Access Control

https://iuk.one/1033-1012

Clemens H. **Cap**
ORCID: 0000-0003-3958-6136

Department of Computer Science
University of **Rostock**
Rostock,Germany
clemens.cap@uni-rostock.de

Version 2

Overview

1. Basic Definitions
2. Discretionary Access Control
3. Flow Models
4. Mandatory Access Control
5. Role Based Access Control
6. Multi Level and Multi Lateral

1. Basic Definitions

We learn some basic vocabulary in access control.

1. Basic Definitions
2. Discretionary Access Control
3. Flow Models
4. Mandatory Access Control
5. Role Based Access Control
6. Multi Level and Multi Lateral

Subjects and Objects

Subject:

- An **active agent** which manipulates an object.
- Example: A human user.
- Example: A running program.

Object:

- A **passive agent** that can be manipulated by a subject.
- Example: A file (which can be *read*, *written*, *appended*, *deleted*)
- Example: A sensor (which can be *read*, *reset*, *calibrated*)
- Example: An actor (which can be *activated*)

Actions

Actions:

- The **variety of manipulations** a subject may exercise on an object.
- Example: A file may be read, written, appended, deleted and more.
- Example: An actor allows lowering or raising a sun screen.

Principals and Groups

Principal:

- An entity which has an **identity** and may be **authenticated**.
- Examples: A person.
- Example: A computer.
- Example: A process, thread or process group.
- Example: A USB stick or hard disc.

Group:

- An abstraction for **sets of entities** from a security point of view.
- Example: Account group wheel (subjects which may escalate privilege to root)
- Example: Account group sysadmin (subjects which may edit the system config)
- Example: System configuration (objects affecting system operation, /etc)
- Example: Action group modify (actions which amount to ability to change file)

2. Discretionary Access Control

The most simple form of access control.

1. Basic Definitions
2. Discretionary Access Control
3. Flow Models
4. Mandatory Access Control
5. Role Based Access Control
6. Multi Level and Multi Lateral

Definition

Definition: In Discretionary Access Control (DAC) the owner is in control.

- Every object `hasAn` owner.
- Access to objects is based on the identity of the subjects or the groups to which they belong.
- Access is granted or revoked by the owner of the object.
- If a program (i.e. a subject) is started by a user (i.e. another subject) it assumes the permissions of the user.

Evaluation:

- Widely implemented in most operating systems.
- Well known.
- Has numerous problems.

Problem 1: Lack of Global Policy

Situation:

- Owners decide on access.
- Owners can decide contrary to company rules (“global policy”).
- DAC cannot enforce consistency of de facto accesses with global policies.

Example:

- Researcher generates and owns files containing lab reports.
- Company cannot enforce access restrictions to new research results.

Problem 2: Malicious and Faulty Software

Situation:

- Owners decide on access.
- Programs assume the permissions of their users.
- Malicious or faulty programs can change access properties of objects on behalf but contrary to the intentions of their users.
- Thus: Confusion of programs as subjects with users as subjects.
- Similar in nature to confused deputy problem.

Problem 3: Confused Deputy Problem

Example:

1. May `root` edit `partitionTable`? yes
2. May `textEditor` edit `partitionTable`? it depends
3. May `textEditor` when run by root edit `partitionTable`? yes
4. May `textEditor` when run by nobody edit `partitionTable`? no
5. May `mineSweeper` when run by root edit `partitionTable` certainly no

Situation:

- Examples (1)-(4) suggest that it might be a good idea to transfer permissions from a subject of type `user` to a subject of type `program`.
- Example (5) shows that this idea can be very dangerous.
- The transfer of rights between users of different types is not a good idea.
- **Name:** A general may start a nuclear war only when acting as a deputy of the president but not when acting on his own.

Problem 4: Information Flow

Example:

- Information can be copied from an object `orig` to an object `copy`.
- Object `orig` is readable to `boss` and not readable to `generalPublic`.
- A program constructs an object `copy` and makes is writeable and readable to `boss` and `generalPublic`.
- As a result we see unwanted information flow to `generalPublic`.

Situation:

- There is no mechanism which prevents unwanted information flows.
- DAC cannot prevent information flow in unwanted directions.

Problem 5: Limited Understanding of the Owner

Situation:

- DAC means: Owner is in control.
- Owner has limited problem awareness.
- Owner has limited understanding of security requirements.

Solutions

- Invent information flow models.
  1. **Biba Model** for integrity.
  2. **Bell-La Padula Model** for confidentiality.
- Additional models in multilateral and compartmental security.
- Invent mandatory access control (MAC).
- Prevent confusion of subject types using roles.

3. Flow Models

Attempting to fix the shortcomings of discretionary access control.

However, flow models have their own shortcomings!

1. Basic Definitions
2. Discretionary Access Control
3. Flow Models
4. Mandatory Access Control
5. Role Based Access Control
6. Multi Level and Multi Lateral

Security Labels

Let S be a set whose elements we call **security labels**.

Let $\sqsubseteq \subseteq S \times S$ be a binary relation on S.

$\sqsubseteq$ is called an **order relation**, iff it satisfies the following 3 axioms:

- **Reflexive:** $\forall x \in S : x \sqsubseteq x$
- **Antisymmetric:** $\forall x, y \in S : x \sqsubseteq y \land y \sqsubseteq x \Rightarrow x = y$
- **Transitive:** $\forall x, y, z \in S : x \sqsubseteq y \land y \sqsubseteq z \Rightarrow x \sqsubseteq z$

An order relation $\sqsubseteq$ is called **linear**, iff any two elements are comparable, i.e. $\forall x, y \in S : x \sqsubseteq y \lor y \sqsubseteq x$

All items under control of a flow model are labeled by elements of a linearly ordered set.

Biba: Integrity

Target: Integrity.

- Data: Is internally consistent and correctly models the real world.
- Systems: Are consistent with their components to the specification.
- Label examples: $\{\texttt{HighlyTrusted}, \texttt{Trusted}, \texttt{Untrusted}\}$

Rationale:

- Items with higher labels may execute transactions of higher importance.
- Items with higher labels must not decide based on data of lower level clearance.

Rules: Prevent information flow from low to high to protect high components.

- **No-Read-Down:** Subject at higher label is not allowed to read data of lower level.
- **No-Write-Up:** Subject of lower label is not allowed to write data into higher level.

Example: Military Chain of Command

- Military commands are communicated from General to Private.
- Operational plan remains intact.

Bell-La Padula: Confidentiality

Target: Confidentiality

- Label examples: $\{\texttt{TopSecret}, \texttt{Secret}, \texttt{Public}\}$

Idea:

- A subject of a higher label is more trustworthy.
- An object of a higher label is more sensitive and dangerous upon disclosure.

Rules: Prevent information flow from high to low.

- **Read-down:** A subject may read only objects of the same or of a lower label.
- **Write-up:** A subject may write only objects of the same or a higher label.

Example: Spy Agency

- Military intelligence is collected by NSA director and not by the employee.

Task: Choice of Information Flow Model

Answer the following questions and give reasons for the decisions!

- Which information flow model is appropriate?
- Which labels should be used?
- What is the order relation should be used on the items?

Case 1: Security of student marks in the study office.

- Items: Students, dean, rector, profs.

Case 2: Security of an operating system with multiple rings:

- Items: Applications, drivers, system progs (update, backup, restore), kernel.

Case 3: Corona test result database.

- Items: Tested patient, clinic director, health politician, journalist.

Problem 1: Semantic Issue

Situation:

- Security models do not understand our intentions and apply rules mechanically.
- Security models apply only to specific goals.
- Distinction of flow directions (Bell-La Padula versus Biba) is not always easy.

Problem 2: Protocols are Bidirectional

Situation: Preventing information flows in particular directions poses a problem for communication.

Example: TCP handshake

- TCP handshake requires a bidirectional packet flow.
- Idea: Use UDP.
- But: Who does the acknowledgments now?
- Idea: Enforce flow rules only on higher ISO layers
- But: Lower ISO layers can exploit covert channels.

Problem 3: Covert Channels

Definition: Covert Channel

A covert channel is a channel allowing information flow in violation of the security policies which should be preventing this information flow.

Situation:

- Even well flow-isolated systems allow covert channels.
- Covert channels can be used to circumvent the flow restrictions.

Covert Channels (1)

Padding Channel: Use fields which usually are zero padding areas.

- Place a payload into these fields.
- Eg: A 2-bit field is placed inside of an 3-bit area (eg. IPv4 header)
- Eg: A packet must have a minimal length (eg. Ethernet)

Timing Channel: Use times of invoking activities,

- Encode information into seemingly random times.
- Make it seem random by using compression and encryption.
- Eg: Times when Ethernet packets are sent.
- Eg: Times when ARP cache entries expire.

Covert Channels (2)

Resource Modulation Channel:

- Exclusive resources have to be waited for since other processes might need use.
- Use exclusive resources and have components see when they are not available.
- Eg: Running out of memory.
- Eg: Keeping busy all available physical CPUs.

Noise Channel:

- Some behavior has inherent statistical noise.
- Use this noise to modulate a signal on top of this noise.
- Needs a error-correcting code.
- Attacker must have a better statistics of noise than the defender.
- Eg: Modify the lowest bits produced by an image sensor. Only the sensor will know that this was not due to sensor noise but due to covert channel activity.
- Eg: Some packets do not arrive due to collision or queue overflow. Use this to build a covert channel.

Covert Channels (3)

Equivalent Means Channel:

- There often are several equivalent ways to achieve the same means.
- Eg: Embed information into a binary by using instructions for $x = x + x$ or $x = 2 \cdot x$.
- Switch these different means to carry information.

Evaluation:

- Numerous possibilities to build low data rate covert channels.
- **Bad:** Cannot prevent covert channels completely
- **Good:** Can attempt to derive an upper limit for their data rate
- **Bad:** Small data rate sufficient to exfiltrate cryptographic keys.
- Compare this with steganographic methods

4. Mandatory Access Control

Fixes the core problem of discretionary access control.

1. Basic Definitions
2. Discretionary Access Control
3. Flow Models
4. Mandatory Access Control
5. Role Based Access Control
6. Multi Level and Multi Lateral

Definition

Definition: In Mandatory Access Control (MAC) a central authority is in control.

- A system wide policy regulates
  which subjects may access which objects by which actions.
- Subjects (as owners of objects) cannot override these policies.

Evaluation:

- Solves many problems of DAC at least to some degree.

Information Labeling

Situation:

- Every item of information is supplied with a label.
- Every container of information is supplied with a label.
- Copy-paste and drag-and-drop is checked for policy conformance.
- Can prevent accidental unwanted info flow by owners.

Evaluation:

- Is it really desirable to have such a strict control on info flow?
- Cannot protect (completely) against bad intent of malicious users.

5. Role Based Access Control

A conceptually and mathematically more advanced concept.

1. Basic Definitions
2. Discretionary Access Control
3. Flow Models
4. Mandatory Access Control
5. Role Based Access Control
6. Multi Level and Multi Lateral

Concepts

Role Based Access Control RBAC is based on three assumptions:

1. **Assignment:** A subject can exercise a privilege only if it has been assigned a role.
2. **Authorization:** A subject can assume a role only if the subject is authorized for this role.
3. **Permission:** A subject can exercise a privilege only if the assigned role of the subject permits exercising this privilege.

Evaluation:

- Reusable management of rights due to separation of the concepts of subjects and roles.
- Roles can model hierarchical structures of an organization.
- Can implement MAC and DAC mechanisms.

Formalization

3 Sets:

1. A set of **subjects** S.
2. A set of **roles** R.
3. A set of **privileges** or rights P.

3 Relations:

1. Relation Γ of **authorized roles**:
   $(s, r) \in \Gamma$ iff subject s is authorized for role r and may be assigned role r.
2. Relation Δ of **permitted privileges**:
   $(r, p) \in \Delta$ iff role r contains the permission to exercise right p.
3. Session relation Σ of **assigned roles**:
   $(s, r) \in \Sigma$ iff subject s is assigned the role r in the current session.

Extension: Separation of Duty

Idea: Specifying pairs of conflicting roles (cannot be assigned at the same time).

- **Static:** Subject must not be authorized for 2 roles at any time.
- **Dynamic:** Subject must not be assigned 2 roles in same transaction.

Example: Treasurer and Auditor

- Every member of an institution may be elected treasurer.
- Every member of an institution may be elected auditor.
- No member of an institution may act as treasurer and as auditor at the same time.

Example: Cashier

- Every cashier may `PrepareBills`.
- Every cashier may `CancelBills`.
- A single cashier may not prepare and cancel a bill on one and the same transaction.

Task: DAC and MAC as RBAC

Demonstrate how DAC and MAC can be implemented in RBAC.

More precisely:

- Provide a practical use case, each for DAC and MAC.
- Give names (Alice, Bob, ...) and practical requirements.
- Explain how each, DAC and MAC, treat the situation.
- Explain for both cases how RBAC treats the situation.
- Provide the formal ingredients for RBAC for both cases.

6. Multi Level and Multi Lateral

Why one set of labels is not enough.

1. Basic Definitions
2. Discretionary Access Control
3. Flow Models
4. Mandatory Access Control
5. Role Based Access Control
6. Multi Level and Multi Lateral

Multi Level Security

Recap:

- There is a (vertical) hierarchy of security levels.
- Higher levels are more sensitive.
- Example: top secret, secret, confidential, public.
- Diverse models regulate information flow between different levels.

Problem 1: Conflict of Interest

Example: Law Firm

- In a law firm, Alice and Bob are bosses and have top-level clearance to documents.
- Anna is client of Alice, Birgit is client of Bob.
- Both bosses can view all documents of the law firm.
- Now Anna sues Birgit...

Example: Insider Secrets

- Would we want the CTO of Apple to become CTO of Google?
- Ann-Kristin Achleitner is member of the board of “Deutsche Börse AG”
- Her husband Paul Achleitner is chairman of the board of “Deutsche Bank AG”
- How can we enforce rules against insider trading?

Example: Clean Room Design in Aviation IT

- Core components must be redundant and designed by independent teams.
- Members of one team must not meet members of other team to prevent bug spread.

Problem 2: Additional Protection Against Compromise

Example: Spy Operations.

- The US spies on Germany and on Russia.
- Should the director for spying on Germany
  know the names of the spies working in Russia?
- Is multilevel security sufficient if there is a chance that one level may be compromised?
- Is it a good idea to have a single level to have access to all data?
  Think of the Snowden incident and the backup use case.

Multilateral or Compartmental Security

Compartmental security introduces different compartments to an organization with multilevel security.

Multilateral security adds horizontal (lateral) structure to a (vertically structured) hierarchy.

Implementation:

- Adopt codewords in addition to (linearly ordered) labels.
- Every subject and object gets a security label plus a set of codewords.
- Subject s may access object o if
  1. **Label condition:** Label of s is higher or equal to the label of o **AND**
  2. **Codeword condition:** Set of codewords of s is superseteq to set of codewords of o

Lattice Based Access Models

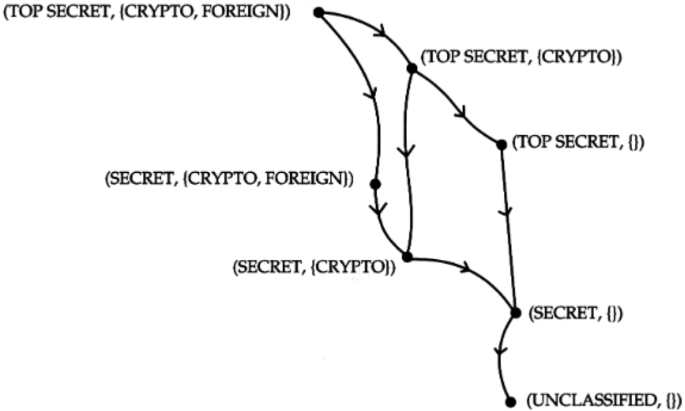

Fig. 1: Every finite distributive lattice can be represented as a lattice of subsets (Birkhoff Representation Theorem). A lattice built with labels and codewords can implement every desired access structure. © Rights see appendix.

Task: Lattice Based Access Models

Task 1: Provide a LBAC model for the US spies on Germany and Russia situation.

Task 2: Provide a LBAC model for the law firm.

- Law suits X and Y of Anna are against external parties.
- Law suits U, V, W of Birgit are against external parties.
- Law suit C is between Anna and Birgit.
- Provide sets of levels and codewords and give attributes to model the situation.

Appendix

Contents of Appendix

List of Figures

List of Rights

Fig. 1 Source: https://www.cl.cam.ac.uk/\protect \unhbox \voidb@x \protect \penalty \@M \ {}rja14/Papers/SE-08.pdf

Terms of Use (1)

Die hier angebotenen Inhalte unterliegen deutschem Urheberrecht. Inhalte Dritter werden unter Nennung der Rechtsgrundlage ihrer Nutzung und der geltenden Lizenzbestimmungen hier angeführt. Auf das Literaturverzeichnis wird verwiesen. Das Zitatrecht in dem für wissenschaftliche Werke üblichen Ausmaß wird beansprucht. Wenn Sie eine Urheberrechtsverletzung erkennen, so bitten wir um Hinweis an den auf der Titelseite genannten Autor und werden entsprechende Inhalte sofort entfernen oder fehlende Rechtsnennungen nachholen. Bei Produkt- und Firmennamen können Markenrechte Dritter bestehen. Verweise und Verlinkungen wurden zum Zeitpunkt des Setzens der Verweise überprüft; sie dienen der Information des Lesers. Der Autor macht sich die Inhalte, auch in der Form, wie sie zum Zeitpunkt des Setzens des Verweises vorlagen, nicht zu eigen und kann diese nicht laufend auf Veränderungen überprüfen.

Alle sonstigen, hier nicht angeführten Inhalte unterliegen dem Copyright des Autors, Prof. Dr. Clemens Cap, ©2020. Wenn Sie diese Inhalte nützlich finden, können Sie darauf verlinken oder sie zitieren. Jede weitere Verbreitung, Speicherung, Vervielfältigung oder sonstige Verwertung außerhalb der Grenzen des Urheberrechts bedarf der schriftlichen Zustimmung des Rechteinhabers. Dieses dient der Sicherung der Aktualität der Inhalte und soll dem Autor auch die Einhaltung urheberrechtlicher Einschränkungen wie beispielsweise Par 60a UrhG ermöglichen.

Die Bereitstellung der Inhalte erfolgt hier zur persönlichen Information des Lesers. Eine Haftung für mittelbare oder unmittelbare Schäden wird im maximal rechtlich zulässigen Ausmaß ausgeschlossen, mit Ausnahme von Vorsatz und grober Fahrlässigkeit. Eine Garantie für den Fortbestand dieses Informationsangebots wird nicht gegeben.

Die Anfertigung einer persönlichen Sicherungskopie für die private, nicht gewerbliche und nicht öffentliche Nutzung ist zulässig, sofern sie nicht von einer offensichtlich rechtswidrig hergestellten oder zugänglich gemachten Vorlage stammt.

Use of Logos and Trademark Symbols: The logos and trademark symbols used here are the property of their respective owners. The YouTube logo is used according to brand request 2-9753000030769 granted on November 30, 2020. The GitHub logo is property of GitHub Inc. and is used in accordance to the GitHub logo usage conditions `https://github.com/logos` to link to a GitHub account. The Tweedback logo is property of Tweedback GmbH and here is used in accordance to a cooperation contract.

Terms of Use (2)

Disclaimer: Die sich immer wieder ändernde Rechtslage für digitale Urheberrechte erzeugt für mich ein nicht unerhebliches Risiko bei der Einbindung von Materialien, deren Status ich nicht oder nur mit unverhältnismäßig hohem Aufwand abklären kann. Ebenso kann ich den Rechteinhabern nicht auf sinnvolle oder einfache Weise ein Honorar zukommen lassen, obwohl ich – und in letzter Konsequenz Sie als Leser – ihre Leistungen nutzen.

Daher binde ich gelegentlich Inhalte nur als Link und nicht durch Framing ein. Lt EuGH Urteil 13.02.2014, C-466/12 ist das unbedenklich, da die benutzten Links ohne Umgehung technischer Sperren auf im Internet frei verfügbare Inhalte verweisen.

Wenn Sie diese Rechtslage stört, dann setzen Sie sich für eine Modernisierung des völlig veralteten Vergütungssystems für urheberrechtliche Leistungen ein. Bis dahin klicken Sie bitte auf die angegebenen Links und denken Sie darüber nach, warum wir keine für das digitale Zeitalter sinnvoll angepaßte Vergütungssysteme digital erbrachter Leistungen haben.

Zu Risiken und Nebenwirkungen fragen Sie Ihren Rechtsanwalt oder Gesetzgeber.

Weitere Hinweise finden Sie im Netz hier und hier oder hier.

Citing This Document

If you use contents from this document or want to cite it,
please do so in the following manner:

Clemens H. Cap: Access Control. Electronic document. https://iuk.one/1033-1012 4. 7. 2021.

Bibtex Information: https://iuk.one/1033-1012.bib

```
@misc{doc:1033-1012,
   author      = {Clemens H. Cap},
   title       = {Access Control},
   year        = {2021},
   month       = {7},
   howpublished = {Electronic document},
   url         = {https://iuk.one/1033-1012}
}
```

Typographic Information:
Typeset on July 4, 2021
This is pdfTeX, Version 3.14159265-2.6-1.40.21 (TeX Live 2020) kpathsea version 6.3.2
This is pgf in version 3.1.5b
This is preamble-slides.tex myFormat©C.H.Cap

List of Slides

Legend:
⧉ continuation slide
○ slide without title header
🖼 image slide