Additionally, it will allow you to control who can login to the LINUX boxes by group memberships within Active Directory. It will give you full 'password' integration, including the ability to change NT passwords from linux. It will also provide redundant Kerberos servers, so that authentication will be available if a primary KDC goes down.

Method	Pros	Cons
Kerberos Only	Fewer packages Credentials protected	No automatic home directory creation, AD support klunky and requires lots of effort to get machine 'into' ad domain Requires schema updates in windows AD
LDAP only	Tends to work 'out of the box'	Credentials sent 'in the clear' unless specifically protected against Requires schema updates in windows AD
Winbind/Kerberos	Automatic mapping of uid/gid pairs from nt/unix, passwords secured by Kerberos, computers appear in AD like normal windows boxes, automatic creation of Home directories, password synchronization, and ability to change password from within LINUX.	Can be more complex, especially if you want to have consistent uid/gid mappings across unix boxes.

Requirements

This paper assumes that you are going to be using CENTOS, a free 'redhat' enterprise clone. It is available at http://www.centos.org

This paper also is not going to cover storing idmap (used for mapping unix ids/groups to nt groups) globally. It will only cover using 'local' idmap files for authentication on UNIX workstations. If you are using NIS, shares, etc. this solution will NOT work, as the GID/UID mappings will be different on each machine.

The recipe:

Winbind- Daemon that maps unix/nt ids, usernames, and allows AD integration
Kerberos- used to authenticate to AD domain by winbind
Samba- has tools (including winbind) that let the whole shebang work
1 AD w3k domain, set to use Kerberos authentication
1 group that you choose to allow to login into the workstation (setup on nt)
1 ntp service setup on UNIX box to ensure time is set correctly (Kerberos will NOT work if there is clock skew!)
1 working CENTOS 4.2 or > installation
1 working pam.d configuration
1 working DNS server (ABSOLUTELY REQUIRED)!

Packages you will need to use this guide: (listed for centos 4.3), names may have changed

Krb5-workstation
Krb5-libs
Pam_krb5
Samba-common
** Samba
NTP (Kerberos is really picky about clock drift!, this service will synchronize your linux box to your Kerberos server (aka Domain controller in microsoftese)

Installing the Necessary Packages

So, to begin the installation, we need to make sure that these packages are installed. Type in this command: (as root!)

rpm –qa | grep krb5

The system should report at least 3 packages:

krb5-workstation
pam_krb5
krb5_libs

If they are missing, simply type in:

yum install packagenamethatismissing

So, if you are missing pam_krb5 you would type in

yum install pam_krb5

These will have some extra numbers after the listing above, versions don't really matter. Finally, we need to ensure that the winbind utilities are installed as well, these are included in SAMBA.

rpm –qa | grep samba

System should return at least 2 packages:

Samba-common
Samba-

If any are missing, just follow the 'yum install' directions above, under the above section.

Samba MUST be > version 3.0.9 to work correctly, as AD integration with 2003 requires samba version 3.

Finally, ensure that NTP is installed:

rpm –qa | grep ntp

You will see if the program is installed. If nothing comes up, then you haven't installed it yet. If it is missing, just follow install directions above.

The package looks like ntp-4.2.0.a the only important part is the ntp- the other information is just the version.

Changing the system config files:

Although I haven't done it directly, centos provides a utility that can do most of the configuration changes for you:

authconfig

I will not cover it here, but I list it for the more adventurous reader here.

krb5.conf

We will be editing the /etc/krb5.conf file first. Here is what the file should look like after we are done:

```
[logging]
 default = FILE:/var/log/krb5libs.log
 kdc = FILE:/var/log/krb5kdc.log
 admin_server = FILE:/var/log/kadmind.log

[libdefaults]
 default_realm = YOUR.DEFAULT.DOMAIN
 dns_lookup_realm = true
 dns_lookup_kdc = true

[realms]
 YOUR.DEFAULT.DOMAIN = {
  kdc = your.primary.dc:88
  kdc = your.secondary.dc:88
  admin_server = your.primary.dc:749
  default_domain = your.default.domain
 }

[domain_realm]
 .your.default.domain = YOUR.DEFAULT.DOMAIN
 your.default.domain= YOUR.DEFAULT.DOMAIN

[kdc]
 profile = /var/kerberos/krb5kdc/kdc.conf

[appdefaults]
 pam = {
   debug = false
   ticket_lifetime = 36000
   renew_lifetime = 36000
   forwardable = true
   krb4_convert = false
 }
```

Several lines deserve discussion:

The [libdefaults] section:

The ‘default’ realm is simply the same as your fully qualified domain within nt, so for the example, corp.somewhere.com is our NT domainname, so Kerberos convention for ‘realm’ is to simply uppercase it. This makes it easier to tell the difference between the ‘realm’ and the ‘domain’ which are really the same thing.

It is CRITICAL to note that if DNS isn’t working, this file will fail MISERABLY due to its reliance on domain name service for resolving Kerberos services. The computer being added should be in DNS as well!

The [realms] sections

The first line is simply your uppercased nt domain name, in this case, YOUR.DEFAULT.DOMAIN

The kdc lines are particularly important. They tell the server where to talk to the Kerberos server for authentication. Note that there are TWO of these. The first is the primary, and the second one is the ‘slave’ or failover kdc should the primary be down. There are no limits to the number of these that you can have. Again, this would be the fully qualified domain name of the primary KDC (aka domain server) in your organization

The admin server line is always the ‘primary’ kdc in your organization.

The default domain, is simply your lowercased NT domain, and the primary ‘realm’, or ‘forest’ that you want users of this server to authenticate to.

The [domain realms] section

This section is basically a mapping between Kerberos and NT. You can provide multiple domains here. This is not covered. Minimally, you need the two entries listed on the previous page, forming :

.your.first.domain = YOUR.FIRST.DOMAIN
your.first.domain = YOUR.FIRST.DOMAIN

the . in the first entry is critical!

The remaining sections simply copy as ‘is’. If you are curious, you can simply type in ‘man krb5.conf’ at a unix prompt for more information.

nsswitch.conf

This file is used to tell the system where to look for users, groups, etc. It lives in the /etc directory. It needs to have the following lines edited/added if they aren’t present:

```
passwd:     files winbind
shadow:     files winbind
group:      files winbind
protocols:  files winbind
services:   files winbind
netgroup:   files winbind
automount:  files winbind
```

These lines tell the system to look to files, then to winbind. It should be noted that if you have a local group, for instance, and the same group in NT, that the LOCAL group will be used first! (Goes to files, then winbind). I found this out when I added a local ‘unix’ group, and couldn’t authenticate because I required users to be a

member of the 'unix' group. The system saw my local file, and the userid from the NT domain wasn't in the local file! So bear this in mind with group memberships.

system-auth (lives in /etc/pam.d directory)

This file is what tells the system how to process authentication information, group and login information etc. Yours should be like this:

```
auth        required      /lib/security/$ISA/pam_env.so
auth        sufficient    /lib/security/$ISA/pam_unix.so likeauth nullok
auth        sufficient    /lib/security/$ISA/pam_krb5.so use_first_pass
auth        sufficient    /lib/security/$ISA/pam_winbind.so use_first_pass
auth        required      /lib/security/$ISA/pam_deny.so

account     required      /lib/security/$ISA/pam_unix.so broken_shadow
account     sufficient    /lib/security/$ISA/pam_succeed_if.so uid < 100 quiet
account     [default=bad success=ok user_unknown=ignore] /lib/security/$ISA/pam_krb5.so
account     [default=bad success=ok user_unknown=ignore]
/lib/security/$ISA/pam_winbind.so
account     required      /lib/security/$ISA/pam_permit.so
account     requisite     /lib/security/$ISA/pam_succeed_if.so user ingroup unix
password    requisite     /lib/security/$ISA/pam_cracklib.so retry=3
password    sufficient    /lib/security/$ISA/pam_unix.so nullok use_authtok md5 shadow
password    sufficient    /lib/security/$ISA/pam_krb5.so use_authtok
password    sufficient    /lib/security/$ISA/pam_winbind.so use_authtok
password    required      /lib/security/$ISA/pam_deny.so

session     required      /lib/security/$ISA/pam_limits.so
session     required      /lib/security/$ISA/pam_unix.so
session     optional      /lib/security/$ISA/pam_mkhomedir.so skel=etc/skel/ umask=0027
session     optional      /lib/security/$ISA/pam_krb5.so
```

There are several lines that deserve explanation:

1. pam_mkhomedir line creates the initial homedirectory if the user doesn't have one. It gets its default files from the /etc/skel directory, and allows user to rw and group to read only, other has no permission at all. To allow other 'read' simply change the 7 to a 2.
2. pam_succeed_if restricts who can login to this box. If you don't have it, anyone in the domain will be able to login. In this case, only users who are a member of the nt group 'unix' will be able to login to this box. The exact documentation for pam_succeed_if lives in /usr/share/doc/pam-0.77/txts/README.pam_succeed_if The name may change, just use the 'locate pam_succeed' and look for readme file.

Smb.conf file (lives in /etc/samba/smb.conf)

This file is where a lot of the magic happens. It allows winbind to register the computer with the domain, and helps in mapping UID-GID pairs from nt to unix. We will only be editing the [global] section of this file

```
[global]

   workgroup = YOUR.DEFAULT.DOMAIN (old style)
   netbios name=LILLY (computer name)
   server string = Samba Server
```

The ‘workgroup’ should be the ‘short name’ of the domain that you are wanting to register the computer with. The ‘netbios’ name is the name that the computer will get when you create the account in AD, as if you were joining a windows workstation to a domain (computer name). The ‘server string’ really doesn’t matter, just leave it.

```
security = ads
realm=YOUR.DEFAULT.DOMAIN
encrypt passwords = yes
smb passwd file = /etc/samba/smbpasswd
allow trusted domains = yes
unix password sync = Yes
passwd program = /usr/bin/passwd %u
passwd chat = *New*password* %n\n *Retype*new*password* %n\n
*passwd:*all*authentication*tokens*updated*successfully*
pam password change = yes
obey pam restrictions = yes
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
dns proxy = no
idmap uid = 16777216-33554431
idmap gid = 16777216-33554431
winbind use default domain = yes
winbind separator = #
winbind enum users = yes
winbind enum groups = yes
template shell=/bin/bash
template homedir = /home/%U
```

The ‘security’ type being set to ads uses active directory for authentication. Also be sure to run smbpasswd utility to create the smbpasswd file!

The unix password sync is really neat because it allows windows and unix passwords to be in sync! You can even use passwd to change your NT password!

The ‘realm’ is set to the uppercased nt domain name e.g. here.is.domain.

The other item that bears some mentioning is the idmap uid and idmap gid entries. These files allow the system to map NT RID ids to Unix GID and UID’s. This is winbind’s job, and these entries allow things to happen. One problem that I had was one of the FAQ’s said to use IDs from 10000-1000000, and these weren’t high enough. The entries above were obtained by using the authconfig utitlity, and patching them into the config, as you see there. I got an error message “unable to allocate group”, when I used incorrect values. Just a note.

Putting it all together

Once these files are added, you need to turn on winbind manually for the test:

/etc/init.d/winbind start

You will now want to try to add the computer to the domain. You will need a account with domain admin privileges to do this:

net ads join –U administrator

This will join the computer to the domain, and will prompt you for the administrator password. It should work successfully. If not, look at logfiles, conf files, and ensure they match this guide, start again!

To see if you are able to see groups, just type in

wbinfo –g (this will list groups in domain)

If this works, you are now almost ready!

The last thing to note is that with the system-auth configuration as noted in this guide requires that users who want to login to this workstation be a member of the 'unix' group in NT. If they aren't you will NOT be able to login. Also, if you change group memberships from a person it is a good idea to restart winbind with the following command:

/etc/init.d/winbind restart

Finally, the RPM should add winbind to the default runlevel, but to check if it is: type in

chkconfig | grep winbind

You will see a listing like:

Winbind 0:off 1:off 2:off 3:on 4:on 5:on 6:off

That is how it should look if it doesn't, here is the command to put it proper runlevels:

chkconfig –level 345 winbind on

That's it. You should now be able to login to the domain using NT accounts, who are a member of the 'unix' group!

Appendix A: Additional Resources

A MUCH more complex and complete howto that I used to do most of this, prepared by Edwin Gnichtel is available at http://www.thelazyadmin.com/index.php?archives/381-LinuxUnix-Active-Directory-Authentication-Integration-Part-1.html

You might want to google for 'winbind howto' as well go to www.samba.org for more information.