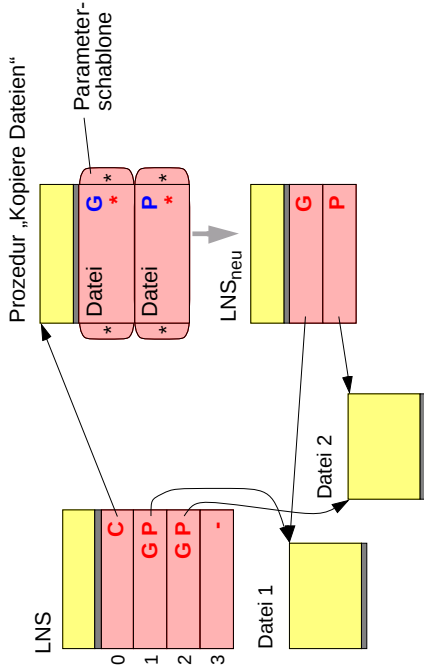


2 Hydra Prozeduraufruf (3)

- Übergabe von Parametern

◆ Beispiel: Prozedur zum Kopieren von Dateiinhalten



SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

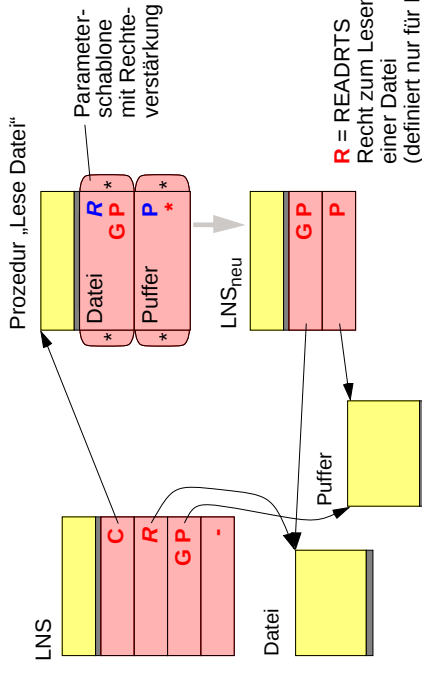
Reproduzieren oder Verwenden dieser Vorlesung ist ohne Genehmigung der Universität Erlangen-Nürnberg strafbar. Bitte die Zustimmung des Autors.

I.52

2 Hydra Prozeduraufruf (3)

- Verstärken von Rechten

◆ Beispiel: Prozedur zum Lesen von Dateiinhalten



SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

Reproduzieren oder Verwenden dieser Vorlesung ist ohne Genehmigung der Universität Erlangen-Nürnberg strafbar. Bitte die Zustimmung des Autors.

I.53

3 Problem: Gegenseitiges Mißtrauen

- Aufrufer mißtraut einer Prozedur
- ◆ Aufrufer möchte der Prozedur nur soviel Rechte einräumen wie nötig
- Aufgerufene Prozedur mißtraut dem Aufrufer
- ◆ Aufrufer soll nur soviel Rechte und Zugang bekommen wie erforderlich
- ★ Hydra Prozeduraufruf unterstützt diese Forderungen direkt
- ◆ Aufrufer übergibt Capabilities, die nötig sind
- ◆ Aufrufer kann Rechte bei der Übergabe maskieren und damit ausschalten
- ◆ Aufrufer erhält nur Zugang zu einem definierten Ergebnis
- ◆ Prozedur kann eigene Capabilities besitzen, die einem LNS zur Verfügung stehen und die dem Aufrufer verborgen bleiben können

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

Reproduzieren oder Verwenden dieser Vorlesung ist ohne Genehmigung der Universität Erlangen-Nürnberg strafbar. Bitte die Zustimmung des Autors.

I.54

3 Problem: Gegenseitiges Mißtrauen (2)

- ▲ Rechteverstärkung als Sicherheitslücke?

◆ Verstärkungsschablone wird nur an vertrauenswürdige Prozeduren ausgegeben und kann nicht einfach erzeugt werden

4 Problem: Modifikationen

- Aufrufer möchte Modifikationen an und über Parametern ausschließen
- ◆ eine Prozedur soll nichts verändern können
- Wegnehmen der entsprechenden Rechte langt nicht
- ◆ Prozedur kann lesend zu neuen Capabilities gelangen und über diese Änderungen vornehmen (Transitivität)
- ◆ Rechteverstärkung könnte angewandt werden

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

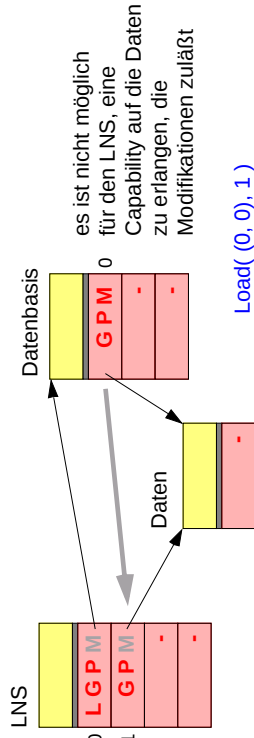
Reproduzieren oder Verwenden dieser Vorlesung ist ohne Genehmigung der Universität Erlangen-Nürnberg strafbar. Bitte die Zustimmung des Autors.

I.55

4 Problem: Modifikation (2)

★ Einführung des Modifikationsrechts **MDFYRTS**

- ◆ für alle modifizierenden Operationen an Datenbereichen und C-Lists muß zusätzlich das Modifikationsrecht vorhanden sein
- ◆ Modifikationsrecht wird automatisch gelöscht, wenn eine Capability über einen Pfad geladen wird, auf dem eine der Capabilities kein Modifikationsrecht besitzt
- ◆ Modifikationsrecht kann nicht über Rechteverstärkung erlangt werden



SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

L56

Reproduction is prohibited. All rights reserved. This document is the property of the University of Erlangen-Nürnberg. It is not to be distributed outside the University of Erlangen-Nürnberg.

4 Problem: Modifikation (3)

■ Parameterübergabe

- ◆ Wegnahme des Modifikationsrecht bei Parametern stellt sicher, daß die aufgerufene Prozedur keinerlei Veränderungen beim Aufrufer durchführen kann

5 Problem: Ausbreitung von Capabilities

■ Aufrufer will verhindern, daß eine übergebene Capability vom Aufgerufenen an einen Dritten weitergegeben wird (*Propagation Problem*)

- ◆ Beispiel: Prozedur „Drucken“ soll niemandem eine Referenz auf die zu druckenden Daten weitergeben können

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

L57

Reproduction is prohibited. All rights reserved. This document is the property of the University of Erlangen-Nürnberg. It is not to be distributed outside the University of Erlangen-Nürnberg.

5 Problem: Ausbreitung von Capabilities (2)

★ Einführung des Environment-Rechts **ENVRTS**

- ◆ für das Speichern oder Anfügen einer Capability an eine C-List muß die Capability selbst das Environment-Recht besitzen
- ◆ Environment-Recht wird automatisch gelöscht, wenn eine Capability über einen Pfad geladen wird, auf dem eine der Capabilities kein Environment-Recht besitzt
- ◆ Environment-Recht kann nicht über Rechteverstärkung erlangt werden

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

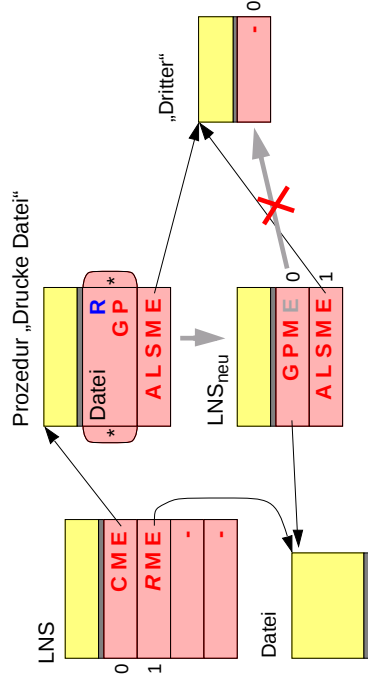
I-Security.doc: 1998-02-04 09:49

L58

Reproduction is prohibited. All rights reserved. This document is the property of the University of Erlangen-Nürnberg. It is not to be distributed outside the University of Erlangen-Nürnberg.

5 Problem: Ausbreitung von Capabilities (3)

■ Versuchte Weitergabe einer Capability an einen Dritten



Call(0, 2, 1, (-E))

Store(0, (1, 0))

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

L59

Reproduction is prohibited. All rights reserved. This document is the property of the University of Erlangen-Nürnberg. It is not to be distributed outside the University of Erlangen-Nürnberg.

6 Problem: Aufbewahrung von Capabilities

- Aufrufer möchte sicher sein, daß Aufruferener keine Capabilities nach der Bearbeitung des Aufrufs zurückbehalten kann (*Conservation Problem*)
- ★ Environment-Recht zusammen mit dem Aufrufmechanismus genügt
 - ◆ Aufruferener kann Capability ohne ENVRTS nicht weitergeben und folglich nicht abspeichern
 - ◆ der LNS des Aufrufs wird mit Beendigung des Aufrufs vernichtet, so daß die übergebenen Capabilities nicht zurückbehalten werden können
 - ◆ ENVRTS wirkt transitiv, so daß auch die über eine Parameter-Capability gewonnenen Capabilities nicht weitergegeben werden können

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

Reproduzieren oder die Weiterverbreitung dieser Unterlagen ist ohne schriftliche Genehmigung der Universität Erlangen-Nürnberg. Kopieren ist strafbar.

L60

7 Problem: Informationsflußbegrenzung

- Aufrufer möchte die Verbreitung von Informationen aus übergebenen Parametern einschränken (*Confinement Problem*)
 - ◆ selektiv: bestimmte Informationen sollen nicht nach außen gelangen
 - ◆ global: gar keine Informationen sollen nach außen gelangen
- ◆ ENVRTS ist nicht ausreichend, da Prozedur den Dateninhalt von Parameterobjekten kopieren könnte (ENVRTS wirkt nur auf die Weitergabe von Capabilities)
- Hydra realisiert nur globale Informationsflußbegrenzung
- ★ Modifikationsrecht auf der Prozedur-Capability
 - ◆ wenn kein Modifikationsrecht vorhanden ist, werden bei allen in den LNS übernommenen Capabilities die Modifikationsrechte ausgeschaltet (gilt jedoch nicht für Parameter)

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

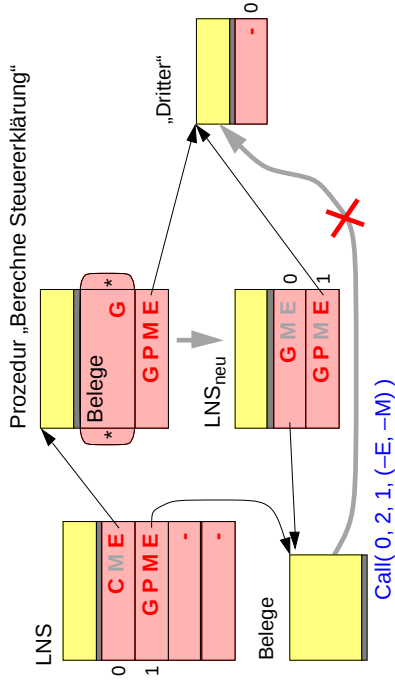
I-Security.doc: 1998-02-04 09:49

Reproduzieren oder die Weiterverbreitung dieser Unterlagen ist ohne schriftliche Genehmigung der Universität Erlangen-Nürnberg. Kopieren ist strafbar.

L61

7 Problem: Informationsflußbegrenzung (2)

- Beispiel: Prozedur zur Steuerberechnung
 - ◆ die übergebenen Beleg- und Buchhaltungsdaten sollen nicht weitergegeben werden können



SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

I-Security.doc: 1998-02-04 09:49

Reproduzieren oder die Weiterverbreitung dieser Unterlagen ist ohne schriftliche Genehmigung der Universität Erlangen-Nürnberg. Kopieren ist strafbar.

L62

8 Problem: Initialisierung

- Initialisierung von Objekten durch Prozeduren
 - ◆ Übergabe eines Objekts und verschiedener Capabilities, mit denen das Objekt initialisiert werden soll
- ◆ Problem: Capabilities müssen Environment-Recht besitzen (sonst ist das zu initialisierende Objekt nicht arbeitsfähig), gleichzeitig soll aber die Ausbreitung solcher Capabilities eingeschränkt werden
- Beispiel: Prozedur zur Initialisierung eines Katalogs bekommt Capabilities auf die entsprechenden Dateien
 - ◆ es soll sichergestellt werden, daß Prozedur die Datei-Capabilities nicht weitergibt
- ★ Environment-Recht auf der Prozedur-Capability
 - ◆ wenn kein Environment-Recht vorhanden ist, werden bei allen in den LNS übernommenen Capabilities die Environment-Rechte ausgeschaltet (gilt jedoch nicht für Parameter)

SP I

Systemprogrammierung I

© Franz J. Hauck, Universität Erlangen-Nürnberg, IMMD IV, 1998

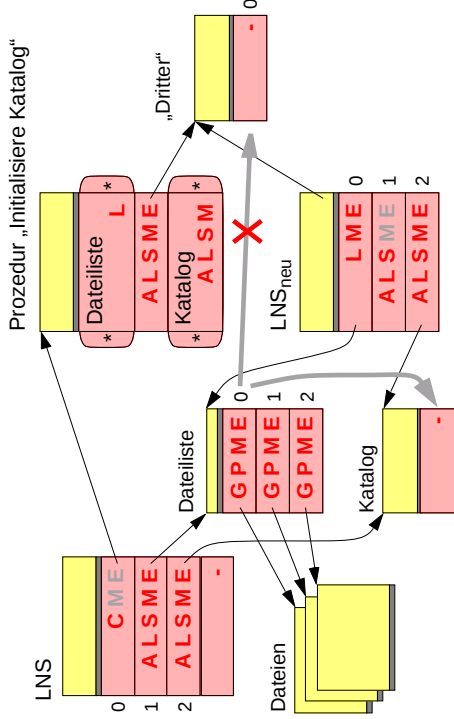
I-Security.doc: 1998-02-04 09:49

Reproduzieren oder die Weiterverbreitung dieser Unterlagen ist ohne schriftliche Genehmigung der Universität Erlangen-Nürnberg. Kopieren ist strafbar.

L63

8 Problem: Initialisierung (2)

- Beispiel: Initialisierung eines Katalogs



9 Rückruf von Capabilities

- Anwender möchte ausgegebene Capabilities für ungültig erklären

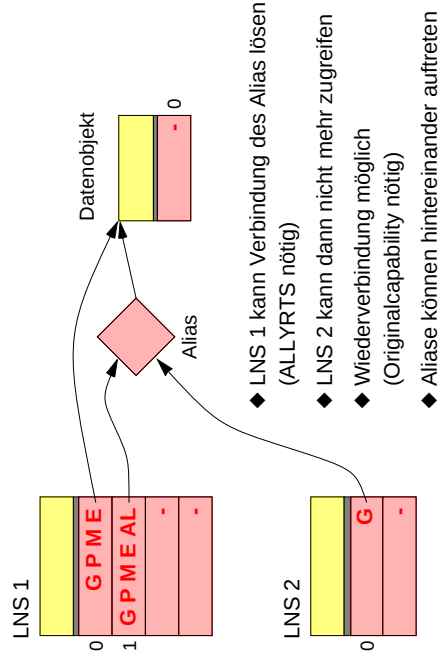
- ◆ sofortiger Rückruf — Rückruf nach einiger Zeit erst wirksam
- ◆ dauerhafter Rückruf — Rückruf nur zeitlich begrenzt wirksam
- ◆ selektiver Rückruf — Rückruf für alle Benutzer eines Objekts
- ◆ partieller Rückruf — Rückruf aller Rechte an einem Objekt
- ◆ Recht zum Rückruf; Rückruf des Rückrufsrechts

- ★ Hydra setzt sogenannte Aliase ein

- ◆ Alias ist eine Indirektionsstufe zu Capabilities
- ◆ Statt auf ein Objekt können Capabilities auf Aliase verweisen und diese wiederum auf andere Aliase oder schließlich auf das eigentliche Objekt
- ◆ Verbindung vom Alias zum Objekt kann gelöst werden: Fehler beim Zugriff
- ◆ Recht zum Lösen der Verbindung **ALLYRTS**

9 Rückruf von Capabilities (2)

- Beispiel: Weitergabe einer rückrufbaren Capability



9 Rückruf von Capabilities (3)

- Beispiel: Aliasketten

