OASIS: Architecture, Model and Management of Policy

Ken Moody

Computer Laboratory, University of Cambridge

Overview – *OASIS* : Architecture, Model and Policy

1. background to the research

   people, projects **(motivation - *EHRs* for the UK *NHS*)**

2. fundamentals of ***OASIS*** architecture

   – *Role-Based Access Control* with parameters

   – *Interoperation of Federated Services*

   – *Support for Active Security*

3. establishing a useful Model for ***OASIS***

   – ***Many-sorted First-Order Predicate Calculus***

4. database and meta-data support for distributed applications

   – *development of an active predicate store on top of **PostgreSQL***

   – *active policy management, meta-policies and verification*

5. *FUTURE WORK*

Experimenting with *OASIS*

people

- *OPERA Group* – Computer Lab, Cambridge (UK)
  - ***Jean Bacon***, ***Ken Moody*** (*Faculty*)
  - ***John H Hine*** – ***sabbatical visitor, 1999*** – VU of Wellington (NZ)
- *PhD students*
  - ***Walt Yao***, ***Wei Wang*** (*employed on EPSRC grants*)
  - ***András Belokosztolszki***, ***David Eyers*** (*independently funded*)
  - ***Nathan Dimmock***, ***Brian Shand*** (*Trust-based access control*)

research grants

- relating more or less specifically to ***RBAC***
  - **(EPSRC)** *evaluating* the use of *OASIS* for *EHRs* in the UK ***NHS***
  - **(EPSRC)** using an ***active database*** to manage ***access control policy***
  - **(EU Framework 5)** ***SECURE*** – Trust-based AC for wide-area computing

OASIS Access Control *"you've gotta ROLL with it . . "* *(pop culture)*

principals (clients?)

- *PERSISTENT* – typically a ***person*** or ***job-title*** – named by ***e.g.*** *NHS_number*
- *TRANSIENT* – a ***computer process*** or ***agent*** – named by ***e.g.*** *session_Public-Key*

scalability of *POLICY expression*

- classify ***clients*** by ***ROLE*** (parametrised?), *ROLE* ***names*** *specific to each service*
  - ***e.g.*** *doctor* , *logged-in_user ("Fred")*
  - potential for giving ***client anonymity*** if required
- specify ***control of access*** in terms of ***ROLEs*** (of *this* and possibly *other services*)
  - as held by ***TRANSIENT PRINCIPALs***
  - *each service* defines its own rules for ***ROLE*** entry

Long-lived rights for *PERSISTENT PRINCIPALs*

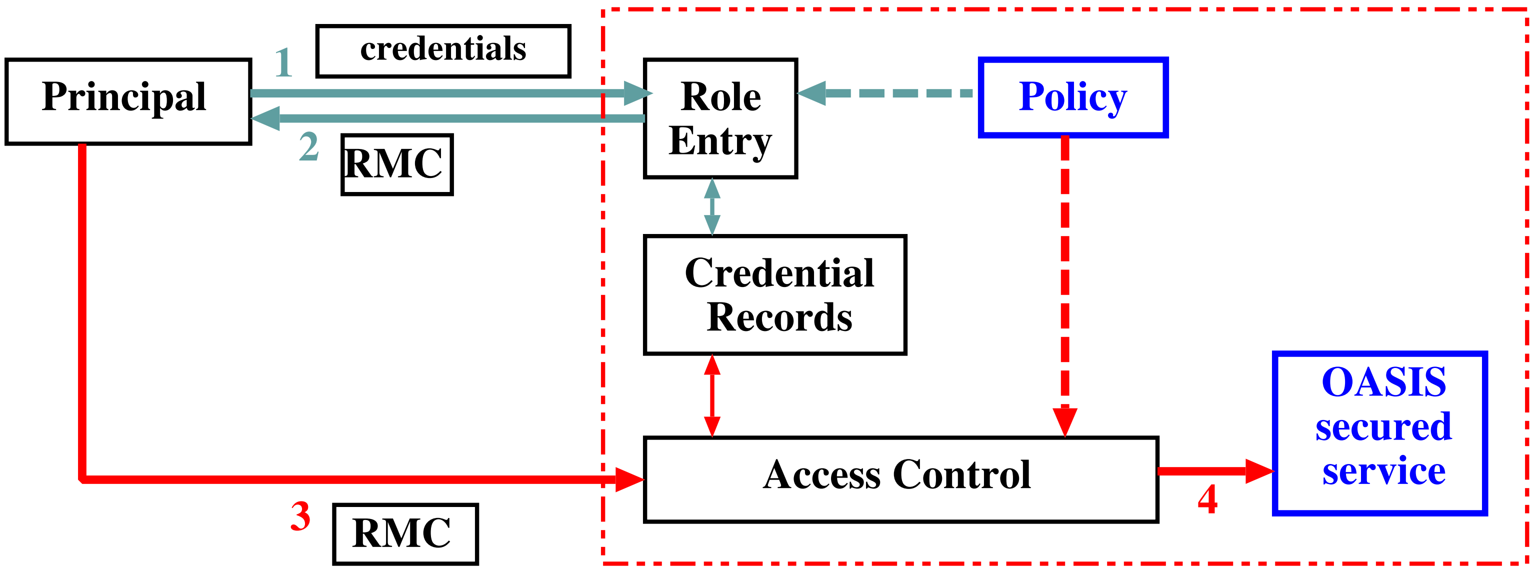
- *APPOINTMENTs* (bound to *PERSISTENT NAMEs*)
  - grant entry to a new *ROLE conditionally on*

    OTHER ROLEs held + *constraints* on their *parameters*
- administered *via* specific *ROLE(s)* (direct expression of *management policy* ?)

Managing *ROLE MEMBERSHIP* and *APPOINTMENT CREDENTIALs*

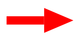
- via a *signed certificate* ("capability") , format determined by the issuing service
  - *issued to* and *managed by* a ***principal*** , ***TRANSIENT*** or ***PERSISTENT***
- a *credential record* (maintained at the issuing service)
  - asserts the *validity* of each issued certificate
  - linked to the active conditions for ***ROLE membership***
  - enables *rapid* and *selective revocation*
    - + dependent on ***asynchronous notification***

A service secured by OASIS access control

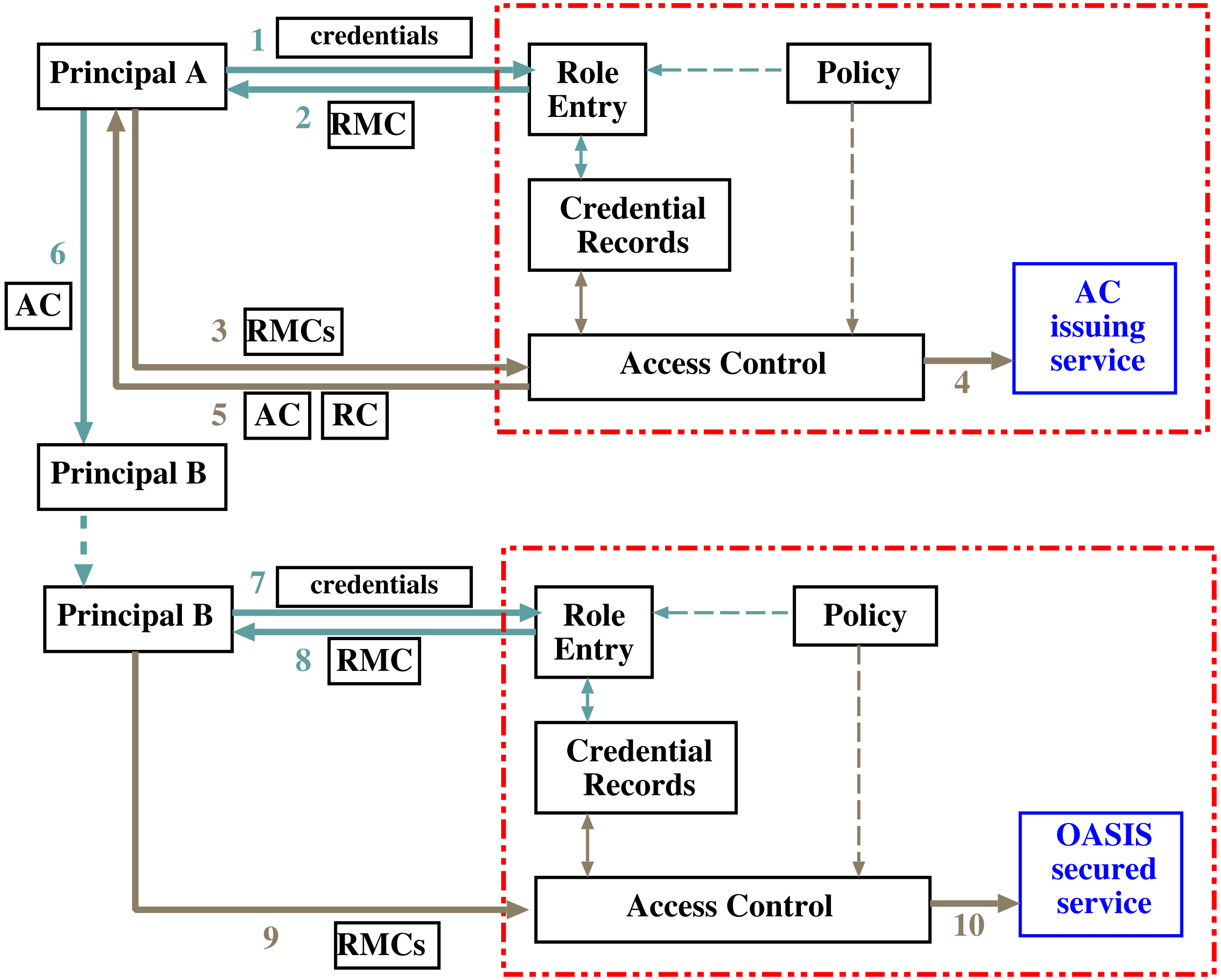


RMC = role membership certificate


= role entry


= use of service

Issuing and Using Appointment Certificates

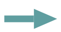


1. principal A enters role *AC-issuer*
2. RMC as *AC-issuer* returned
3. *AC-issuer* requests an AC for principal B
4. validated request passed on
5. AC and RC returned to principal A
6. principal A passes AC to principal B but keeps RC

. . . .

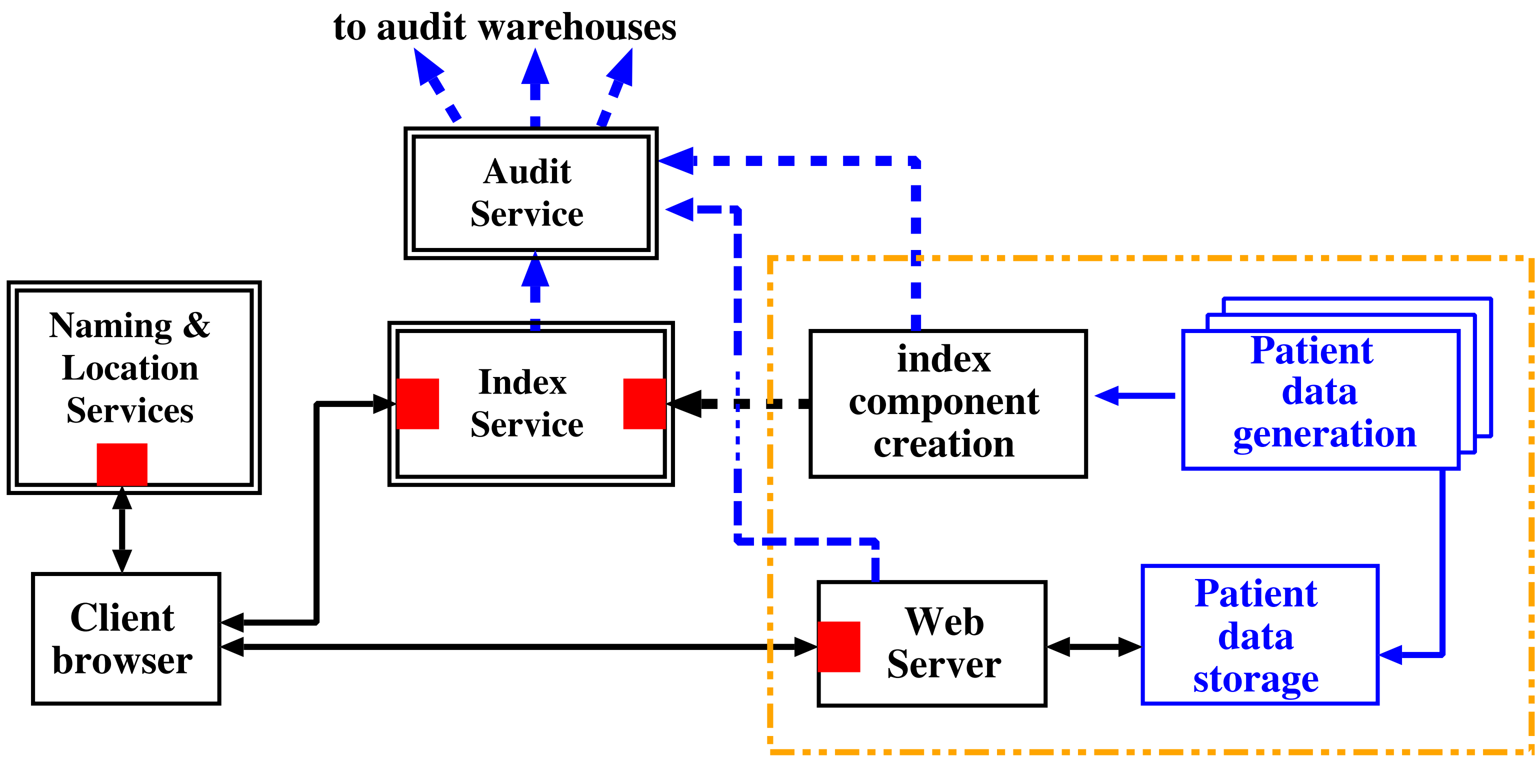
7. principal B enters a role using AC as one credential
8. RMC returned to principal B
9, 10. standard use of OASIS secured service

RMC = role membership certificate, AC = appointment certificate, RC = revocation certificate

 = obtaining and using credentials for role entry

 = use of service

Overall EHR Architecture



reliable message transport

service secured with Oasis access control

reliable, distributed, replicated national services

data-provider architecture

end systems including legacy systems

The OASIS Model

- Based on ***Many-Sorted First Order Predicate Calculus***
  - ***sorts*** correspond to the datatypes in parameter value domains
  - predicate constants are interpreted as ***access control system entities***
    - \+ ***environmental constraints*** which test context
  - rules are conjunctive (non-recursive Horn clauses)
  - ***Many-Sorted*** algebra of terms (no surprises)
    - \+ ***function symbols*** context sensitive
    - \+ ***constants*** 0-ary functions (e.g. ***current_time***)
- syntax for parameter slots depends on the predicate type and the position in the rule
  - can include ***named variables*** as parameters (modes ***in-*** and ***out-***)
  - ***variable instances*** must match during rule interpretation (unification)
  - no theorem proving required, an efficient plan can be derived statically

Predicates taking part in rule evaluation

- **Access Control System Entities**
  - ***Role Membership Certificates*** have typed parameters
  - ***Appointment Certificates*** also have typed parameters
  - ***Privileges*** (correspond to e.g. ***method invocations***)
    - granularity of ***privileges*** may be coarser

- **Environmental Constraints**
  - standard example is ***database lookup*** (use modes ***in-*** **and** ***out-***)
  - explicit predicates for testing ***time*** (various aspects)
  - for efficiency require support from an ***active platform*** (***COBEA***)
    - in order to support ***role membership conditions***
    - also helpful for caching ***authorising conditions***

Role Activation Rules

- **Syntax**

$$r_1 \ , \ r_2{*} \ , \ \ldots \ a_1 \ , \ a_2 \ , \ \ldots \quad e_1 \ , \ e_2{*} \ , \ \ldots \quad \vdash \quad r_{\mathrm{T}}$$

   – where each r_i is a ***Role Membership Certificate*** predicate
   – and each a_j is an ***Appointment Certificate*** predicate
   – and each e_k is an ***Environmental Constraint***

These are the *preconditions* (* indicates that the condition must remain valid)

   + r_{T} is the ***Target Role***

- **Interpretation**

   – r_i and a_j are simply matched against the required certificates
   – e_k invoke predicates to test ***the current context*** (e.g. ***active database***)

   – matched parameters give values for slots in the ***Target RMC***

Authorisation Rules

- **Syntax**

$$r_1 \,, \quad e_1 \,, \; e_2 \,, \; \ldots \quad \vdash \quad p_T$$

  - where r_1 is the authorising ***Role Membership Certificate***
  - and each e_k is an ***Environmental Constraint***

These are the *authorising conditions*.

  + Here p_T is the ***Target Privilege Instance***

- Interpretation
  - the ***Target Privilege Instance*** is derived from the invocation
  - parameter values are set by pattern matching from r_1 and p_T
  - can cache values of e_k with support from an ***active platform***

Aims of the OASIS Model

- **High-level goals**
  - the rules should express policy precisely, and it should be explicable
  - the model should act as a target for high-level policy languages
    - have experimented with ***Attempto controlled English***
  - the consistency of policies derived from multiple sources should be decidable
  - it must be easy to provide tools to support managers of applications
    - ***support for interoperation*** across changes of policy locally
    - via ***active predicate*** extension to the PostgreSQL DBMS
  - rule evaluation **must** be efficient (particularly for ***authorization***)
    - ***static analysis*** to establish a plan for parameter matching
    - ***caching*** of results of ***environmental predicates***
- **System-related goals – continuous monitoring of security conditions**
  - use ***snapshot semantics*** to reason about policy (no explicit transitions)
  - use ***platform properties*** to reason about the behaviour under partition

Work in progress related to the OASIS Model

- **Supporting a federation of management domains**
  - applications such as EHRs must accept policy from multiple sources
  - require tools so that applications can discover how to obtain privileges
  - require ***conventions*** for naming external environmental constraints
  - **must** check consistency of policies derived from multiple sources
    - + ***generate*** a policy synthesis automatically

- **Use of an active predicate store**
  - coordinating policy change in a federated management structure
    - + automatic generation of ***Service Level Agreements***
  - storing access control meta-data to support a ***policy adviser***
    - + for ***policy administrators*** , ***application programmers***
  - implementing environmental predicates efficiently for ***authorization***

The problems of reasoning within the OASIS Model

- **Expressive power of the computational model**
  - in general ***environmental constraints*** can express arbitrary computations
    - hence ***environmental predicates*** are not in general decidable
    - **but** support in ***active*** PostgreSQL extensions for ***binary relations***
    - ***conjunctive form*** of rules $\Rightarrow$ ***predicates*** can only ***restrict*** access
  - **need** for ***decidable*** sublanguages to express e.g. ***temporal constraints***
  - ***opaqueness*** of the binding of ***predicates*** to their ***implementations***
  - **need** for a ***formal specification*** (*assertion*) of the properties of ***predicates***
    - requires ***integration*** into the ***policy store*** technology

- **Implicit behaviour of the active platform**
  - monitoring ***membership conditions*** requires a ***notification*** mechanism
    - **mustn't be any** *side effects* on the *Access Control System*
  - validity of ***external predicates*** depends on the ***integrity*** of the network
    - ***network partition*** is detected using a ***heartbeat protocol***
  - in what sense is the procedure of ***falsification*** under ***partition*** a ***safe*** one ?

Meta-policies as a means of coordinating a policy federation

Reference: András Belokosztolszki and Ken Moody

"Meta-Policies for Distributed Role-Based Access Control Systems" ,

Proc. Policy 2002 (Monterey, June 2002), IEEE CS Press, pp. 106-115.

- **Intuition behind our approach to meta-policies** (***decidable*** and ***compositional***)
  - formalization of an ***interface specification*** at ***policy level***
    - specify ***invariance properties*** to which local managers must ***comply***
    - allow ***certification*** of participants in a ***federated application*** (***NHS***)
    - provide a ***stable framework*** to support ***interoperation*** of domains
  - **components** comprising the ***formal specification*** of a ***meta-policy***
    - ***type system*** information – ***data types*** , ***objects*** , ***functions***
    - ***access control system*** signatures – ***roles*** , ***appointments***

- **Current progress with the experimental framework** (proving ***hard!***)
  - matching ***policy instances*** against a ***meta-policy*** (checking ***compliance***)
  - managing ***service level agreements*** automatically across ***change of policy***

This talk: http://www.cl.cam.ac.uk/~km/UofHull-talk.pdf

Other talks:
http://www.cl.cam.ac.uk/~km/Active_DB-AB.pdf
http://www.cl.cam.ac.uk/~km/MW2000-talk.pdf
http://www.cl.cam.ac.uk/~km/MW2001-talk.pdf
http://www.cl.cam.ac.uk/~km/NL_policy.pdf

Computer Laboratory OPERA Group Web pages

http://www.cl.cam.ac.uk/Research/SRG/opera/publications/index.html

(all of these papers can be downloaded from the publications pages)

Research Overviews

Jean Bacon, Ken Moody, John Bates, Richard Hayton, Chaoying Ma, Andrew McNeil, Oliver Seidel, Mark Spiteri,
"Generic Support for Distributed Applications"
IEEE Computer, March 2000, pp. 68-76.

Jean Bacon, Ken Moody,
"Towards Open, Secure, Widely Distributed Services"
Communications of the ACM, June 2002, pp. 59-64.

Other papers most relevant to this talk

R. Hayton, J. Bacon, and K. Moody **ARCHITECTURE**
"OASIS: Access Control in an Open, Distributed Environment"
Proc IEEE Symposium on Security and Privacy, Oakland CA, May 1998, pp. 3-14.

J. Hine, W. Yao, J. Bacon, and K. Moody
"An Architecture for Distributed OASIS Services" Proceedings of Middleware2000,
LNCS 1795, Springer-Verlag, Heidelberg and New York, April 2000, pp. 107-123.

J. Bacon, M. Lloyd, and K. Moody
"Translating Role-based Access Control within Context" Proceedings of Policy2001,
LNCS 1995, Springer-Verlag, Heidelberg and New York, Jan 2001, pp. 107-119.

J. Bacon, K. Moody, and W. Yao (expanded from SACMAT 2001) **MODEL**
"A Model of OASIS Role-Based Access Control and its Support for Active Security"
ACM TISSEC, Vol. 5, No. 4, November 2002, pp. 492-540.

J. Bacon, K. Moody, and W. Yao **IMPLEMENTATION**
"Access Control and Trust in the Use of Widely Distributed Services"
Proceedings of Middleware2001, LNCS 2218, Springer, Nov 2001, pp. 295-310.