

DOKUMENTATION SQUID PROXY CACHE MIT EINBINDUNG WINDOWS AD

Debian Lenny 5.0.6

Linux 2.6.26-2-686 #1 SMP Mon Aug 30 07:01:57 UTC 2010 i686 GNU/Linux

Squid 2.7.STABLE3

Samba 3.2.5-41

Kerberos krb5-config 1.22 / krb5-user 1.6.dfsg.4

Winbind 3.2.5-41

Für das Verständniss dieses Dokuments wird der Umgang mit Netzwerken, Linux Debian, Windows Domänen und VMware ESXI Servern vorausgesetzt. Hilfreich sind Kenntnisse in Kerberos, Samba und Squid.

INHALTSVERZEICHNIS

1	Installation Server	3
1.1	Installation VM	3
1.2	Installation benötigte Programme	5
1.3	Konfiguration der Programme	7
1.3.1	Allgemeine Hinweise	7
1.3.2	etc/samba/smb.conf	7
1.3.3	etc/krb5.conf – Kerberos Konfiguration.....	8
1.3.4	Konfiguration Winbind	10
1.3.5	etc/nsswitch.conf	10
1.4	Testen der Konfiguration	11
1.5	Konfiguration Squid	13
1.6	Last Words	15
2	Anhang	16
2.1	http://wiki.samba.org/index.php/Samba_&_Active_Directory	16
2.2	http://www.cyberciti.biz/faq/squid-ntlm-authentication-configuration-howto/	23

1 INSTALLATION SERVER

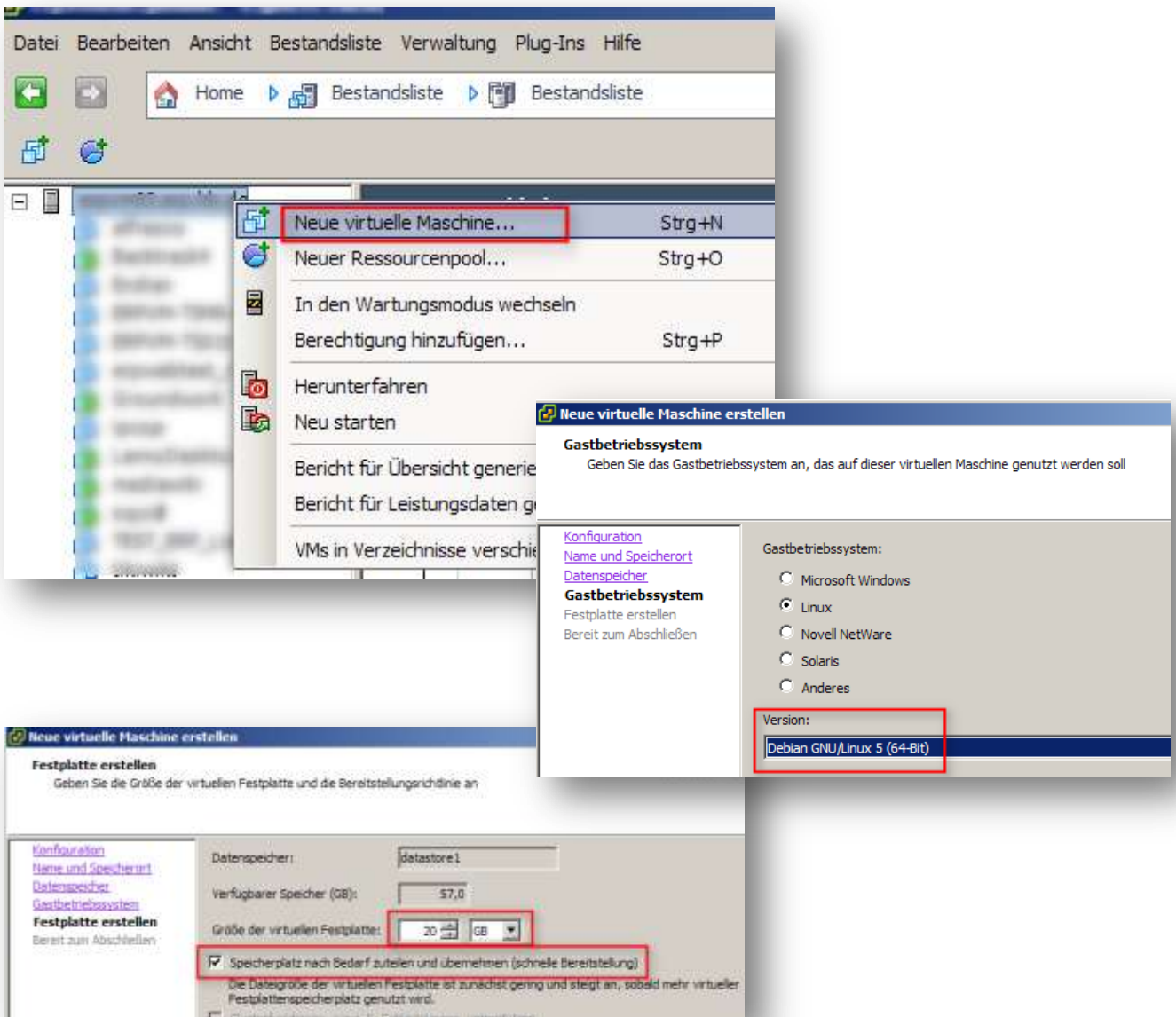
Als Basis wird ein aktuelles Debian Netinst. verwendet. Zu beziehen ist dies unter <http://www.debian.org/CD/netinst/>

Es wird eine VM unter ESXi installiert. Ich empfehle Squid erst ohne Authentifizierung, zur besseren Fehlersuche, zu testen

Eine funktionierende DNS Umgebung ist für die Kommunikation mit dem AD *zwingende* Voraussetzung.

1.1 Installation VM

Im Folgenden nur die „wichtigen“ Screenshots.

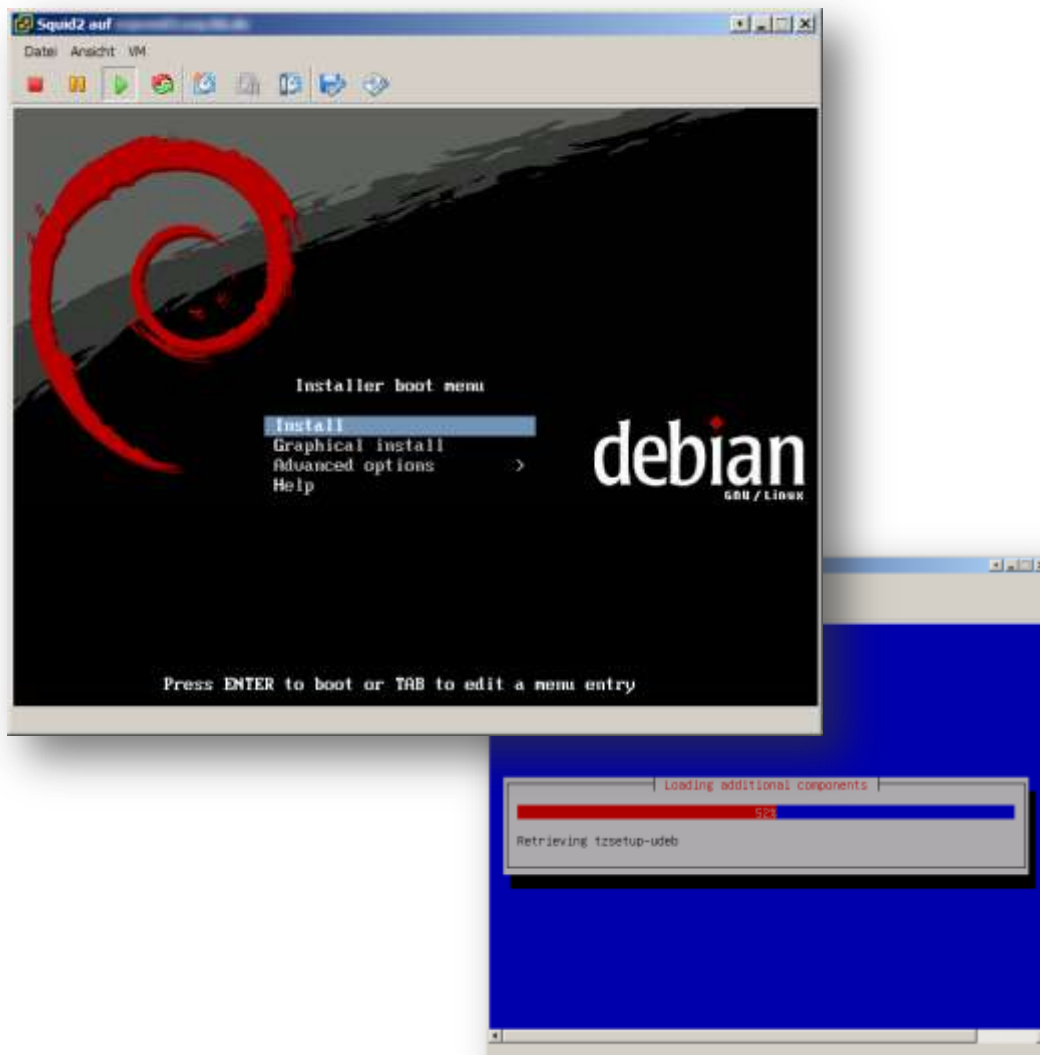


Für Testzwecke ist ein „Speicherplatz nach Bedarf zuteilen“ in Ordnung. Für die produktive Maschine sollte man die Festplatte komplett zuweisen.

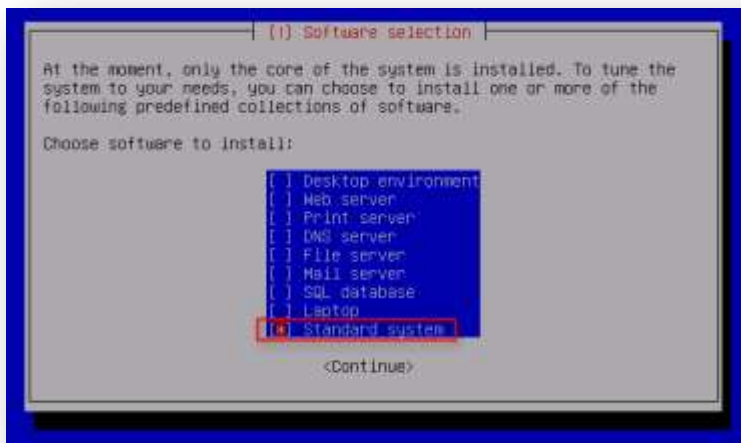
Anschließend in den Eigenschaften der VM noch die passenden Einstellungen vornehmen (Arbeitsspeicher, CPUs, CD/DVD Laufwerk (vorweg geladenes Debian Netinst. ISO einbinden)).



Nun kann die virtuelle Maschine gestartet werden.



Als Vorschlag zur Installation nur das Standard System auswählen (SPACE wählt an und ab)



Als root anmelden und ssh installieren, die weiteren Schritte sind dann mit Putty einfacher.

```
apt-get install ssh
```

1.2 Installation benötigte Programme

```
squid2:~# apt-get update
Hit http://ftp.de.debian.org lenny Release.gpg
Ign http://ftp.de.debian.org lenny/main Translation-en_US
Hit http://ftp.de.debian.org lenny Release
Hit http://security.debian.org lenny/updates Release.gpg
Ign http://security.debian.org lenny/updates/main Translation-en_US
Ign http://ftp.de.debian.org lenny/main Packages/DiffIndex
Hit http://security.debian.org lenny/updates Release
Ign http://ftp.de.debian.org lenny/main Sources/DiffIndex
Ign http://security.debian.org lenny/updates/main Packages/DiffIndex
Hit http://ftp.de.debian.org lenny/main Packages
Ign http://security.debian.org lenny/updates/main Sources/DiffIndex
Hit http://volatile.debian.org lenny/volatile Release.gpg
Hit http://ftp.de.debian.org lenny/main Sources
Hit http://security.debian.org lenny/updates/main Packages
Hit http://security.debian.org lenny/updates/main Sources
Ign http://volatile.debian.org lenny/volatile/main Translation-en_US
Hit http://volatile.debian.org lenny/volatile Release
Ign http://volatile.debian.org lenny/volatile/main Packages/DiffIndex
Ign http://volatile.debian.org lenny/volatile/main Sources/DiffIndex
Hit http://volatile.debian.org lenny/volatile/main Packages
Hit http://volatile.debian.org lenny/volatile/main Sources
Reading package lists... Done
```

```
squid2:~# apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

```
# apt-get install httpd ntp squid samba winbind krb5-config krb5-user
```

Im weiteren Verlauf folgen ein paar Fragen zu den installierenden Paketen, hier kann entweder der Default Wert eingetragen werden oder sinnvolle Werte. Die entsprechenden Eintragungen in den Configdateien von Samba, Winbind und Kerberos werden im weiteren Verlauf angepasst.

```
##### Samba Server #####
A
A If your computer gets IP address information from a DHCP server on the
A network, the DHCP server may also provide information about WINS servers
A ("NetBIOS name servers") present on the network. This requires a change
A to your smb.conf file so that DHCP-provided WINS settings will
A automatically be read from /etc/samba/dhcp.conf.
A
A The dhcp3-client package must be installed to take advantage of this
A feature.
A
A Modify smb.conf to use WINS settings from DHCP?
A
A      <Yes>          <No>
A
#####
```

```
##### Configuring krb5-config #####
A
A Enter the hostnames of Kerberos servers in the [realm] Kerberos realm separated by spaces.
A
A Kerberos servers for your realm:
A
A _____
A
A
A      <Ok>
A
#####
```

```
##### Configuring krb5-config #####
A
A Enter the hostname of the administrative (password changing) server for the [realm] Kerberos realm.
A
A Administrative server for your Kerberos realm:
A
A _____
A
A
A      <Ok>
A
#####
```

OK, die benötigten Programme sind installiert.

Htop – ist ein persönlicher Favorit von mir

Ntp – wird benötigt damit die Uhrzeit synchronisiert ist, damit im weiteren Verlauf die Kommunikation mit der Windows AD Domäne funktioniert.

Squid – ja, wir wollten ja einen Proxy installieren ;-)

samba winbind krb5-config krb5-user – werden für die Anbindung an die AD benötigt.

1.3 Konfiguration der Programme

Zuallererst müssen die Daemons gestoppt werden.

```
squid2:~# /etc/init.d/samba stop
Stopping Samba daemons: nmbd smbd.
```

```
squid2:~# /etc/init.d/winbind stop
Stopping the Winbind daemon: winbind.
```

1.3.1 Allgemeine Hinweise

Die Konfiguration der Dienste ist diffizil und fehleranfällig. Deswegen im weiteren Verlauf Beispiele der Configdateien für ein Copy und Paste. Diese Dokumentation ist auf einer neuen Serverversion erfolgreich getestet worden. Ich hoffe die Namen für workgroup, realm, etc. sind sprechend.

1.3.2 /etc/samba/smb.conf

```
squid:~# cat /etc/samba/smb.conf
```

```
[global]
workgroup = DOMAENEKURZNAME
realm = DOMAENE.FQDN.TLD
preferred master = no
security = ADS
encrypt passwords = true
winbind separator = +
idmap uid = 600-20000
idmap gid = 600-20000
client ntlmv2 auth = yes
```

```
[homes]
valid users = %S
```

Kurze Erklärung:

Wichtig ist die GROßSCHREIBUNG der workgroup und des realms, sonst funktioniert später die Einbindung in die AD nicht.

Preferred master ist optional, würde ich aber auf jeden Fall setzen – wenn der Linux host der Master ist wird wahrscheinlich das anmelden in der Domäne schwierig ;-)

Der winbind seperator spielt nachher in der Kerberos und Squid Konfiguration eine wichtige Rolle, der gewohnte Backslash „\“ ist kaum brauchbar, da man immer daran denken müsste ihn durch einen vorangestellten „\“, also „\\“ in den Configs brauchbar zu machen. Von daher ist ein Pluszeichen i.O.

Die Parameter idmap_uid und idmap_gid haben mit den Informationen zu tun die der Linux Server aus der AD zieht, ist z.B der erste Wert (600) zu hoch gesetzt können keine Gruppen oder Benutzerinformationen übertragen werden. Diese Werte stammen aus dem Samba Wiki (http://wiki.samba.org/index.php/Samba_&_Active_Directory) und funktionieren. Die komplette Anleitung findet sich im Anhang (2.1).

Client ntlmv2 auth legt die Kommunikation mit dem AD auf das sichere NTLMv2 fest.

1.3.3 /etc/krb5.conf – Kerberos Konfiguration

```
squid:~# cat /etc/krb5.conf
[logging]
default = FILE:/var/log/krb5libs.log
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log

[libdefaults]
    default_realm = DOMAENE.FQDN.TLD
    dns_lookup_realm = false
    dns_lookup_kdc = false
    ticket_lifetime = 24h
    forwardable = yes

# The following krb5.conf variables are only for MIT Kerberos.
    krb4_config = /etc/krb.conf
    krb4_realms = /etc/krb.realms
    kdc_timesync = 1
    ccache_type = 4
    forwardable = true
    proxiable = true

# The following encryption type specification will be used by MIT Kerberos
# if uncommented.  In general, the defaults in the MIT Kerberos code are
# correct and overriding these specifications only serves to disable new
# encryption types as they are added, creating interoperability problems.
#
# This is only time when you might need to uncomment these lines and change
# the enctype is if you have local software that will break on ticket
# caches containing ticket encryption types it doesn't know about (such as
# old versions of Sun Java).

#    default_tgs_enctypes = des3-hmac-sha1
#    default_tkt_enctypes = des3-hmac-sha1
#    permitted_enctypes = des3-hmac-sha1

# The following libdefaults parameters are only for Heimdal Kerberos.
    v4_instance_resolve = false
    v4_name_convert = {
        host = {
            rcmd = host
            ftp = ftp
        }
        plain = {
            something = something-else
        }
    }
    fcc-mit-ticketflags = true

[realms]
    DOMAENE.FQDN.TLD = {
        kdc = dc.domaene.tld
        admin_server = dc.domaene.tld
```

```

}

[domain_realm]
.mit.edu = ATHENA.MIT.EDU
mit.edu = ATHENA.MIT.EDU
.media.mit.edu = MEDIA-LAB.MIT.EDU
media.mit.edu = MEDIA-LAB.MIT.EDU
.csail.mit.edu = CSAIL.MIT.EDU
csail.mit.edu = CSAIL.MIT.EDU
.who.edu = ATHENA.MIT.EDU
who.edu = ATHENA.MIT.EDU
.stanford.edu = stanford.edu
.slac.stanford.edu = SLAC.STANFORD.EDU

[login]
krb4_convert = true
krb4_get_tickets = false

```

Ich habe zu der Standardconfig die Domäne hinzugefügt, und teilweise die Einträge belassen, zur besseren Übersicht hier nochmals die wichtigen Einstellungen separiert.

```

[realms]
DOMAENE.FQDN.TLD = {
    kdc = dc.domaene.tld
    admin_server = dc.domaene.tld
}

[libdefaults]
default_realm = DOMAENE.FQDN.TLD
dns_lookup_realm = false
dns_lookup_kdc = false
ticket_lifetime = 24h
forwardable = yes

```

Auch das Logging habe ich hinzugefügt.

```

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

```

Falls Fragen auftauchen empfehle ich nach /etc/krb5.conf und den entsprechenden Schlagwörtern (kdc, admin_server) zu googlen, eine ausreichende Erklärung ist in diesem Dokument nicht möglich. Das Thema ist zu umfangreich.

Die Einstellungen hierzu sind wieder aus dem Samba Wiki

1.3.4 Konfiguration Winbind

```
4450 proxy 20 0 7912 5548 1760 S 0.0 0.5 0:09.23 (squid) -D -YC
4471 proxy 20 0 1612 324 268 S 0.0 0.0 0:00.00 (unlinkd)
4608 proxy 20 0 9096 2160 1800 S 0.0 0.2 0:00.36 (ntlm_auth) --helper
4609 proxy 20 0 9048 1940 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4610 proxy 20 0 9048 1936 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4611 proxy 20 0 9048 1936 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4613 proxy 20 0 9048 1936 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4616 proxy 20 0 9048 1940 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4617 proxy 20 0 9048 1936 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4618 proxy 20 0 9048 1936 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4619 proxy 20 0 9048 1940 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4620 proxy 20 0 9048 1940 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4621 proxy 20 0 9048 1936 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4622 proxy 20 0 9048 1940 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4623 proxy 20 0 9048 1940 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4624 proxy 20 0 9048 1944 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
4626 proxy 20 0 9048 1936 1608 S 0.0 0.2 0:00.00 (ntlm_auth) --helper
F1Help F2Setup F3Search F4Invert F5Tree F6SortBy F7Nice -F8Nice +F9Kill F10Quit
```

Squid und seine Helfer laufen als User Proxy auf dem Linuxserver, daher muss mit

`gpasswd -a proxy winbindd_priv` auf der Kommandozeile (Putty) der User proxy der Gruppe winbindd_priv hinzugefügt werden.

1.3.5 /etc/nsswitch.conf

In der /etc/nsswitch.conf muss winbind zu passwd, group und shadow hinzugefügt werden, hier die Ansicht der bearbeiteten Datei.

```
squid:~# cat /etc/nsswitch.conf
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the `glibc-doc-reference' and `info' packages installed, try:
# `info libc "Name Service Switch"' for information about this file.

passwd:          compat winbind
group:           compat winbind
shadow:          compat winbind

hosts:           files dns
networks:        files

protocols:       db files
services:        db files
ethers:          db files
rpc:             db files

netgroup:        nis
```

1.4 Testen der Konfiguration

Die Dienste bitte in der folgenden Reihenfolge starten

```
/etc/init.d/samba start  
/etc/init.d/winbind start
```

Jetzt kann der Linuxserver als Mitgliedsserver in die Windows AD gefahren werden.

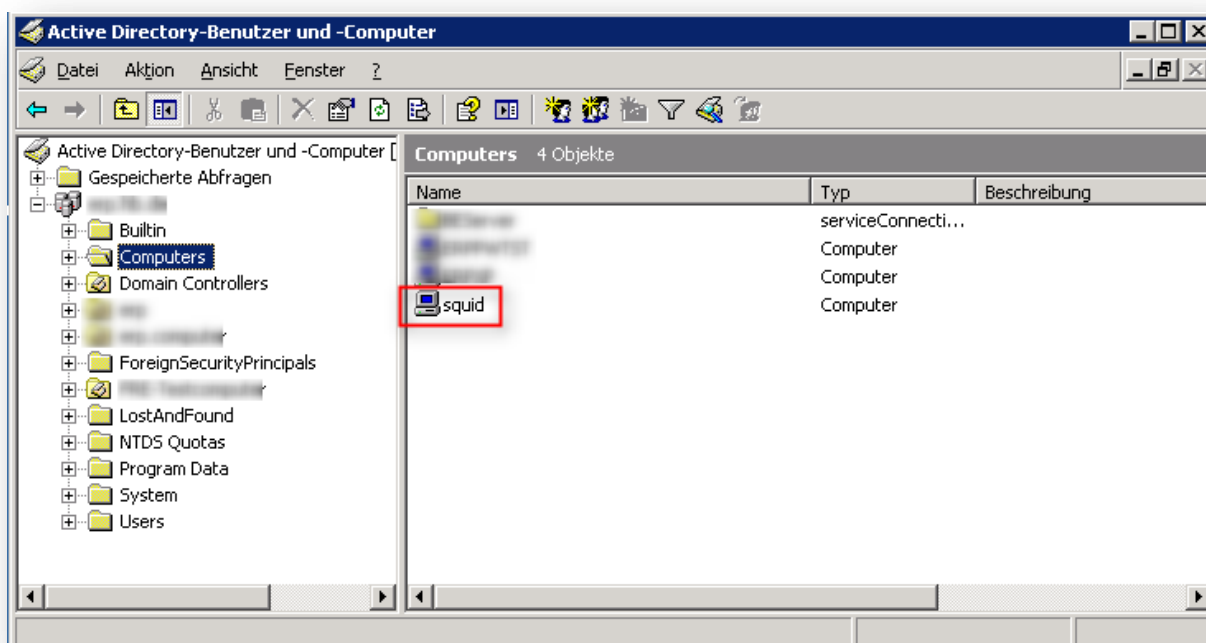
```
net ads join -U Adminaccount
```

Hierbei wird das Passwort abgefragt.¹

Hat das funktioniert liefert das folgende Kommando einen Erfolg.

```
squid:~# net ads testjoin  
Join is OK
```

Gleichzeit sollte der Server in der AD auftauchen.



Folgend ein paar Kommandos, mit denen sich eine erfolgreiche Verbindung testen lässt.

```
squid:~# kinit testuser@DOMAENE.FQDN.TLD  
Password for testuser@DOMAENE.FQDN.TLD:
```

```
squid:~# kinit -V  
Password for testuser@DOMAENE.FQDN.TLD:  
Authenticated to Kerberos v5
```

...zeigt eine erfolgreiche Erstellung eines Kerberos Tickets, was will der Admin mehr ;-)

¹ Ich habe bei dieser Gelegenheit mit tcpdump die Pakete mitgeschnitten und diese mit Wireshark untersucht. Mein Passwort konnte ich nicht entdecken, daher scheint die Verschlüsselung zu funktionieren ;-)

An dieser Stelle ist ein Neustart des Servers erforderlich, fragt mich nicht warum ;-)) ich weiss es nicht. Ich habe nur reproduzierbar festgestellt das wbinfo erst Info's liefert, nach dem der Server neu gestartet wurde.

wbinfo -u liefert die User der Domäne

```
squid:~# wbinfo -u
DOMAENE+testuser
DOMAENE+gast
DOMAENE+support_388945a0
DOMAENE+testuser2
DOMAENE+administrator
```

wbinfo -t liefert das Ergebniss zum Trust

```
squid:~# wbinfo -t
checking the trust secret via RPC calls succeeded
```

wbinfo -g liefert die Gruppen

```
squid:~# wbinfo -g
DOMAENE+sqlserver2005sqlbrowseruser$...
DOMAENE+sqlserver2005mssqlserveradhelperuser$...
DOMAENE+sqlserver2005mssqluser$erpc02$...
DOMAENE+sqlserver2005msfteuser$erpc02$...
DOMAENE+domÄnencomputer
DOMAENE+domÄnencontroller
DOMAENE+zertifikatherausgeber
DOMAENE+domÄnen-admins
DOMAENE+domÄnen-benutzer
...
DOMAENE+inet
DOMAENE+domÄnen-gÄste
DOMAENE+ras- und ias-server
DOMAENE+wins-benutzer
DOMAENE+dnsadmins
```

Wbinfo -a macht eine Probeanmeldung

```
squid:~# wbinfo -a ERP+testuser%Passwort
plaintext password authentication failed
Could not authenticate user testuser with plaintext password
challenge/response password authentication succeeded
```

Bei dieser Ausgabe ist zu sehen das Plaintext nicht funktioniert (Domäneneinstellung), eine Anmeldung aber doch (challenge/response password authentication succeeded (in diesem Fall über das sichere NTLMv2))

Okay, damit ist die Testphase abgeschlossen, der Server ist Mitglied der Domäne und kann erfolgreich Nutzer authentifizieren. Jetzt kann Squid eingerichtet werden.

1.5 Konfiguration Squid

Die Squid Konfiguration ist in der Datei `/etc/squid/squid.conf` gesammelt. Diese Datei ist sehr umfangreich und nicht einfach zu konfigurieren. Anbei meine geänderten Zeilen die eine erfolgreiche Authentifizierung mit der Domäne bewirken. (Bei Zeilenumbrüchen bitte die Zeile als Ganzes betrachten). Ich gehe die `squid.conf` von oben nach unten durch.

Zur besseren Übersichtlichkeit kann man nach dem **# fettgedruckten Text** suchen, dann ist man schneller im entsprechenden Block

Hinweise zur Formatierung:

Ist die Zeile in der Configdatei

Sind meine Anmerkungen (darunter)

Änderungen /etc/squid/squid.conf

```
# WELCOME TO SQUID 2.7.STABLE3 -----
```

OPTIONS FOR AUTHENTICATION

```
auth_param ntlm program /usr/bin/ntlm_auth --helper-protocol=squid-2.5-ntlmssp --
require-membership-of="DOMAENE+inet"
```

--require-membership-of="DOMAENE+inet" steht für die Gruppe in der Domäne, in diesem Fall also die Gruppe INET. Benutzer die Mitglieder dieser Gruppe sind wird der Zugang über den Proxy gewährt

```
#auth_param ntlm program /usr/bin/ntlm_auth --helper-protocol=squid-2.5-ntlmssp
```

Dieser Parameter ist auskommentiert, bedeutet aber das jedem authentifizierten Nutzer das Surfen erlaubt wird.

```
auth_param ntlm children 15
```

Anzahl der Threads, die Squid öffnen darf um Anfragen an das AD zu stellen. Bei Bedarf sollten diese erhöht werden. Erfahrungen dazu habe ich noch nicht.

```
auth_param ntlm keep_alive on
```

TAG: acl

```
# Defining an Access List
```

```
...
```

```
acl AuthorizedUsers proxy_auth REQUIRED
```

Damit wird die ACL Auth.Users kreiert, über `proxy_auth` wird auf `auth_param ntlm` verwiesen. Es heisst also nichts anderes als definiere eine Gruppe Auth.User, die sich zuvor über `proxy_auth` authentifiziert haben.

```
acl SSL_ports port 443 # https
acl SSL_ports port 563 # snews
acl SSL_ports port 873 # rsync
acl Safe_ports port 80 # http
acl Safe_ports port 21 # ftp
acl Safe_ports port 443 # https
acl Safe_ports port 70 # gopher
acl Safe_ports port 210 # wais
acl Safe_ports port 1025-65535 # unregistered ports
```

```

acl Safe_ports port 280          # http-mgmt
acl Safe_ports port 488          # gss-http
acl Safe_ports port 591          # filemaker
acl Safe_ports port 777          # multiling http
acl Safe_ports port 631          # cups
acl Safe_ports port 873          # rsync
acl Safe_ports port 901          # SWAT
acl purge method PURGE
acl CONNECT method CONNECT

```

Bei den Ports habe ich nichts verändert, die Liste scheint vollständig. Bei Bedarf an weiteren Ports, diese hier anfügen.

TAG: http_access

```
http_access allow all AuthorizedUsers
```

Über diese http_access Regel wird den Auth.Usern der Zugriff gewährt (weiter oben wurde Auth.Users definiert). Benutzer müssen in der Gruppe DOMAENE\inet sein.

```

# http_access allow localnet
http_access allow localhost

```

Localhost darf auch

```

# And finally deny all other access to this proxy
http_access deny all

```

der verbleibende Rest darf nicht mehr

TAG: http_port

```

#      Usage:  port [options]
#             hostname:port [options]
#             1.2.3.4:port [options]
# Squid normally listens to port 3128
http_port 3128

```

Port bei Bedarf verändern, z.B. 8888

MEMORY CACHE OPTIONS

```

# -----
# TAG: cache_mem          (bytes)
#Default:
cache_mem 16 MB
# maximum_object_size_in_memory 8 KB
# memory_replacement_policy lru

```

DISK CACHE OPTIONS

```

# -----
# TAG: cache_replacement_policy
#Default:
# cache_replacement_policy lru

#Default:
# cache_dir ufs /var/spool/squid 100 16 256
#Default:

```

```
# minimum_object_size 0 KB

#Default:
# maximum_object_size 20480 KB

#Default:
# cache_swap_low 90
# cache_swap_high 95
```

LOGFILE OPTIONS

```
# -----

# TAG: logformat

# The default formats available (which do not need re-defining) are:
#
#logformat squid %ts.%03tu %6tr %>a %Ss/%03Hs %<st %rm %ru %un %Sh/%<A %mt
#logformat squidmime %ts.%03tu %6tr %>a %Ss/%03Hs %<st %rm %ru %un %Sh/%<A %mt
[%>h] [%<h]
#logformat common %>a %ui %un [%tl] "%rm %ru HTTP/%rv" %Hs %<st %Ss:%Sh
#logformat combined %>a %ui %un [%tl] "%rm %ru HTTP/%rv" %Hs %<st "%{Referer}>h"
"%{User-Agent}>h" %Ss:%Sh

access_log /var/log/squid/access.log squid

#Default:
cache_log /var/log/squid/cache.log

#Default:
cache_store_log /var/log/squid/store.log
```

Das access_log kann bei Bedarf verändert werden, in dieser Einstellung sieht eine Zeile so aus:

```
1284045145.911 910 123.123.123.136 TCP_MISS/200 55024 GET http://www.ngud.de/ DOMAENE+testuser2
DIRECT/87.230.78.108 text/html
```

Wenn nicht alle relevanten Informationen zu sehen sind kann das Log über „logformat squid“ angepasst werden.

OPTIONS FOR FTP GATEWAYING

```
# -----
#Default:
ftp_user Squid@domain.tld
```

Ist das editieren der squid.conf beendet kann man mit `/etc/init.d/squid restart` (oder `reload`) die Konfiguration testen.

Ich persönlich ziehe noch einen kompletten Serverneustart vor, um sicherzugehen das alle Dienste in der richtigen Reihenfolge geladen weden.

1.6 Last Words

Nun sollte ein User, der in der Gruppe „inet“ ist surfen können. Ein Nutzer, aus der gleichen Domäne, der nicht in der Gruppe ist, nicht.

Happy Testing :-)

2.1 http://wiki.samba.org/index.php/Samba_&_Active_Directory

(auf Formatierung wird verzichtet, dient nur der Vollständigkeit)

Samba & Active Directory

From SambaWiki

Contents

[hide]

- * 1 How much AD integration do you want?
- * 2 Which steps must be done to run Samba with AD-Integration
 - o 2.1 Prerequisites
 - o 2.2 Steps
 - o 2.3 Slightly Fuller Explanation
 - + 2.3.1 system-Config-authentication
 - + 2.3.2 Setting Up Kerberos
 - + 2.3.3 Setting up Samba
 - + 2.3.4 Adding this list to the password list.
 - + 2.3.5 Setting up PAM Authentication for Active Directory.
 - + 2.3.6 Home Directories
 - + 2.3.7 Creating home directories manually
 - o 2.4 Authenticating share users and groups against active directory
- * 3 Advanced Configuration
 - o 3.1 Windows 2003 R2 Active Directory
 - + 3.1.1 Configuring Windows
 - + 3.1.2 Configuring Samba
 - o 3.2 access control
 - + 3.2.1 permissions
 - + 3.2.2 password changes

[edit]

How much AD integration do you want?

You can get Linux Login's and Samba Shares to both authenticate against AD. For that, keep reading this document.

If, instead, you want to have Linux Login's authenticated either natively or against OpenLDAP, but have Samba Shares authenticated by AD, while still using UID/GID's defined internally or by OpenLDAP, read this: Samba, Active Directory & LDAP

[edit]

Which steps must be done to run Samba with AD-Integration

[edit]

Prerequisites

1. Software
 - * Samba > 3.0.20
 - * Kerberos MIT/Heimdal
 - * ntp
 - * often cups-Server
2. Permissions/Users
 - * root-user on the server
 - * an AD user with the permission to join AD ([Explanation]).

[edit]

Steps

1. The time between DC's and the Samba server must be in sync
 - * use ntp
2. configure your Kerberos environment kinit must be running fine
3. configure your smb.conf
 - * security = ADS
4. join into the domain
 - * kinit
 - * net ads join

5. start the services

- * nmbd
- * smb
- * winbind

[edit]

Slightly Fuller Explanation

Taken from <http://ask.java23.co.uk> (by the Author)

To connect Linux (specifically RHEL) to Active Directory, you must have Kerberos (krb5), Winbind and Samba installed. Samba must be newer than 3.08, or it doesn't work. It will also be helpful during the testing to take the firewall down, to facilitate things working. A working firewall will be posted as soon as one has been worked out.. You will also need to configure PAM and nsswitch to get it authenticating against the Active Directory.

[edit]

system-Config-authentication

On RHEL system, (or, presumably on Fedora core systems), you can use the system-config-authentication module. Simply chose winbind, and remember to chose "use kerberos". The Domain will be your windows Domain, and the Realm will be your ADS realm (eg WINDOWS.JARA23.CO.UK). You should then be able to click "join domain". This should connect you happily to the domain. Unfortunately, you won't see any errors until you exit the program. Normally, the biggest problem is with clock skew, ensure your clock time matches that of your ActiveDirectory server (for example by using ntpdate my.activedirectory.time.server.com). A fully populated and properly configured DNS system (including SRV records for your realm) for your LAN will save you many WTF! moments.

[edit]

Setting Up Kerberos

The first thing to do is to set up the kerberos keys so that they work. Remember that kerberos is time-dependent, so you may have to make sure that the machine time is correct using a protocol like NTP.

Windows Servers should automatically update their clocks and Windows Workstations (2000 and later) synchronize their clocks to the Active Directory server. To emulate this behavior in Linux add the line server ad-server-name in your /etc/ntp.conf file and comment out all other server lines.

Below is a working krb5.conf file.

ALERT! Capitals are important here. Without capitalization of your realms and .domain_realm, kerberos won't be able to connect.

[logging]

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

[libdefaults]

```
default_realm = WINDOWS.JARA23.CO.UK
dns_lookup_realm = false
dns_lookup_kdc = false
ticket_lifetime = 24h
forwardable = yes
```

[realms]

```
WINDOWS.JARA23.CO.UK = {
  kdc = server.windows.java23.co.uk
  admin_server = server.windows.java23.co.uk
  default_domain = windows.java23.co.uk
}
```

[domain_realm]

```
.kerberos.server = WINDOWS.JARA23.CO.UK
.windows.java23.co.uk = WINDOWS.JARA23.CO.UK
```

[kdc]

```
profile = /var/kerberos/krb5kdc/kdc.conf
```

[appdefaults]

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

Save the file. Once the file is saved you can test it with this command: `kinit admin@WINDOWS.JARA23.CO.UK`. Remember, again, capitals are important. This should ask you for the password for the user account "admin", and then tell you if you have successfully logged in.

[edit]
Setting up Samba

Samba is the software that allows you to connect Linux and UNIX clients to a Window's domain in the same way as you would a Windows 2000/XP machine. There are three important components, `smbd`, `nmbd`, and `winbind`, which all use the same configuration file: `/etc/samba/smb.conf`. Check the example configuration file below:

```
#GLOBAL PARAMETERS
[global]
  workgroup = MIDGARD
  realm = WINDOWS.JARA23.CO.UK
  preferred master = no
  server string = Linux Test Machine
  security = ADS
  encrypt passwords = yes
  log level = 3
  log file = /var/log/samba/%m
  max log size = 50
  printcap name = cups
  printing = cups
  winbind enum users = Yes
  winbind enum groups = Yes
  winbind use default domain = Yes
  winbind nested groups = Yes
  winbind separator = +
  idmap uid = 600-20000
  idmap gid = 600-20000
  ;template primary group = "Domain Users"
  template shell = /bin/bash
```

```
[homes]
  comment = Home Direcotries
  valid users = %S
  read only = No
  browseable = No
```

```
[printers]
  comment = All Printers
  path = /var/spool/cups
  browseable = no
  printable = yes
  guest ok = yes
```

A few important switches that might need a bit of explanation.

* `winbind use default domain = Yes` removes the domain prefix from usernames, so you can login as Username instead of `DOMAIN\Username` or in some cases `DOMAIN+Username` (see next explanation).

* `winbind separator = +` : This is the separator used to separate domain from username. Generally in documentation you will find this set to `.`. When you run `testparm` this will throw a warning, but that should be okay. It will mean that when you list the users you will see them in the form `"MIDGARD+phb"`.

* `idmap uid = 600-2000` and `idmap gid = 600-2000` set where the users from the AD will map onto the local system. the starting value for this should be the highest number of the last local user on your system. for example: if in your `passwd` the last user listed is "bob" with a

uid of "740," the starting value of your idmap entries should probably be about 800. if local uid's and gid's overlap mapped ad uid's and gid's, then the user will be evaluated in accordance with nsswitch.conf order. setting your idmaps lower than 100 is ill advised: This can lock you out of your root account if your nsswitch order specifies a query to winbindd first.

* template primary group = "Domain Users" sets the default group for users coming into the system. not required for most windows 2003 domain controllers unless they are in "mixed mode."

* template shell = /bin/bash gives the default shell to users logging onto your system. As this is not filled in by Active Directory, winbind does it all for you, locally.

* winbind enum groups and winbind enum users allow the command "getent" to return with groups and users respectively.

winbind enum users and groups should be used with caution in active directories greater than 200 users or groups, as enumeration is an expensive process and likely to timeout and cause login failures. during login, the full passwd and group will be "enumerated" every time from your active directory server. enumeration is not required for a successful login.

Now, test the parameters file, and correct any syntax errors, using the command "testparm". It should print out that everything is okay, and a warning about the + sign possibly causing problems with domain joins. This can be safely ignored. The next thing is to start the services. All the documents on the web suggest starting them in order NMB, SMB, then Winbind. On Fedora Core 7 (and redhat enterprise linux 5), using the service smbd start and service windbind start works fine. prior versions will require an additional "d" on winbind (service winbindd)

Now to join your machine to the active directory. You will need the user-name and password to a Domain Administrator account to do this. The command you need to join the domain is net ads join -U sadwrn. This should then ask you for a password, and print a domain join notice.

To check that you have succesfully joined the domain, there are several things you can test.

* net ads testjoin Test the connection to the Active Directory.

* wbinfo -u Should now list all the members of the domain. TIP You might see a laod of machine names followed by \$. eg myserver\$. This is normal. You might want to try piping the output to more. a wbinfo -u may fail once or twice if your directory is large.

* wbinfo -g Should now list all the groups available in the domain. You might note that if you have more than one domain, that the members of the other domain will appear in the form "DOMAIN+mygroup". This is normal, and expected! the command may also fail once or twice if your directory is large.

* wbinfo -a username%password checks to see if username using password can connect to the domain. Remember the password, you have to type it as part of the command; it won't ask you for it later.

* should wbinfo fail to return all groups or users in the active directory, simply increase the idmap gid upper boundary and restart winbind and SMB until all users and groups are produced in the list.

At this point you are presumably authenticating against the AD. authentication failures have the potential to lock your Active Directory account.

[edit]

Adding this list to the password list.

The next step is to get the passwd command to check the winbind list for usernames and groups. This is fairly straight forward as it only involves changing one file, /etc/nsswitch.conf and at that fairly minimally. Of course, backup this file before changing it.

passwd: files winbind

shadow: files winbind

group: files winbind

#hosts: db files nisplus nis dns

hosts: files dns wins

Example - obey only what nisplus tells us...

#services: nisplus [NOTFOUND=return] files

#networks: nisplus [NOTFOUND=return] files

#protocols: nisplus [NOTFOUND=return] files

#rpc: nisplus [NOTFOUND=return] files

#ethers: nisplus [NOTFOUND=return] files

#netmasks: nisplus [NOTFOUND=return] files

bootparams: nisplus [NOTFOUND=return] files

ethers: db files
netmasks: files
networks: files dns
protocols: db files
rpc: files
services: files

netgroup: files

publickey: nisplus

automount: files
aliases: files nisplus

As you can see, the file is configured to check the local passwd file first. This is so that things like system and root accounts don't lag waiting for a response from the Active Directory, when won't be forthcoming. It also ensures that if there is a problem, Root and services will still function (for example, samba can't look up it's own account over Active Directory during startup!).

Note that the following files (and symlinks) must be present in the system /lib directory:

libnss_winbind.so
libnss_winbind.so.2 -> libnss_winbind.so
libnss_wins.so
libnss_wins.so.2 -> libnss_wins.so

If you compiled from source you will probably have to copy these files manually after `make install` You should now be able to run getent passwd and see the local password list, and on the end of it, those that have been imported from the Active Directory. Now all that remains is setting up PAM authentication.

[edit]

Setting up PAM Authentication for Active Directory.

ALERT! Before you start, backup you you /etc/pam.d directory. Failure at this stage can lock the entire machine. Log in a root account on a virtual terminal, and LEAVE IT LOGGED IN until such time as you have tested the new configuration. Perhaps log in TWO root accounts incase of mistakes.

On RedHat, changing the PAM configuration is as easy as changing one file, the /etc/pam.d/system-auth file. This file is responsible for directing the services that require authentication to the right mechanism to get a response. Change the file as follows:

```
##%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth required /lib/security/$ISA/pam_env.so
auth sufficient /lib/security/$ISA/pam_unix.so likeauth nullok
auth sufficient /lib/security/$ISA/pam_winbind.so use_first_pass
auth required /lib/security/$ISA/pam_deny.so

account required /lib/security/$ISA/pam_unix.so
account sufficient /lib/security/$ISA/pam_succeed_if.so uid < 100 quiet
account sufficient /lib/security/$ISA/pam_winbind.so use_first_pass
account required /lib/security/$ISA/pam_permit.so

password requisite /lib/security/$ISA/pam_cracklib.so retry=3 type=
password sufficient /lib/security/$ISA/pam_unix.so nullok use_authok md5 shadow
password sufficient /lib/security/$ISA/pam_winbind.so use_first_pass
password required /lib/security/$ISA/pam_deny.so

session required /lib/security/$ISA/pam_limits.so
session required /lib/security/$ISA/pam_unix.so
session required /lib/security/$ISA/pam_winbind.so use_first_pass
```

Save the file, and change to another virtual terminal. Try logging in as a Member of the Active Directory. This should work, though you WILL see an error about missing home-directory (don't worry about that, we'll fix that later). If you have had a previous user account on that machine that matches the log-in from the Active Directory, you will need to comment it out. (comment, not delete, that way you can restore if things go wrong). Check as many users as you can, until you feel comfortable that the mechanism works. ALERT! Remember to ensure that ROOT can still log in.

[edit]

Home Directories

Once all the above is working, you might be tempted to reboot, and try to log in. Bad Idea. Currently no users have home-directories.

[edit]

Creating home directories manually

By default the home directories are created under /home/DOMAIN/username. So, in our example, user phb's home directory will be found in /home/MIDGARD/shadowknight. You can specify the root of the home directory with the "template homedir = /home/%U" (or similar) option; this example is similar to *nix local user home directory structure and eliminates the (%D reference and) domain name from users' home directories' paths. Since we're using the "winbind use default domain = yes" option, we're only planning to resolve the accounts for one domain anyway; there is no danger of account name overlap among trusted domains. Another Gotcha is that there is no group "ShadowKnight". So when you create the directory, as root, and then attempt to chown (shadowknight:shadowknight) it, it throws an error. What you need to do is to create the Directory MIDGARD, then create the directory shadowknight, then chown -R shadowknight:"Domain Users" shadowknight. This, of course, is a bit of a security problem. It leaves the home directory open to anyone in "Domain Users" unless you remove permissions from that group. Implementation of Posix ACLs on the underlying file system partition also makes management and inheritance of complex ACLs much easier. With proper implementation of Posix ACLs (including default ACLs) and no security restrictions in smb.conf, Samba very closely approximates NT ACLs (including full configuration from within Windows Explorer Security tabs).

---ShadowKnight 06:37, 15 September 2006 (CDT)

If you'd rather have users home directories generated on the fly when they first login to your Linux machine, add this line of code in /etc/pam.d/system-auth.

```
session required /lib/security/pam_mkhomedir.so
```

through the specification of skel= you can also control special skel files provided to dc users. --Nimbus 13:11, 31 October 2006 (CST)

[edit]

Authenticating share users and groups against active directory

Yeah, this one took me about a day too.

[Pictures]

```
comment = Directory for storing pictures by jims users
path= /usr/local/pictures
Valid Users =@NETWORK+archival NETWORK+billybob NETWORK+jane
; public=no
writable=yes
browseable=yes
```

So what has this done? @NETWORK+archival gives any member of the archival group on NETWORK access to this share. NETWORK+billybob NETWORK+jane gives billybob and jane, both single user members of NETWORK, access to this share.

--Nimbus 13:11, 31 October 2006 (CST)

[edit]

Advanced Configuration

[edit]

Windows 2003 R2 Active Directory

In the case that your AD is running on Windows 2003 R2 and the UNIX integration is enabled (RFC2307 schema), you can use the information in the AD for the Unix accounts provided by winbind.

Thus all user information (Windows & Unix) is maintained in a single place (the Active Directory) and you don't have to bother about ID mapping any more.

[edit]

Configuring Windows

I am still working on that ...

Notes:

* Samba ignores the member list of Unix groups but rather follows the Windows group relationships (including nested groups). If you also use LDAP/NIS somewhere else, you must continue to fill in the Unix group lists to match the Windows group membership information. This is due to the stupid design of the Windows SFU extension and Samba is doing the right thing.

[edit]

Configuring Samba

The Samba configuration (tested on 3.0.24) should contain these values:

[global]

```
passdb backend = tdbsam
idmap backend = ad
idmap uid = 100-20000000
idmap gid = 100-20000000
winbind nss info = rfc2307
```

Notes:

- * passdb backend is not strictly required, but recommended to manage the BUILTIN and LOCAL accounts
- * the idmap uid and idmap gid ranges should be chosen to not conflict with existing Unix users on the Unix server but to include all existing Unix IDs already in your AD
- * You must make sure that the primary group of the Unix users in the AD is also Unix enabled (with a GID) (A user whose primary group is not also a Unix group will not show up on Unix at all !)
- * winbind nss info instructs Samba to use the RFC2307 schema in the AD for the NSS information (shell, home dir ...)

Please note that from 3.0.25 on these values look different as one needs to use the new idmap stuff !

--Schlomo 05:59, 1 May 2007 (CDT)

[edit]

access control

not every domain user should be allowed to gain access to every linux box on the network. in pam, winbind can be configured to allow only a certain group access to the server example:

```
auth required /lib/security/$ISA/pam_winbind.so use_first_pass require_membership_of=vpn-informatiotechnology
```

now, only members of the AD group vpn-informatiotechnology can access this server. multiple winbind entries in pam can allow multiple groups access to a single server as normal users. whitespaces arent allowed in group names.

access.conf as well as /etc/sudoers can also be controlled using active directory groups and users. however, once more, no groups with spaces are permitted.

[edit]

permissions

AD users and groups may be designated as file and directory owners, and whitespace may be used in group names hwoever must be escaped by backslash. chown, chgrp, setfacl, and getfacl all function with active directory users and groups.

[edit]

password changes

it is possible to change your password using samba AD integration, but not recommended. it results in a curious condition in that seemingly two passwords become assigned to one username.

both the old, as well as the new password come into effect and are accepted by linux as well as some windows systems utilizing AD as an authentication mechanism (ex: altiris). Windows logon however only accepts the new password assigned by the user.

the madness continues with windows' ability to "extinguish" silently the old password... Once a linux user logs back into their windows system with this new password (assigned in linux), the old password automagically becomes null and void; no longer existing for use anywhere.

Squid NTLM authentication configuration using ntlm_auth

by Vivek Gite · 16 comments

Q. How do I configure squid for NTLM authentication?

A. You need to use squid ntlm_auth helper tool. It o allow external access to Winbind's NTLM authentication function. ntlm_auth uses winbind to access the user and authentication data for a domain.

Make sure winbindd is working

winbindd is a daemon that provides a number of services to the Name Service Switch capability found in most modern C libraries, to arbitrary applications via PAM and ntlm_auth and to Samba itself. If you are not sure about winbindd, refer to official Samba documentation for configuration.

Configure squid for NTLM authentication

Open squid configuration file - squid.conf, enter:

```
# vi squid.conf
```

Append following configuration directive:

```
auth_param ntlm program /usr/lib/squid/ntlm_auth --helper-protocol=squid-2.5-ntlmssp
auth_param basic program /usr/lib/squid/ntlm_auth --helper-protocol=squid-2.5-basic
auth_param basic children 5
auth_param basic realm Squid proxy-caching web server
auth_param basic credentialsttl 2 hours
```

To setup ntlm_auth for use by squid 2.5 with group limitation, add:

```
auth_param ntlm program /usr/lib/squid/ntlm_auth --helper-protocol=squid-2.5-ntlmssp --require-membership-
of="WORKGROUP\Domain Users"
auth_param basic program /usr/lib/squid/ntlm_auth --helper-protocol=squid-2.5-basic --require-membership-of="WORKGROUP\Domain
Users"
auth_param ntlm children 5
auth_param ntlm max_challenge_reuses 0
auth_param ntlm max_challenge_lifetime 2 minutes
```

OR You can also pass DOMAIN/PDC name:

```
auth_param ntlm program /usr/lib/squid/ntlm_auth DOMAINNAME/PDC
auth_param ntlm children 5
auth_param ntlm max_challenge_reuses 0
auth_param ntlm max_challenge_lifetime 2 minutes
```

Now add ACL configuration for ntlm_auth helper

```
acl ntlm_users proxy_auth REQUIRED
http_access allow ntlm_users
http_access deny all
```

Save and close the file. Restart Squid:

```
# /etc/init.d/squid restart
```

For more information:

- * Read ntlm_auth man page
- * Samba winbindd - Use of Domain Accounts

Updated for accuracy.