Access control types for agents

Rohit Chadha and Matthew Hennessy

University of Sussex

Overview: an agent calculus

- We consider an extension of π-calculus
- It has two named entities
  - Channels used for communication
  - Agents use channels to communicate
- Channels are the resources in this calculus
- Types are used to control access to the channels
  - The type of a channel names the agents that can access the channel

Syntax

- A system has a two-level structure
  - At the lower level, there are extended π-processes
  - At the higher level, there are agents running π threads
- At the level of processes, we add a primitive for sender authentication
  - The process $u?\{y\}\,(X : \mathsf{T})\,P$ inputs X along the channel u and y is bound to the name of the sender
  - The details of the authentication are abstracted away
- A typical system looks like $(\mathsf{new}\, e : \mathsf{E})(a[\![P]\!] \mid b[\![Q]\!])$
  - a and b are agents which share the name e
  - a is executing the thread P and b is executing the thread Q

Communication

- There are no sites and communication occurs globally
- There are two types of communication
  - Standard communication:
    $$a[\![c!\langle V\rangle\, P]\!] \mid b[\![c?(X : \mathsf{T})\, Q]\!] \longrightarrow a[\![P]\!] \mid b[\![Q\{\!|^{V}/_{X}|\!\}]\!]$$
  - Authenticated input:
    $$a[\![c!\langle V\rangle\, P]\!] \mid b[\![c?\{y\}\,(X : \mathsf{T})\, Q]\!] \longrightarrow a[\![P]\!] \mid b[\![Q\{\!|^{a}/_{y}, ^{V}/_{X}|\!\}]\!]$$
    b learns the identity of the sender.

Overview: types

- Channels are typed as list of input and output capabilities
  - An input capability $\mathsf{r}_u\langle \mathsf{F} \rangle$ means u can read on the channel
  - An output capability $\mathsf{w}_u\langle \mathsf{F} \rangle$ means u can write on the channel
- There is subtyping relation $<:$ on channel types
  - A channel type, C is a subtype of C' if C is less restrictive than C'
- The type agent classifies a name as an agent

Capability Types

- In capability types, $\mathsf{r}_u\langle \mathsf{F}\rangle$ or $\mathsf{w}_u\langle \mathsf{F}\rangle$, F may be
  - A normal transmission type T
    - $\mathsf{r}_u\langle \mathsf{T}\rangle$ means u can read values of at *least* type T
    - $\mathsf{w}_u\langle \mathsf{T}\rangle$ means u can read values of at *most* type T
  - An authenticated transmission type $\mathsf{Adep}(x : \mathsf{agent})\,\mathsf{T}$
    - $\mathsf{r}_u\langle \mathsf{F}\rangle$ means u can read authenticated values
      Values read must have at least type $\mathsf{T}\{\!|{}^{v}\!/\!_{x}|\!\}$, if v is the sender
    - $\mathsf{w}_u\langle \mathsf{F}\rangle$ means u can write authenticated values
      Values written must have at most $\mathsf{T}\{\!|{}^{u}\!/\!_{x}|\!\}$

The wild card *

- In place of an identifier u, a capability type may also have a special symbol $*$
  - $\mathsf{r}_*\langle \mathsf{F} \rangle$ means anybody can read on the channel
  - $\mathsf{w}_*\langle \mathsf{F} \rangle$ means anybody can write on the channel

Type judgements

- A type judgement for a system in the agent calculus takes the form

$$\Gamma \vdash M$$

- Γ, the type environment, is a list of identifiers
  - a : agent, meaning that a is an agent
  - u : C, meaning that u is a channel that has capability list C
- $\Gamma \vdash M$ if in the execution of M, an agent a in Γ accesses a channel u : C in Γ, only when allowed by C

Typing values and processes

- The typing judgement uses two other judgements
- A judgement for typing values, $\Gamma \vdash v : \mathsf{T}$
  - Keeps track of access

    For example, if $\Gamma \vdash u : \mathsf{r}_a\langle \mathsf{T} \rangle$ then a is allowed to input values of type T on u
- A judgement for typing process threads, $\Gamma \vdash_a P$
  - a is allowed by Γ to perform the possible input/output while executing P

Type inference for communication

- Output on a channel

$$\frac{\Gamma \vdash V : \mathsf{T} \qquad \Gamma \vdash u : \mathsf{w}_a\langle \mathsf{T} \rangle \qquad \Gamma \vdash_a P : \mathbf{proc}}{\Gamma \vdash_a u!\langle V \rangle\, P : \mathbf{proc}}$$

- Input from a channel

$$\frac{\Gamma \vdash u : \mathsf{r}_a\langle \mathsf{T} \rangle \qquad \Gamma, \{X : \mathsf{T}\} \vdash_a P : \mathbf{proc}}{\Gamma \vdash_a u?(X : \mathsf{T})\, P : \mathbf{proc}}$$

Authenticated communication

- Output on an authenticated channel

$$\frac{\Gamma \vdash V : \mathsf{T}\{\!|{}^{a}/_{y}|\!\} \qquad \Gamma \vdash u : \mathsf{w}_a\langle \mathsf{Adep}(y : \mathsf{agent})\, \mathsf{T}\rangle \qquad \Gamma \vdash_a P : \mathbf{proc}}{\Gamma \vdash_a u!\,\langle V\rangle\, P : \mathbf{proc}}$$

- Input from an authenticated channel

$$\frac{\Gamma, \vdash u : \mathsf{r}_a\langle \mathsf{Adep}(y : \mathsf{agent})\, \mathsf{T}\rangle \qquad \Gamma, y : \mathsf{agent},\ \{X : \mathsf{T}\} \vdash_a P : \mathbf{proc}}{\Gamma \vdash_a u?\{y\}\,(X : \mathsf{T})\, P : \mathbf{proc}}$$

Simple examples

- Consider the system
  $M = a[\![c!\langle b\rangle\,\mathsf{stop}]\!] \mid d[\![c?(x:\mathbf{bool})\,.stop]\!]$
- $\Gamma \vdash M$ if Γ is
  $a:\mathsf{agent}, d:\mathsf{agent}, b:\mathbf{bool}, c:\mathsf{w}_a\langle\mathbf{bool}\rangle, \mathsf{r}_d\langle\mathbf{bool}\rangle$
- $\Gamma \not\vdash M$ if Γ is
  $a:\mathsf{agent}, d:\mathsf{agent}, b:\mathbf{bool}, e:\mathsf{agent},$
  $c:\mathsf{w}_e\langle\mathbf{bool}\rangle, \mathsf{r}_d\langle\mathbf{bool}\rangle$

Simple examples continued...

- Consider the system
  $M = a[\![c!\langle b\rangle\, \mathsf{stop}]\!] \mid d[\![c?(x : \mathbf{bool})\,.stop]\!]$
- $\Gamma \vdash M$ if Γ is $a : \mathsf{agent}, d : \mathsf{agent}, b : \mathbf{bool}, e : \mathsf{agent}, c : \mathsf{w}_a\langle\mathbf{bool}\rangle, \mathsf{w}_e\langle\mathbf{bool}\rangle, \mathsf{r}_d\langle\mathbf{bool}\rangle$

If a channel type C *lists more elements than* C′*, then it is less restrictive*

Simple examples continued...

- Consider the system
  $M = a[\![c!\langle b\rangle\,\mathsf{stop}]\!] \mid d[\![c?(x : \mathbf{bool})\,.stop]\!]$
- $\Gamma \vdash M$ if Γ is

  $a : \mathsf{agent}, d : \mathsf{agent}, b : \mathbf{bool}, c : \mathsf{w}_*\langle\mathbf{bool}\rangle, \mathsf{r}_d\langle\mathbf{bool}\rangle$ or,

  $a : \mathsf{agent}, d : \mathsf{agent}, b : \mathbf{bool}, c : \mathsf{w}_a\langle\mathbf{bool}\rangle, \mathsf{r}_*\langle\mathbf{bool}\rangle$

  $\mathsf{w}_*\langle\mathsf{F}\rangle$ *is less restrictive than* $\mathsf{w}_a\langle\mathsf{F}\rangle$ *and* $\mathsf{r}_*\langle\mathsf{F}\rangle$ *is less restrictive than* $\mathsf{r}_a\langle\mathsf{F}\rangle$

Handover of capabilities

- Consider the system
  $M = a[\![c!\langle b\rangle\, \mathsf{stop}]\!] \mid d[\![c?(x : \mathsf{T})\,.stop]\!]$
- Let T be $\mathsf{w}_d\langle\mathbf{bool}\rangle$
- $\Gamma \vdash M$ if Γ is
  $a : \mathsf{agent}, d : \mathsf{agent}, b : \mathsf{w}_*\langle\mathbf{bool}\rangle,$
  $c : \mathsf{w}_a\langle\mathsf{w}_d\langle\mathbf{bool}\rangle\rangle, \mathsf{r}_d\langle\mathsf{w}_d\langle\mathbf{bool}\rangle\rangle$
- a hands over the capability of writing on b to d

Handover of capabilities continued..

- In particular, b may be a channel that only a knows at $\mathsf{w}_*\langle\mathbf{bool}\rangle$
- Consider the system, M
  - $((\mathsf{new}\, b : \mathsf{T})\, a[\![c!\langle b\rangle\, \mathsf{stop}]\!]) \mid d[\![c?(x : \mathsf{T})\, .stop]\!]$
- Let T be $\mathsf{w}_*\langle\mathbf{bool}\rangle$
- $\Gamma \vdash M$ if Γ is

  $a : \mathsf{agent}, d : \mathsf{agent}, c : \mathsf{w}_a\langle\mathsf{w}_d\langle\mathbf{bool}\rangle\rangle, \mathsf{r}_d\langle\mathsf{w}_d\langle\mathbf{bool}\rangle\rangle$

Handover of capabilities continued..

- a can demand payment for the capability
  - $((\mathsf{new}\, b : \mathsf{T})\, a[\![pay?\{y\}\,(z : \mathsf{T}')!\langle b\rangle\,.stop]\!])$
  - a gets payment from y, who also sends a return channel z
  - a sends back the name b on z
- In order for this to work, we can choose
  - T' as $\mathsf{w}_a\langle \mathsf{w}_y\langle \mathsf{T}\rangle\rangle$; a returns b on z, allowing only the paying agent to write on the channel
  - Γ contains $a : \mathsf{agent}, pay : \mathsf{r}_a\langle \mathsf{Adep}(y : \mathsf{agent})\,\mathsf{T}'\rangle, \mathsf{w}_*\langle \mathsf{Adep}(y : \mathsf{agent})\,\mathsf{T}'\rangle$

Handover of capabilities continued..

- The above system can be thought of as a repository of papers
- b a channel on which a paying agent could request papers
- For example, in order to get the write permission on b, an agent d can execute the following code
  - $(\mathsf{new}\, ret : \mathsf{T''})(pay!\langle ret\rangle\, .ret?(get : \mathsf{w}_d\langle \mathsf{T}\rangle)\, .get!\ldots)$
    where T” is $\mathsf{w}_a\langle \mathsf{w}_d\langle \mathsf{T}\rangle\rangle, \mathsf{r}_d\langle \mathsf{w}_d\langle \mathsf{T}\rangle\rangle$

Conclusions

- An agent calculus with two named entities, channels and agents
- The calculus allows for sender authentication
- A type system that controls of access to channels
  - The type of a channel explicitly names the agents allowed to access
  - A special type to allow everybody to access the channel
  - A dependent type to model sender authentication

Ongoing and future work

- A typed modal logic that allows us to specify the desired properties
  - A modal μ-calculus with a past operator
- Proof techniques to show that systems satisfy these properties
- Investigate relationship between the logic and types
- Extensions with sites, delegation, etc..