Automount von Sambalaufwerken beim Login - inkl. Kerberos und Nomachine Terminalserver

Du möchtest dich gerne für unsere Hilfe erkenntlich zeigen . Gerne. Wir bedanken uns bei dir für deine Spende!

Hauseigenes Apt-Repo: https://apt.iteas.at

GITLAB Enterprise:

+ + +

Diese Anleitung bezieht sich auf folgende Konstellation:

- Serverumgebung Univention Corporate Server auf Debian Basis (Active Directory/LDAP/Kerberos)

Als Clients kommen folgende Systeme getestet zum Einsatz:

- KDE Neon 20.04 mit Nomachine Freeedition/Enterpise Client
- Ubuntu 18.04/20.04 Server (über SSH)
- Lubuntu 20.04 (LXQT) als Nomachine Workstation Enterprise

Auch in Linuxumgebungen ist die Automation von Desktops voll und ganz ein Standard. Da dies aber meist nur größeren Enterpriseumgebungen vorbehalten ist, findet man im Internet hierüber sehr wenig ausführlich gute Dokumentation. Aus diesem Grund hab ich mir mal gedacht ich schreib darüber doch auch mal einen Artikel um dich daran auch teilhaben zu lassen.

Voraussetzung

Voraussetzung ist eine laufende funktionierende UCS4.4.x Umgebung inkl. Kerberos. Solltest du eine andere Umgebung als AD/LDAP benutzen ist das selbstverständlich auch ok. Hast du das nicht, installiere dir diese mal schnell nach.

Überprüfung am Client

Bist du am Linuxclient eingeloggt, öffne eine Konsole und tippe `klist`. Dies zeigt dir dein aktuelles Kerberosticket und die Gültigkeit. Würde als Output so etwas kommen,

```
klist: No ticket file: /tmp/krb5cc_0
```

bist wohl nicht richtig an deinem ActiveDirectory registiert.

Funktioniert alles gut, ist an dieser Stelle ein kleines Helferlein zu erwähnen. Installiere dir gleich `krb5-auth-dialog` nach. Das erneuert bequem deine Tickets im laufenden Betrieb. Immer dann wenn du wo dein Passwort eingeben sollest, z.B. wenn du deinen Bildschirm entsperrst.

Installation und Konfiguration am Client

Als erstes sind ein paar Pakete zu installiern.

```
apt install libpam-mount davfs2 keyutils -y
```

Optional das Paket `refresh-krb5mounts` von https://apt.iteas.at. Damit bekommt man ein Icon das nachdem ein Kerberosticket ausgelaufen ist, die Mountpoints auffrischt. Oft funktioniert hier ein Klick nicht. Die CMD schafft Abhilfe.

Dies bearbeitet die Pamconfiguration. Dies lässt du zu, oder editierst sie später manuell, falls du mal selbst Änderungen vorgenommen haben solltest. `nano /etc/pam.d/common-auth`. Die folgende Zeile kommt vor der Zeile `auth optional pam_cap.so`.

```
...
auth    optional        pam_mount.so
```

In der Datei `/etc/pam.d/common-password` fügst du vor der Zeile `password optional pam_gnome_keyring.so` folgendes ein:

```
...
password        optional        pam_mount.so disable_interactive
...
```

Verwendest du Nomachine auf deinem Rechner inkl. SDDM als Loginmanager, musst du noch einen weiteren spezielle Eintrag in der Pamkonfiguration vor der Zeile `session optional pam_systemd.so` setzen. Die normale Pammountzeile kommentierst du aus. Den Artikel von Nomachine darüber findest du hier.

```
nano /etc/pam.d/common-session
```

```
...
session optional        pam_mount.so disable_interactive
#session        optional        pam_mount.so
session optional        pam_systemd.so
```

```
...
```

Setzten der pam_mount.conf.xml

Die Standardoptionen haben bei meinen Konfigurationen immer funktioniert. In diesem File trägst du alle Laufwerke ein die in deiner Umgebung erreichbar sind. Loggst du dich mit deinem Benutzer ein, werden automatisch alle Laufwerke in dein Home eingebunden. Beim Logout, wieder sauber getrennt. Die Funktion kann mit **Cifs, NFS4.x** und **Webdav** wissentlich umgehen.

```
/etc/security/pam_mount.conf.xml
```

Hier nun ein Beispiel mit Samba/Windowslaufwerken:

pam_mount.conf.xml

```
...
<!-- pam_mount parameters: Volume-related -->
<volume fstype="cifs" server="server.tux.lan"
options="vers=3.0,sec=krb5,cruid=%(USERUID)" path="Downloads"
mountpoint="~/Downloads"> <not><user>root</user></not>
<not><user>sddm</user></not> <not><user>nx</user></not> </volume>
<volume fstype="cifs" server="server.tux.lan"
options="vers=3.0,sec=krb5,cruid=%(USERUID)" path="Dokumente"
mountpoint="~/Dokumente"> <not><user>root</user></not>
<not><user>sddm</user></not> <not><user>nx</user></not> </volume>
<volume fstype="cifs" server="server.tux.lan"
options="vers=3.0,sec=krb5,cruid=%(USERUID)" path="Bilder"
mountpoint="~/Bilder"> <not><user>root</user></not>
<not><user>sddm</user></not> <not><user>nx</user></not> </volume>
<volume fstype="cifs" server="server.tux.lan"
options="vers=3.0,sec=krb5,cruid=%(USERUID)" path="%(USER)"
mountpoint="~/MYHOME"> <not><user>root</user></not>
<not><user>sddm</user></not> <not><user>nx</user></not> </volume>
<mkmountpoint enable="1" remove="true" />
...
```

Die User Root, Sddm und NX werden ignoriert. Sprich diese machen keine Mountabfrage. Macht ja auch keinen Sinn. **Der große Vorteil:** Angabe von Passwort und Benutzer sind natürlich nicht notwendig. Um das kümmert sich LDAP/Kerberos von UCS, und erledigt beim Loginvorgang den Rest.

Macht man das ganze mit PAM ohne Kerberos, könnte man jetzt wieder paranoid reagieren und sagen, „ja die Passworteingabe wird ja beim Login PAM übergeben, das ist doch unsicher!“. Ja das stimmt schon wenn man es aus technischer Sicht ganz genau betrachten würde, ist das so. Aber das lass ich jetzt mal im Raum so stehen 🤐

Hier noch ein Beispiel für Wedav, CIFS und NFS. In dieser Konfig gehen diese nicht über Kerberos.

pam_mount.conf.xml

```
...
<volume fstype="davfs" path="https://daten.tux.com/webdav-daten"
mountpoint="~/webdav-daten"
options="username=%(USER),rw,nosuid,nodev,uid=%(USER)">
<not><user>root</user></not> <not><user>sddm</user></not> </volume>
<volume fstype="nfs" server="servername.bla.at" path="/home/Dokumente"
mountpoint="~/Dokumente" />
<volume fstype="cifs" server="servername.bla.at" options="vers=3.0"
path="Organisation" mountpoint="~/Organisation" />
```

Du kannst schon erkennen, das der Mechanismus sehr mächtig ist. Die Manpage ist hier sehr ausführlich. Mit ein wenig Zeit und tun, kannst du dir hier etwas schönes bauen.

```
man pam_mount.conf
```

Ab diesem Zeitpunkt bekommst du deine Laufwerke beim Login über dem Displaymanager SDDM oder Lightdm bereits eingebunden.

SSH Login über Kerberos und Automount aller berechtigten Netzlaufwerke

Sehr konfortabel ist der Login mit SSH über Kerberos. Hierbei ist lediglich die Keytab vom Univention Master Server zu exportieren. Die Keytab wird bei Kerberosanbindung/Default Domänenanbindung automatisch erstellt. Dieser Objekttyp kostet auch Stückzahl „1“ der lizensierten Domänenservices.

```
samba-tool domain exportkeytab --principal=host/pc-peter.tux.at@TUX.AT
/root/pc-peter-host.keytab
```

Die Keytab kopiert man am Client auf `/etc/krb5.keytab` und muss die Rechte root:root mit 600 besitzen. Dies ist der Platz der Defaultkeytab und jegliche Anwendung die Kerberos, spricht greift drauf zu und beforzugt Kerberos als erste Auth-Möglichkeit.

Den Inhalt der Keytab kannst du mit

```
ktutil -k /etc/krb5.keytab list
```

auflisten. Um zu prüfen ob die Keytab richtig funktioniert, kannst du als Root die Keytab testen. z.B. mit

```
kinit -k host/pc-peter.tux.at@TUX.AT
```

Damit muss du nun ein Ticket ohne Passworteingabe bekommen haben. Ist das nicht der Fall, funktioniert etwas noch nicht richtig.

Um am UCS Masterserver zu sehen welche Principal es der Zeit gibt, verwendest du diesen Befehl:

```
univention-s4search '(|(userPrincipalName=*)(servicePrincipalName=*))'
userPrincipalName servicePrincipalName
```

Essentiell ist auch noch das in der `/etc/hosts` der FQDN für Localhost gesetzt ist:

```
# Standard host addresses
127.0.0.1  localhost
::1        localhost ip6-localhost ip6-loopback
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters
# This host address
127.0.1.1  pc-peter.tux.at pc-peter
```

Die Quelle des Befehls findest du hier.

Folgenes ist damit möglich

- Passwortloses login via SSH
- Mitnahme des Tickets auf andere Server
- Automatisches mounten sämtlicher Dateifreigaben (Cifs, NFS4, DAV, ...) via Kerberos über SSH, TTY, Displaymanager, was auch immer.
- Automatisches transperentes Einbinden von Laufwerken über den Dolphin Netzwerkmanger inkl. Remotemounts
- Kerberoslogin über Nomachine Enterprise
- uvm.

Konfiguration des SSH-Servers und Client

Hier ist in der `/etc/ssh/sshd_config` lediglich folgender Eintrag zu setzten:

```
GSSAPIAuthentication yes
```

Alles andere kann default belassen werden. Danach den SSH Server neu starten: `systemctl restart sshd`. Am client müssen diese zwei Zeilen aktiviert werden:

```
GSSAPIAuthentication yes
GSSAPIDelegateCredentials yes
```

Nach einem Restart des SSH-Servers, bekommst du deine Laufwerke auch darüber bequem eingebunden.

Meinen Beitrag im Forum findest du hier.

Login am NX Terminalserver via Kerberos

Um dies zu ermöglichen mußt du am NXServer ein paar kleine Configänderungen in der `/usr/NX/etc/server.cfg` vornehmen.

```
EnableNXKerberosAuthentication 1
NXGssapiLibraryPath "/usr/lib/x86_64-linux-gnu/libgssapi_krb5.so.2"
NXKerberosLibraryPath "/usr/lib/x86_64-linux-gnu/libkrb5.so.3"
```

Nun noch den Server neu starten: `systemctl restart nxserver.service`.

Am Nomachine Client mußt du noch unter „Konfiguration Authentifizierung“, **Kerberos ticketbasierende Authentifizierung verwenden**, auswählen. Im Untermenü nun noch die Option „**Authentifizierung weiterleiten**“ Checkbox aktivieren. Ab nun wird man mit dem Kerberos Ticket automatisch eingeloggt und auch Sambalaufwerke werden wie oben beschrieben in einer virtuellen Desktopsitzung automatisch mit dem Ticket eingebunden.

Bekannte BUGS

Sollte man bei einer Maschine die Meldung bekommen das kein Display verfügbar ist, so ist die `/etc/pam.d/nx` wie folgt zu modifzieren:

```
auth include su
account include su
password include su
#session include su
session required pam_loginuid.so
session optional pam_env.so
session optional pam_umask.so
session required pam_unix.so
session optional pam_mount.so disable_interactive
```

Nomachine neustarten.
Danach sollte das Display verfügbar sein und auch Pammount seinen Dienst tun. Vielen Dank an dieser Stelle an BenSommer der diesen Workaround erarbeitet hat.