16.Authentifizierung

Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

Lesen der Informationsspeicher für Authentifizierungsinformationen

- Routinen und Programme für /etc/passwd und /etc/shadow
- PAM
- NIS
- NIS+
- LDAP
- RADIUS-Server
- SASL
- SAMBA-Server

Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

Files

/etc/passwd, /etc/shadow

Der Zugriff auf /etc/passwd ist in allen Systemen direkt möglich.

Es gibt keinerlei zusätzliche Schutzmechanismen für diese Datei.

Der Zugriff auf /etc/shadow (und ähnliche Dateien) ist nur durch „root“ möglich. Dadurch ist ein gewisser Schutz gegeben.

Beide Files können ohne zusätzliche Systeme (SASL, ..) nur lokal für die Zugriffssteuerung benutzt werden. Es sollten aber immer für den Fall der Fälle der Nutzer „root“ und ein „nicht-root“-Nutzer konfiguriert sein.

- für „root“: lokales Login am Terminal bei Netzausfall über ein „secure“ Terminal.
- für „nicht-root“: remote Login bei Ausfall des zentralen Authentifizierungssystems

Alle Programme benutzen zum Zugriff auf diese Files die Routinen.

- getpwent, getpw, getpwent_r, getpwnam, getpwuid, ...

16. Authentifizierung

Zugriffssteuerung mit /etc/nsswitch.conf

Die Datei /etc/nsswitch.conf wird bei vielen UNIX-Systemen für die Auswahl der benutzten Informationssysteme für die einzelnen Systemfiles (Datenbasen) benutzt.

Folgende Informationssysteme (Quellen) werden unterstützt:

Files, NIS, NIS+, LDAP, DNS

Der Zugriff zu folgenden Datenbasen wird durch /etc/nsswitch.conf gesteuert:

passwd, group, hosts, ipnodes, networks, protocols

rpc, ethers, netmasks, bootparams, publickey, netgroup

automount, aliases, services, sendmailvars, printers

auth_attr, prof_attr, project

Realisierung durch dynamische Libraries für die jeweilige Datenbasis:

(lib)nss_files.so, (lib)nss_nis.so, (lib)nss_nisplus.so,

(lib)nss_dns.so, (lib)nss_ldap.so, (lib)nss_compat.so,

16. Authentifizierung

Zugriffssteuerung mit /etc/nsswitch.conf

Ein Eintrag in /etc/nsswitch.conf hat folgenden Aufbau:

eintrag ::= *base***:** *src [* **[** *cond* **]** *] { src [* **[** *cond* **]** *] }*

base ::= **passwd** | **group** | **hosts** | **ipnodes** | **networks** | **rpc** | **ethers** | **netmasks** | **bootparams** | **publickey** | **netgroup** | **automount** | **aliases** | **services** | **printers** | **sendmailvars** | **auth_attr** | **prof_attr** | **project | passwd_compat | group_compat**

src :: = **files** | **nis** | **nisplus** | **ldap** | **compat**

cond ::= *returncode***=***aktion* { *returncode***=***aktion* }

Beispiel:

passwd: files

passwd: compat

passwd_compat: ldap

Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

Zugriffssteuerung mit /etc/nsswitch.conf

cond ::= returncode=*action { returncod*=*action }*

returncode ::= **SUCCESS** | **NOTFOUND** | **UNAVAIL** | **TRYAGAIN**

aktion ::= **return** | **continue** | **forever** | *nn*

Die Returncodes haben folgende Bedeutungen:

SUCCESS - Eintrag in der Quelle gefunden

NOTFOUND - Eintrag in der Quelle nicht gefunden

UNAVAIL - die Quelle ist auf dem System nicht vorhanden

TRYAGAIN - die Quelle reagiert nicht oder ist beschäftigt

Die Aktionen bewirken folgendes:

return - Rückkehr mit dem gefundenen Ergebnis

continue - nächste Quelle durchsuchen

forever - diese Quelle weiter durchsuchen

nn - diese Quelle *nn*-mal durchsuchen

16. Authentifizierung

Zugriffssteuerung mit /etc/nsswitch.conf

Die Defaultwerte für die Bedingungen cond sind wie folgt gesetzt:

DNS und NIS als Quelle:

[SUCCESS=return NOTFOUND=continue UNAVAIL=continue TRYAGAIN=3]

Alle anderen Quellen:

[SUCCESS=return NOTFOUND=continue UNAVAIL=continue TRYAGAIN=forever]

Für die Datenbasen **passwd, group** und **hosts** gibt es noch die Quelle **compat** (Kompatibilitätsmodus). Diese beinhaltet als erstes das Durchsuchen der entsprechenden Quelle **file** mit Auswertung von „+*wert*“, „-*wert*“, „+*@wert*“, „-*@wert*“ für die anschließende Durchsuchung der zweiten Quelle. *@wert* greift dabei auf die Datenbasis **netgroup** zurück.

Die Datenbasen **passwd_compat** und **group_compat**, bestimmen die zweite Datenquelle für den Kompatibilitätsmodus.

16. Authentifizierung

Zugriffssteuerung mit /etc/nsswitch.conf

Beispiele:

```
passwd: compat
passwd_compat: nis
group: files nis
hosts: files dns
hosts: nis [NOTFOUND=return] files        !!!! sollte man nicht benutzen
netgroup: nis
protocols: files nis
```

/etc/passwd (für compat):

```
+bell
+@rbg
-@nouser
+@umx500::::::/bin/false:
```


Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

PAM - Pluggable Authentication Moduls

Bestimmte Programme müssen die Nutzer authentifizieren können (Nutzername und Passwort oder Zertifikat). Mittels PAM kann der Systemadministrator die Authentifizierungsregeln für die verschiedenen Programme individuell festlegen ohne die Programme selbst neu zu kompilieren. Die Konfiguration für PAM befindet sich in der Regel im File **/etc/pam.conf** oder im Directory **/etc/pam.d**. Für jedes Programm, das PAM benutzt, wird dann in dem Directory ein File angelegt, in dem die Authentifizierungsregeln aufgelistet sind.

Vorteile von PAM:

- gemeinsame Authentifizierungsverfahren für verschiedene Programme
- Große Flexibilität
- Trennung von Authentifizierung und Programm

Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

PAM

Konfigurationsdateien

Ein Eintrag in dem Konfigurationsfile ***/etc/pam.d/****service* hat folgenden Aufbau:

type control module-name modul-arguments

oder

type **include** *filename*

Ein Eintrag in dem Konfigurationsfile **/etc/pam.conf** hat folgenden Aufbau:

service type control module-name modul-arguments

service ist dabei der Name des Programms, das PAM benutzen will (su, login, passwd, sshd, telnetd, rlogind) Mehrere Eintragungen für einen Service sind möglich.

16. Authentifizierung

PAM

Eintrag in die Konfigurationsdatei

type control modul-name modul-arguments

type gibt die Managementklasse an. Folgende Managementklassen sind zulässig:

auth - authentication service: Modul realisiert die Authentifizierung, Zulassungen werden gespeichert

account - account managment: Modul betrifft Accountverwaltung (Gültigkeit, Zeit zulässig, Passwort abgelaufen)

session - Verwalten von Sessions. Start und Ende einer Login-Sitzung.

password - Passwortverwaltung, Passwort darf geändert werden, Passwort muß geändert werden, Passwort ist zu leicht.

16. Authentifizierung

PAM

Eintrag in die Konfigurationsdatei

type control modul-name modul-arguments

Das *control*-Feld enthält Informationen über die Wirkung des Resultats eines Moduls und über die weitere Abarbeitung der Module innerhalb des Modulfolge. Alle Module des gleichen Typs gehören zu einer Folge und werden in Abhängigkeit des *control*-Feldes abgearbeitet.

binding - required Modul. Wenn der Modul success meldet und alle vorangegangen required Module ebenfalls, wird die PAM-Authentifizierung mit success beendet. Andernfalls wird die Modulfolge fortgesetzt und bei einem Fehler dies als required-Fehler vermerkt.

optional - optional Modul. Der PAM-Authentifizierung wird fortgesetzt. Ein Fehler wird als optional Fehler vermerkt.

16. Authentifizierung

PAM

Eintrag in die Konfigurationsdatei

type control modul-name modul-arguments

control-Feld

required - required Modul. PAM-Authentifizierung wird fortgesetzt. Bei success wird dies für required Module vermerkt. Bei nosuccess wird der Fehler gemerkt, wenn dies der erste required-Fehler war.

requisite - required Modul. Bei success wird der PAM-Authentifizierung fortgesetzt. Bei Fehler wird die Modul-Folge sofort beendet und der 1.required-Fehler gemeldet (notwendig).

sufficient - optional Modul. Bei success wird die Modul-Folge sofort mit success beendet unabhängig von vorangegangen Ergebnissen. Bei nosuccess wird ein optinaler Fehler gemeldet und die Modul-Folge fortgesetzt.

include - einfügen des angegeben Files in das Konfigurationsfile *modul-name*

Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

PAM

Eintrag in die Konfigurationsdatei

type control modul-name modul-arguments

module-name ist der Name des entsprechenden Moduls, der die Überprüfung vornehmen soll. Der Name kann absolute oder relativ zu **/lib/security** bzw. **/usr/lib/security** angegeben werden.

modul-arguments sind verschieden Optionen, die je nach Modul angegeben werden können und zusätzliche Eigenschaften aktivieren, z.B.:

pam_unix2: **debug, nullok, not_set_pass, use_first_pass, use_authok**

pam_pwcheck: **blowfish, debug, not_set_pass, nullok, md5, bigcrypt**

pam_unix_auth: **nowarn, server_policy, nolock**

16. Authentifizierung

PAM

Beispiel: login

```
auth       requisite    pam_nologin.so   # kein login wenn /etc/nologin existiert
auth       [user_unknown=ignore success=ok ignore=ignore auth_err=die
                  default=bad] pam_securetty.so  # root-login-tty   in  /etc/securetty
auth        include      common-auth
account    include      common-account
session    required     pam_loginuid.so     # setzen von uid
session    include      common-session
session    required     pam_lastlog.so nowtmp # /var/log/lastlogin füllen
session    required     pam_resmgr.so
session   optional      pam_mail.so standard    # you have new mail
```

16. Authentifizierung

PAM

Beispiel: common-auth

```
auth       required     pam_env.so
auth       sufficient   pam_unix2.so              # in /etc/passwd, ok
auth       required     pam_ldap.so    use_first_pass    #  ldap
```

Beispiel: common-account

```
account    requisite    pam_unix2.so
account    sufficient   pam_localuser.so        # in /etc/passwd, ok
account    required     pam_ldap.so  use_first_pass    # ldap
```


Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

NSCD - Name Service Cache Daemon

Da das Lesen von Informationen aus einer Datenbasis für die verschiedenen Informationssysteme relativ lange dauert, wurde in den Zugriffsmechanismus für die Datenbasis bei neueren Systemen der **NSCD** eingeschoben. Der **NSCD** speichert alle Anfragen und Antworten an die Datenbasen für **passwd**, **group**, **hosts** und **ipnodes** und gibt selbstständig Antworten auf Anfragen, wenn diese bereits im Speicher sind, ohne das Informationssystem erneut abzufragen!!!

Folgende Bibliotheksrufe benutzen einen vorhandenen **NSCD**:

gethostbyname, **getipnodebyname**, **gethostbyaddr**, **getspnam**, **getpwuid**, **getgrid**, **getgrgid**

Wird der **NSCD** nicht gestartet, führt jeder Zugriff auf die Datenbasis zu einer Abfrage der entsprechenden Informationssysteme.

Achtung: Das **NSCD** benutzt eigene Timeouts, so dass eine Änderung an der Datenbasis sich nicht unmittelbar auswirkt!!!!

16. Authentifizierung

NSCD - **N**ame **S**ervice **C**ache **D**aemon

Starten von **NSCD**:

/usr/sbin/nscd [**-f** *conf-file*] [**-g**] [**-e** *cachename*, **yes** | **no**] [**-i** *cachename*]

-f conf-file - Konfigurationsfile. Standard: **/etc/nscd.conf**
-g - Cache-Status ausgeben
-e *cachename*,**yes** - Cache aktivieren
-e *cachename*,**no** - Cache deaktivieren
-i *cachename* - Cache löschen

Beispiele:

```
nscd -i hosts  # Cache für hosts löschen
nscd -g        # Cache-Status ausgeben
```

16. Authentifizierung

NSCD - Name Service Cache Daemon

Konfigurationsfile **/etc/nscd.conf**

```
# logfile                 /var/log/nscd.log
debug-level               0
# parameter               cache    value
# enable-cache            hosts    no
poitive-time-to-live      hosts    3600     # Gültigkeit bei gefundenem Wert
negative-time-to-live     hosts    5        # Gültigkeit bei nicht gefunden
suggested-size            hosts    211      # erwartete Größe des Caches
                                            # Primzahl
keep-hot-count            hosts    200      # Anzahl der Werte im Cache
check-files               hosts    yes      # Prüft Änderungen von /etc/hosts
# cachenames: passwd, group, ipnodes, exec_attr, user_attr, prof_attr
```

16. Authentifizierung

NSCD - **N**ame **S**ervice **C**ache **D**aemon

Achtung:

Verträglichkeit der verschiedenen Informationsquellen (Files, NIS, NIS+, LDAP, DNS) untereinander und mit NSCD auf dem jeweiligen System überprüfen!!!!

Bekannte Unverträglichkeiten:

Linux: hosts: files nis dns

hosts: files ldap

Bei Unverträglichkeit hilft oft das Abschalten des **NSCD**.

16. Authentifizierung

NIS – Network Information Service

Zugriff erfolgt über die üblichen Zugriffsroutinen (**/etc/nsswitch.conf** gesteuert).
Nutzer und Administrator können mit ypcat und ypmatch auf alle Daten zugreifen.

ypcat [**-k**] [**-x**] [**-d** *ypdomain*] *mapname*

Anzeigen der angegebenen Map mit dem Namen *mapname*. Der Mapname kann als voller Name (**hosts.byname**, **hosts.byaddr**) oder als Nickname (**passwd**, **hosts**) angegeben werden.

-k - mit Schlüssel ausgeben

-x - anzeigen der Nicknames

-d *ypdomain*

Beispiele:

ypcat -k auto.home

ypcat -k passwd

16. Authentifizierung

NIS

ypmatch [**-k**] [**-t**] [**-d** *ypdomain*] *key* { *key* } *mapname*

ypmatch -x

Anzeigen der Eintragung mit den angegebenen Schlüsseln key der Map mit dem Name mapname. Der Mapname kann als voller Name (**hosts.byname**, **hosts.byaddr**) oder als Nickname (passwd, hosts) angegeben werden.

-k - mit Schlüssel ausgeben

-t - verbieten von Nicknames

-x - anzeigen der Nicknames

-d *ypdomain*

Beispiel:

ypmatch -k bell tbell passwd

ypmatch bell passwd.byname

16. Authentifizierung

NIS+ - Network Information System +

Zugriff erfolgt über die üblichen Zugriffsroutinen. Zusätzlich können sich Nutzer und Administratoren mittels **niscat** und **nismatch** die für sie zugelassenen Eintragungen und Tabellen anschauen.

niscat [**-AhLMv**] [**-s** *sep*] *tablename*

Ausgabe aller Eintragungen der Tabelle mit dem Namen *tablename*.

- **-A** - Ausgabe aller Eintragungen
- **-h** - Ausgabe einer Kopfzeile
- **-L** - Folgen von Links
- **-M** - Daten vom Masterserver holen
- **-v** - Ausgabe von binary-Daten
- **-s** *sep* - Seperatorzeichen sep für die Ausgabe benutzen

Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

NIS+

niscat [-**AhLMv**] [**-s** *sep*] *tablename*

Beispiele:

niscat -h hosts.org_dir
niscat passwd.org_dir
niscat -A hosts.org_dir
niscat -M hosts.org_dir
niscat -L hosts.org_dir
niscat -o hosts.org_dir

Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

NIS+

nismatch [**-AchMoP**v] [**-s** *sep*] *key tablename*

nisgrep [**-AchMoPv**] [**-s** *sep*] *keypattern tablename*

Ausgabe von durch Schlüssel *key* oder Schlüsselmuster spezifzierten Eintragungen einer Tabelle mit dem Namen tablename.

-A - alle gefundenen Eintragungen ausgeben

-c - nur die Anzahl der gefundenen Eintragungen ausgeben

-h - mit Header

-M- Eintragungen vom Master holen

-o - interne Repräsentation ausgeben

-P - Pfad folgen

-v - verbose, ausgabe von binären Daten

-s *sep* - Tabellenseperatorzeichen sep

Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

NIS+

nismatch [**-AchMoPv**] [**-s** *sep*] *key tablename*
nisgrep [**-AchMoPv**] [**-s** *sep*] *keypattern tablename*

Beispiele:

nismatch bellus hosts.org_dir
nisgrep bellus hosts.org_dir
nismatch -h bellus hosts.org_dir
nisgrep -h bellus hosts.org_dir
nismatch -c bellus hosts.org_dir
nismatch -o bellus hosts.org_dir

16. Authentifizierung

LDAP

Zum Auslesen der UNIX-typischen Informationen aus einer LDAP-Datenbank wird normalerweise die Software von PADL benutzt. Sie besteht aus den Bibliotheken nss_ldap.so und pam_ldap.so und wird durch Eintragungen in **/etc/nsswitch.conf** bzw. in **/etc/pamd/...** aktiviert. Die Konfiguration des Zugriffs auf die Eintragungen im LDAP-Server erfolgt über die verschiedenste **CONF**-Files:

/etc/ldap.conf

/etc/openldap.conf

/etc/ldaprc

~home/ldaprc

~home/.ldaprc

Häufig werden die Libraries für spezielle Anwendungsfälle individuell mit eigenen Konfigurationsfiles übersetzt, z.B. **/etc/pam_ldap.conf**

16. Authentifizierung

LDAP

Eintragungen in den Konfigurationsfiles:

/etc/openldap/ldap.conf

/etc/ldaprc (sollten beide existieren - link ok, **ldapsearch**)

```
BASE   dc=informatik,dc=hu-berlin,dc=de
URI   ldaps://141.20.20.7:636
TLS_CACERTDIR /etc/openldap/CACerts
TLS_CACERT      /etc/openldap/CACerts/cacert.pem
TLS_CIPHER_SUITE   HIGH:MEDIUM:+SSLv2:+SSLv3
TLS_REQCERT demand
```

16. Authentifizierung

LDAP

Eintragungen in den Konfigurationsfiles:

/etc/ldap.conf (**nss_ldap.so**) (1)

```
URI   ldaps://141.20.20.8:636
base  ou=all,dc=informatik,dc=hu-berlin,dc=de
ldap_version 3
pam_password   crypt
nss_base_passwd ou=People,ou=all,dc=informatik,dc=hu-berlin,dc=de
nss_base_shadow ou=People,ou=all,dc=informatik,dc=hu-berlin,dc=de
nss_base_group  ou=Group,ou=all,dc=informatik,dc=hu-berlin,dc=de
nss_base_netgroup ou=Netgroup,ou=all,dc=informatik,dc=hu-berlin,dc=de
```

16. Authentifizierung

LDAP

Eintragungen in den Konfigurationsfiles:

/etc/ldap.conf (**nss_ldap.so**) (2)

```
nss_map_attribute   uniqueMember member
pam_filter    objectclass=posixAccount
TLS_CACERTDIR  /etc/openldap/CACerts
TLS_CACERT       /etc/openldap/CACerts/cacert.pem
TLS_CIPHER_SUITE  HIGH:MEDIUM:+SSLv2:+SSLv3
TLS_REQCERT    demand
TLS_CHECKPEER   no
```

16. Authentifizierung

LDAP

Das allgemeine Suchprogramm für den Nutzer und den Administratoren in der LDAP-Datenbasis ist **ldapsearch**. Für den Administrator steht außerdem noch das nur offline zu nutzende Programm *slapcat* zur Verfügung.

ldapsearch [**-n**] [**-u**] [**-v**] [**-k**] [**-K**] [**-t**] [**-A**] [**-L**[**L**[**L**]]] [**-M**[**M**]] [**-d** *debuglevel*] [**-f** *file*]
[**-D** *binddn*] [**-W**] [**-w** *passwd*] [**-y** *passwdfile*] [**-H** *ldapuri*] [**-h** *ldaphost*]
[**-p** *ldapport*] [**-P 2**|**3**] [**-b** *searchbase*] [**-l** *timelimit*] [**-z** *sizelimit*]
[**-s base**|**one**|**sub**] [**-a never**|**always**|**search**|**find**] [**-I**] [**-Q**] [**-U** *authcid*]
[**-O** *scurity-properies*] [**-R** *realm*] [**-x**] [**-X** *authzid*] [**-Y** *mech*] [**-Z**[**Z**]]
filter [*attribute*]

Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

LDAP

Beispiele:

```
ldapsearch -x -D cn=Manager,dc=informatik,dc=hu-berlin,dc=de -W uid=tbell
ldapsearch -x -D cn=Manager,dc=informatik,dc=hu-berlin,dc=de -w geheim  \
                                uid=tbell
ldapsearch -x uid=tbell
ldapsearch -x -b ou=People,ou=all,dc=informatik,dc=hu-berlin,dc=de uid=tbell
ldapsearch -x ipHostNumber=141.20.20.14
ldapsearch -x iphostnumber=141.20.20.14
ldapsearch -x cn=fipo
slapcat
```

16. Authentifizierung

RADIUS

Radius ist ein Protokoll, das den **AAA** Prozess unterstützt. **AAA** steht für

Authentication - verifizieren eines Nutzers oder Rechners mittels eines Accounts und eines Passwortes oder eines Zertifikates.

Authorization - Zuordnung einer Rolle oder von Zugriffsrechten für einen Nutzer, der authentifiziert wurde.

Accounting - Erfassung von Abrechnungsdaten (login, logout)

Radius ist ein Client-Server-Protokoll, bei dem der Client :

1. einen Nutzer verifizieren will
2. Informationen über den Nutzer erhalten will
3. Abrechnungs-Informationen speichern will.

Der Client erhält dabei keinen Zugriff auf die eigentlichen Nutzerdaten. Er kennt lediglich einen vertrauenswürdigen Server. Dieser Server kann weitere Server befragen. Der Client bekommt nur ja oder nein als Antwort und eine Rolle.

16. Authentifizierung

RADIUS

Grundlegende Merkmale:

- Radius basiert auf dem UDP-Protokoll
- Radius-Pakete werden jeweils zwischen zwei Partnern verschlüsselt
- Radius ist zustandslos
- Radius unterstützt PAP und CHAP
- Radius benutzt MD5 für die Verschlüsselung
- Radius ist gedacht für Netzwerktechnik
- Radius unterstützt herstellerspezifische Attribute
- Radius erlaubt den Anschluss an verschiedenste Informationsspeicher

Es existieren verschiedenste Implementationen. Standard ist z.Z. www.freeradius.org

16. Authentifizierung

RADIUS

Bestandteile von FreeRadius:

Server-Programme: **radiusd**, **radrelay**

Diverse Wartungsprogramme: **radacct**, **radclient**, **radlast**, **radlogin** **radstatus**,**radtest**, **radwho**, **radzap**

Directory **/etc/raddb** mit diversen Konfigurationsfiles:

acct_users, **attrs, certs** (Directory), **clients.conf**, **dictionary**, **eap.conf**, **hints**, **huntgroups**, **ldap.attrmap**, **mssql.conf, naspasswd**, **oraclesql.conf**, **preproxy**_users, **radiusd.conf**, **realms**, **snmp.conf**, **users**

Früher erfolgte die Konfiguration in einem File, jetzt gibt es für jede Komponente ein eigenes Konfigurationsfile - sehr aufwendig, nicht sehr leicht verständlich.

16. Authentifizierung

RADIUS

Die wichtigsten Konfigurationsfiles:

acct_users - Accountingaktionen bei Start und Stopp

attrs - Standardattribute für alle Realms

clients.conf - Enthält Informationen über die zugelassenen Clienten einschließlich des Verschlüsselungspasswortes

dictionary - Wörterbuch mit der Beschreibung der Attribute für die einzelnen Hersteller

proxy.conf - Beschreibung weitere Radius-Server

radiusd.conf - Konfigurationsfile des Radiusservers, benutzte Directories

Humboldt Universität Informatik
Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

RADIUS

Beispiele für Konfigurationsfiles:

mail:/opt/csw/etc/raddb

Protokollfiles:

mail:/opt/csw/var/log/radius

radius.log - allgemeines Protokollfile

radacct/141.20.20.41/detail-20070125

- detailliertes Accountingfile

16. Authentifizierung

SASL

Kein eigenes Authentifizierungssystem. Setzt auf bestehende Authentifizierungssysteme auf, z.B. pam, ldap, nis.

Bestandteile:

- saslauthd - Authentifizierungsdaemon
  - /etc/saslauthd.init - Basiskonfigurationsfile
  - /etc/saslauthd.conf - Extended Konfigurationsfile
  - /lib/sasl2 - Direktory für Konfigurationsfiles
  - /lib/sasl2/Sendmail.conf - Konfigurationsfile für „sendmail“
  - /etc/sasldb2 - eigener Nutzer-/Passwortspeicher
- saslpasswd2 - Tool zum Erzeugen von Nutzern in /etc/sasldb2
- testsaslauthd - Tool zum Testen der Funktionsfähigkeit von saslauthd

16. Authentifizierung

SASL

/etc/saslauthd.init oder /etc/default/saslauthd - Basiskonfigurationsfile für Startscript

MECHANISM=ldap

CONFIG_FILE="/opt/csw/etc/saslauthd.conf"

SASL_OPTIONS="-n 5 -c"

/opt/csw/etc/saslauthd.conf - Einstellungen für saslauthd mit LDAP

ldap_servers: ldaps://141.20.20.7

ldap_search_base: ou=all,dc=informatik,dc=hu-berlin,dc=de

ldap_bind_dn: uid=sasladmin,ou=People,ou=all,dc=informatik,....

ldap_bind_pw: geheim

ldap_auth_method: bind

ldap_ssl: on

/opt/csw/lib/sasl2/Sendmail.conf

pwcheck_method: saslauthd

mech_list: login plain cram-md5 digest-md5

16. Authentifizierung

SAMBA

Komplexes System zum Ersetzen von Active Directory von Windows unter Unix-Umgebungen. Aktuelle Produktionsversion ist Samba 4.1.3. Seit 2006 wird an Samba 4 gearbeitet.

Leistungen:

- Bereitstellen von Authentifizierungsinformationen (Active Directory)
- Bereitstellen von Netzwerklaufwerken für Windows-Rechner
- Bereitstellen von Druckerdiensten für Windows-Rechner
- Verbindung von UNIX-Welt mit Windows-Welt

16. Authentifizierung

SAMBA

Installation

Notwendige Pakete

- openssl
- openldap
- (kerberos)
- sasl2
- zlib
- ncurses
- samba

16. Authentifizierung

SAMBA

Konfigurationsfiles

smb.conf - Konfigurationsfile
secrets.tdb - Samba eigene Passwörter
smbpasswd - lokale Samba-Passwortdatei
smbusers - mapping Unix-Nutzer ↔ Windows-Nutzer

Mögliche Direktories für Konfigurationsfiles
/etc/samba/
/usr/local/etc/samba/
/usr/local/samba/lib/
/opt/csw/etc/samba/

16. Authentifizierung

SAMBA

Beispiel(1):

/opt/csw/etc/samba/smb.conf

/opt/csw/samba/lib/smb.conf

```
[global]
    workgroup = HOME
    netbios = HOME
    dns proxy = YES
    os level = 65
    passdb backend = ldapsam:ldap://repl201.informatik.hu-berlin.de
    ldap ssl = start_tls
    ldap admin dn = smbadmin,ou=People,dc=...
    encrypt passwords = YES
    ....
```

16. Authentifizierung

SAMBA

Beispiel(2):

/opt/csw/etc/samba/smb.conf

/opt/csw/samba/lib/smb.conf

```
[global]
    ....
    ldap machine suffix = ou=Hosts
    ldap user suffix = ou=People
    ldap group suffix = ou=Groups
    winbind nested groups = no
    security = user
    hosts allow = 127. 141.20.20.0/24
    log file = /var/log/samba/log.%m
    log level = 1
```

16. Authentifizierung

SAMBA

Beispiel(3):

/opt/csw/etc/samba/smb.conf

/opt/csw/samba/lib/smb.conf

```
[global]
    ....
    max log size = 50
    domain master = NO
    printcap = /etc/printers.conf
    follow symlinks = no
    locking = no
    smb ports = 445
```

16. Authentifizierung

SAMBA

Beispiel(4):

/opt/csw/etc/samba/smb.conf

/opt/csw/samba/lib/smb.conf

```
[homes]
 comment = Home Directories
 browseable = NO
 writable = YES
 invalid users = adm root bin
 path = %H
[ … ]
```


Systems Architecture Group
http://sar.informatik.hu-berlin.de

16. Authentifizierung

SAMBA

Systems Architecture Group
http://sar.informatik.hu-berlin.de