Authentifizierung und Autorisierung unter Linux/Solaris: PAM und NSS

Vortrag im Proseminar „Konzepte von Betriebssystemkomponenten“

Clemens Lang

sicslang@stud.cs.fau.de

09. Juni 2010

Motivation

Problem

- Linux: Benutzer in der Datei `/etc/passwd`
- Setup mit mehreren Rechnern:
  Authentifizierung von Benutzern ohne Anpassung von `/etc/passwd` auf jedem Rechner?

Herangehensweise

Wie funktioniert die Authentifizierung unter Linux?

Motivation

Problem

- Linux: Benutzer in der Datei `/etc/passwd`
- Setup mit mehreren Rechnern:
  Authentifizierung von Benutzern ohne Anpassung von `/etc/passwd` auf jedem Rechner?

Herangehensweise

Wie funktioniert die Authentifizierung unter Linux?

Gliederung

Gliederung

Problem

Problem

- Services (z.B. *ssh*, *login*, *gdm*, *ftp*, *screensaver* . . .) müssen Benutzer authentifizieren
- Services bestimmen zunächst selbst, welche Authentifizierungsmethoden sie nutzen
- Neue Authentifizierungsmethoden oder Bugfixes müssen in jedem Programm implementiert werden

Lösung

Abstrahierung der Authentifizierungsmethoden von den Services

Problem

Problem

- Services (z.B. *ssh*, *login*, *gdm*, *ftp*, *screensaver* . . .) müssen Benutzer authentifizieren
- Services bestimmen zunächst selbst, welche Authentifizierungsmethoden sie nutzen
- Neue Authentifizierungsmethoden oder Bugfixes müssen in jedem Programm implementiert werden

Lösung

Abstrahierung der Authentifizierungsmethoden von den Services

PAM abstrahiert Authentifizierungsmethoden

Was ist PAM?

- PAM steht für *Pluggable Authentication Modules*
- Abstraktionsebene zwischen sog. „system-entry services“ und Authentifizierungsmethoden
- PAM erlaubt Administratoren Authentifizierungsmethoden zu wählen

Entstehungsgeschichte

- Entwickelt von Sun als interne Komponente von Solaris
- 1996 durch Samar spezifiziert und publiziert [Sam96]
- Spezifikation in Linux implementiert

PAM abstrahiert Authentifizierungsmethoden

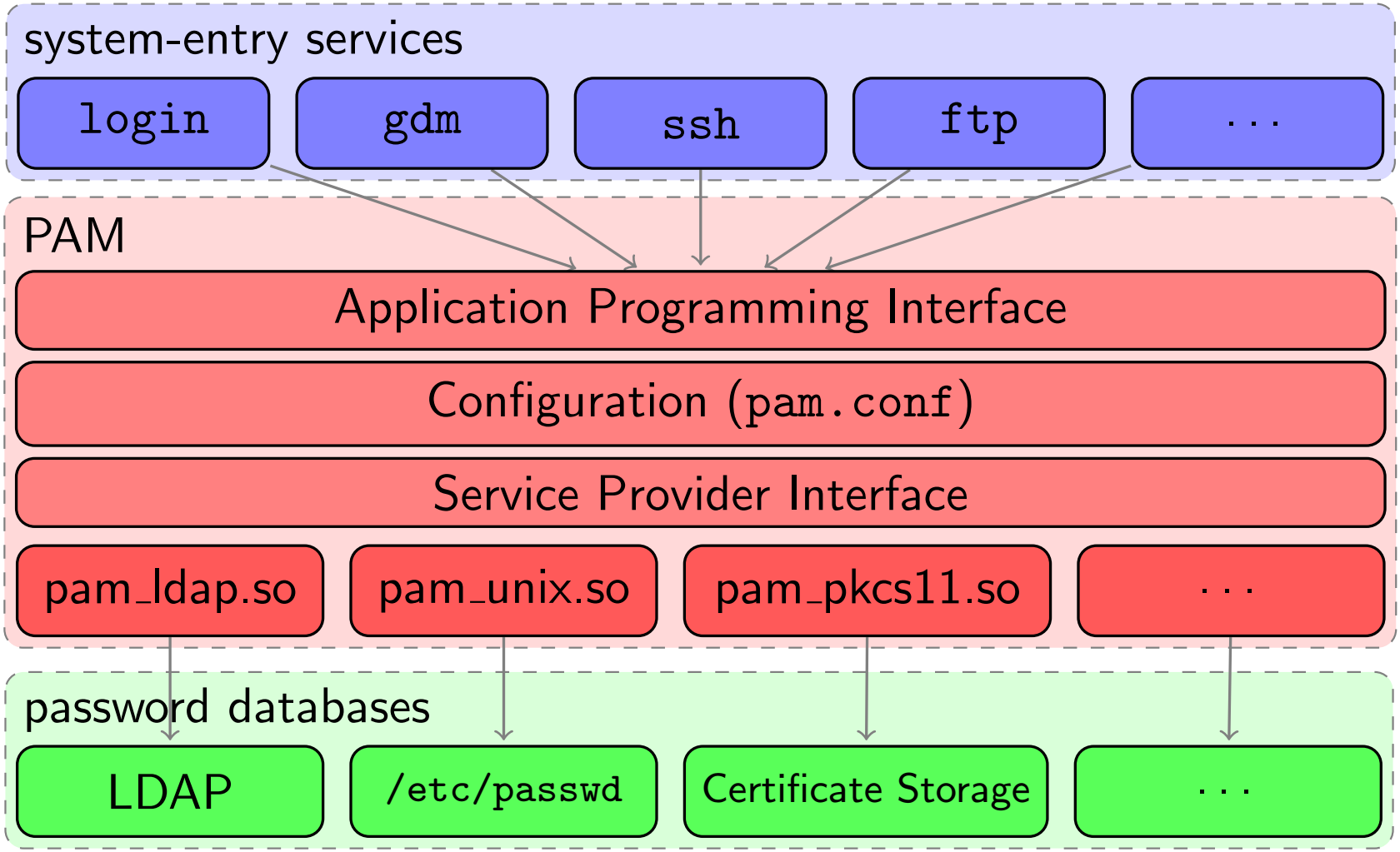
Was ist PAM?

- PAM steht für *Pluggable Authentication Modules*
- Abstraktionsebene zwischen sog. „system-entry services“ und Authentifizierungsmethoden
- PAM erlaubt Administratoren Authentifizierungsmethoden zu wählen

Entstehungsgeschichte

- Entwickelt von Sun als interne Komponente von Solaris
- 1996 durch Samar spezifiziert und publiziert [Sam96]
- Spezifikation in Linux implementiert

Schematischer Aufbau



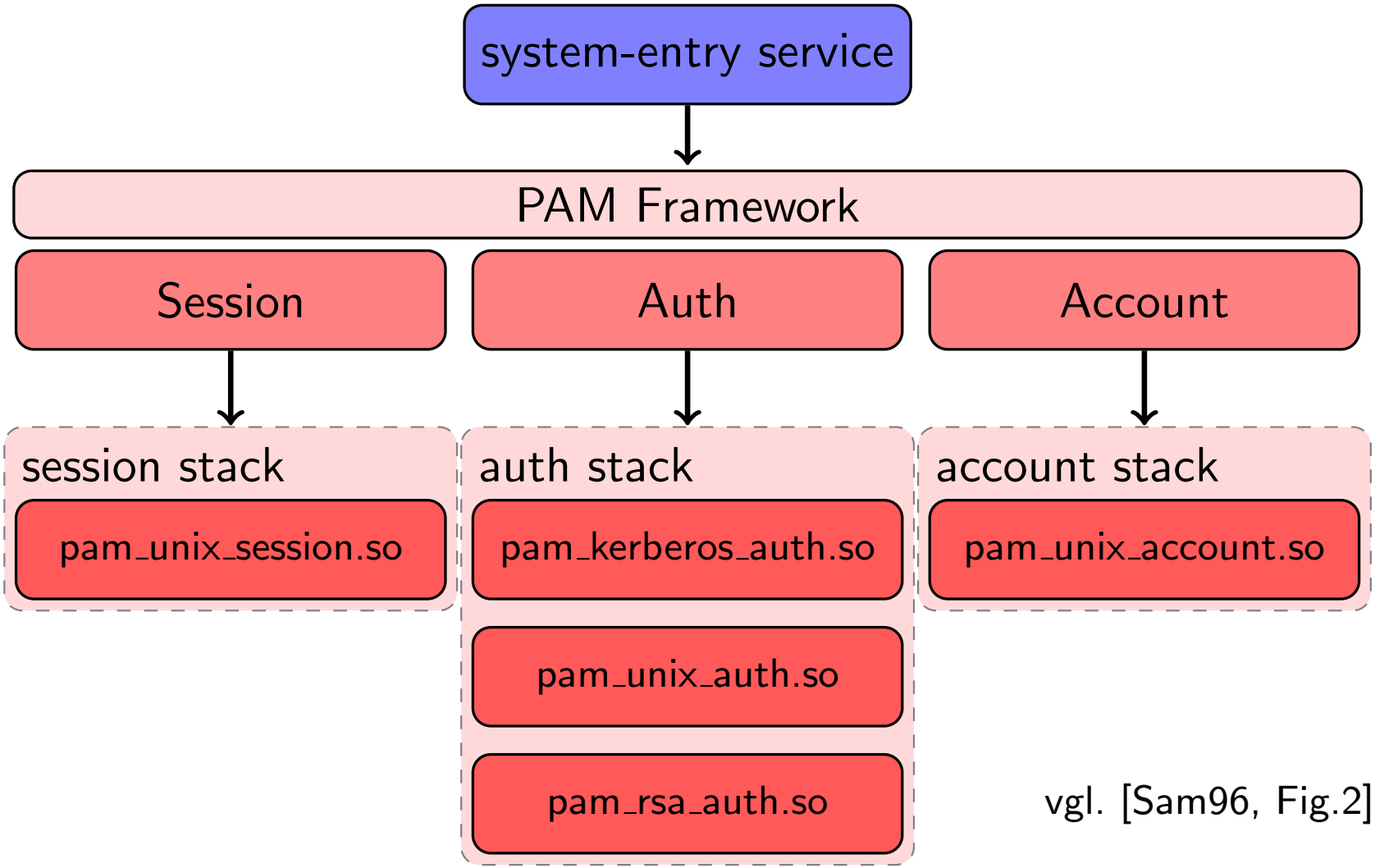
Logischer Aufbau

Logischer Aufbau

Authentifizierung	Ist der Benutzer, wer er vorgibt zu sein?
Kontoverwaltung	Ist der Account abgelaufen oder gesperrt?
Sitzungsverwaltung	Ist der Benutzer bereits eingeloggt?
	Wie lange ist der Benutzer eingeloggt?
Passwortverwaltung	Änderung des Passworts

- Unabhängig voneinander konfigurierbar
- Nutzung durch Services optional

Stapeln von Modulen



vgl. [Sam96, Fig.2]

Konfiguration: pam.conf bzw. pam.d/

Beispielkonfiguration von PAM

Service	Type	Flags	Module Path	Options
login	auth	required	pam_kerb_auth.so	debug
login	auth	required	pam_unix_auth.so	use_mapped_pass
login	auth	optional	pam_rsa_auth.so	try_first_pass

[Sam96]

Funktionsweise aus Anwendungssicht

Beispielhafter Ablauf einer Authentifizierungsanforderung

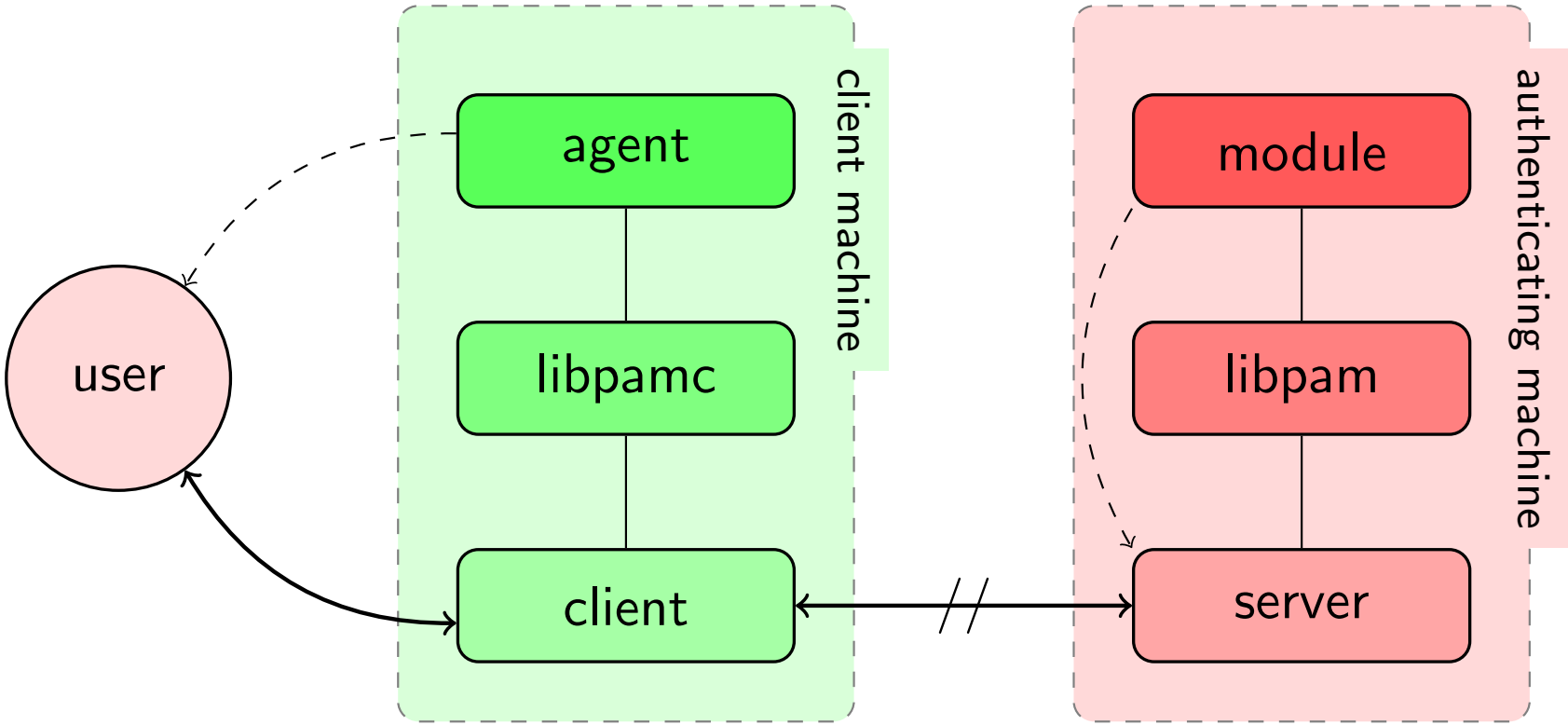
`pam_start()`	Erzeugen eines PAM Handles
`pam_authenticate()`	Authentifizierung des Nutzers
`pam_acct_mgmt()`	Prüfen das Kontos des Nutzers
`pam_open_session()`	Benachrichtigung über neue Session
`pam_setcred()`	Setzen von Eigenschaften, z.B. Starten eines `ssh-agent`
`pam_end()`	Beenden des Authentifizierungsprozesses

Schnittstelle zum User: Conversation Function

Schnittstelle zum User

- Benutzerinteraktion durch Callback-Funktion
- 4 Nachrichten möglich:
  - Eingabeaufforderung (versteckt oder sichtbar)
  - Fehlermeldung
  - informative Meldung
- Authentifizierung (nach dem Standard) nur durch Eingaben
- $\Rightarrow$ z.B. Token/Fingerabdruck nur lokal nutzbar

Erweiterungen in Linux-PAM



[Mor01]

Erweiterungen in Linux-PAM: libpamc

client-side support für PAM

- libpamc: Client-seitige Unterstützung für PAM [Mor01]
- Theoretisch: Token/Fingerabdruck/... durch sog. *Agent* am Client auch entfernt nutzbar
- Praktisch: Keine Implementierung

Gliederung

Name Service Switch

Benutzerdatenbank jenseits der Authentifizierung

- Wie zeigt z.B. `ls -l` Benutzernamen an?
- GNU C Library `pwd.h`
  $\Rightarrow$ libc muss Zugriff auf die Benutzerdatenbank haben

Name Service Switch [lib]

- Konfigurierbare Schnittstelle
- Leitet Datenbankabfragen an Module weiter
- Module sind Shared Libraries, z.B. `libnss_ldap.so`

Name Service Switch

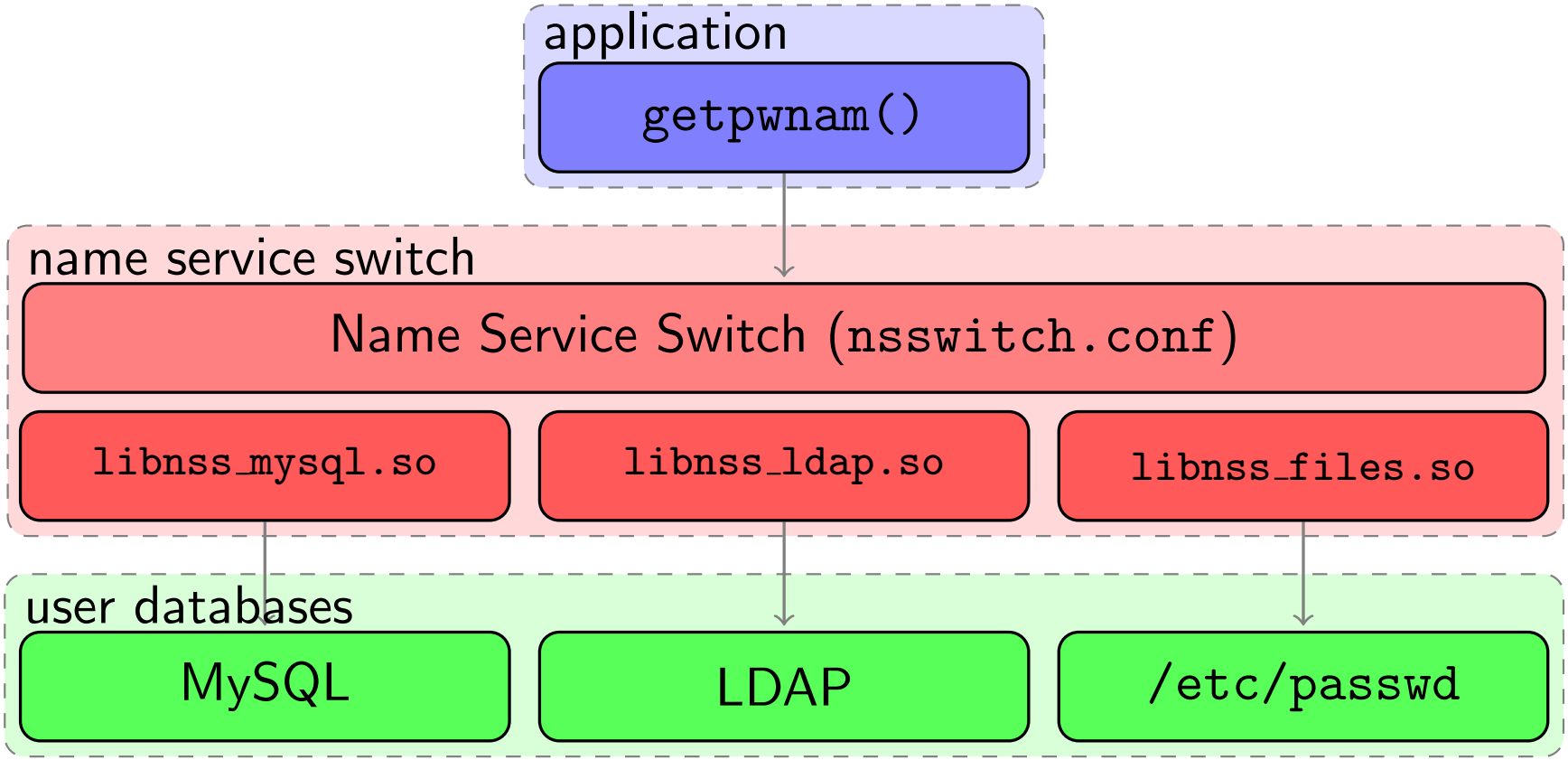
Benutzerdatenbank jenseits der Authentifizierung

- Wie zeigt z.B. `ls -l` Benutzernamen an?
- GNU C Library `pwd.h`
  $\Rightarrow$ libc muss Zugriff auf die Benutzerdatenbank haben

Name Service Switch [lib]

- Konfigurierbare Schnittstelle
- Leitet Datenbankabfragen an Module weiter
- Module sind Shared Libraries, z.B. `libnss_ldap.so`

Funktionsweise



Gliederung

Zusammenfassung

- **Ausgangspunkt**
  Mehrbenutzerumgebung
- **Authentifizierung mit PAM**
  Abstrahierung der Authentifizierung von „system-entry services“, Aufbau, Modul-Stacking, Schnittstelle zum Benutzer, libpamc
- **Name Service Switch**
  Konfigurierbare Datenbankschnittstelle der libc, Funktionsweise
- **Zurück zum Ausgangspunkt**
  Umsetzung von Mehrbenutzersystemen möglich

Quellen I


The Open Group.
X/open single sign-on service (xsso) – pluggable authentication modules.
X/open preliminary specification, The Open Group, Mar 1997.


Charlie Lai.
Making login services independent of authentication technologies.
Presentation published by Linux-PAM on kernel.org, Nov 1996.


The GNU C Library Manual.
Section 28 System Databases and Name Service Switch.

Quellen II


Andrew G. Morgan and Thorsten Kukuk.
The Linux-PAM Guides, 1.1.1 edition, Dec 2009.
Manuals for Linux-PAM.


Andrew G. Morgan.
Pluggable Authentication Modules (PAM), Dec 2001.
Draft for the Open-PAM working group PAM standard.


Vipin Samar.
Unified login with pluggable authentication modules (pam).
In *CCS '96: Proceedings of the 3rd ACM conference on Computer and communications security*, pages 1–10, New York, NY, USA, 1996. ACM.

Q&A

Danke für die Aufmerksamkeit

Questions & Answers