Security Types for Dynamic Web Data

Mariangiola Dezani-Ciancaglini[1], Silvia Ghilezan[2], Svetlana Jakšić[2], Jovanka Pantović[2], Daniele Varacca[3]

Dipartimento di Informatica, Università di Torino

Faculty of Technical Sciences, University of Novi Sad

Universite Paris Diderot

BETTY meeting, Rome

Distributed systems - decentralised peer-to-peer networks

- secure management of distributed data (XML)
- different processes have different access rights
- dynamic changes of access rights
- different access policies in different locations
- exchange between data and processes preserving access control

- One solution - typed models
  - control of access
  - control of movements rights

Distributed systems - decentralised peer-to-peer networks

- secure management of distributed data (XML)
- different processes have different access rights
- dynamic changes of access rights
- different access policies in different locations
- exchange between data and processes preserving access control

- One solution - typed models
  - control of access
  - control of movements rights

Related work

- X$d\pi$ calculus - Gardner, Maffeis
  - localised mobile processes
  - distributed, dynamic, semi-structured web data

Philippa Gardner and Sergio Maffeis.
Modelling dynamic web data.
Theoretical Computer Science, 342(1):104–131, 2005.

- Variety of type systems for $d\pi$ and related calculi
  - controlling the use of accesses and mobility of processes

Security levels

- Security types for X$d\pi$
  Dezani, Ghilezan, Pantović, Varacca, 2008
  - partially ordered set (with bottom) as security levels
  - communication, movement and data usage control

Mariangiola Dezani-Ciancaglini, Silvia Ghilezan, and Jovanka Pantovic.
Security types for dynamic web data.
TGC, volume 4661 of *Lecture Notes in Computer Science*, pages 263–280.
Springer, 2006.

Mariangiola Dezani-Ciancaglini, Silvia Ghilezan, Jovanka Pantovic, and Daniele Varacca.
Security types for dynamic web data.
Theor. Comput. Sci., 402(2-3):156–171, 2008.

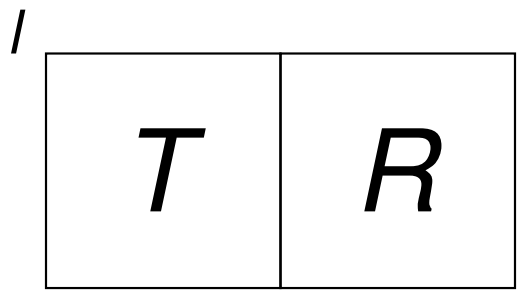
Roles

- Security types for ®X$d\pi$-calculus
  Dezani, Ghilezan, Jakšić, Pantović, 2011
  - partially ordered set of roles, RBAC
  - dynamic change of access rights
  - communication, movement and data usage control

Mariangiola Dezani-Ciancaglini, Silvia Ghilezan, Svetlana Jakšić, and Jovanka Pantović.
Types for Role-Based Access Control of Dynamic Web Data.
In *WFLP'10*, volume 6559 of *LNCS*, pages 1–29. Springer, 2011.

Silvia Ghilezan, Svetlana Jakšić, Jovanka Pantović, and Mariangiola Dezani-Ciancaglini.
Types and Roles for Web Security.
Transactions on Advanced Research, 8(2):16–21, 2012.

Locations and networks

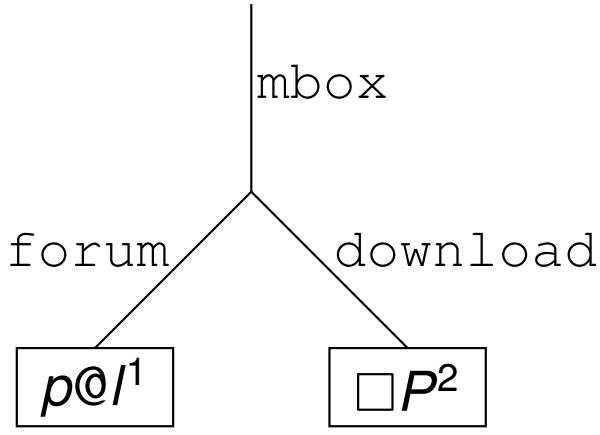


- Each location consists of data in a form of a tree and a process
- A well-formed network is a parallel composition (|) of *locations* with different names.

$$\mathbf{N} ::= \mathbf{0} \mid \mathbf{N} \mid \mathbf{N} \mid l[T \parallel R] \mid (\nu c^{Tv})\mathbf{N}$$

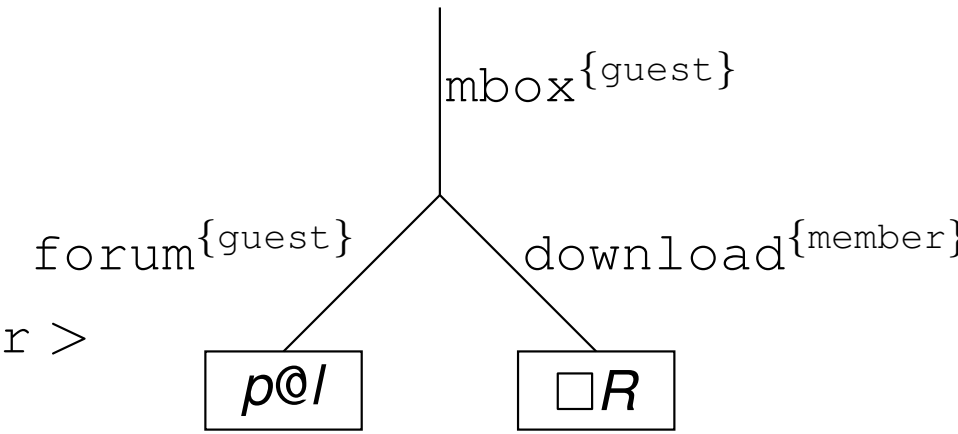
Data

```
< mbox >
    < forum >
        p@l^1
    < /forum >
    < download >
        □P^2
    < /download >
< /mbox >
```



Data

```
< mbox role = guest >
    < forum role = guest >
        p@λ
    < /forum >
    < download role = member >
        □R
    < /download >
< /mbox >
```



Processes

π-calculus

P	$::=$	0	the nil process
	$\mid$	$P \mid P$	composition of processes
	$\mid$	$\bar{c}^{Tv}\langle v \rangle$	output value v on a channel c
	$\mid$	$c^{Tv}(x).P$	input parameterized by a variable x
	$\mid$	$!c^{Tv}(x).P$	replication of an input process

$d\pi$-calculus

P	$::=$	$\texttt{go}\ \lambda.R$	migrate to location λ, continue as R

X$d\pi$-calculus

P	$::=$	$\texttt{run}_p$	run command
	$\mid$	$\texttt{read}_p(\chi).P$	read command
	$\mid$	$\texttt{change}_p(\chi, V).P$	change command

Processes

®X$d\pi$-calculus

P	$::=$	$\texttt{enable}(r)_p.P$	gives permission to the role r to access data on the path p
	$\mid$	$\texttt{disable}(r)_p.P$	removes the role r from roles that are allowed to access data on the path p
R	$::=$	$P^{\neg\rho}$	single process with roles ρ
	$\mid$	$R \mid R$	parallel composition of processes with roles
	$\mid$	$(\nu c^{Tv})R$	restriction of channel name c

Type system

Main goals

- to control communication of values
- to control migration and activation of processes
- to control access to data and their modification
- to control change of access rights

Location Policy

$$\mathcal{P} = (\sigma, \mathcal{E}, \mathcal{D})$$

- σ: set of minimal roles which can access data
- $\mathcal{E}$: policy for role enabling
- $\mathcal{D}$: policy for role disabling

Type system

Main goals

- to control communication of values
- to control migration and activation of processes
- to control access to data and their modification
- to control change of access rights

Location Policy

$$\mathcal{P} = (\sigma, \mathcal{E}, \mathcal{D})$$

- σ: set of minimal roles which can access data
- $\mathcal{E}$: policy for role enabling
- $\mathcal{D}$: policy for role disabling

Syntax of types

Ch(*Tv*)	type of channels communicating values of type *Tv*
Loc($\mathcal{P}$)	type of locations with the policy $\mathcal{P}$
Script($\mathcal{P}$)	type of scripts which can be activated at locations with the policy $\mathcal{P}$
Path(α)	type of paths having the last edge with the set of roles α
Pointer(α)	type of pointers whose path is typed by *Path*(α)
Tree($\mathcal{P}, \tau, \zeta$)	type of trees, which can stay at locations with the policy $\mathcal{P}$, with initial branches asking τ and which can be completely accessed by processes with at least one role of ζ
Proc($\mathcal{P}, \rho$)	type of pure processes, which can stay at locations with the policy $\mathcal{P}$ and which can be assigned roles ρ
ProcRole($\mathcal{P}$)	type of processes with roles which can stay at locations with the policy $\mathcal{P}$

Typing rules

$$\frac{\Gamma \vdash p : \mathit{Path}(\alpha) \quad \Gamma \vdash P : \mathit{Proc}(\mathcal{P}, \rho) \quad \alpha \leq \rho \qquad \Gamma \cup \Gamma_\chi \vdash \begin{cases} V : \mathit{Script}(\mathcal{P}) \text{ or} \\ V : \mathit{Pointer}(\beta) \text{ or} \\ V : \mathit{Tree}(\mathcal{P}, \tau', \zeta') \quad \alpha \leqq \tau' \end{cases} \qquad \text{if } \chi = x^{(\mathcal{P},\tau,\zeta)} \text{ then } \zeta \leq \rho}{\Gamma \vdash \mathtt{change}_p(\chi, V).P : \mathit{Proc}(\mathcal{P}, \rho)} \quad \text{(Change)}$$

$$\frac{\vdash l : \mathit{Loc}(\mathcal{P}) \quad \vdash T : \mathit{Tree}(\mathcal{P}, \tau, \zeta) \quad \vdash R : \mathit{ProcRole}(\mathcal{P})}{\vdash l[\![\ T \parallel R\ ]\!] : \mathit{Net}} \quad \text{(NetLoc)}$$

Safety properties

- (Subject reduction) If $\vdash \mathbf{N} : \mathit{Net}$ and $\mathbf{N} \rightarrow \mathbf{N}'$, then $\vdash \mathbf{N}' : \mathit{Net}$.

- Properties of location policies and communication:
  - **P0** All trees and processes in a location agree with the location policy;
  - **P1** A process with roles can communicate only values with characteristic roles accessible to the process.
- Properties of migration between locations:
  - **P2** A process with roles can migrate to another location only if it is well typed for that location.
- Properties of access of processes to local data trees:
  - **P3** A process with roles looks for a path in the local tree only if the path is accessible to the process.

Safety properties

- Properties of manipulation of local data trees by processes:
  - **P4** A script is activated in a location only if the corresponding process with roles can stay in that location;
  - **P5** A process with roles generated by a read command in a location can stay in that location;
  - **P6** A process with roles can erase a subtree of data only if it can access all data;
  - **P7** A tree built by a change command in a location can stay in that location;
  - **P8** A process with roles can add a role to an edge in the local tree only if this is allowed by the location policy;
  - **P9** A tree built by an enable command in a location can stay in that location;
  - **P10** A process with roles can erase a role from an edge in the local tree only if this is allowed by the location policy;
  - **P11** A tree built by a disable command in a location can stay in that location.

Future work

- extension of type system
- security and privacy properties of systems with data in other formats