A Framework for Linear Authorization Logics

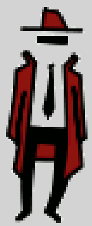
Vivek Nigam
Universidade Federal da Paraíba

Based on a LICS'12 paper. An extended version is available on my homepage.

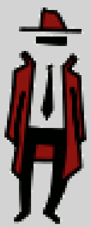
Proof-Carrying Authorization (PCA) [Appel and Felten CCS'99]

Proof-Carrying Authorization (PCA) [Appel and Felten CCS'99]

Alice

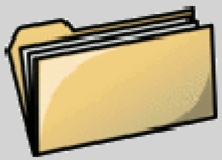


Bob

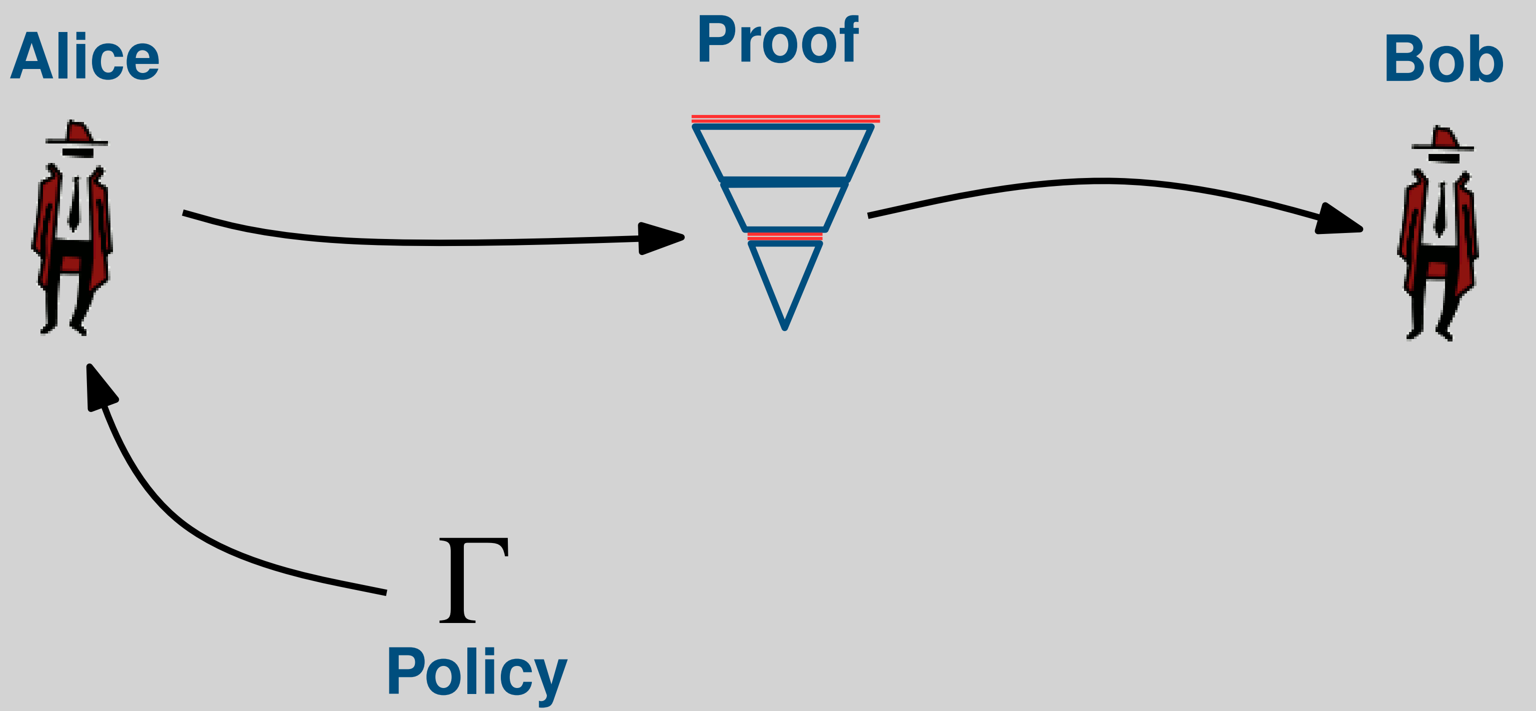


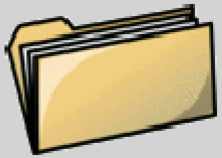
Γ

Policy

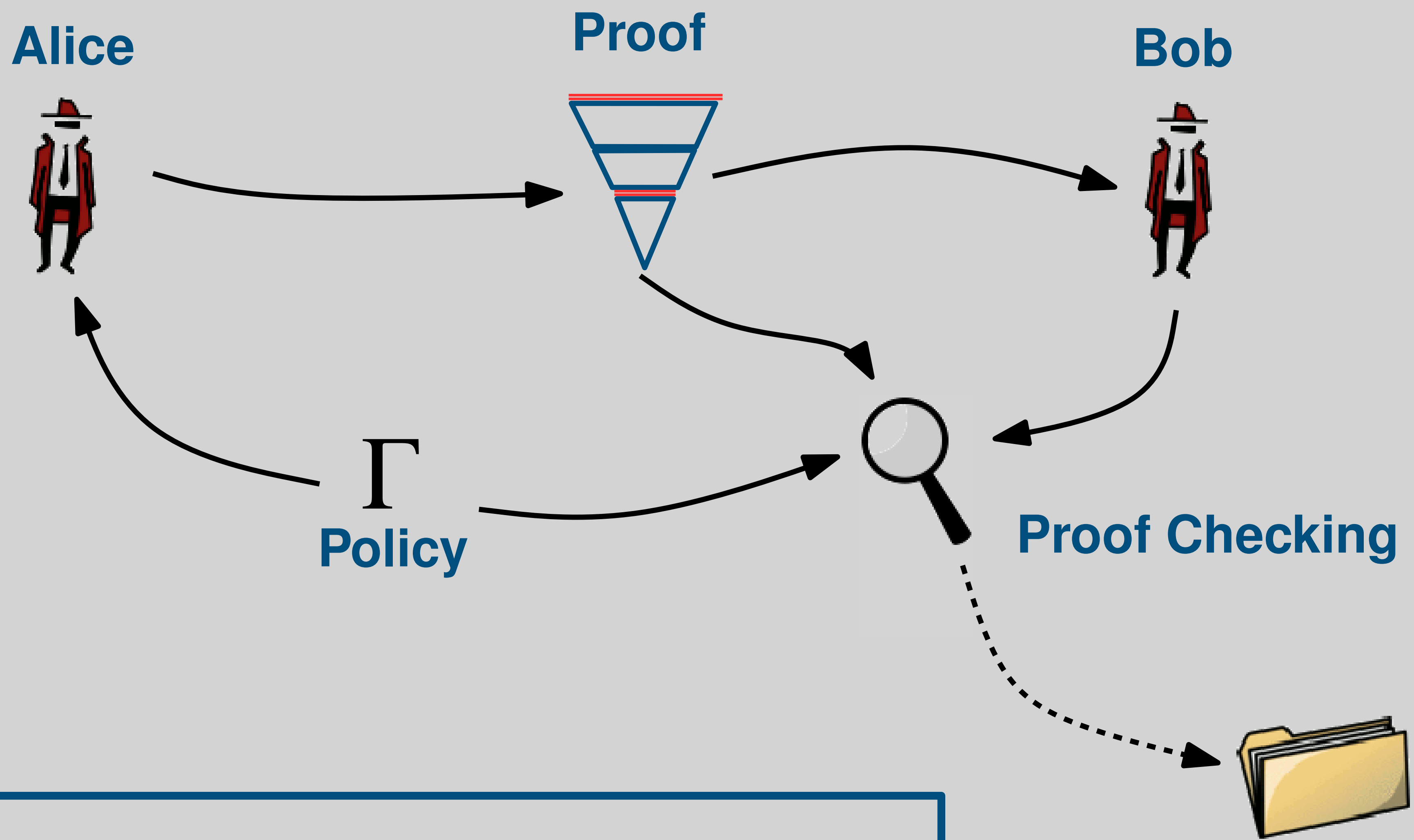


Proof-Carrying Authorization (PCA) [Appel and Felten CCS'99]





Proof-Carrying Authorization (PCA) [Appel and Felten CCS'99]



At the center of PCA lie the policies and the use of formal proofs.

Proof-Carrying Authorization (PCA)

Γ

Policy

Authorization Logics

Proof-Carrying Authorization (PCA)

Γ

Policy

Authorization Logics

Access control logics for distributed systems [Abadi et al. '93].

Modal Logics:

$$P \supset K \text{ says } P$$

$$K \text{ says } (P_1 \supset P_2) \supset K \text{ says } P_1 \supset K \text{ says } P_2$$

$$K \text{ says } (K \text{ says } P) \supset K \text{ says } P$$

Proof-Carrying Authorization (PCA)

Γ

Policy

Authorization Logics

In many situations, we would like to express effect-based policies

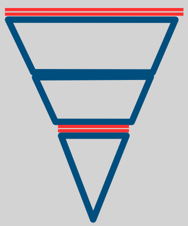
"A principal may have access to a room at most once."

"A principal may not withdraw more money than the money available in her bank account."

Linear Authorization Logics [Garg et al. ESORICS'06]

Our main contributions

Proof



Authorization Logics

We propose a logical framework where different linear authorization logics may live together. We show that in this framework one can express a wider range of policies.

"A principal may use a set of (low-ranked) policy rules, but not a set of (high-ranked) policy rules."

Our main results

Complexity Results

Provability Problem for LAL

Our main results

Complexity Results

Provability Problem for LAL	
MELL	Undecidable

Notice that for MELL the same problem is still open.

Our main results

Complexity Results

Provability Problem for LAL	
MELL	Undecidable
FOL Balanced Bipoles	PSPACE-complete

Notice that for MELL the same problem is still open.

Propositional Classical auth. logics is also PSPACE-complete

Agenda

- **Linear Authorization Logic**
- Undecidability
- Proof search and MSR
- PSPACE-completeness
- Conclusions and Future Work

Linear Logic Basics

Linear Logic Basics

Multiplicative Fragment

$$\frac{\Gamma, F, G \longrightarrow H}{\Gamma, F \otimes G \longrightarrow H} \otimes_L \qquad \frac{\Gamma_1 \longrightarrow F \quad \Gamma_2 \longrightarrow G}{\Gamma_1, \Gamma_2 \longrightarrow F \otimes G} \otimes_R$$

$$\frac{\Gamma_1 \longrightarrow F \quad \Gamma_2, G \longrightarrow H}{\Gamma_1, \Gamma_2, F \multimap G \longrightarrow H} \multimap_L \qquad \frac{\Gamma, F \longrightarrow G}{\Gamma \longrightarrow F \multimap G} \multimap_R$$

Linear Authorization Logics[Garg et al.]

Linear Authorization Logics[Garg et al.]

Three Families of Modalities

K says P K knows P K has P

Linear Authorization Logics[Garg et al.]

Three Families of Modalities

K says P

Linear Authorization Logics[Garg et al.]

Three Families of Modalities

K says P

A lax modality denoting that the principal K affirms the formula P:

$$\frac{\Gamma, P \longrightarrow K \text{ says } G}{\Gamma, K \text{ says } P \longrightarrow K \text{ says } G}\ \text{says}_L \qquad \frac{\Gamma \longrightarrow P}{\Gamma \longrightarrow K \text{ says } P}\ \text{says}_R$$

Linear Authorization Logics[Garg et al.]

Three Families of Modalities

K knows P

Linear Authorization Logics [Garg et al.]

Three Families of Modalities

$$K \text{ knows } P$$

Since knowledge is unrestricted, one is allowed to contract as well as weaken it:

$$\frac{\Gamma \longrightarrow G}{\Gamma, K \text{ knows } P \longrightarrow G}\ W \qquad \frac{\Gamma, K \text{ knows } P, K \text{ knows } P \longrightarrow G}{\Gamma, K \text{ knows } P \longrightarrow G}\ C$$

Linear Authorization Logics [Garg et al.]

Three Families of Modalities

$$K \text{ knows } P$$

$$\frac{\Gamma, P \longrightarrow G}{\Gamma, K \text{ knows } P \longrightarrow G}\ \text{knows}_L \qquad \frac{\Psi \longrightarrow P}{\Psi \longrightarrow K \text{ knows } P}\ \text{knows}_R$$

where Ψ contains only formulas of the form $K \text{ knows } F$.

Linear Authorization Logics [Garg et al.]

Three Families of Modalities

K has P

A restricted modality denoting that the principal K has the consumable resource P:

$$\frac{\Gamma, P \longrightarrow G}{\Gamma, K \text{ has } P \longrightarrow G}\ \text{has}_L \qquad \frac{\Psi, \Delta \longrightarrow P}{\Psi, \Delta \longrightarrow K \text{ has } P}\ \text{has}_R$$

where Ψ contains only formulas of the form K knows F, while Δ contains only formulas of the form K has F.

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{\mathsf{b}}, !^{\mathsf{r}}$ and $?^{\mathsf{b}}, ?^{\mathsf{r}}$:

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{\mathsf{b}}, !^{\mathsf{r}}$ and $?^{\mathsf{b}}, ?^{\mathsf{r}}$:

$$!^{\mathsf{b}} F \not\equiv !^{\mathsf{r}} F \qquad\qquad ?^{\mathsf{b}} F \not\equiv ?^{\mathsf{r}} F$$

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{\mathsf{b}}, !^{\mathsf{r}}$ and $?^{\mathsf{b}}, ?^{\mathsf{r}}$:

$$!^{\mathsf{b}}F \not\equiv\, !^{\mathsf{r}}F \qquad ?^{\mathsf{b}}F \not\equiv\, ?^{\mathsf{r}}F$$

All other connectives are canonical.

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{\mathsf{b}}, !^{\mathsf{r}}$ and $?^{\mathsf{b}}, ?^{\mathsf{r}}$:

Subexponentials

$!^{\mathsf{b}} F \not\equiv !^{\mathsf{r}} F \qquad ?^{\mathsf{b}} F \not\equiv ?^{\mathsf{r}} F$

All other connectives are canonical.

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{\mathsf{b}}, !^{\mathsf{r}}$ and $?^{\mathsf{b}}, ?^{\mathsf{r}}$:

Subexponentials

$!^{\mathsf{b}} F \not\equiv\, !^{\mathsf{r}} F \qquad ?^{\mathsf{b}} F \not\equiv\, ?^{\mathsf{r}} F$

All other connectives are canonical.

Subexponential Signature

$$\langle I, \preceq, U \rangle$$

where $U \subseteq I$ and is closed under $\preceq$.

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{\mathsf{b}}, !^{\mathsf{r}}$ and $?^{\mathsf{b}}, ?^{\mathsf{r}}$:

Subexponentials

$!^{\mathsf{b}} F \not\equiv !^{\mathsf{r}} F \qquad ?^{\mathsf{b}} F \not\equiv ?^{\mathsf{r}} F$

All other connectives are canonical.

Subexponential Signature

$$\langle I, \preceq, U \rangle$$

where $U \subseteq I$ and is closed under $\preceq$.

Subexponentials with index $\mathsf{a} \in U$ can weaken and contract:

$$\frac{\Gamma, !^{\mathsf{a}} P, !^{\mathsf{a}} P \longrightarrow G}{\Gamma, !^{\mathsf{a}} P \longrightarrow G}\ C \qquad \frac{\Gamma \longrightarrow G}{\Gamma, !^{\mathsf{a}} P \longrightarrow G}\ W$$

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{b}, !^{r}$ and $?^{b}, ?^{r}$: **Subexponentials**

$!^{b} F \not\equiv !^{r} F \qquad ?^{b} F \not\equiv ?^{r} F$

All other connectives are canonical.

Subexponential Signature

$$\langle I, \preceq, U \rangle$$

where $U \subseteq I$ and is closed under $\preceq$.

Subexponentials with index $a \in U$ can weaken and contract:

$$\frac{\Gamma, !^{a} P, !^{a} P \longrightarrow G}{\Gamma, !^{a} P \longrightarrow G}\ C \qquad \frac{\Gamma \longrightarrow G}{\Gamma, !^{a} P \longrightarrow G}\ W$$

Introduction Rules

$$\frac{!^{x_1} F_1, \ldots !^{x_n} F_n \longrightarrow G}{!^{x_1} F_1, \ldots !^{x_n} F_n \longrightarrow !^{a} G}\ !^{a}{}_{R}$$

$$\frac{!^{x_1} F_1, \ldots !^{x_n} F_n, F \longrightarrow ?^{x_{n+1}} G}{!^{x_1} F_1, \ldots !^{x_n} F_n, ?^{a} F \longrightarrow ?^{x_{n+1}} G}\ ?^{a}{}_{L}$$

where $a \preceq x_i$ for all i.

Linear Logic with Subexponentials [NM'09, DJS'93]

Linear Logic Exponentials are Not Canonical

$!^{\mathsf{b}}, !^{\mathsf{r}}$ and $?^{\mathsf{b}}, ?^{\mathsf{r}}$:

Subexponentials

$$!^{\mathsf{b}} F \not\equiv !^{\mathsf{r}} F \qquad ?^{\mathsf{b}} F \not\equiv ?^{\mathsf{r}} F$$

All other connectives are canonical.

Subexponential Signature

$$\langle I, \preceq, U \rangle$$

where $U \subseteq I$ and is closed under $\preceq$.

Subexponentials with index $\mathsf{a} \in U$ can weaken and contract:

$$\frac{\Gamma, !^{\mathsf{a}} P, !^{\mathsf{a}} P \longrightarrow G}{\Gamma, !^{\mathsf{a}} P \longrightarrow G}\ C \qquad \frac{\Gamma \longrightarrow G}{\Gamma, !^{\mathsf{a}} P \longrightarrow G}\ W$$

Introduction Rules

$$\frac{!^{\mathsf{x}_1} F_1, \ldots !^{\mathsf{x}_n} F_n \longrightarrow G}{!^{\mathsf{x}_1} F_1, \ldots !^{\mathsf{x}_n} F_n \longrightarrow !^{\mathsf{a}} G}\ !^{\mathsf{a}}{}_R$$

$$\frac{!^{\mathsf{x}_1} F_1, \ldots !^{\mathsf{x}_n} F_n, F \longrightarrow ?^{\mathsf{x}_{n+1}} G}{!^{\mathsf{x}_1} F_1, \ldots !^{\mathsf{x}_n} F_n, ?^{\mathsf{a}} F \longrightarrow ?^{\mathsf{x}_{n+1}} G}\ ?^{\mathsf{a}}{}_L$$

where $\mathsf{a} \preceq \mathsf{x}_i$ for all i.

Theorem: For any subexponential signature, Σ, SELL_Σ admits cut-elimination.

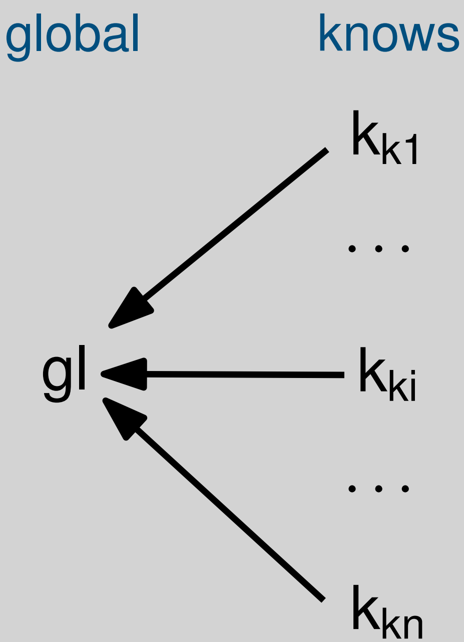
Encoding Linear Authorization Logics

Encoding Linear Authorization Logics

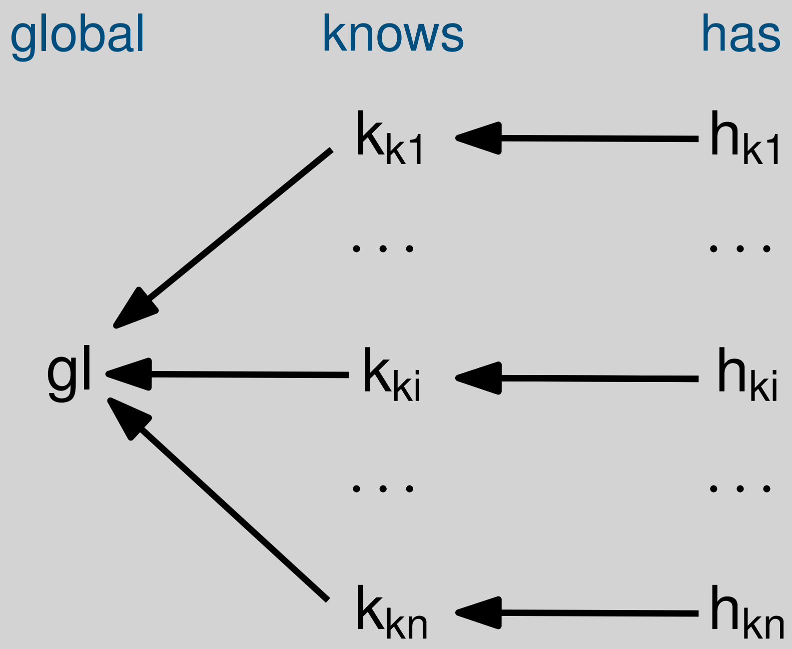
global

gl

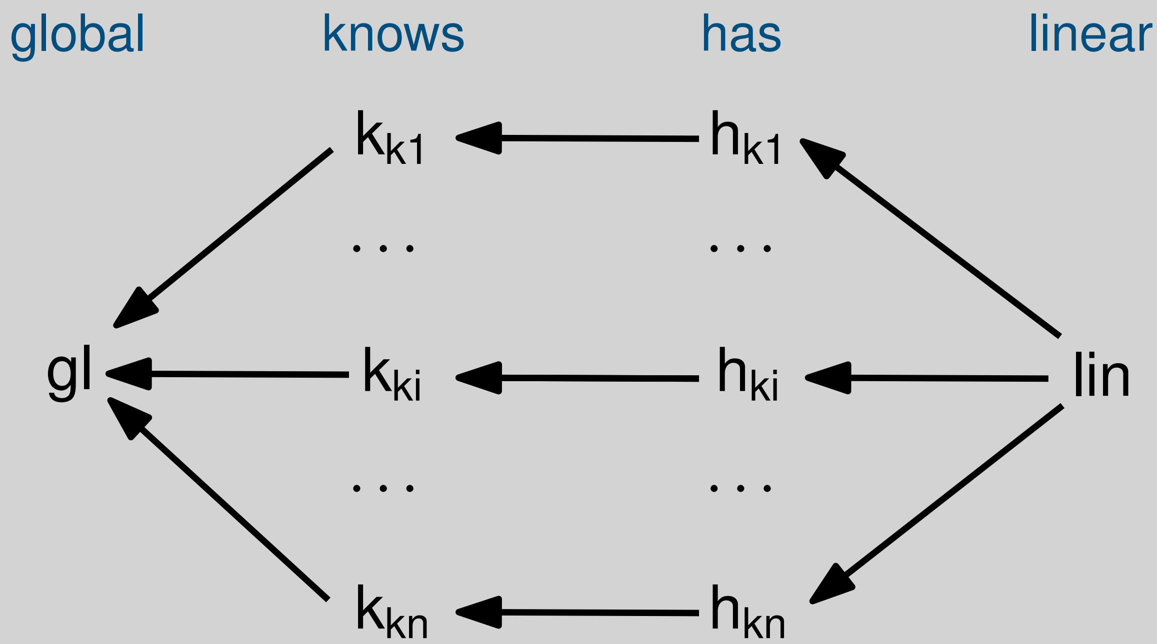
Encoding Linear Authorization Logics



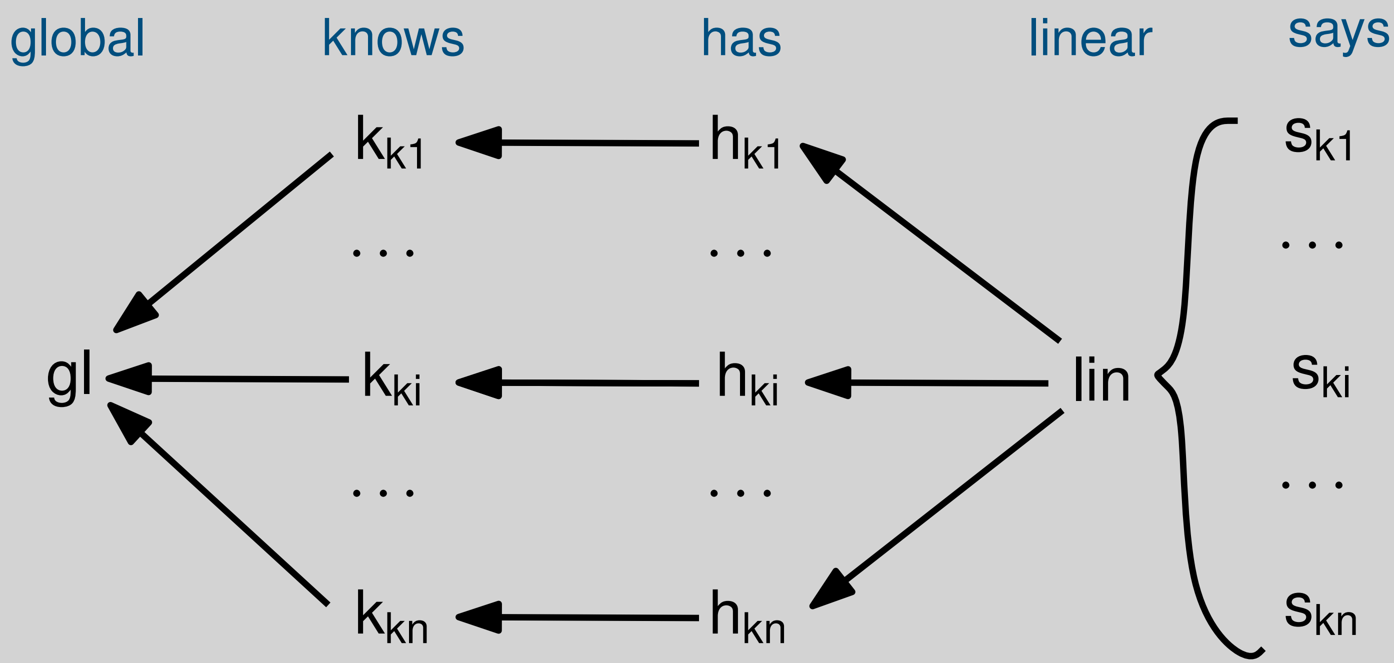
Encoding Linear Authorization Logics



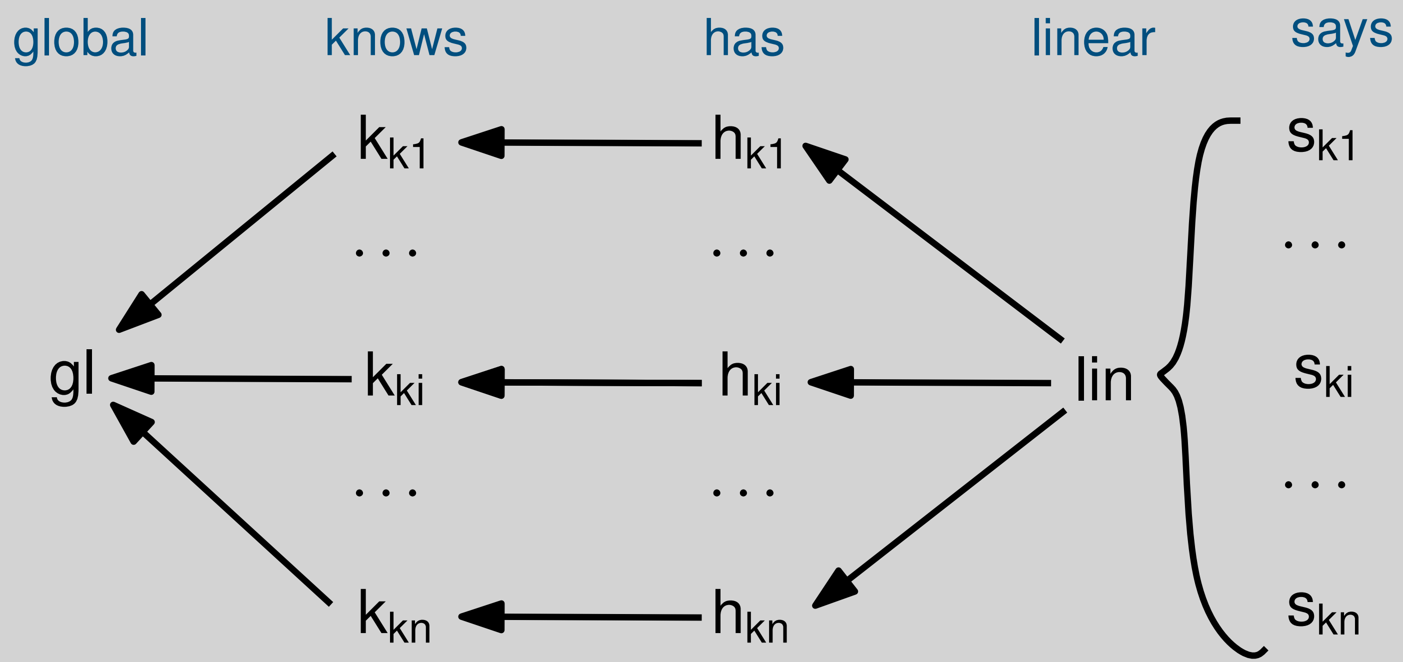
Encoding Linear Authorization Logics



Encoding Linear Authorization Logics



Encoding Linear Authorization Logics

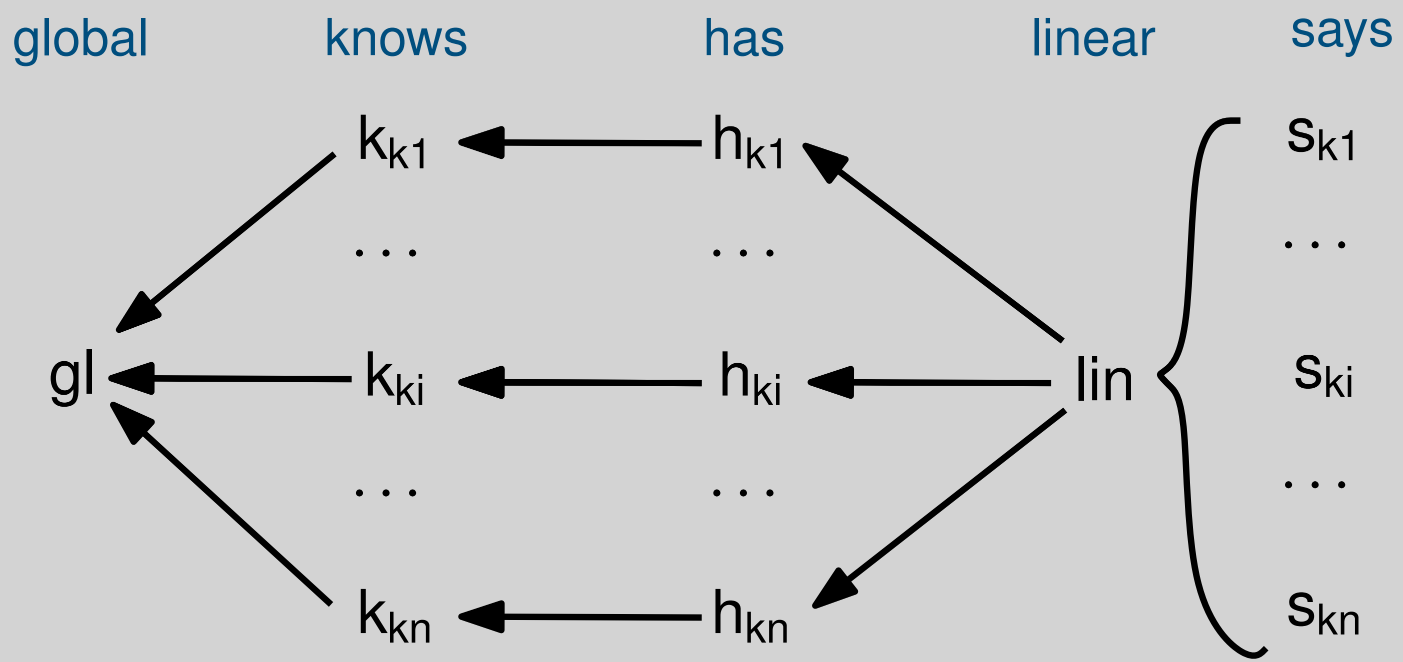


$$[\![F \text{ knows } K]\!]_L = !^{\mathsf{k}_K}[\![F]\!]_L \qquad [\![F \text{ knows } K]\!]_R = !^{\mathsf{k}_K}[\![F]\!]_R$$

$$[\![F \text{ has } K]\!]_L = !^{\mathsf{h}_K}[\![F]\!]_L \qquad [\![F \text{ has } K]\!]_R = !^{\mathsf{h}_K}[\![F]\!]_R$$

$$\frac{!^{\mathsf{gl}}\{\Theta\}, !^{\mathsf{k}_K}\{\Gamma\} \longrightarrow F}{!^{\mathsf{gl}}\{\Theta\}, !^{\mathsf{k}_K}\{\Gamma\} \longrightarrow !^{\mathsf{k}_K}F} \qquad \frac{!^{\mathsf{gl}}\{\Theta\}, !^{\mathsf{k}_K}\{\Gamma\}, !^{\mathsf{h}_K}\{\Delta\} \longrightarrow F}{!^{\mathsf{gl}}\{\Theta\}, !^{\mathsf{k}_K}\{\Gamma\}, !^{\mathsf{h}_K}\{\Delta\} \longrightarrow !^{\mathsf{h}_K}F}$$

Encoding Linear Authorization Logics

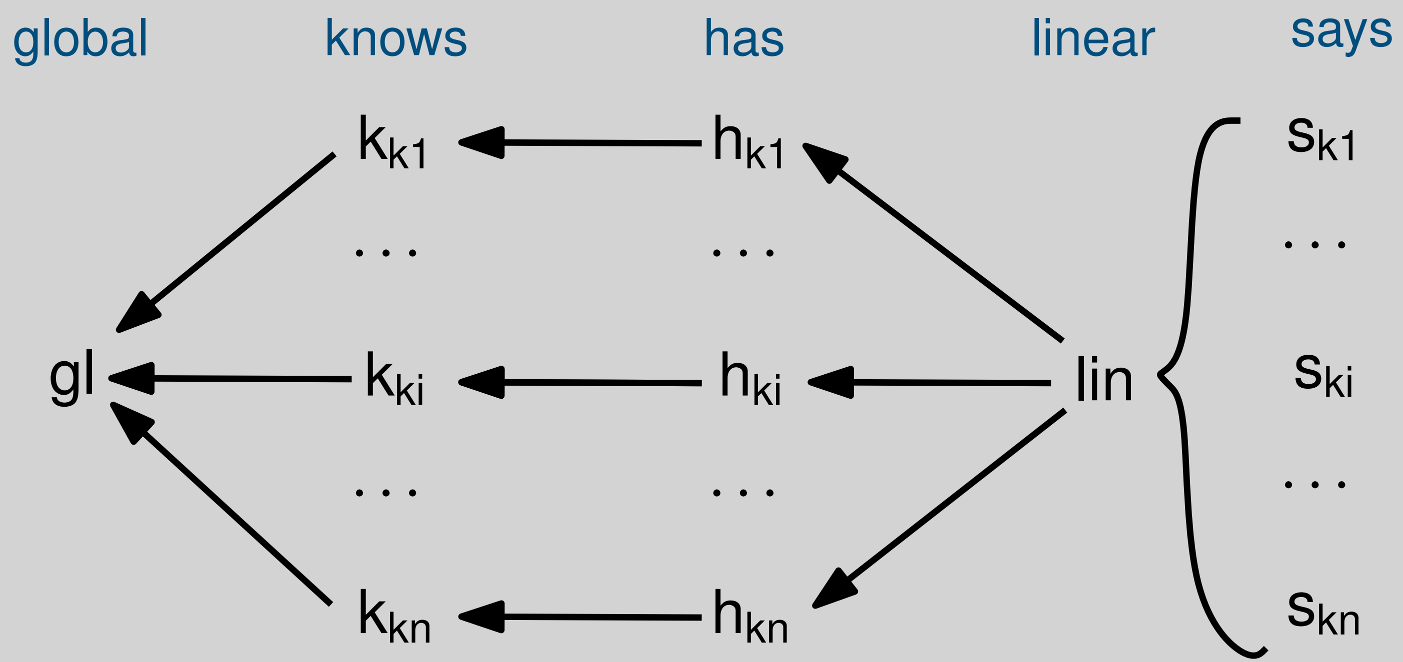


$$[\![F \text{ says } K]\!]_L = !^{\text{lin}} ?^{\text{s}_\text{k}} [\![F]\!]_L$$

$$[\![F \text{ says } K]\!]_R = ?^{\text{s}_\text{k}} [\![F]\!]_R$$

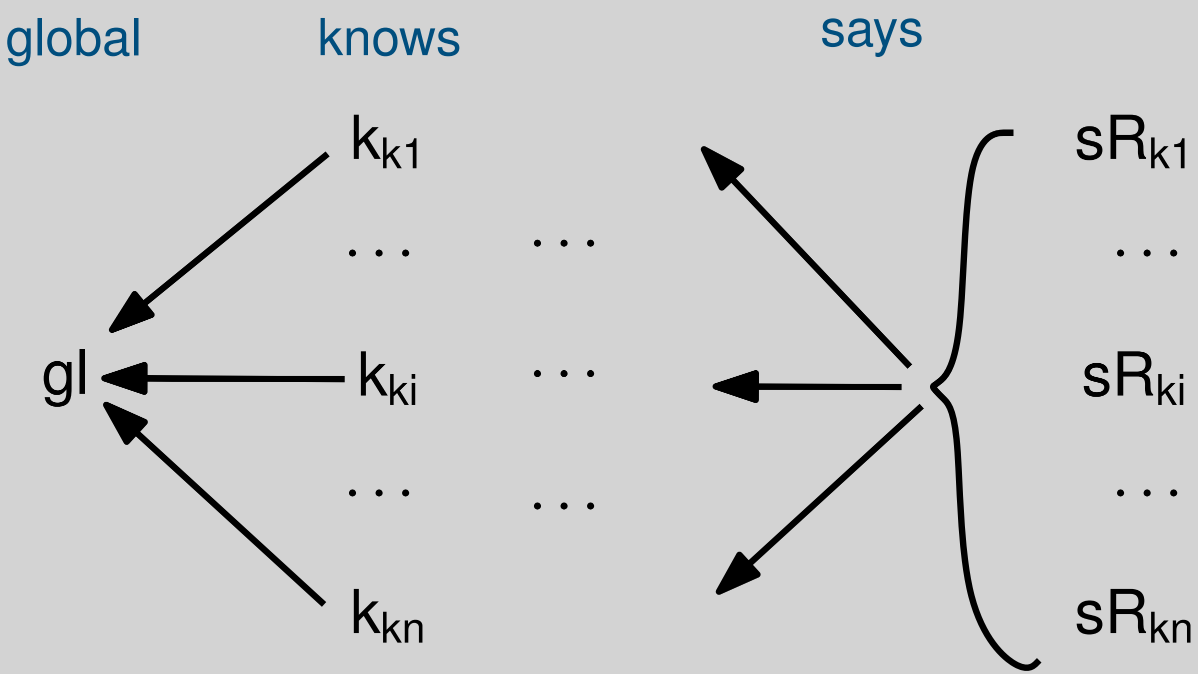
$$\frac{\Gamma, P \longrightarrow K \text{ says } G}{\Gamma, K \text{ says } P \longrightarrow K \text{ says } G} \qquad \frac{[\![\Gamma]\!]_L, [\![P]\!]_L \longrightarrow ?^{\text{s}_\text{k}} [\![G]\!]_R}{[\![\Gamma]\!]_L, !^{\text{lin}} ?^{\text{s}_\text{k}} [\![P]\!]_L \longrightarrow ?^{\text{s}_\text{k}} [\![G]\!]_R}$$

Encoding Linear Authorization Logics

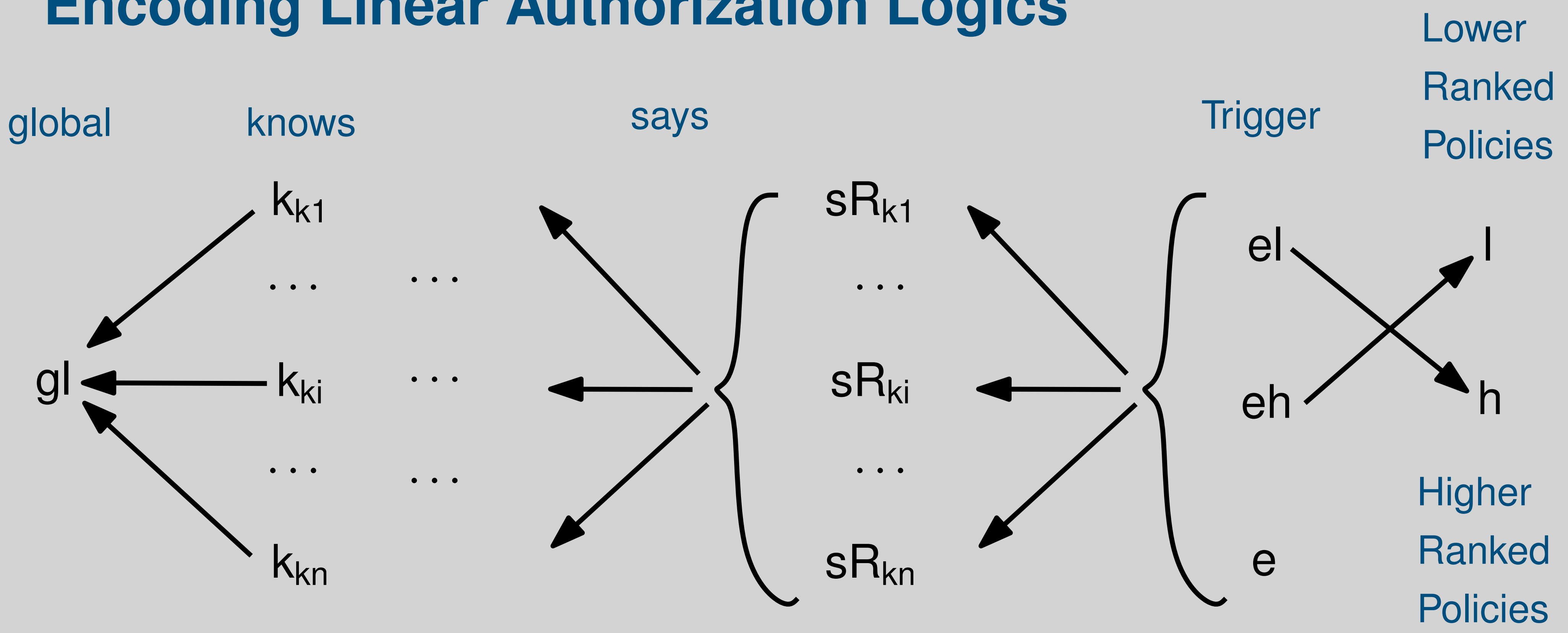


Theorem: The sequent $\Gamma \longrightarrow F$ is provable in linear authorization logic if and only if the sequent $[\![\Gamma]\!]_L \longrightarrow [\![F]\!]_R$ is provable in SELL.

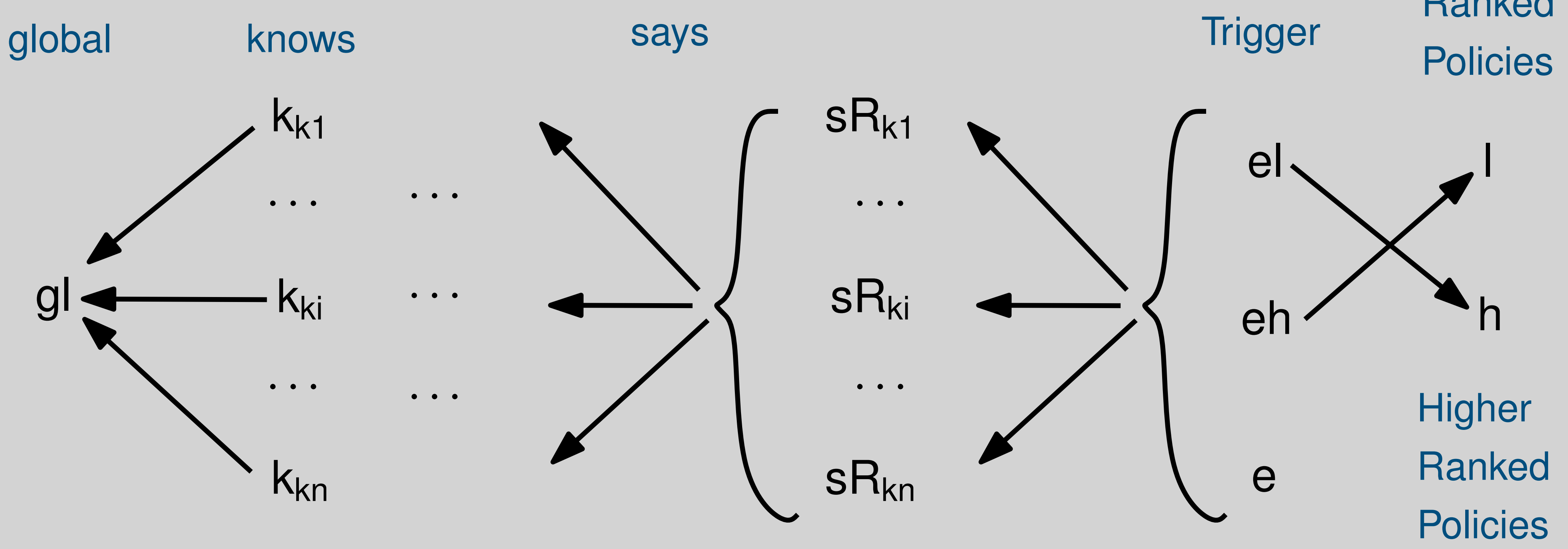
Encoding Linear Authorization Logics



Encoding Linear Authorization Logics

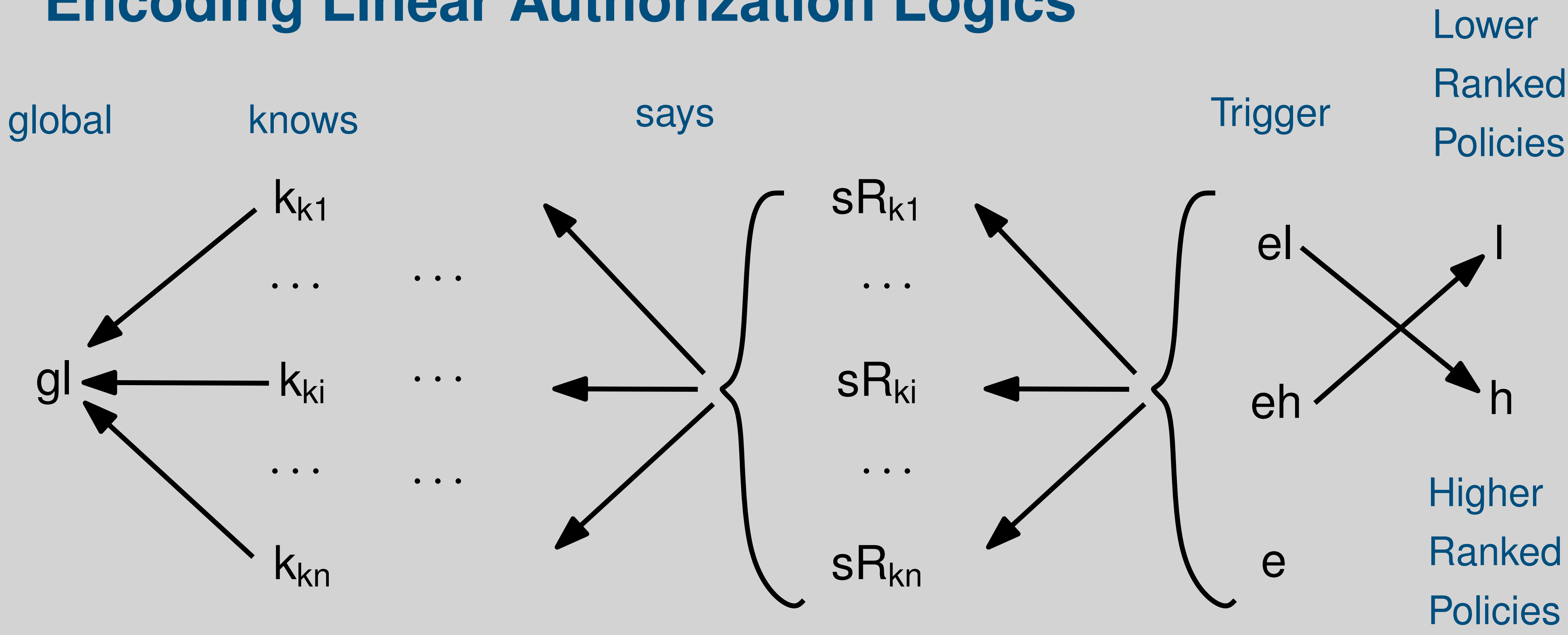


Encoding Linear Authorization Logics



$$\dfrac{\dfrac{\Gamma \longrightarrow F}{\Gamma \longrightarrow !^{\mathsf{el}} F}\ !^{\mathsf{el}}{}_R}{\Gamma, !^{\mathsf{l}}\{\Gamma_L\} \longrightarrow !^{\mathsf{el}} F}\ n \times W$$

Encoding Linear Authorization Logics



$$\text{admin knows (superuser}(K_1)) \otimes K_1 \text{ says } (K_2 \text{ has } P) \multimap K_2 \text{ has } P$$

$$\text{admin knows (user}(K_1)) \otimes\, !^{eh} K_1 \text{ says } (K_2 \text{ has } P) \multimap K_2 \text{ has } P$$

Agenda

- Linear Authorization Logic
- **Undecidability**
- Proof search and MSR
- PSPACE-completeness
- Conclusions and Future Work

Two counter machine

Two counter machine

Instructions (uniquely labelled)

(Add r_1) a_k: $r_1 = r_1 + 1$; goto b_j
(Add r_2) b_k: $r_2 = r_2 + 1$; goto a_j
(Sub r_1) a_k: $r_1 = r_1 - 1$; goto b_j
(Sub r_2) b_k: $r_2 = r_2 - 1$; goto a_j

(0-test r_1) a_k: if $r_1 = 0$ then goto b_{j_1} else goto b_{j_2}
(0-test r_2) b_k: if $r_2 = 0$ then goto a_{j_1} else goto a_{j_2}
(Jump$_1$) a_k: goto b_j
(Jump$_1$) b_k: goto a_j

Undecidability of Multiplicative Linear Authorization Logic

Two counter machine

Instructions (uniquely labelled)

(Add r_1)	a_k:	$r_1 = r_1 + 1$; goto b_j
(Add r_2)	b_k:	$r_2 = r_2 + 1$; goto a_j
(Sub r_1)	a_k:	$r_1 = r_1 - 1$; goto b_j
(Sub r_2)	b_k:	$r_2 = r_2 - 1$; goto a_j
(0-test r_1)	a_k:	if $r_1 = 0$ then goto b_{j_1} else goto b_{j_2}
(0-test r_2)	b_k:	if $r_2 = 0$ then goto a_{j_1} else goto a_{j_2}
(Jump$_1$)	a_k:	goto b_j
(Jump$_1$)	b_k:	goto a_j

Computations

$$\langle a_1, n, 0\rangle \xrightarrow{a_1} \cdots \xrightarrow{b_j} \langle a_i, n_i, m_i\rangle \xrightarrow{a_i} \langle b_k, n_k, m_k\rangle \xrightarrow{b_k} \cdots$$

Undecidability of Multiplicative Linear Authorization Logic

Two counter machine

Instructions (uniquely labelled)

(Add r_1) a_k: $r_1 = r_1 + 1$; goto b_j
(Add r_2) b_k: $r_2 = r_2 + 1$; goto a_j
(Sub r_1) a_k: $r_1 = r_1 - 1$; goto b_j
(Sub r_2) b_k: $r_2 = r_2 - 1$; goto a_j

(0-test r_1) a_k: if $r_1 = 0$ then goto b_{j_1} else goto b_{j_2}
(0-test r_2) b_k: if $r_2 = 0$ then goto a_{j_1} else goto a_{j_2}
(Jump$_1$) a_k: goto b_j
(Jump$_1$) b_k: goto a_j

Computations

$$\langle a_1, n, 0\rangle \xrightarrow{a_1} \cdots \xrightarrow{b_j} \langle a_i, n_i, m_i\rangle \xrightarrow{a_i} \langle b_k, n_k, m_k\rangle \xrightarrow{b_k} \cdots$$

Final State

$$\langle a_0, 0, 0\rangle$$

Undecidability of Multiplicative Linear Authorization Logic

Two counter machine

Instructions (uniquely labelled)

(Add r_1) a_k: $r_1 = r_1 + 1$; goto b_j
(Add r_2) b_k: $r_2 = r_2 + 1$; goto a_j
(Sub r_1) a_k: $r_1 = r_1 - 1$; goto b_j
(Sub r_2) b_k: $r_2 = r_2 - 1$; goto a_j

(0-test r_1) a_k: if $r_1 = 0$ then goto b_{j_1} else goto b_{j_2}
(0-test r_2) b_k: if $r_2 = 0$ then goto a_{j_1} else goto a_{j_2}
(Jump$_1$) a_k: goto b_j
(Jump$_1$) b_k: goto a_j

Computations

$$\langle a_1, n, 0\rangle \xrightarrow{a_1} \cdots \xrightarrow{b_j} \langle a_i, n_i, m_i\rangle \xrightarrow{a_i} \langle b_k, n_k, m_k\rangle \xrightarrow{b_k} \cdots$$

Final State

$$\langle a_0, 0, 0\rangle$$

The termination problem for two-counter machines is undecidable.

Translation

Translation

Assume two principals A and B, where A is responsible for the register 1 and B for the register 2.

Translation

Assume two principals A and B, where A is responsible for the register 1 and B for the register 2.

Configurations (similar for b-states)

$$\langle a_i, n_i, m_i \rangle$$

$$\underbrace{A \text{ has } r_1, \ldots, A \text{ has } r_1}_{n_i \text{ copies}}, \underbrace{B \text{ has } r_2, \ldots B \text{ has } r_2}_{m_i \text{ copies}} \longrightarrow A \text{ has } a_i$$

Undecidability of Multiplicative Linear Authorization Logic

Translation – Instructions

ADD$_1$: $(A \text{ has } r_1 \multimap B \text{ says } b_j) \multimap A \text{ says } a_k$
ADD$_2$: $(B \text{ has } r_2 \multimap A \text{ says } a_j) \multimap B \text{ says } b_k$
SUB$_1$: $(A \text{ has } r_1 \otimes B \text{ says } b_j) \multimap A \text{ says } a_k$
SUB$_2$: $(B \text{ has } r_2 \otimes A \text{ says } a_j) \multimap B \text{ says } b_k$
0-IF$_1$: $B \text{ has } (B \text{ says } b_{j_1}) \multimap A \text{ says } a_k$
0-IF$_2$: $A \text{ has } (A \text{ says } a_{j_1}) \multimap B \text{ says } b_k$
0-ELSE$_1$: $(A \text{ has } r_1 \multimap B \text{ says } b_{j_2}) \otimes A \text{ has } r_1 \multimap A \text{ says } a_k$
0-ELSE$_2$: $(B \text{ has } r_2 \multimap A \text{ says } a_{j_2}) \otimes B \text{ has } r_2 \multimap B \text{ says } b_k$
JUMP$_1$ $B \text{ says } b_j \multimap A \text{ says } a_k$
JUMP$_2$ $A \text{ says } a_j \multimap B \text{ says } b_k$
FINAL $A \text{ has } \top \otimes B \text{ has } \top \multimap A \text{ says } a_0$

Completeness

$$\mathsf{ADD}_1 : (A\ \mathsf{has}\ r_1 \multimap B\ \mathsf{says}\ b_j) \multimap A\ \mathsf{says}\ a_k$$

Completeness

$$\mathrm{ADD}_1 : (A \text{ has } r_1 \multimap B \text{ says } b_j) \multimap A \text{ says } a_k$$

Backchaining

$$\dfrac{\dfrac{}{A \text{ says } a_k \longrightarrow A \text{ says } a_k}\, I \qquad \dfrac{\Gamma, A \text{ has } r_1 \longrightarrow B \text{ says } b_j}{\Gamma \longrightarrow A \text{ has } r_1 \multimap B \text{ says } b_j}\, \multimap_R}{\Gamma \longrightarrow A \text{ says } a_k}\, \mathrm{ADD}_1$$

Completeness

0-IF$_1$: B has $(B$ says $b_{j_1})$ $\multimap$ A says a_k

Completeness

$$\text{0-IF}_1 : B \,\mathsf{has}\, (B \,\mathsf{says}\, b_{j_1}) \multimap A \,\mathsf{says}\, a_k$$

Backchaining

$$\dfrac{\dfrac{}{A \,\mathsf{says}\, a_k \longrightarrow A \,\mathsf{says}\, a_k}\, I \qquad \dfrac{\Gamma \longrightarrow B \,\mathsf{says}\, b_{j_1}}{\Gamma \longrightarrow B \,\mathsf{has}\, (B \,\mathsf{says}\, b_{j_1})}\, \mathsf{has}_R}{\Gamma \longrightarrow A \,\mathsf{says}\, a_k}\, \text{0-IF}_1$$

Soundness

For soundness, we need more invariants on how says formulas move while splitting the context.

Lemma: Sequents of the form below are not provable:

$$!^{\mathsf{gl}}\{\Theta_M\}, C \,\mathsf{says}\, q_i, D \,\mathsf{says}\, q_j, \Gamma \longrightarrow E \,\mathsf{says}\, q_k$$

Lemma: If the sequent of the following form is provable:

$$!^{\mathsf{gl}}\{\Theta_M\}, D \,\mathsf{says}\, q_j, \Gamma \longrightarrow C \,\mathsf{says}\, q_k,$$

then

$$\langle q_k, m, n\rangle \longrightarrow^* \langle qj, 0, 0\rangle$$

without any transition using the if case of zero instructions.

Main Result

Theorem The encoding of two counter machines is sound and complete.

Corollary The propositional multiplicative fragment for linear authorization logics with two principals and no function symbols is **undecidable**.

Agenda

- Linear Authorization Logic
- Undecidability
- **Proof search and MSR**
- PSPACE-completeness
- Conclusions and Future Work

Can we interpret policies as multiset rewrite rules?

Can we interpret policies as multiset rewrite rules?

States

$$T \;::=\; K \,\textsf{says}\, A \mid K \,\textsf{has}\, A \mid K \,\textsf{says}\, T \mid K \,\textsf{has}\, T$$

Can we interpret policies as multiset rewrite rules?

States

$$T ::= K \text{ says } A \mid K \text{ has } A \mid K \text{ says } T \mid K \text{ has } T$$

No knowledge as one can easily use it to encode the existential Horn implication problem, which is undecidable.

Can we interpret policies as multiset rewrite rules?

States

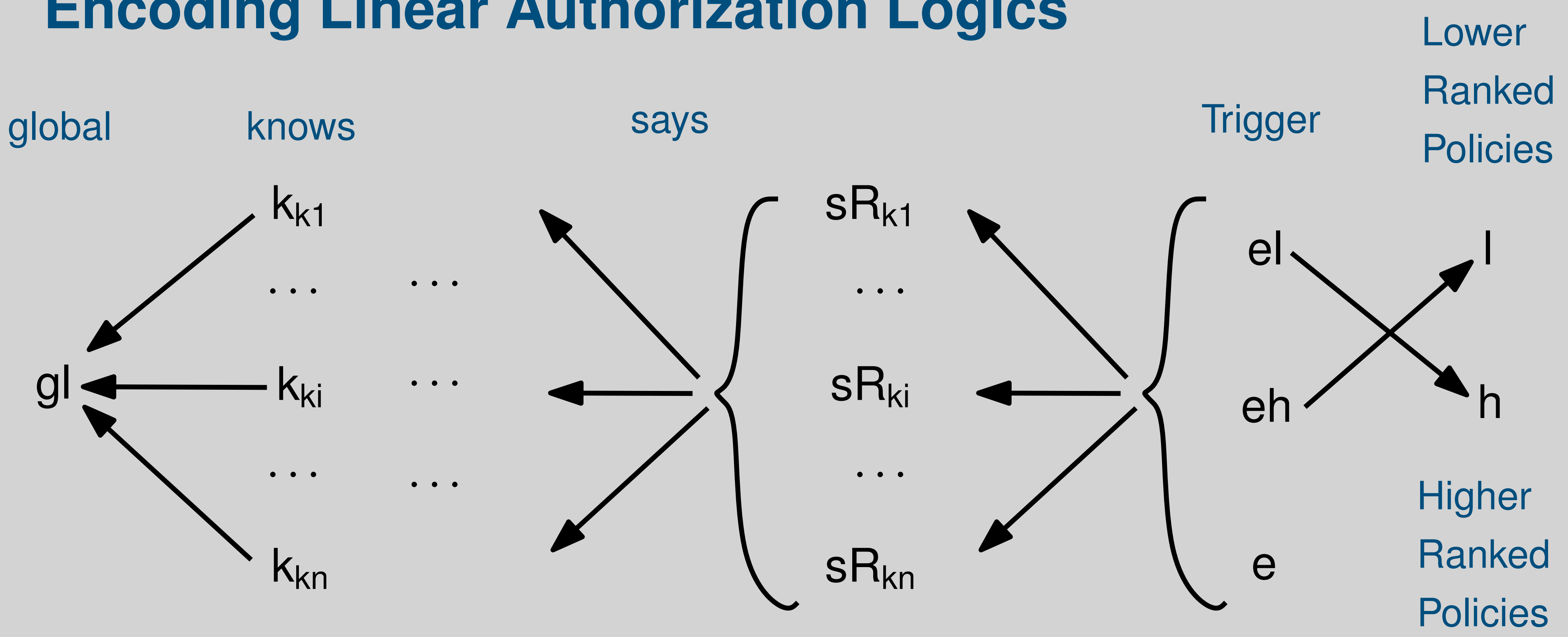
$$T ::= K \text{ says } A \mid K \text{ has } A \mid K \text{ says } T \mid K \text{ has } T$$

Policy Rules (Bipoles)

$$\forall \vec{y}\underbrace{[!^{\mathsf{e}} T_1 \otimes \cdots \otimes !^{\mathsf{e}} T_m] \multimap}_{\text{Pre-condition}} \exists \vec{x}.\underbrace{[T'_1 \otimes \cdots \otimes T'_n]}_{\text{Post-condition}}$$

Fresh Values

Encoding Linear Authorization Logics



Can we interpret policies as multiset rewrite rules?

States

$$T \ ::= \ K \,\textsf{says}\, A \mid K \,\textsf{has}\, A \mid K \,\textsf{says}\, T \mid K \,\textsf{has}\, T$$

Policy Rules (Bipoles)

$$\forall \vec{y}\,\underbrace{[!^{\mathsf{e}} T_1 \otimes \cdots \otimes !^{\mathsf{e}} T_m] \multimap}_{\text{Pre-condition}}\ \exists \vec{x}.\underbrace{[T'_1 \otimes \cdots \otimes T'_n]}_{\text{Post-condition}}$$

Fresh Values

Simple proofs!

$$\frac{T''_1 \longrightarrow T_1 \qquad \cdots \qquad T''_m \longrightarrow T_m \qquad !^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T'_1, \ldots, T'_k \longrightarrow G}{!^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T''_1, T''_2, \ldots, T''_m \longrightarrow G}$$

Can we interpret policies as multiset rewrite rules?

Simple proofs!

$$\frac{T_1'' \longrightarrow T_1 \quad \cdots \quad T_m'' \longrightarrow T_m \qquad !^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T_1', \ldots, T_k' \longrightarrow G}{!^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T_1'', T_2'', \ldots, T_m'' \longrightarrow G}$$

Lemma: Checking whether a sequent of the form $T \longrightarrow T'$ is provable is **in NP**. It is bounded by the number of modalities in T and T'.

Can we interpret policies as multiset rewrite rules?

States

$$T \ ::= \ K \,\mathsf{says}\, A \mid K \,\mathsf{has}\, A \mid K \,\mathsf{says}\, T \mid K \,\mathsf{has}\, T$$

Goals

$$!^{\mathsf{e}} T_G \otimes \top$$

Can we interpret policies as multiset rewrite rules?

States

$$T \ ::= \ K \,\mathsf{says}\, A \mid K \,\mathsf{has}\, A \mid K \,\mathsf{says}\, T \mid K \,\mathsf{has}\, T$$

Goals

$$!^{\mathsf{e}} T_G \otimes \top$$

Simple proofs!

$$\dfrac{T'' \longrightarrow T_G \qquad \dfrac{}{!^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T} \longrightarrow \top}\ \top_R}{!^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T'' \longrightarrow\ !^{\mathsf{e}} T_G \otimes \top}$$

Can we interpret policies as multiset rewrite rules?

States

$$T \ ::= \ K \,\mathsf{says}\, A \mid K \,\mathsf{has}\, A \mid K \,\mathsf{says}\, T \mid K \,\mathsf{has}\, T$$

Goals

$$!^{\mathsf{e}} T_G \otimes \top$$

Simple proofs!

$$\dfrac{T'' \longrightarrow T_G \qquad \dfrac{}{!^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T} \longrightarrow \top}\ \top_R}{!^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T'' \longrightarrow\ !^{\mathsf{e}} T_G \otimes \top}$$

Theorem: Proof search using only derivations of the forms above is sound and complete.

Can we interpret policies as rewrite rules?

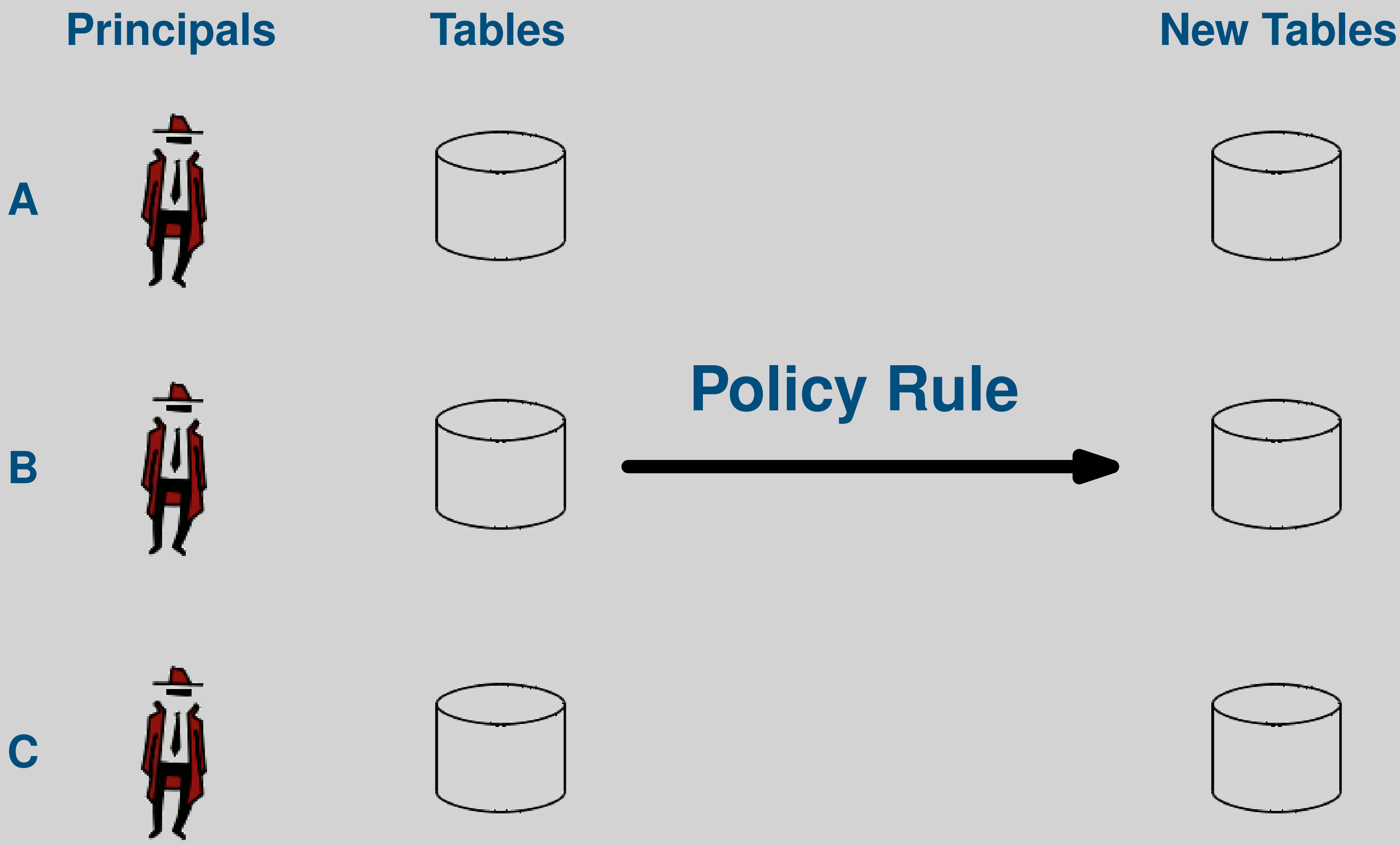
Principals

A 

B 

C 

Can we interpret policies as rewrite rules?



Agenda

- Linear Authorization Logic
- Undecidability
- Proof search and MSR
- **PSPACE-completeness**
- Conclusions and Future Work

Restriction based on [Kanovich, Rowe, Scedrov]

Restriction based on [Kanovich, Rowe, Scedrov]

Balanced Bipoles

$$\forall \vec{y}[!^{e} T_1 \otimes \cdots \otimes !^{e} T_m] \multimap \exists \vec{y}.[T_1' \otimes \cdots \otimes T_n']$$

$$n = m$$

Restriction based on [Kanovich, Rowe, Scedrov]

Balanced Bipoles $\forall \vec{y}[!^{\mathsf{e}} T_1 \otimes \cdots \otimes !^{\mathsf{e}} T_m] \multimap \exists \vec{y}.[T'_1 \otimes \cdots \otimes T'_n]$

$n = m$

$$\frac{T''_1 \longrightarrow T_1 \quad \cdots \quad T''_m \longrightarrow T_m \qquad !^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T'_1, \ldots, T'_n \longrightarrow G}{!^{\mathsf{h}}\{\Gamma_H\}, \mathcal{T}, T''_1, T''_2, \ldots, T''_m \longrightarrow G}$$

Number of T-formulas to the left-hand-side of sequents is **always the same**.

Parameters based on [Kanovich, Ban Kirigin, Nigam, and Scedrov]

- $\mathcal{L}$ is finite first-order alphabet without function symbols with J predicate symbols and D constant symbols;
- k is an upper bound on the arity of predicate symbols;
- $\mathcal{P}$ is a finite set of **balanced bipoles** specifying the policy rules;
- $\mathcal{T}$ is a multiset of exactly m T-formulas specifying the initial contents of the sequent.
- G is G-formula appearing at the right-hand-side of the sequent.

Problem

The sequent $!^{\mathsf{h}}\{\mathcal{P}\}, \mathcal{T} \longrightarrow G$ is provable or not in SELL

Theorem: There is an algorithm that determines whether a sequent $!^{\mathsf{h}}\{\mathcal{P}\}, \mathcal{T} \longrightarrow G$ is provable or not and runs in **PSPACE** with respect to the parameters above.

PSPACE-completeness

PSPACE lower bound

Easy sound and complete encoding of a Turing Machine that accepts in space n.

PSPACE upper bound

Lemma: Checking whether a sequent of the form $T \longrightarrow T'$ is provable is **in NP**. It is bounded by the number of modalities in T and T'.

Lemma: The upper bound M on the number of modalities in a T-formula appearing in a sequent $\mathcal{S}$ is the same as the upper bound in any one of its cut-free proofs.

Lemma: There are at most $MJ(D + 2mk)^k$ different T-formulas.

Theorem: There is an algorithm that determines whether a sequent $!^{\mathsf{h}}\{\mathcal{P}\}, \mathcal{T} \longrightarrow G$ is provable or not and runs in **PSPACE** with respect to the parameters above.

Conclusions and Future Work

We proposed a logical framework for linear authorization logics.

We showed that the MELL fragment of LAL is undecidable.

We proposed a novel first-order fragment of LAL for which provability is PSPACE-complete.

Future Work

Investigate the use of subexponentials on formulas appearing in the postcondition of rules. [CONCUR'13]

Decidable fragments when using knows modalities.

Questions