http://www.therealgang.de/

Titel :	Zusammenfassung SAMBA Vorlesung
Author :	Michaela Uhl
Kategorie :	Sonstige-Skripte

Zusammenfassung des SAMBA - Skripts

SAMBA

Kapitel 1

Samba lässt jeden Unixrechner in der Netzwerkumgebung von Windows erscheinen. Der Client merkt keinen Unterschied on Windowsserver oder Unixserver.

Eigenschaften von Samba:

- Dateifreigaben können einfach erstellt werden
- Benutzer kann Dateien im eigenen HOME und in freigegeben Verzeichnisse ablegen
- Drucker, die unter Unix ansprechbar sind, können als Netzwerkdrucker in Windows angesprochen werden.

Samba bietet verschiedene Dienste, die sonst nur von Windows NT geleistet werden:

- WINS Server kann einfach eingerichtet werden
- Computersuchdienst (stabiler Server)
- Diagnosewerkzeug (effektive Werkzeuge zur Vereinfachung von Fehlersuche)

Vorteile von Samba, die von Unix geerbt wurden:

- entfernte Administration (von Kommando - Zeile aus)
- zentrale Konfiguration(eine einzige Textdatei)
- Stabilität von Unix

SMB – Server Message Protokoll

Client/Server Protokoll, das nach dem Frage/Antwort Prinzip arbeitet. Client stößt Aktion durch Anfrage an. Server gibt Antwort/Fehlermeldung zurück.

Konfiguration

Samba wird mit der Datei smb.conf konfiguriert. Sie ist unter `/etc/smb.conf` oder unter `/etc/samba/smb.conf` zu finden. Wenn die Datei angelegt wurde, müssen zwei Dämonen gestartet werden: der `nmbd` und der `smbd`. Sie liegen unter `/etc/init.d/smb`. Die Konfigurationsdatei gleicht den `.ini` Dateien unter Windows und haben den gleichen Aufbau:

- sie sind in mehrere Abschnitte unterteilt
- Abschnitte werden mit Abschnittsnamen eingeleitet
- Namen dürfen nur 1x vorkommen und werden in `[]` Klammern gesetzt
- Inhalt der Abschnitte bestehen aus Parameterzuweisungen

[homes] = Freigabe der Heimatverzeichnisse
Alle außer [global] sind Verzeichnis/Druckerfreigaben([public], [homes], [cdrom]).
Zeilen unter den Abschnittsnamen beziehen sich ausschließlich auf dieselbige.
[global] beschreibt keine Freigaben, sondern beinhaltet Festlegungen, die den Server als ganzes betreffen. Z.B Server Mitglied einer Workgroup oder Domäne, verschlüsselte Passwörter ja/nein etc. Jede Option die in dem [global] Abschnitt festgelegt wird, ist für alle Freigaben wirksam. Tauchen Optionen sowohl unter [global] als auch in anderen Abschnitten, in denen Freigaben festgelegt werden auf, hat die lokale Option im Abschnitt der Freigabe Vorrang. Für jeden Benutzer, die Freigaben der Samba Server nutzen möchten muss ein Eintrag in der `smbpasswd` gemacht werden, wenn verschlüsselte Passwörter vom Server verlangt werden. (encrypt passwords = yes)

Kapitel 2

Bestandteile von Samba:
Servertolls

Die beiden Dämonen smbd und nmbd stellen die eigentliche Dienste zur Verfügung.
SMBD = zentraler Serverprozess, der für die Datei+ Druckerfreigaben zuständig ist. Es gibt mehrere smbds im System. Einer hört auf den Port 139 und nimmt neue Verbindungen entgegen. Jede Verbindung stößt einen neuen smbd Prozess an. Sollen Verbindungen getrennt werden, müssen nur mit `smbstatus` die Prozessnummer der zuständigen smbds erfragt und gelöscht werden.

NMBD = ist für die NetBios Namens, - und Datagrammdienste zuständig. Dieser Prozess reserviert beim Start von Samba die entsprechenden NetBios Namen. Er kann WINS Server sein und ist für den Computersuchdienst zuständig.

TESTPARM = Programm, das die smb.conf auf syntaktische Korrektheit überprüft. Es liest die Konfigurationsdatei und gibt Fehler aus, falls es unbekannte Parameter findet.

SMBPASSWD = serverseitige Pflege der verschlüsselten Passwörter.

SMBCONTROL = Programm, mit dem Dämonen kontrolliert werden.

WINBINDD = Dämon, der eingesetzt wird, wenn Samba als reiner Dateiserver eingesetzt werden soll.

Weitere Komponenten, aus denen smbd + nmbd ihre Informationen beziehen und über die sie miteinander kommunizieren:

- /var/lock/samba: temporäre Lockdatei + Datenbank Ablage
- smbpasswd: Passwortdatenbank von Samba, sofern verschlüsselte Passwörter
- secret.tdb: wichtige Informationen über Zustand von Samba und Zusammenspiel mit Windows NT – Dämonen. Z.B Passwort der Workstation/Kontos. Samba in der Rolle des PDC speichert in dieser Datei die Domänen SID.
- /var/log/samba: Log Dateien von nmbd und smbd

Clientseitige Tools
Samba kann nicht nur als Server, sondern kann auch als Client auf Windows Rechner zugreifen.

- *smbclient:* Programm, mit dem man auf Freigaben von NT-Rechner zugreifen kann z.B. drucken auf von NT zur Verfügung gestellten Drucker. Mit smbclient kann auch die Liste der Server im Netz erfragt werden.
- *nmblookup*: Diagnosetool für Namensauflösung. (Wenn 2 PCs sich nicht finden können, kann mit diesem Befehl deren Versuche sich zu finden, nachgestellt werden.)
  WINS Server können befragt werden und ein **NetBios Status Request** abgefragt werden.
  (analog zu nbtstat unter Windows)
- *wbinfo:* Diagnoseprogramm für winbindd. Mit ihm wird geprüft, ob die Verbindung der Samba Serber zum DC funktioniert.

Kapitel 3

NetBios

Net Bios = Softwareschnittstelle zur Kommunikation von Rechner.
Wenn Windows Rechner Netzwerklaufwerke verbinden, sich gegenseitig in der Netzwerkumgebung sehen oder Drucker freigeben, funktioniert ihre Kommunikation untereinander über NetBios. Mit dieser Schnittstelle werden Programme unterschiedlicher Dienste zur Kommunikation zur Verfügung gestellt.
NetBios 1984 von IMB erfunden um in kleinen, lokalen Netzen Rechner Datenbereiche für den gegenseitigen Gebrauch freizugeben. Aber Transportprotokoll fehlte, also entstand eine Erweiterung des NetBios, das NetBeui. Durch TCP/IP Verwendung im stark wachsenden Internetbereich wurden auch dahingehende Anpassungen notwendig. Zusammenfassung beider Protokolle war allerdings problematisch, da TCP/IP mit Zahlen und NetBios mit Namen arbeitet. Zur Identifizierung entstand NBT (NetBios over TCP/IP).

NBT Standard beschreibt derzeit drei Netzwerkdienste:
Namensdienst, Datagrammdienst und Sitzungsdienst.

NetBios Dienste

- *Namensdienst* auf Port 137: gegenseitige Identifizierung der Rechner im Netz. Wollen Anwendungen zwischen Rechner im Netz kommunizieren, müssen sie sich zuerst gegenseitig identifizieren können. Dazu werden 16 Byte lange Namen gebraucht. Von den 16 sind 15 nutzbar, das letzte Byte kennzeichnet den Ressourcentyp des Namens. NetBios Namen existieren im einen, flachen Namensraum, d.h es gibt kein Äquivalent zu Domänen Bezeichnungen. Namen dürfen alphanumerisch sein und gewisse Sonderzeichen enthalten. Jede Anweisung kann für sich beliebig viele Namen reservieren. Unter einem Namen werden Verbindungen aufgebaut und Daten ausgetauscht. Die Reservierung von Namen gilt für Clients als auch für Server.
Wollen 2 Anwendungen per NetBios kommunizieren, muss der Server zuerst seine Bereitschaft, Verbindungen entgegenzunehmen, kundtun. Dazu reserviert er im Netz per Broadcast seinen Namen, so dass alle im Subnetz mithören. Dieser Vorgang, (Reservierung per Broadcast) passiert insgesamt 3x. Erfolgt daraufhin kein Protest, so sieht der Server seinen Namen als reserviert an. Verfahren der Namensreservierung bei Clients identisch, allerdings muss der Client zuerst die MAC - Adresse des Servers herausfinden.

- *Datagrammdienst* auf Port 138: Dienst, um schnelle, einfache Nachrichten zu versenden. Es erfolgt keine Verbindung zwischen den Computern beim Versand der Daten (~ ist ein verbindungslos – orientierter Dienst). Daten werden in einzelne Pakte zerlegt und an den Zielrechner ausgeliefert. Pakete können verloren gehen, in vertauschter Reihenfolge oder mehrfach ankommen. U.U erhält der Sender auch nicht einmal eine Benachrichtigung über verlorengegangene Pakete. Einziger Vorteil: geringer Aufwand beim Verschicken von Daten und gleichzeitiges Verschicken von Datagrammen an mehrere Rechner.

- *Sitzungsdienst* auf Port 139: entspricht Telefonverbindung, zwischen den beiden kommunizierenden Applikationen besteht eine Verbindung ⇔ Datagrammdienst.
Es wird eine NetBios Sitzung vereinbart. Daten kommen mit dem ~ auf jeden Fall richtig und auch in der richtigen Reihenfolge an. Falls nicht, erhält die versendende Applikation eine entsprechende Fehlermeldung. Dieser Zuverlässigkeit steht ein höherer Aufwand beim Sitzungsaufbau und Abbau gegenüber.

NetBios Implementationen

NetBios kann mit unterschiedlichen Protokollen implementiert werden: NetBeui, IPX oder TCP/IP.

- *NetBeui*: Client findet Server nur über Broadcasts. Der Server, der sich für den gesuchten Namen verantwortlich fühlt, antwortet, nachdem er seine MAC - Adresse ausgelesen hat. Mit NetBeui können nur Rechner miteinander kommunizieren, die in der gleichen Broadcastdomäne liegen.

- *TCP/IP*: Client muss IP des Servers herausfinden. Dies kann über Broadcast im lokalen Netz geschehen. Befindet sich der Rechner im gleichen Subnetz, kann direkt eine ARP Anfrage nach der MAC - Adresse ausgelöst werden. Andernfalls muss der entsprechende Router anhand der Routingtabelle herausgefunden werden und dann dessen Mac-Adresse per ARP festgestellt werden.

ARP steht für Adress Resolution Protocol und läuft in 4 Schritten ab:

1.Datenpaket an Ethernet Netzwerk Schnittstelle der eigenen Station übergeben. Sie sucht MAC - Adresse in der eigenen Tabelle. Ist ein gültiger Eintrag vorhanden, wird ein Paket mit der gefundenen Adresse versehen und gesendet.
2.Ist in der eigenen Tabelle kein gültiger Eintrag vorhanden, wird ein ARP - Broadcast mit der IP – Adresse des Zielhosts erzeugt und gesendet.
3.Alle im LAN erhalten das Broadcast Paket und vergleichen die darin enthaltene Ziel IP mit ihrer eigenen. Die Station, mit der gesuchten IP - Adresse sendet ein ARP – Paket, das die gesuchte Ethernet-Adresse enthält an den anfragenden Rechner.
4.Anfragender Host trägt nach dem Empfang MAC – Adresse in seine Tabelle ein und sendet das IP Paket direkt an den gesuchten Host.

Verwaltung der ARP Tabelle erfolgt dynamisch:

Dienst	Protokoll	Port	Samba Prozess
Namensdienst	UDP	137	nmbd
Datagrammdienst	UDP	138	nmbd
Sitzungsdienst	TCP/IP	139	smbd

Kapitel 5

NetBios Namensanfrage wird mit nmblookup ausgelöst. Paket wird an die Broadcastadresse im lokalen Subnetz gesendet. Nmblookup entnimmt konkrete Broadcastadresse der Zeile `interfaces = smb.conf`. Unter Windows ist eine isolierte Namensanfrage nicht möglich. Es muss eine Verbindung aufgebaut werden. Eine Anzeige der reservierten Namen erfolgt mit der Operation **Node Status Request**. Ein Rechner hat gleich mehrere Namen für sich reserviert. Jeder Name steht für eine andere Anwendung. (Unterscheidung am 16.Byte)
Es gibt NetBios Gruppen, - und Einzelnamen. Einerseits müssen bestimmte Dienste benannt werden, andererseits müssen manche Anwendungen mit mehr als einem Partner gleichzeitig kommunizieren.

Einzelname: existieren nur ein einziges Mal im gesamten Netz.

- `computername<00>` = Client tut seine Existens kund, dient zur eindeutigen Identifizierung
- `computername<20>` = Name für Serverdienst, Funktion eines Servers
- `benutzer<03>` = Anmeldung des Nachrichtendienstes des Rechners
- `arbeitsgruppe<1d>` = LMB

Gruppenamen: existieren mehrfach im Netz, „Broadcast für Arbeitsgruppe“. Existieren Gruppennamen, können unter diesem Namen alle Rechner dieser Arbeitsgruppe mit einem Datagramm erreicht werden.

- `arbeitsgruppe>00>` = Zugehörigkeit zu einer Arbeitsgruppe
- `arbeitsgruppe<1c>` = der Domän Logon Server reserviert diesen Namen für sich
- `arbeitsgruppe<1e>` = alle Rechner, die LMB werden können, reservieren diesen Namen für sich
- `.._MSBROWSE_.<01>` = alle LMBs um sich gegenseitig zu finden

Kapitel 6

NetBios: mit Hilfe von ~ sind Rechner im Netz ansprechbar und können verschiedene Dienste anbieten.
Arbeitsgruppe: Liste von Rechner, nur als reines Transportmedium mit NetBios zu tun.
Domäne: gemeinsam genutzte Benutzerdatenbank von Rechner

LMB
Rechner, der die Netzwerkumgebung pflegt, wird gewählt, nicht bestimmt, Wahl wird von dem Rechner angestoßen, der als 1. merkt, dass es kein solcher LMB gibt.
Will ein Rechner die Netzwerkumgebung anschauen, kontaktiert dieser den LMB über den NetBios Namen `arbeitsgruppe<1d>.` Server, die angezeigt werden wollen finden den LMB auf die gleiche Weise.
Wahl zum LMB: per Datagramm an Gruppennamen `<arbeitsgruppee<1e>`
Kriterien:
- OS Level
- Betriebssystem, dass besser ist wie ein anderes verschickt Wahlpaket mit Parameter
- Uptime, Rechner, der am längsten läuft „gewinnt“
- NetBios Name, alphabetischer Reihenfolge

Kapitel 7

NetBios über Subnetzgrenzen
Rechner die hinter Routern liegen, können über Broadcast nicht erreicht werden, denn Broadcasts verbleiben nur im Subnetz.

1.LMHOSTS: einfachste Weg Namensauflösung über Subnetzgrenzen hinweg zu realisieren. Sie ist eine statisch zu verwaltende Datei und liegt unter /etc/hosts. Der Zusatz von #PRE bewirkt direktes Verwenden des Werts in der LMHOSTS. Ohne den Zusatz wird zuerst eine konventionelle Namensauflösung durchgeführt. Der Nachteil der ~ ist, dass sie auf jedem Rechner statisch zu pflegen ist.

2.Möglichkeit mit WINS Server: dynamische Datenbank auf zentralem Server zur Pflege der NetBios Namen. Jede NetBios Applikation muss sich im Netz mit eigenem Namen anmelden. IP dieses Servers muss jedem Rechner mitgeteilt werden. Dies geschieht bei Samba durch den Eintrag `wins server = <ipadresse>` im Abschnitt `<global>` in der `smb.conf` .Sobald ein Rechner die IP - Adresse des WINS Servers kennt, ist es egal ob sich dieser im gleichem Subnetz befindet oder nicht. Namenreservierung passiert nicht mehr über Broadcast sondern per gerichteten UDP – Paket an den WINS Server. Router leitet gerichtetes Paket wie jedes andere Paket an den WINS Server weiter. Möchte ein Rechner einen Namen reservieren, der schon vergeben ist, fragt der WINS Server nach, ob der Name noch gebraucht wird. Wird der Name noch gebraucht, bekommt der anfragende Rechner eine Ablehnung. Wird der Name nicht mehr

gebraucht, oder bekommt der WINS Server keine Antwort so bekommt der anfragende Rechner eine positive Benachrichtigung. Diese Vorgehensweise ist dafür gedacht, um mit abgestürzten Rechner sauber umgehen zu können. Die Anfrage an den WINS Server erfolgt mit nmblookup.
Z.B `nmblookup 192.168.1.5 Samba = WINS Server`
Server, der die IP 192.168.1.5 hat, wird nach dem Namen Samba befragt.

Kapitel 8

NetBios Anwendungen = Windows Programme um Laufwerke mit Server zu verbinden. Die gesamte Netzwerkverbindung gehört ebenfalls zu den NetBios Anwendungen.
System schaut im NetBios Namenscache nach, ist ein WINS Server konfiguriert, wird dieser befragt. Kann der Name nicht aufgelöst werden, so wird eine Broadcast Anfrage ausgelöst. Es wird in der LMHOSTS Datei nachgesehen. Falls DNS Auflösung für NetBios in den TCP/IP Eigenschaften aktiviert ist, wird das Auflösungssystem für TCP/IP Anwendungen übergeben. Namenseinträge in LMHOSTS werden erst nach den WINS und Broadcast Timeouts berücksichtigt.

TCP/IP Anwendungen = Anwendungen, die es nur in der TCP/IP Protokollfamilie gibt. Namensauflösung funktioniert etwas anders als bei NetBios Anwendungen. Es wird in der LMHOSTS nachgesehen, ist ein DNS Server konfiguriert, wird dieser befragt. DNS Name wird an die NetBios Namensauflösung übergeben.

Samba kann sowohl als Client, als Server und auch als Domänenmitglied auftreten. Als Client und als Server muss Samba Namen auflösen. Samba kennt wie Windows 4 Mechanismen um dies zu tun: Broadcast, WINS, LMHOSTS und die Unix Namensauflösung.

Kapitel 9

Wenn eine einheitliche Arbeitsgruppe über Subnetzgrenzen hinweg gewünscht wird, muss ein weiterer Dienst installiert werden: DMB
Domän Master Browser = Rechner, der die Serverlisten von allen LMBs einsammelt und auf Anfrage wieder herausgibt. Der DMB wartet nur passiv darauf, dass ein LMB sich mit ihm synchronisieren will. Die LMBs haben die Aufgabe sich regelmäßig danach zu erkundigen, wo der DMB sitzt und mit ihm diesem die Serverlisten abzugleichen. Damit ein Samba Server die Aufgaben eines DMB übernehmen kann, ist innerhalb der smb.conf der Parameter `domain master = yes` in der [global] Section zu setzten.

Kapitel 10

Virtuelle Samba Server
Einen einzigen Server auf einer Maschine laufen zu lassen, nutzt einen PC heute bei weitem nicht mehr aus. Ein Samba Server ist in der Lage mehrere Identitäten gleichzeitig anzunehmen. Zur Serverkonsolidierung kann es nötig sein, unter mehreren Namen in der Netzwerkumgebung zu erscheinen. Eine andere Konfiguration ist die Einbindung von virtuellen Samba Servern in eine Hochverfügbarkeitsumgebung.

Kapitel 12

SMB Sitzungen

Um Fehlerdiagnose zu betreiben, ist das Wissen um die genaue Fehlerursache wertvoll.

NetBios Namensauflösung

Namen des Servers eingeben beim Aufbau einer Verbindung:
- Doppelklick in der Netzwerkumgebung auf einen Rechner
- von der Kommandozeile aus `net use h: \\server\freigabe`
- Netzlaufwerke verbinden
- ausführen im Startmenü \\server = Anzeige der Freigaben des Servers

TCP Verbindungen

Wenn die Adresse, zu der verbunden werden soll klar ist, wird eine TCP Verbindung zu Port 139 des Servers aufgebaut. Um vorhandene Verbindungen sich anzeigen zu lassen, gibt es das Werkzeug `netstat`. Ob die TCP Verbindung geklappt hat, prüft man mit `telnet <ip> 139`.
=> Entweder Fehlermeldung oder Verbindungen sind ESTABLISHED.

NetBios Sitzungen

Alle Anwendungen haben für sich Namen reserviert und sind unter der IP – Adresse des Rechners und dem TCP Protokoll auf dem Port 139 zu erreichen. Anhand des TCP _ Verbindungsaufbaus ist nicht klar, welche Serverapplikation angesprochen werden soll. Die Unterscheidung wird durch den Servernamen getroffen, der in der TCP - Verbindung als erstes übertragen wird.

Negotiate Protocol

Innerhalb einer NetBios Sitzung wird eine SMB Sitzung schrittweise aufgebaut:
Die erste Anfrage die der Client an den Server schickt, ist ein **Negotiate Protocol Request**. Er schickt eine Liste der Protokollvarianten, die er beherrscht. Der Server wählt eine Protokoll für die weitere Kommunikation aus der Liste aus und schickt Index des jeweiligen an den Client zurück. Außerdem werden zwei weitere Einstellungen verschickt:
- die Zugriffssteuerung auf Benutzer - oder auf Freigabeebene (`security = share` bedeutet auf Freigabeebene und `security = user` auf Benutzerebene)
- der Zeitpunkt, zu dem der Benutzer ein Passwort liefern muss (direkt beim Session Setup oder erst danach beim Tree Connect)
- Verwendung von Klartextpasswörtern oder verschlüsselten Passwörter (werden verschlüsselte Passwörter verwendet wird eine Herausforderung für das Challenge Response mitgeschickt)

Es ist nicht möglich für einige Benutzer Klartextpasswörter und für andere verschlüsselte Passwörter zu verwenden.

Session Setup

Nachdem Abhandeln der Protokollversion wird vom Client ein Session Setup verschickt. Darin enthalten ist der Benutzername des Clients und falls vom Server vorher verlangt(security = user) auch das Passwort. Damit ist der Server in der Lage die Identität des Benutzers festzustellen.

Tree Connect

Als letztes legt der Client fest, welche Freigabe er ansprechen will. Der Entsprechende Aufruf heißt ~. Wenn security = share angegeben wurde, wird der Server an dieser Stelle das Passwort überprüfen.

Kapitel 13

Rechte an Freigaben

Ist bei Samba security = users gesetzt, so hat der Server die Möglichkeit anhand des angemeldeten Benutzers Zugriffsrechte zu vergeben und zu verweigern. Wenn bei der Einstellung einer Freigabe keine Parameter für die Zugriffsrechte gesetzt sind, hat jeder korrekt angemeldete Benutzer Leserecht. Mit den Optionen zur Rechtevergabe an Freigaben hat man die Möglichkeit einzelnen Benutzer und ganzen Unixgruppen Rechte zu geben oder zu nehmen.

Alle Benutzer haben gleichen Zugriff

[projekt] path = /data/projekt = alle angemeldeten Benutzer mit Name und Passwort haben Lesezugriff auf die Freigabe. Schreibrecht vergibt man mit `writeable = yes`.

Einige Benutzer haben gleichen Zugriff

`[projekt] path = /data/projekt valid users = mueller , uhl` (Einschränkung auf Benutzer Müller und Uhl) Optional kann ihnen auch noch Schreibrecht gegeben werden, writeable = yes

Root muss wie jeder Benutzer in die Liste aufgenommen werden!

Mit valid users können auch ganze Unixgruppen in den Zugriff aufgenommen werden (@ Zeichen) z.B. `valid users = root, @users`

Einige Benutzer haben Leserecht andere Schreibrecht

```
[projekt] path = /data/projekt
valid users = @users, @admins
write list = @admins
```

Kapitel 14

Zugriffsrechte im Dateisystem

Benutzer muss sowohl durch eine Freigabe -, als auch durch Dateirechte zu Operationen berechtigt sein. Nach erfolgreichem Verbund mit der Freigabe nimmt der Benutzer seine ganz normalen Rechte als Unix User wahr. Will ein Benutzer in eine Datei schreiben, muss ihm dies sowohl durch die Freigabe als auch durch die Dateisystemrechte erlaubt sein. Von Samba vergebenen Rechte können darunter liegende Unixrechte nicht erweitern. Die Einschränkung durch Unixrechte ist ein wichtiges Prinzip von Samba. Im Dateisystem implementiert Samba keine eigenen Zugriffskontrollen, sondern verlässt sich auf die Unixmechanismen.

2 Gründe

- Zugriffsrechte sind im Betriebsystem bereits vorhanden implementiert
- es ist nicht möglich Samba ACLs synchron mit dem Unix – Dateisystem zu halten (falls sich Verzeichnisstrukturen ändern, wie soll dann die Samba ACLs angepasst werden können?!)

DOS Attribute

Diese Attribute sind Eigenschaften von Dateien, die es in dieser Form unter Unix nicht gibt. Insgesamt kennt DOS 4 verschiedene Attribute, die für Dateien vergeben werden können:

- `Read – only` (Inhalt kann nur gelesen werden, nicht geschrieben und gelöscht werden)
- `System` (für Betriebssystemzwecke)
- `Hidden` (diese Dateien werden mit dem Kommando dir nicht angezeigt)
- `Archiv` (wird bei jedem Schreibzugriff gesetzt)

Abbildung von DOS Attributen zu Unix Rechten

Unterschied zwischen DOS/Windows und Unix ist die Behandlung von Dateiberechtigungen.

Unix führt bei jeder Datei einen Satz von Zugriffsberechtigungen mit. (Besitzer, Gruppe, Rest – schreiben, lesen, ausführen)
Windows kennt nur die 4 Bits, die nicht getrennt für mehrere Benutzer einstellbar sind (schreibgeschützt, System, Archiv, versteckt). Windows kennt demnach kein Bit, das ausführbare Dateien kennzeichnet (es erkennt sie an den Dateierweiterungen).
Samba darf die drei Bits für die Ausführbarkeit der Dateien unter Unix seinen DOS/Windows Clients nicht mitteilen. Dennoch muss Samba die DOS Attribute beibehalten, wenn es Dateien auf einem Unix System ablegt. Samba benutzt für drei der DOS Attribute die drei Unix Bits, die die Ausführbarkeit der Dateien kennzeichnet. Unter [data] entscheiden drei Optionen darüber, ob Samba die Bit-Zuordnung verwenden soll: `map archive, map system, map hidden`.

DOS Attribut	Unix Recht	Maske	Parameter	Standard
Schreibschutz	**schreibrecht Besitzer**	**200**	**-**	**immer**
Archiv	**Ausführung Besitzer**	**100**	**map archive**	**yes**
System	**Ausführung Gruppe**	**010**	**map system**	**no**
Versteckt	**Ausführung Andere**	**001**	**map hidden**	**no**

Kapitel 16

Opportunistic Locks (Oplocks)
Dateizugriffe über Netzwerk deutlich langsamer als auf einer lokalen Festplatte.
Zugriffe von Clients auf Freigaben müssen koordiniert werden, da ein Cache auf Netzwerkdateien nicht davon ausgehen kann, die Datei alleine zu benutzen.
Opportunistic Locks (Oplocks) sind Mechanismen, mit dem Clients erlaubt werden kann, Dateiinhalte zu cachen. Mit einem Oplock bekommt der Client eine Datei solange exklusiv für sich, bis der Server ihn auffordert, die Änderungen zurückzuschreiben und die Sperre freizugeben. Oplock ist die Zusage, dass niemand sonst auf die Datei zugreifen kann. Damit muss ein Client weder bei jedem Lesezugriff den Server befragen, noch muss er jeden Schreibzugriff unverzüglich an den Server liefern. Wenn ein weitere Client auf die Freigabe zugreifen möchte, schickt der Server dem Client A ein so genanntes Oplock Break. Dies ist die Anweisung sämtliche Änderungen zurückzuschreiben und den Schreibcache auf diese Datei in Zukunft auszuschalten. Erst nachdem die Änderungen zurückgeschrieben wurden, kann Client B auf die Datei zugreifen. Die Änderungen sind sofort sichtbar. Dieses Schema funktioniert innerhalb von Samba hervorragend. Sobald Unix Prozesse ebenfalls auf Dateien zugreifen müssen, die von Samba freigegeben wurden, gibt es Probleme.

Kapitel 17

Benutzerauthentifizierung muss vor allem 2 Dinge leisten:
- Benutzer muss beweisen, dass er sein Passwort kennt
- Authentifizierungsprotokoll kann es dabei ermöglichen, dass das Passwort nicht übertragen werden muss.

Es gibt 2 Verfahren zur Verschlüsselung:

- symmetrische Verschlüsselung

Nachricht wird zerstückelt, so dass niemand sie mehr lesen kann (außer jemand kennt das Verfahren, mit dem verschlüsselt wurde), es existiert zu der Verschlüsselung ein Gegenstück, das wieder die originale Nachricht herstellt. D.h es muss nicht unbedingt das Verfahren bekannt sein, mit dem verschlüsselt wurde, sondern nur der Schlüssel.

- asymmetrische Verschlüsselung
Schlüsselpaar: private + public key
Mit Hilfe von public key Nachricht verschlüsselt
Empfänger kann mit private key Nachricht wieder entschlüsseln
Schlüsselpaar wird mit Hilfe von Programmen erstellt.

Challenge Response Verfahren
Bevor Verbindung zum Server aufgebaut wird, muss Client sich am Server anmelden mit Benutzername und Passwort. In der Antwort auf das Negotiate Protocol Response schickt der Server dem Client eine Zufallszahl, die Herausforderung genannt wird. Mit ihr wird nun nicht das Passwort verschlüsselt, sondern umgekehrt. Das Passwort wird als Schlüssel benutz um die Herausforderung zu verschlüsseln. Die mit dem Passwort verschlüsselte Herausforderung schickt der Client in der Session Setup zusammen mit Benutzernamen an den Server. Der Server liest nun aus seiner Benutzerdatenbank das Passwort des Benutzers aus und entschlüsselt damit die Herausforderung wieder und vergleicht das Ergebnis mit der Zufallszahl, die er dem Client geschickt hat und sich gemerkt hat. Somit kann der Server simpel überprüfen, ob das Passwort des Benutzers stimmt, oder nicht.

Die smbpasswd
Die Verschlüsselung im Challenge Response Verfahren erfolgt mit einem Hashwert des Passworts. Er wird vom Client direkt nach Eingabe des Passwortes gebildet und gespeichert.
In der smbpasswd müssen keine echten Klartextpasswörter gespeichert werden, sondern nur diese Hashwerte.
Struktur der smbpasswd:
- Username
- UID
- LAN Manager Password Hash: 32 Bit lange Zahl in hexadezimal, die das Kennwort für Windows 95/98 Clients repräsentiert.
- NT Password Hash: 32 Bit lange Zahl in hexadezimal, die das Kennwort für NT Clients repräsentiert.
- Account Flags: u gewöhnliches Konto, d Konto momentan deaktiviert, n besitzt kein Passwort, w Vertrauensstellung zu einer Arbeitstation
- Last Change Time: Zeichen LTC und hexadezimale Darstellung der Sekunden, die seit Begin des 1.Januars 1970 vergangen sind.

Kapitel 18

Um Drucker unter Samba zur Verfügung zu stellen, müssen diese von Unix aus ansprechbar sein. Dies geschieht durch die Einträge in der /etc/printcap. Alle Drucker, die dort definiert sind, kann man als Netzwerkdrucker für Windowsclients freigeben. Druckerfreigaben werden genauso behandelt wie andere Freigaben auch.