Privacy-Enhanced Internet

Yves Deswarte
deswarte@laas.fr

LAAS-CNRS

Summary

- "Privacy": Definition & Legislation
- Example: e-commerce transaction
- PETs : Privacy Enhancing Technologies
  - IP address protection
  - Location protection
  - Anonymous access to services
  - Privacy-preserving authorization

Privacy definition

- Intimacy, protection of private domain, of personal data
- Common criteria (ISO 15408): "*protection against discovery and misuse of identity by other users*"

  One functional class, 4 families:
  - Anonymity: ensures that other users or subjects are unable to determine the identity of a user bound to a subject or operation.
  - Pseudonymity: idem, but ensures that this user is still accountable for its actions.
  - Unlinkability: ensures that users and/or subjects are unable to determine whether the same user caused certain specific operations.
  - Unobservability: ensures that users and/or subjects cannot determine whether an operation is being performed.

Legislation examples

- Protection of nominative data:
  French law "Informatique et Libertés" (Jan. 1978): mandatory declaration of files, forbidden info (race, religion, political & union opinions, …)
- Protection of personal data:
  European convention (Jan. 1981) + directives 95/46/EC (*free movement*) & 97/66/EC (*telecoms*)
- Data Protection Agencies
- French laws on professional secret & correspondence secret

But…

- Internet spans international borders
- Application of privacy laws relies on a voluntary effort (cost prohibitive, bad return on investment)
- Trade-off between Internet security and users' privacy

PETs : Privacy Enhancing Technologies

- "Need-to-know" principle
  Personal data should be available only to those that need them to accomplish a given task (and only for the task duration)
- … on the Internet like in the real world …
- …with limits:
  - Law enforcement (e.g., money laundering, terrorism, …)
  - Dispute resolution

  some personal data must be made available under judicial control: pseudonymity rather than anonymity

Example : e-commerce transaction(1)

- Involved parties:
  customer, merchant, delivery company, banks, credit card company, ISPs, …
- The merchant does not NEED the customer's identity (usually), but must be ensured that the money order is valid.
- The delivery company does not NEED to know the customer's identity, or what goods are purchased (except their physical characteristics), only the delivery addressee.

Example : e-commerce transaction(2)

- The customer's bank does not NEED to know the merchant's identity or what goods are purchased, only the merchant's account reference.
- The merchant's bank does not NEED to know the customer …
- The ISPs do not NEED to know the transaction content, only the technical characteristics of the connection …

Internet : 4 kinds of PETs

- IP address protection
- Location protection
- Anonymous service access
- Privacy-preserving authorization

IP address protection (1)

- An IP address can be nominative data

Example:

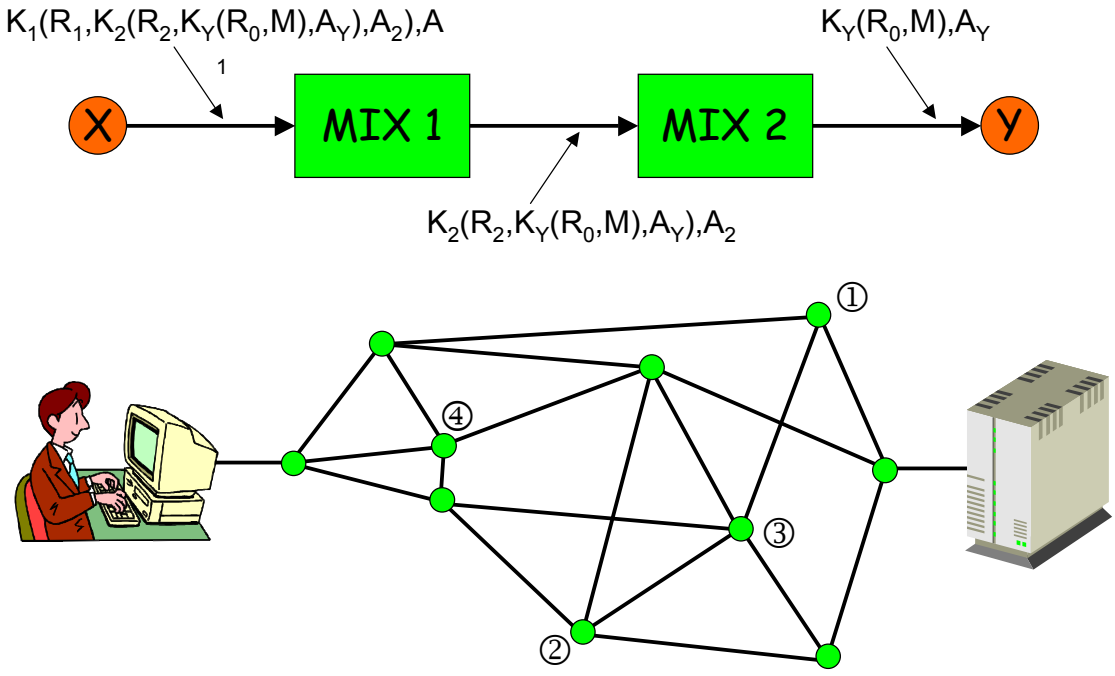
```
Return-Path: <Yves.Deswarte@laas.fr>
Received: from laas.laas.fr (140.93.0.15) by mail.libertysurf.net (6.5.026)
      id 3D518DEF00116A4D for yves.deswarte@libertysurf.fr; Tue, 13 Aug
2002 13:44:40 +0200
Received: from [140.93.21.6] (tsfyd [140.93.21.6])
    by laas.laas.fr (8.12.5/8.12.5) with ESMTP id g7DBid1D001531
    for <yves.deswarte@libertysurf.fr>; Tue, 13 Aug 2002 13:44:39 +0200
(CEST)
User-Agent: Microsoft-Entourage/10.1.0.2006
Date: Tue, 13 Aug 2002 13:44:38 +0200
Subject: test
From: Yves Deswarte <Yves.Deswarte@laas.fr>
To: <yves.deswarte@libertysurf.fr>
Message-ID: <B97EBDC6.2052%Yves.Deswarte@laas.fr>
Mime-version: 1.0
Content-type: text/plain; charset="US-ASCII"
Content-transfer-encoding: 7bit
```

IP address protection (2)

- PET: dynamic IP address assignment (DHCP, PPP, NAT, …)
- Anonymous routers:
  - MIX (David Chaum)
  - Onion Routing
  - Crowds
  - …

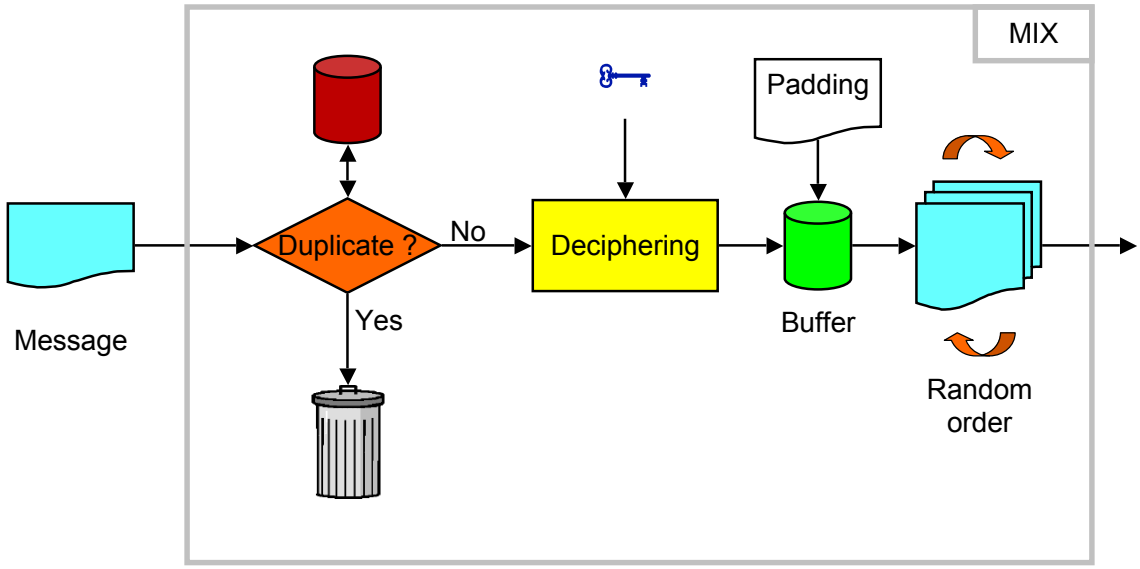
MIX / Onion Routing / Crowds

http://www.vote.caltech.edu/wote01/pdfs/juels2-wote.ppt



MIX : how does it work ?

http://www.inf.tu-dresden.de/



Location protection

- Today: one IP address <-> one location (topological routing)
- Many service providers know their customers location
  - Today: cellular phone operators, ISPs, …
  - Currently being deployed: fleet management, navigation, …
- Tomorrow: IP everywhere (*ubiquitous computing, ambiant intelligence …*) : every (nomadic) device will have an IP@, every person will own several devices, connected to nearby devices (ad-hoc networks), which will identify each others, route their communications, etc.
- New PETs should be developed to address these problems (e.g., session identities, …)

Anonymous access to services

- *Anonymity proxies*:
  - Web
  - ftp
  - e-mail
  - …
- Pseudonymity servers:
  - e-mail
  - Multiple identities provided by ISPs
  - Multiple virtual identities -> Liberty Alliance <http://www.projectliberty.org> vs. Microsoft Passport

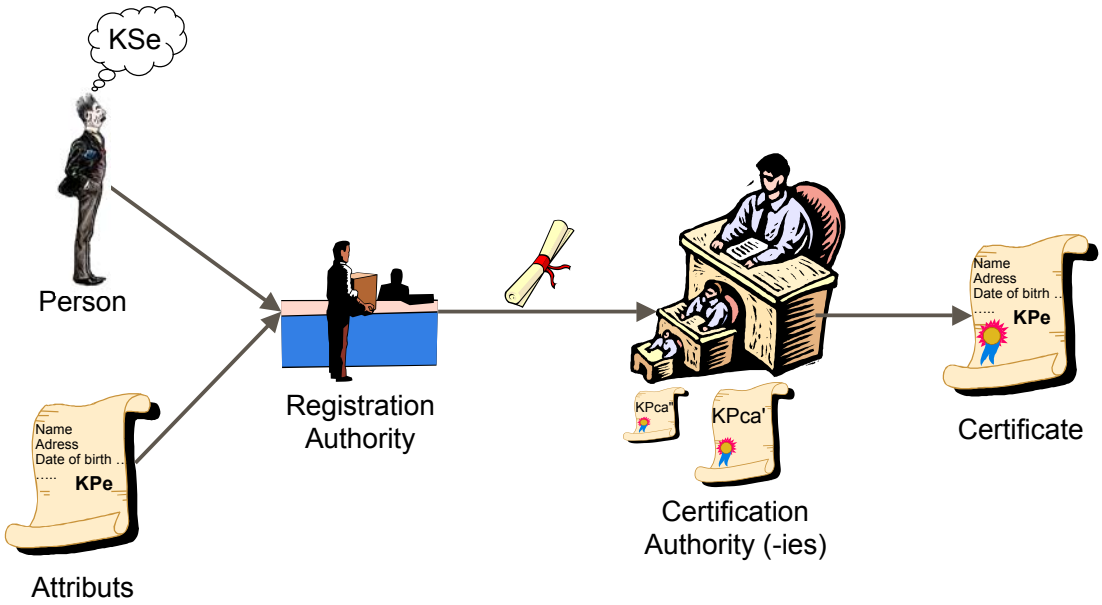
Internet authorization

- Today: client-server paradigm:
  - the server grants or denies privileges to the client according to the client's claimed identity (possibly verified by an authentication service)
  - The server collects all possible information on the transaction, to use it as evidence in case of dispute
  - Such information can be misused (clients profiling, direct marketing, spamming, black mailing, …)
- P3P : *Platform for Privacy Preferences Project* (W3C) automatic negotiation based on (claimed) privacy policies

This paradigm is outdated

- Internet transactions usually involve more than two parties (e.g., e-commerce)
- These parties have different (or even opposing) interests: they are mutually suspicious
- Privacy intrusive:
  it does not obey the need-to-know principle

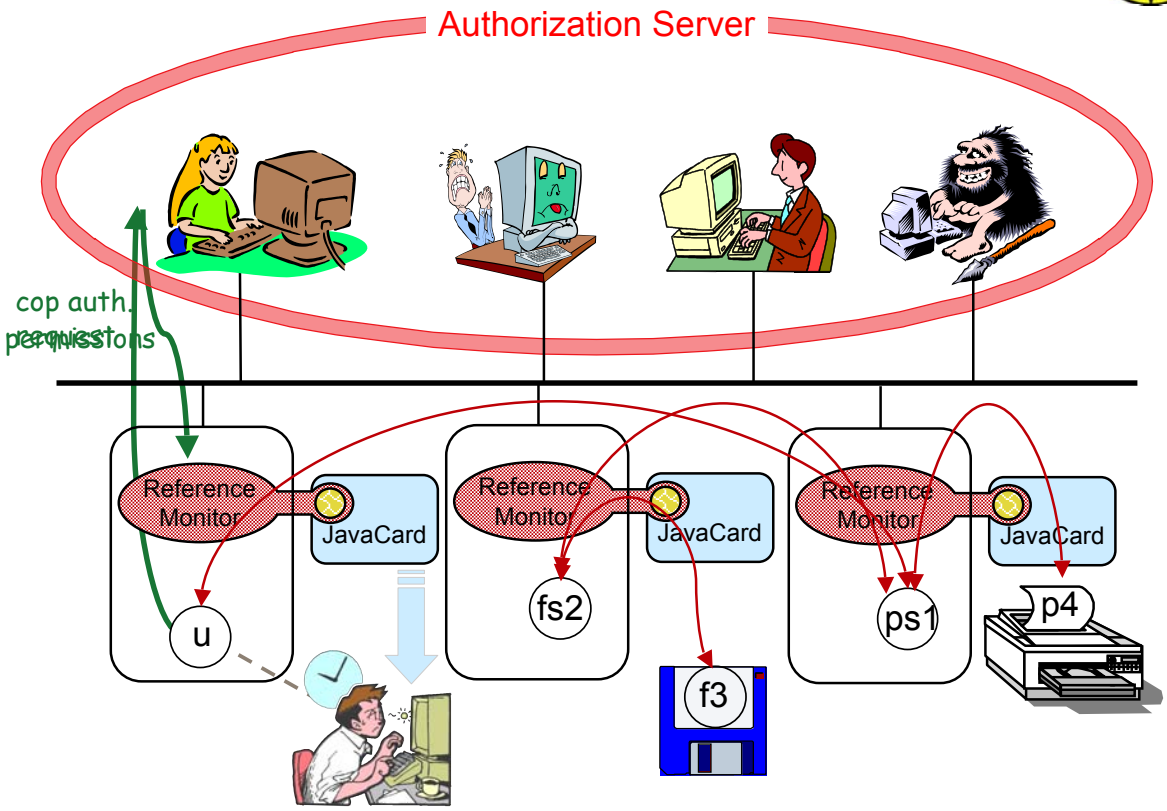
Certificates



Authorization proofs (1)

- Multiple certificates:
  - Subscription cards, association cards, …
  - Driving license, voter registration card, …
- Restricted certificates:
  - SPKI : attribute certificates/authorization certificates
  - "Partial Revelation of Certified Identity", F. Boudot, CARDIS 2000
- Problems: linkability (a single public key or multiple public keys for multiple certificates?), management of certificates and keys, authentcation, evidence collection, revocation, …

MAFTIA Authorization Scheme



Bibliography

- David Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", *Communications of the ACM*, 24/2 (1981) 84-88.
- David M. Goldschlag, Michael G. Reed, and Paul F. Syverson, "Onion Routing for Anonymous and Private Internet Connections," *Communications of the ACM*, vol. 42, num. 2, February 1999.
- M. K. Reiter and A. D. Rubin. "Crowds: Anonymity for web transactions", *ACM Transactions on Information and System Security,* 1(1):66-92, November 1998.
- Fabrice Bodot, "Partial Revelation of Certified Identity", *4th IFIP WG8.8 Working Conference on Smart Card and Advanced Applications* (*CARDIS*-2000),Sept. 2000, Bristol (UK), Kluwer (Eds: J. Domingo-Ferrer, D. Chan, A. Watson), pp.257-269.
- Yves Deswarte, Noreddine Abghour, Vincent Nicomette, David Powell, "An Internet Authorization Scheme using Smartcard-based Security Kernels", in Smart Card Programming and Security, Eds. Isabelle Attali and Thomas Jensen, Proc. e-Smart 2001, Cannes (France), 19-22 septembre 2001, Springer, LNCS n°2140, pp. 71-82.
- MAFTIA Delivrable D6 <http://www.research.ec.org/maftia/deliverables/index.html>

MAFTIA

IST Dependability Initiative
Cross Program Action 2
Dependability in services and technologies

Malicious- and Accidental-Fault Tolerance for Internet Applications

University of Newcastle (UK)	Brian Randell, Robert Stroud
University of Lisbon (P)	Paulo Verissimo
DSTL + QinetiQ (ex-DERA) (UK)	Tom McCutcheon, Sadie Creese
University of Saarland (D)	Birgit Pfitzmann
LAAS-CNRS, Toulouse (F)	Yves Deswarte, David Powell
IBM Research, Zurich (CH)	Marc Dacier, Michael Waidner

c. 55 man-years, EU funding c. 2.5M€
Jan. 2000 -> Dec. 2002 (Feb. 2003)

Objectives

- Architectural framework and conceptual model (WP1)
- Mechanisms and protocols:
  - dependable middleware (WP2)
  - large scale intrusion detection systems (WP3)
  - dependable trusted third parties (WP4)
  - distributed authorization mechanisms (WP5)
- Validation and assessment techniques (WP6)

MAFTIA Authorization Server

Makes use of MAFTIA middleware:

- Non-confidential information is replicated (atomic multicast)
- Confidential information is shared securely (threshold crypto)
- Global consensus is achieved (majority voting / Byzantine agreement)
- Authorization proofs are distributed to local reference monitors (threshold signatures)

Local protection

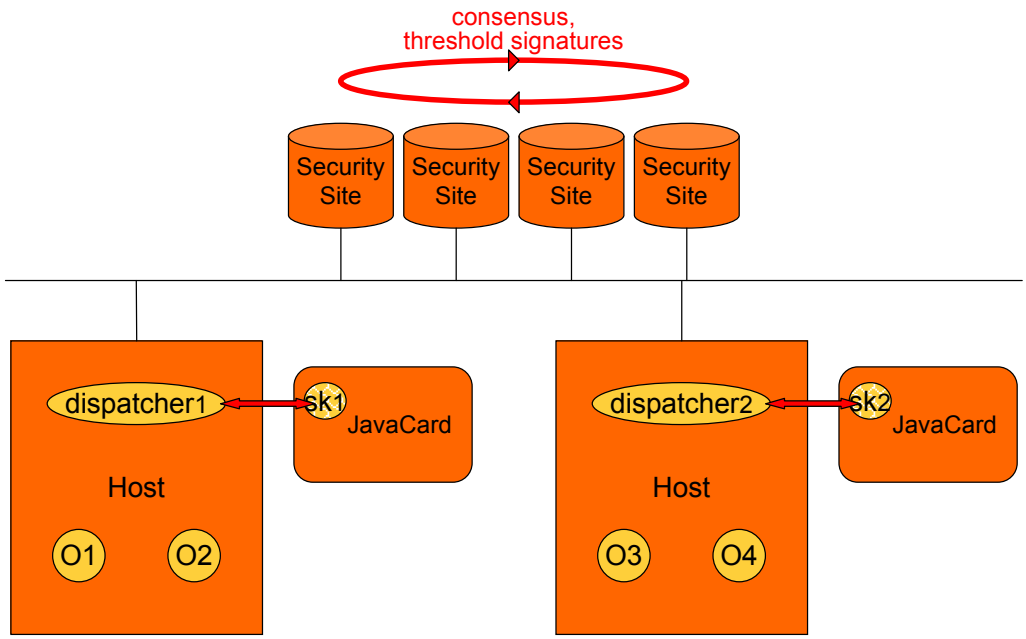
- Internet applications: heterogeneous platforms => no modification of user workstations or even servers
- => no trusted security kernel, but JVM
- Local Reference Monitor =
  - A local dispatcher (not trusted)
  - A JavaCard -> "security kernel"

Security properties

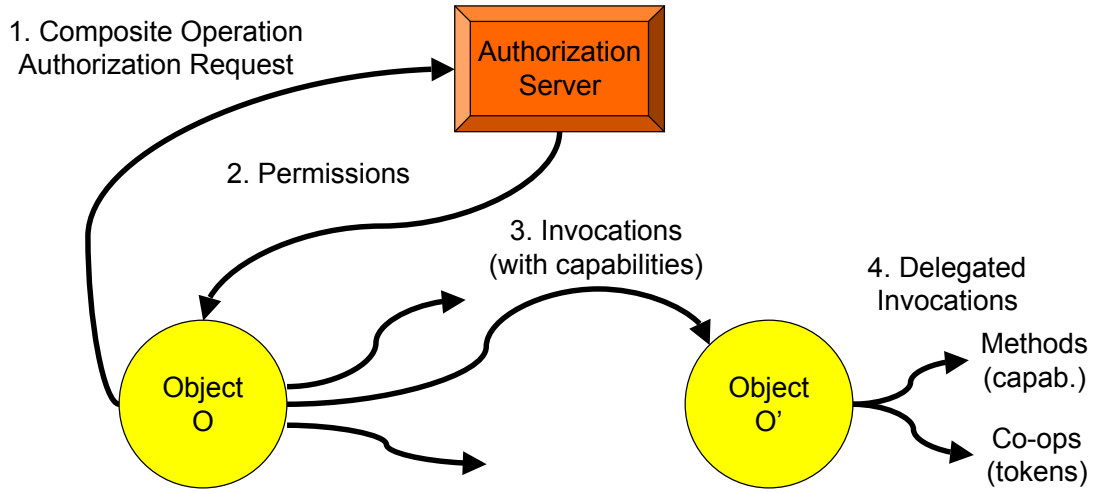
- Authorization server:
  - AS1: The AS generates only valid authorization proofs
  - AS2: It is not possible to prevent AS from generating valid authorization proofs
- Local reference monitors
  - RM1: Only valid operations will be executed on a non-faulty host
  - RM2: It is not possible to prevent non-faulty hosts from executing valid operations

- Assumption1: no network denial-of-service
- Assumption2: Java Card tamperproof

Architecture



Permissions



Permissions(O) = <{Perm(O,$O'.m$)*;Perm(O,cop)*}>$_{SKas}$
Perm(O, $O'.m$) = {O;$O'.m$($parC$);**cap**(O;$O'.m$($parC$))[;**vouch**($O'.m$)]}
Perm(O,cop) = {O;cop($parC$);**token**(O;cop($parC$))}
cap(O;$O'.m$($parC$)) = <{O;$O'.m$;$parC$;$nonce$}>$_{SKas,PKhost(O')}$
vouch($O'.m$) = <{Perm(O',$O''.m'$)*;Perm(O',cop)*}>$_{SKas}$
token(O;cop($parC$)) = <{O;cop;$parC$;$nonce$}>$_{SKas,PKas}$

Cop Authorization Checks

Symbolic rights: corresponding to the authorization for an object to execute composite operations

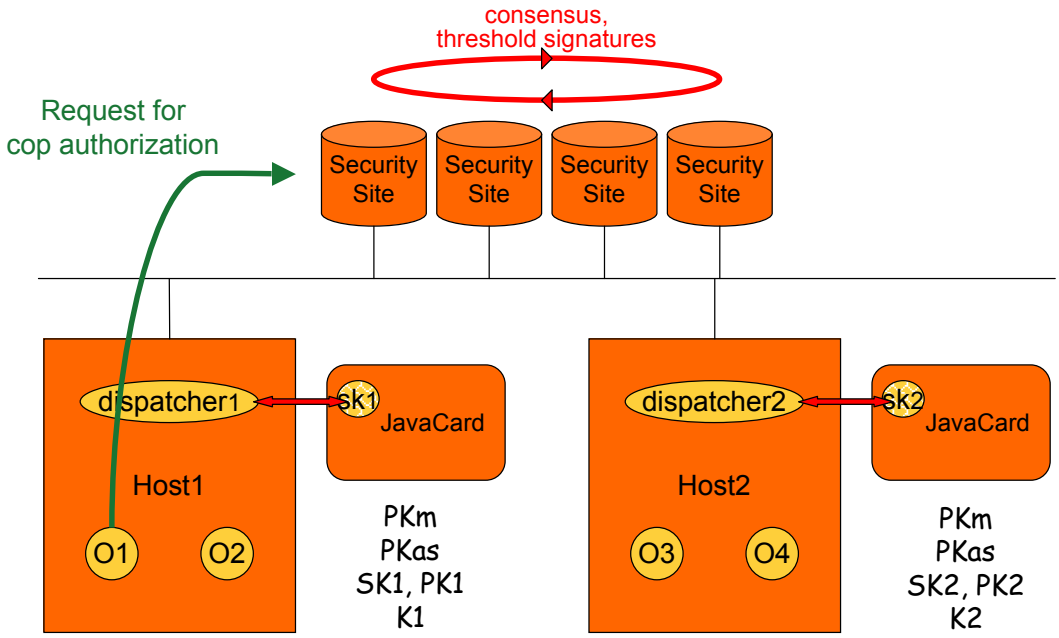
(a simple method execution is a particular case of cop)

	ps1	fs2	f 3	p 4
u			PF(this,PRINTER)	PF(FILE,this)
ps1				print
fs2				

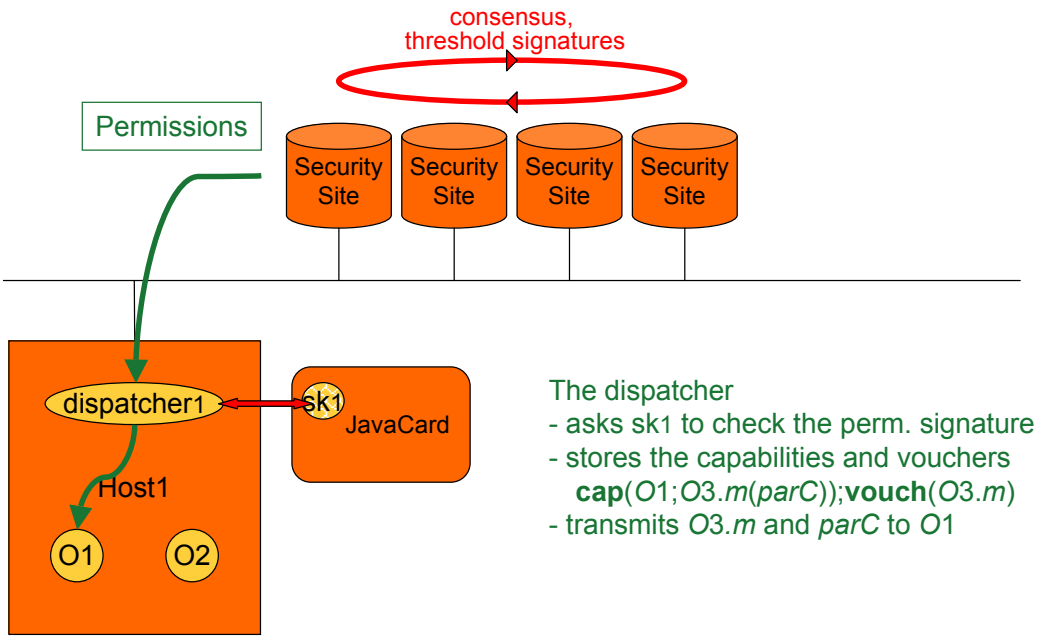
Symbolic right rules: to check authorization for composite operations

Permission creation rules: to generate permissions (capabilities and *vouchers*, tokens) to enable all methods executions

Architecture

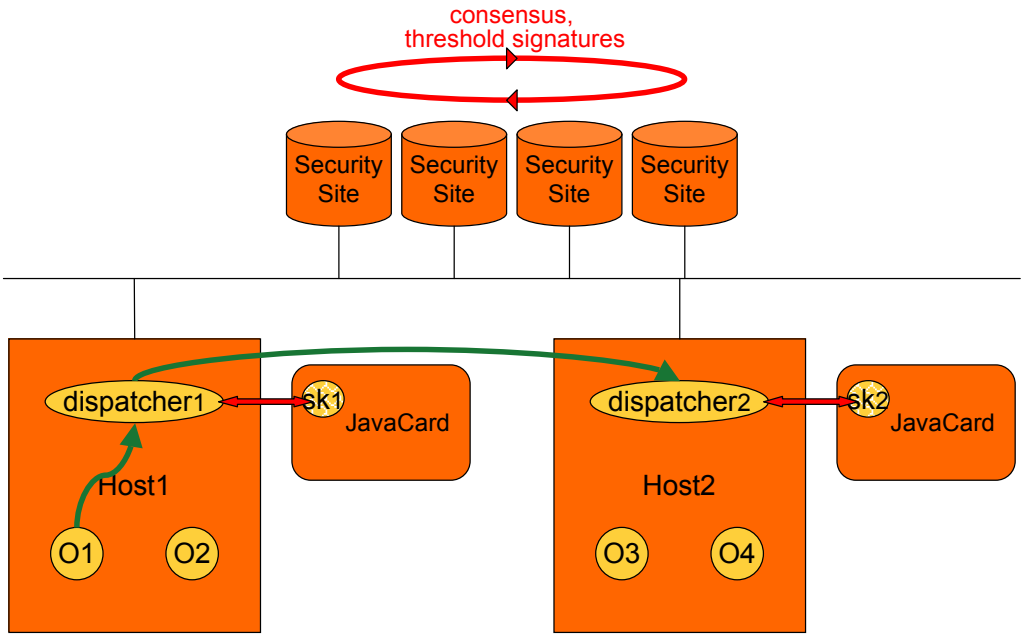


Architecture



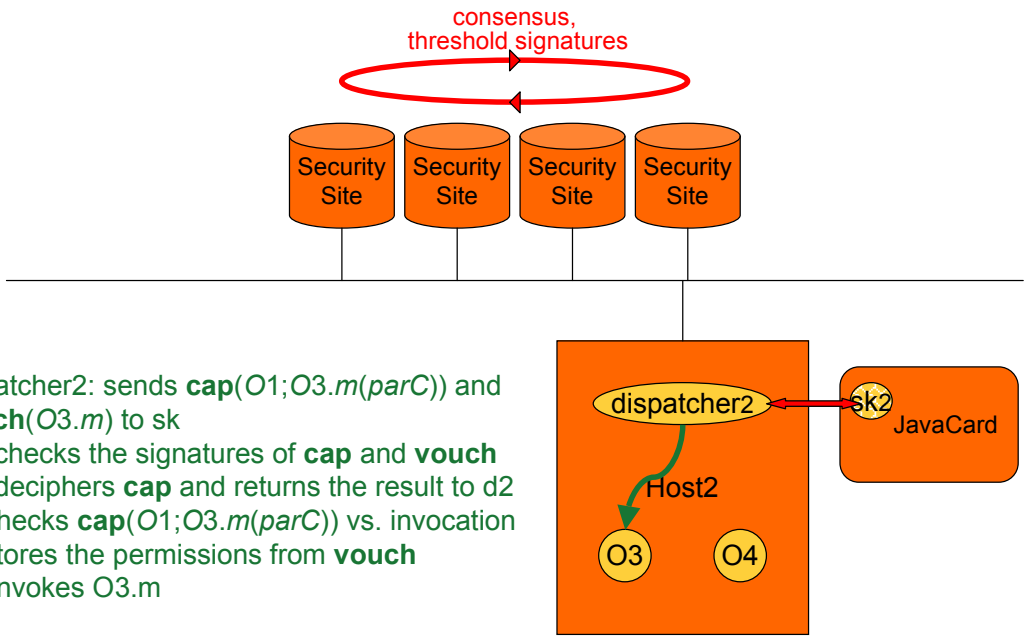
The dispatcher
- asks sk1 to check the perm. signature
- stores the capabilities and vouchers
  cap(*O*1;*O*3.*m*(*parC*));**vouch**(*O*3.*m*)
- transmits *O*3.*m* and *parC* to *O*1

Architecture



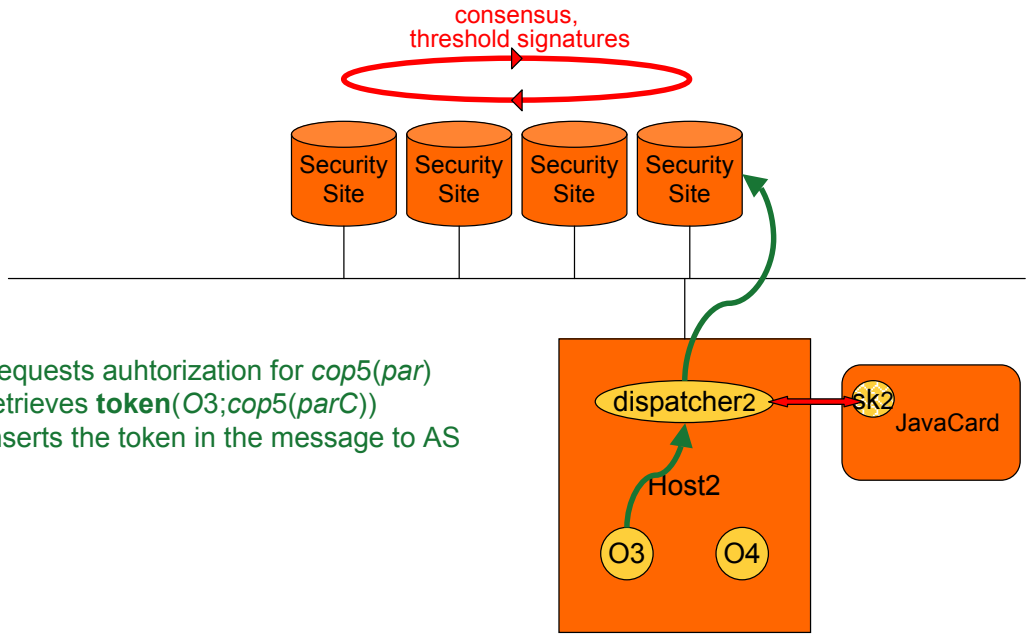
*O*1 invokes *O*3.*m* with {parameters}
dispatcher1 retrieves {**cap**(*O*1;*O*3.*m*(*parC*));**vouch**(*O*3.*m*)}, inserts it in the message to disp.2

Architecture



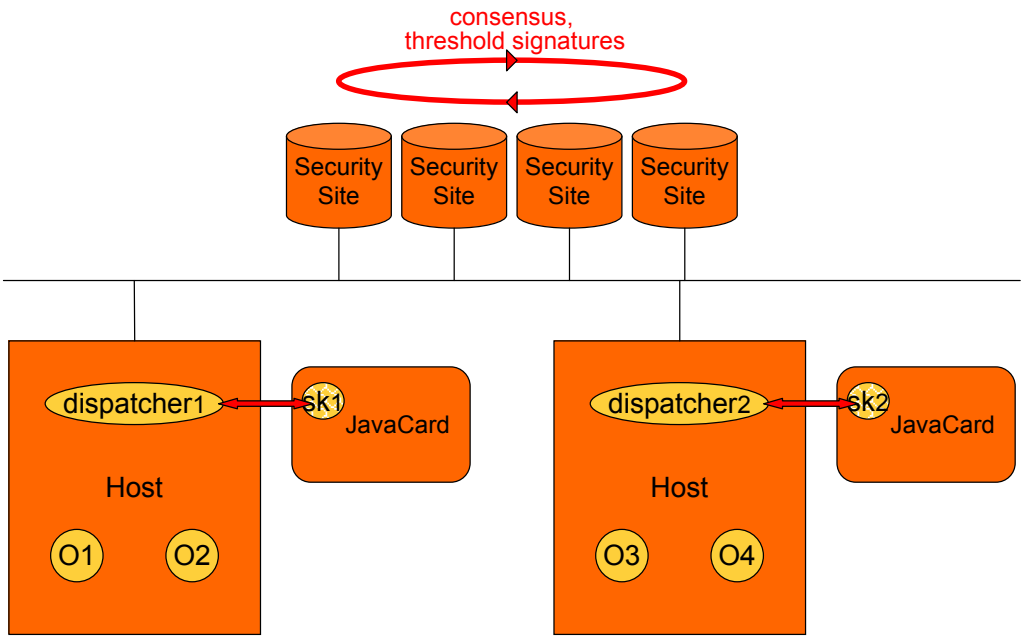
- dispatcher2: sends **cap**(*O*1;*O*3.*m*(*parC*)) and **vouch**(*O*3.*m*) to sk
- sk2 checks the signatures of **cap** and **vouch**
- sk2 deciphers **cap** and returns the result to d2
- d2 checks **cap**(*O*1;*O*3.*m*(*parC*)) vs. invocation
- d2 stores the permissions from **vouch**
- d2 invokes O3.m

Architecture

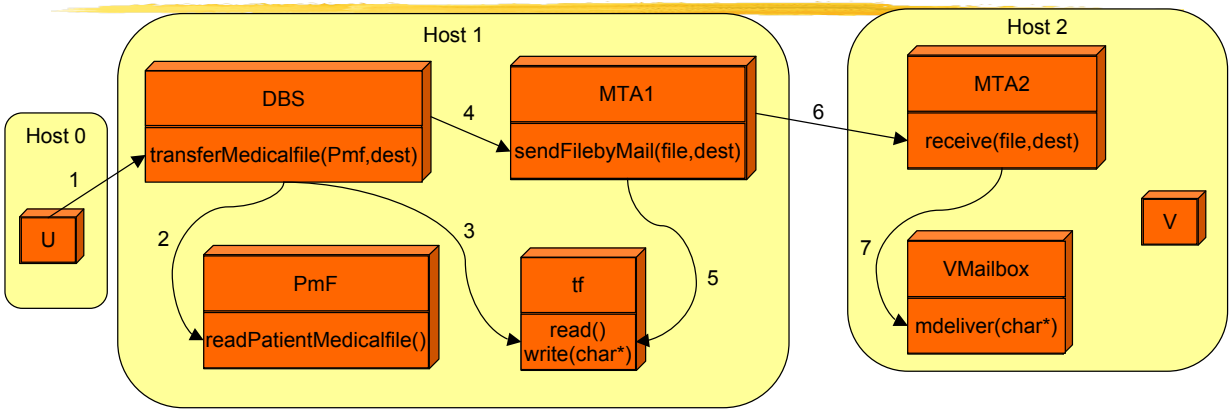


- *O*3 requests auhtorization for *cop*5(*par*)
- d2 retrieves **token**(*O*3;*cop*5(*parC*))
- d2 inserts the token in the message to AS

Architecture



Example:

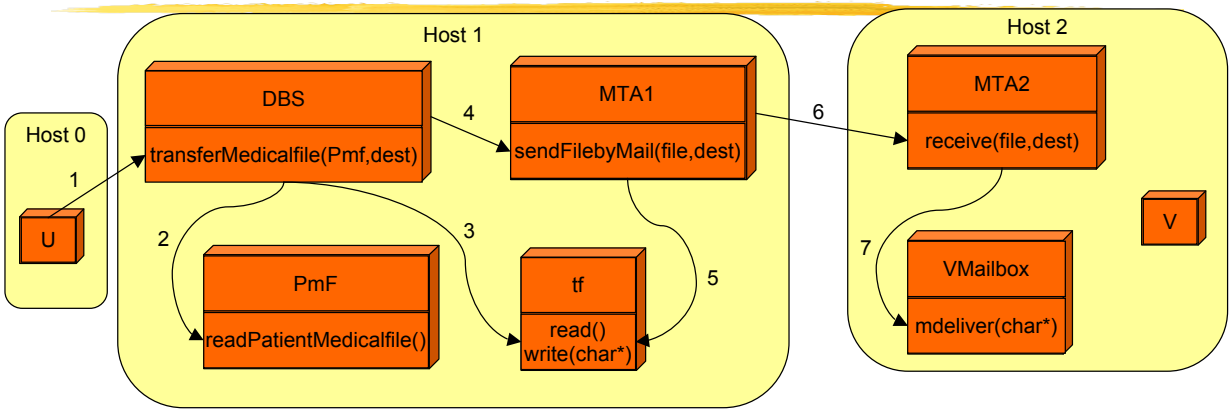


$$U \rightarrow AS : \text{Request}\left(SendPatientMedicalFile\left(Pmf_1, V\right)\right)$$

	…	*HCP Role*	*{Pmf(U)}*
U		$SPmf\left(\left\{Pmf(U)\right\}, this\right)$	*read, write(*), SPmf(this, HCPRole)*
DBS	…		
…			

$$AS \rightarrow D_0 : \left\{\begin{array}{c} U; DBS.transferPatientMedicalfile\left(Pmf_1, V\right); \mathbf{Cap}\left(U; DBS.transferPatientMedicalfile\left(Pmf_1, V\right)\right); \\ \mathbf{Vouch}\left(DBS.transferPatientMedicalfile\right) \end{array}\right\}$$

Example:

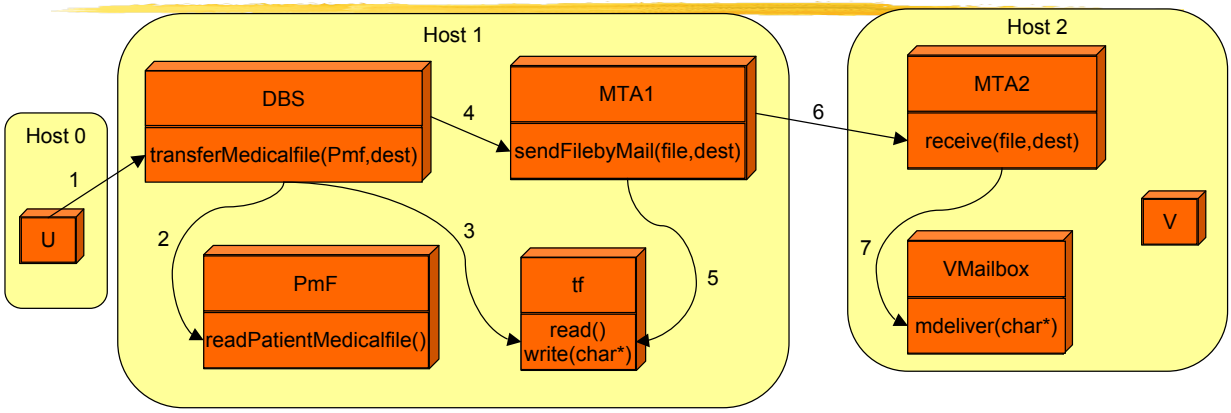


$$U \overset{1}{\mapsto} DBS : \left\{\begin{array}{c} \mathbf{RMI}\left(DBS.transferPatientMedicalfile\left(Pmf_1, V\right)\right); \\ \mathbf{Cap}\left(U; DBS.transferPatientMedicalfile\left(Pmf_1, V\right)\right); \mathbf{Vouch}\left(DBS.transferPatientMedicalfile\right) \end{array}\right\}$$

$$D_1 \rightarrow JC_1 : \left\{\mathbf{Cap}\left(U; DBS.transferPatientMedicalfile\left(Pmf_1, V\right)\right); \mathbf{Vouch}\left(DBS.transferPatientMedicalfile\right)\right\}$$

$$\mathbf{Vouch}(\text{DBS...}) = \left\{\begin{array}{c} DBS; Pmf1.readPatientMedicalfile; \mathbf{Cap}\left(DBS; Pmf1.readPatientMedicalfile\right); \\ DBS; MTA1.sendFilebyMail(*V); \mathbf{Cap}\left(DBS; MTA1.sendFilebyMail(*V)\right); \\ \mathbf{Vouch}\left(MTA1.sendFilebyMail\right) \end{array}\right\}$$

Example:

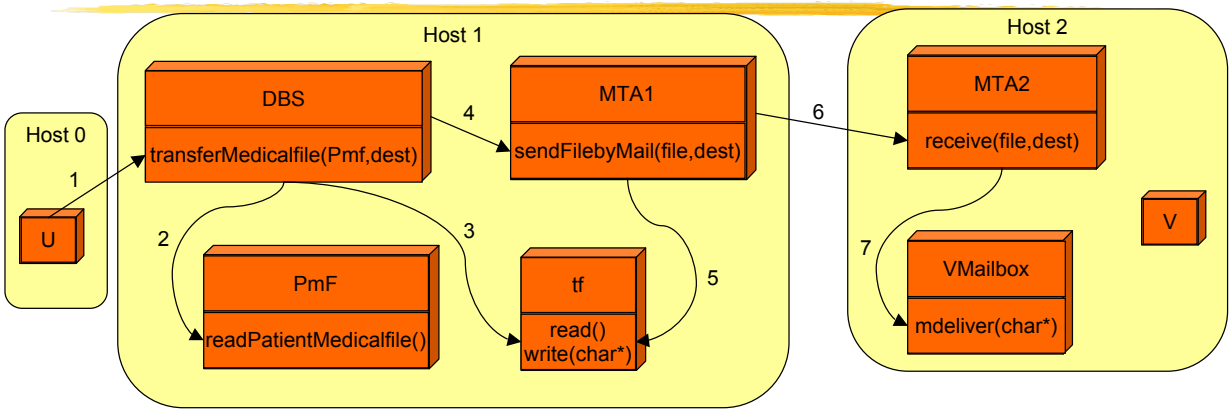


$$DBS \overset{2}{\mapsto} Pmf_1 : \left\{ \mathbf{RMI}\left(Pmf_1.readPatientMedicalfile(\ ) \right); \mathbf{Cap}\left(DBS; Pmf_1.readPatientMedicalfile \right) \right\}$$

$$DBS \overset{3}{\mapsto} tf : \left\{ \mathbf{RMI}\left(tf.write(\langle \text{content of } Pmf_1 \rangle) \right) \right\}$$

$$DBS \overset{4}{\mapsto} MTA1 : \left\{ \begin{array}{c} \mathbf{RMI}\left(DBS.sendFilebyMail(tf,V) \right); \\ \mathbf{Cap}\left(DBS; MTA1.sendFilebyMail(*V) \right); \mathbf{Vouch}\left(MTA1.sendFilebyMail \right) \end{array} \right\}$$

Example:

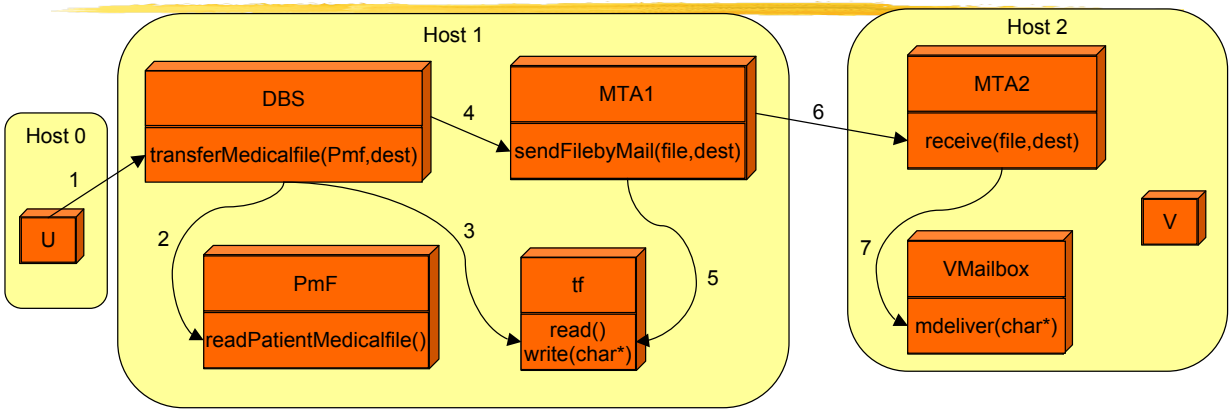


$$MTA1 \rightarrow AS : \text{Request}\left(DeliverFilebyMail(*,V) \right); \mathbf{Token}\left(MTA1; DeliverFilebyMail(*,V) \right)$$

$$AS \rightarrow D_1 : \left\{ MTA1; MTA2.receive(*V); \mathbf{Cap}\left(MTA1; MTA2.receive(*V) \right); \mathbf{Vouch}\left(MTA2.receive \right) \right\}$$

$$MTA1 \overset{6}{\mapsto} MTA2 : \left\{ \begin{array}{c} \mathbf{RMI}\left(MTA2.receive(\langle \text{content of } tf \rangle, V) \right); \\ \mathbf{Cap}\left(MTA1; MTA2.receive(*V) \right); \mathbf{Vouch}\left(MTA2.receive \right) \end{array} \right\}$$

Example:



$$D_2 \rightarrow JC_2 : \{\mathbf{Cap}(MTA1; MTA2.receive(*V)); \mathbf{Vouch}(MTA2.receive)\}$$

$$MTA2 \overset{7}{\mapsto} VMailbox : \{\mathbf{RMI}(VMailBox.mdeliver(\langle \text{content of } tf \rangle)); \mathbf{Cap}(MTA2; VMailbox.mdeliver(*))\}$$

http://www.research.ec.org/maftia/