Konzepte von Betriebssystem-Komponenten
Schwerpunkt Sicherheit

Unix-Benutzerverwaltung: Grundlagen, OpenLDAP

Daniel Bast
daniel.bast@gmx.net

1 Einleitung

Ein zentrales Thema bei der Sicherheit von Rechnersystemen ist die Benutzerverwaltung. Gerade bei Unix-Systemen, zu denen oft zahlreiche Nutzer einen Zugang haben, gewinnt dieses Thema an Bedeutung. Hierzu existieren Mechanismen die Benutzerdaten lokal oder zentral auf einem oder mehreren Rechnern zu verwalten.
Im folgendem möchte ich zuerst als Grundlage die klassischen Benutzerverwaltung erläutern, um dann im zweiten Teil auf die Funktionsweise von OpenLDAP einzugehen.

2 Grundlagen: Klassische Benutzerverwaltung

2.1 Aufbau der Dateien: `/etc/passwd`, `/etc/shadow`, `/etc/group`

Bei der klassischen Benutzerverwaltung werden die Benutzerdaten lokal in den Dateien `/etc/passwd`, `/etc/shadow` und `/etc/group` gespeichert.

In der Benutzerdatei `/etc/passwd` wird der Nutzer eines Unix-System definiert. Die Datei besteht aus Zeilen mit sieben durch Doppelpunkt getrennten Einträgen:

```
Login-ID:Passwort:User-ID:Group-ID:Kommentar:Verzeichnis:Shell
```

Login-ID:	Nutzerkennung mit der sich der Benutzer anmeldet
Passwort:	Checksumme des Login- Passworts (die ersten 2 Stellen (= Salt) fließen zur Variation in das Hashverfahren mit ein; Anzahl der Variationen: 64^2=4096) ist dieser Eintrag leer → Nutzer hat kein Passwort ist dieser Eintrag einstellig (meist X, *) → kein Login möglich steht hier ein „x“ → es existiert die Datei /etc/shadow mit Passwort
User-ID:	eindeutige bijektiv zuordnungsbare Nummer des Benutzers; normale Nutzer beginnen ab 50, 100 oder 500; kleinere Nummern sind für Systemdienste; root hat 0
Group-ID:	Nummer der Hauptgruppe, der der Benutzer angehört
Kommentar:	weitere Informationen zum Benutzer z.B.: Name
Verzeichnis:	Home Verzeichnis des Benutzers. Hier befindet er sich nach dem Anmelden
Shell:	voller Pfad zu der Kommandoshell des Nutzers

Die Datei /etc/passwd muss für alle Nutzer zugänglich sein. Zum Beispiel ordnet `ls` anhand dieser Datei den User-IDs die Login-IDs zu. Das bringt den Nachteil mit sich, dass auch die Passwort Checksummen von allen lesbar sind. Durch Erzeugen von Checksummen für beliebige Zeichenketten und anschließendem Vergleich mit dem Wert in der passwd könnte man eines der möglichen Passwörter erraten. Das Anlegen eines großen Wörterbuches bestehend aus Zeichenfolgen mit zugehörigem Hash erfordert aber einen relativ großen Aufwand, da man für jede Zeichenfolge 4096 (= Anzahl der Verfahren) Hashes eintragen müsste.

Um eine höhere Systemsicherheit zu gewährleisten hat man die Datei /etc/shadow eingeführt. Diese Datei ist nur von root lesbar und enthält neben dem Passwort zusätzlich noch weitere Einträge. Sie besteht aus Zeilen mit neun durch Doppelpunkt getrennten Einträgen:

- Login-ID: die gleiche Nutzerkennung wie in /etc/passwd
- Checksumme des Passworts
- Tag des letzten Kennwortwechsels (Tage seit 01.01.1970)
- Mindestgültigkeitsdauer des Passworts in Tagen
- Maximalgültigkeitsdauer des Passworts in Tagen
- Anzahl der Tage, an denen vor dem Ende der Gültigkeitsdauer gewarnt wird
- Anzahl der Tage nach dem Ablauf der Gültigkeitsdauer bis der Zugang verweigert wird
- Anzahl der Tage seit 01.01.1970, an denen der Zugang gesperrt ist
- unbenutzt / reserviert

Neben der in /etc/passwd bestimmten Hauptgruppe kann ein Nutzer auch weiteren Gruppen angehören. Dies wird in der Datei /etc/group festgelegt. Diese Datei besteht aus Zeilen mit vier durch Doppelpunkt getrennten Einträgen:

- Gruppenname
- Gruppenpasswort
- Gruppen-ID
- Komma getrennte Liste der angehörigen User-IDs der Nutzer, die die Gruppe nicht als Hauptgruppe haben

2.2 Weitere Konfigurationsdateien:

Zusätzlich zu den bisher genannten Konfigurationsdateien gibt es noch weitere die zwar nicht unbedingt Nutzer spezifisch sind, es aber evtl. dennoch Sinn machen würde sie als solche zu verwalten (→ OpenLDAP).

`/etc/profile`	:	Nutzer spezifisches Script, das beim Login ausgeführt wird (außer C-Shell) und Umgebungsvariablen und diverse Einschränkungen (z.B.: max Stackspeicher, Dateigröße) setzen kann
`/etc/aliases`	:	Mail Aliases
`/etc/auto.master`	:	File System Table → Informationen für Automounter
`/etc/hosts`	:	IP-Adressen bekannter Hosts
`/etc/networks`	:	IP-Bereiche bekannter Netze

2.1.4 Beurteilung

Positiv:

- transparent
- schnell ← Konfiguration ist auf wenige Dateien mit Klartext beschränkt
- betriebsicher ← nur Hash des Passworts wird gespeichert; Änderungen nur root erlaubt

Negativ:

- Synchronisationsprobleme
- Pflegeaufwand linear zur Rechneranzahl ← keine zentrale Verwaltung der Konfiguration
- nicht erweiterbar ← Aufbau der Konfig.-Dateien fest vorgeben
- Unix-spezifisch

3 Moderne Benutzerverwaltung: OpenLDAP

3.1 Einleitung

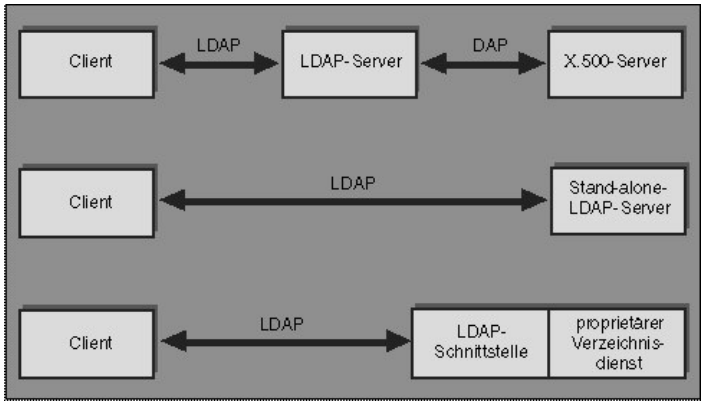
Heutige Rechner-Infrastrukturen werden immer umfangreicher und somit die Verwaltung dieser Rechner und ihr Informationsaufkommen immer komplizierter. Nicht nur im Internet verlieren sich die Informationen, die unstrukturiert ins Web gestellt werden. Auch in Unternehmensnetzen drohen Informationen verloren zu gehen, da sie entweder nicht mehr gefunden werden und gar deren Existenz nicht mehr bekannt ist. Es wird deshalb ein Informationsdienst gebraucht der folgende Anforderungen erfüllt:

plattformunabhängig – zentral u. lokal wartbar – einfach handhabbar – sicher – effektiv

Eine Lösung für diese Anforderungen sind Verzeichnisdienste. Hier diesem Fall OpenLDAP. Der Verzeichnisdienst LDAP wurde für diese Anforderungen entwickelt und bietet sich mit seinem Unterbau als zentraler Informationsdienst an.

3.2 OpenLDAP - ein Verzeichnisdienst

LDAP steht für Lightweight Directory Access Protocol und ist ein Verzeichnisdienst auf Basis einer baumähnlichen Datenbankstruktur. OpenLDAP ist die Open Source Implementierung von LDAP. Es wurde 1995 als leichtgewichtige Alternative zu X.500DAP von Young, Howes und Kille an der Universität Michigan spezifiziert. Ursprünglich war es nur ein Zugriffsprotokoll auf X.500 Server. So entstanden LDAP-Server die Anfragen vom Klienten entgegennehmen, in DAP-Anfragen umsetzen und an ein X.500 Server weiterleiten. Später wurden dann Stand-alone-LDAP-Server entwickelt, die selbst die Daten in einem Verzeichnis vorhalten.



Aufgaben von LDAP [3]

3.3 OpenLDAP im Detail

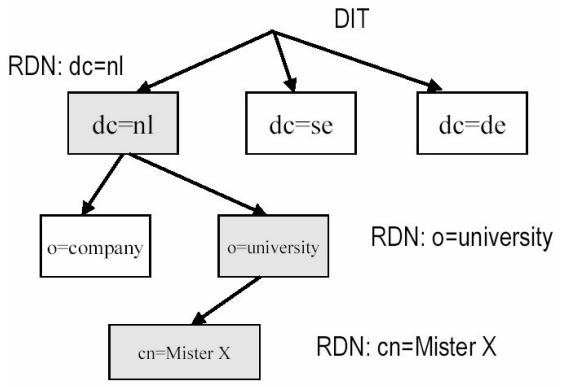
LDAP verwaltet seine Informationen in Objekten, die sowohl Personen, Geräte oder logische Objekte wie Gruppen sein können. Jedes Objekt hat einen oder mehrere Werte. Welche Attribute ein Objekt haben kann, bestimmt seine Objektklasse.

LDAP-Informationsmodell:

- Objekt und Attribut: Ein Objekt ist ein Datensatz, der aus Attributen mit einem oder mehren Werten besteht. Ein Attributtyp hat verschiedene Vergleichsregeln und unterliegt seinem zugehörigen Attributsyntax.
- Objektklassen: Es gibt viele verschiedene nach RFC standardisierte Objektklassen. z.B.: person (RFC 2256), organisation (RFC 2256) usw.
- Schema: Eine Zusammensetzung aus Objektklassen, Attributen und Syntaxregeln nennt man Schema. Man kann auch einfach selbst ein solches Schema definieren. Zur globalen Verwendung ist aber eine Registrierung oder Standardisierung sinnvoll (IETF-RFC).

LDAP- Namensraummodell:

- DIT, DN und RDN: Objekte werden als Baumknoten (Directory Information Tree; DIT) mit 0-n Kinderknoten und genau einem Elternknoten gespeichert. Jedes Objekt hat einen eindeutigen Namen. Alle Namen von der Wurzel zum Objekt bilden den Distinguished Name (DN); alle Namen von einem Elternknoten zum Objekt bilden den Relativ Distinguished Name (RDN).
- Beispiel:



aus [8]

Authentifizierung:

- Simple Bind: Authentifizierung mit DN und Passwort
- Simple Bind + SSL: Aufbau einer verschlüsselten Session → Authentifizierung dann mit DN und Passwort
- Alternative Authentifizierung mit SASL:
  - Simple Authentification und Security Layer
  - Vorgeschrieben: Digest MD5 (challenge response)
  - Andere SASL-Mechanismen möglich

3.4 Zugangskontrolle

mit ACL: (=Access Control List) Die Zugriffsrechte werden in der LDAP Serverkonfiguration gespeichert.

Nachteile: Server muss bei Änderung neu gestartet werden; keine automatische Replikation der ACL, weil nicht in der LDAP Datenbank

mit ACI: (=Access Control Information) Zugriffsrechte werden mit in der Datenbank gespeichert.

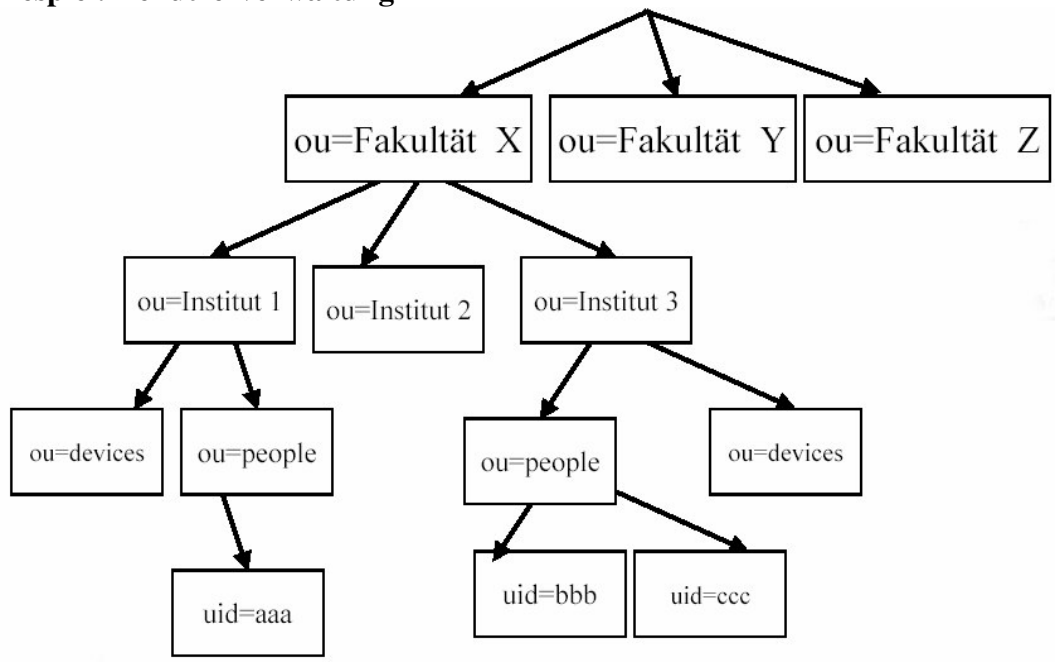
Vorteile: Änderungen on the fly möglich; Replikation über die Datenbank; Programme können den Access Level von Objekten auslesen

Nachteile: nicht so flexibel wie ACL (keine regulären Ausdrücke, Vererbung); hoher Verwaltungsaufwand, da jedes kontrollierte Objekt seine eigenen ACIs hat; noch im experimentellen Status

3.5 Anwendungen

- Benutzerverwaltung im heterogenen Netzwerk
- Authentifizierungsdienste
- Ressourcenverwaltung
- Public Key Verwaltung
- Kontaktverwaltung (Adressen, Telefonnummern, eMail Adressen…)
- ...

Bespiel: Benutzerverwaltung



aus [8]

3.6 Beurteilung

Vorteile:

- Objektorientierte Datenmodulierung
- Unabhängigkeit von Herstellern durch offenen Standard und Open Source Implementierung
- beliebige Skalierbarkeit durch Verteilung
- Ausfallsicherheit durch Replikation
- hohe Sicherheit durch Zugriffskontrolle und Authentifizierung
- Daten sind über TCP/IP basiertes Netzwerkprotokoll zugänglich
- keine unnötige Redundanz, da Daten von verschiedenen Anwendungen verwendbar

Nachteile:

- Konsistenzerhaltung schwierig: Da die Datenbank beliebige erweiterbar ist, kann das Wildwuchs zu folge haben → Erstellung strikter Schemata erforderlich
- Unterstützung von ACIs befindet sich noch im experimentellen Status

Literatur:

[1] Unix Benutzerverwaltung, Arnold Willemer
http://www.willemer.de/informatik/unix/unsyuser.htm

[2] Betriebssystem UNIX/Linux, Prof. Jürgen Plate
http://www.fs.ei.tum.de/admin/howto/unix/

[3] Linux: Installation, Konfiguration, Anwendung, Michael Kofler
3. Auflage Addison-Wesley, 1998

[4] LDAP Quellen: Wissen, Anleitungen, Artikel
http://www.sendung.de/ldap/wissen/

[5] LDAP – Überblick über das Chaos, Peter Wachtler, Linux-Magazin 09/1998
http://www.linux-magazin.de/Artikel/ausgabe/1998/09/LDAP/ldap.html

[6] Universalservice, Markus Jünemann, iX 8/1997
http://www.heise.de/ix/artikel/1997/08/118/

[7] OpenLDAP Documentation
http://www.openldap.org/doc/

[8] LDAP – Eine Einführung von Petra Haberer
http://www.mitlinx.de/ldap/assets/LDAP.pdf

[9] LDAP and OpenLDAP (on the Linux Platform), Adam Tauno Williams 2001
ftp://kalamazoolinux.org/pub/pdf/ldapv3.pdf