Credential Management in Event-Driven Healthcare Systems

Jatinder Singh David M. Eyers Jean Bacon
Computer Laboratory
University of Cambridge
Cambridge CB3 0FD, UK
{firstname.lastname}@cl.cam.ac.uk

ABSTRACT

Health processes generate data that must be both stored and shared—often across organisational boundaries. Emerging initiatives in healthcare require the dynamic formation of care teams spanning widely-distributed, heterogeneous infrastructure. These environments suit decoupled communication paradigms such as publish/subscribe. Health information is sensitive, thus access control is critically important. This paper focuses on the management of credentials in event-driven healthcare environments. We describe the integration of credential management mechanisms with a context-sensitive data control model to provide fine-grained specification of data disclosure policy.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Systems and Software

General Terms

Security, Management

Keywords

confidentaility, credential management, data control, event-based middleware, healthcare, privacy, publish/subscribe

1. INTRODUCTION

Information is central to healthcare, where it—somewhat paradoxically—must be shared yet must remain confidential. Medical professionals must collaborate with others to provide proper care, yet also must keep private the information collected as part of the care process. Information control concerns the circumstances in which it is *appropriate* to release information: on a *need-to-know*[1] basis. A primary concern is *who* receives the data, which is qualified by circumstance (*when*). Credentials provide information on principals acting in a system, asserting identity, qualifications and relationships. Therefore, credential management is paramount to data control in healthcare environments.

The English National Health Service (NHS) is currently developing technical infrastructure to support national-level health services. The current focus is on the interoperability of current health practices. Future directions in health involve provisioning care services closer to home environments. As these scenarios are particularly amenable to event-driven infrastructure, we have integrated data control mechanisms into publish/subscribe (pub/sub) environments [**?**, **?**].

This paper considers credential management in relation to event dissemination. We outline how OASIS, a role based access control model, interacts with a policy-based pub/sub middleware and proposed NHS infrastructure to control information disclosure. We describe the interplay of credentials and context in providing local environments with fine-grained control over data release. We subsequently illustrate deployment of our approach using some example scenarios.

2. HEALTH INFORMATION CONTROL

Confidentiality underpins the carer-patient relationship. Medical information is highly sensitive: its protection imposed by codes of conduct and law. However, healthcare is a collaborative environment that requires information sharing. Generally, health information may be shared where there is consent; though this may be implied. Information 'users' are typically staff (carers) that operate on information 'belonging' to others (patients). Care providers are *responsible* for sharing information with carers and organisations to ensure proper care, while respecting the confidentiality of personal information. The question for providers is: Is it *appropriate* to share information? The answer depends on context. This research contributes by allowing policy to define the circumstances for data dissemination.

2.1 Event-Driven Healthcare

Healthcare providers need to be informed of incidents—happenings of interest—as they occur. These might reflect changes in state, such as a sudden downturn in a patient's vital signs (emergency), or the actions of carers, such as prescribing a drug or ending a shift. A single incident may be relevant to many entities, perhaps across administrative domains, depending on their role in the care process; e.g. drug administration is relevant to carers, a pharmacy and NHS accountants. The NHS is currently designing a range of systems for integrating care services at an organisational level, some of which will be event-driven.

[1] To mean as appropriate given patient consent, legislation, codes of conduct and/or other circumstance. Our focus is on general infrastructure, irrespective of political standpoint.

With the ageing population, there is a push to improve the quality of care/life while reducing the burden on resources. The movement is towards the provision of care services outside of traditional care institutions (e.g. hospitals) [?]. Homecare environments are dynamic, created on demand to cater for specific aspects of a patient's well being [?]. They are very much data-driven, and therefore well-suited to event-based architectures. The management policy and involvement of service providers (entities) is customised to the particular homecare instance.

As homecare environments operate outside of a central administrative domain, infrastructure must allow *entities* (e.g. doctors, managerial staff, and secondary service staff such as researchers and auditors) to receive information relevant to their tasks. This relevance depends on a number of factors such as their role in the care process, credentials, managing organisation, in addition to patient particulars (conditions, demographics) and the current environmental state (context: vital-signs, for example). Active notification is important for ensuring data consistency and to allow rapid response to particular situations, where a single incident might be relevant to multiple entities, depending on the services required.

2.2 Importance of Credentials in Healthcare

Given the sensitivity of health information, access must be controlled. Credentials are the primary consideration in determining rights and information privileges. While many event-based scenarios are relatively anonymous, healthcare requires information regarding principals. This is reflected in the proposed NHS infrastructure, where credentials will be used to assign privilege. All NHS communications occur through the N3 network, which validates the identity of a principal through a smartcard and PIN. Role-Based Access Control (RBAC) mechanisms are then used to assign privileges to the principals. Legitimate Relationship services ensure a connection between a carer and a particular patient. Consent mechanisms are used to restrict certain information from being revealed except to those with specific credentials, in line with patient requests. Credentials provide the means on which to base decisions of information disclosure. Current design documents focus on request-based interactions, concerning general privileges. This work uses credentials and environmental state to allow granular control over event-based data transmission.

3. BACKGROUND

This section introduces the access control and event-based technologies we employ.

3.1 Access Control

Access control defines privilege. Privileges are generally a function of the *role* a principal plays within an organisation, rather than being based on their identity. Role-based Access Control (RBAC) systems provide an abstraction between the identities of individuals and the collections of privileges that they require, facilitating privilege management. Roles are essentially a form of capability; for example, they can be delegated to other individuals if appropriate.

This paper employs parameterised RBAC, where attributes may be attached to an active role. These attributes can differentiate context for the enforcement of dynamic constraints, such as to encode a relationship between a doctor and the patients they treat. When parameters are introduced to roles, an inference mechanism is required to bind attribute values in requested roles from prerequisite credentials and the environment.

3.1.1 OASIS

The *Open Architecture for Secure Interworking Services (OASIS)* is a distributed RBAC implementation using a first-order logic-based model [?] to describe policy enabling users to acquire privileges, authorising them to use services by activating appropriate roles.

OASIS roles are activated in the context of sessions. Roles and rules are managed in a decentralised manner: a distributed object is linked on an OASIS service. These services operate in an asynchronous manner, cooperating through the use of event channels, thus supporting active security features such as proactive role revocation.

Role activation rules specify how to assign a role to a user activated within a session. A role activation rule takes the form: $r_1, r_2, ..., r_{n_r}, ac_1, ..., ac_{n_{ac}}, e_1, ..., e_{n_e} \vdash r$ where r_i represents prerequisite role certificates, ac_j are appointment certificates (generally persistent) and e_k are environmental constraint predicates. Predicate expressions (preconditions) on the left hand side of the rule must be valid for a given user to activate the target role (r). If marked as a membership condition, a precondition must remain true—violation causes deactivation. Role and appointment certificates are valid if they have not been revoked. Environmental predicates most all be true at that time.

Authorisation rules assign a privilege to a role. They take the form: $r, e_1, ..., e_{n_e} \vdash p$ for target privilege p and only allow one prerequisite role r. Various environmental constraints e_k can be specified. Both role activation and authorisation rules are in Horn-clause form, and negative privileges are not supported. A set of rules define the policy for an OASIS service.

To support context-aware security, OASIS roles are parameterised. As discussed above, policy rules may contain environmental predicates. These allow rule computation to incorporate data from outside the OASIS environment. They can also be used to bind role parameter values. We have not included the role parameters, nor details of delegation in this introduction to OASIS. See [?] for formal details.

3.1.2 Access Control in Healthcare—NHS

Access control is central to healthcare. NHS documents categorise access control into roles and relationships. Roles tend to encapsulate an individual's long-term access rights. The NHS uses RBAC to define the activities available to a principal. This is defined on job-role, optionally qualified by a work area. Role allocations are centrally defined, where service providers (domains) may customise (augment) tasks to better suit the local environment. Relationships establish a connection between the principal and data, based on circumstance. The NHS defines a Legitimate Relationships service to map care providers/entities to patients through a grouping structure (workgroup), which is intended to be automatically constructed as part of the care process.

Data control definitions must reference both defined organisational processes (job-roles and activities) and data particulars. The NHS model links Legitimate Relationships to RBAC, where the roles define general privilege and the relationship defines access to a (patient's) data record. Access must also consider consent and patient requests.

This work aims to provide more flexible definitions of context to give fine-grained control over the event-based information dissemination.

3.2 Content-Based Publish/Subscribe

An *event* provides information on an occurrence, or incident, in a system. Each event is an instance of an event type. Typically event type definitions specify a name, and provide a set of (attribute name, attribute data-type) pairs.

Pub/sub [**?**] communication models facilitate multi-way event delivery. Principals take the role of publishers and/or subscribers that connect to the pub/sub middleware in order to communicate. Subscribers register their interest in events of particular types through a subscription that may specify additional conditions (filters) on event content.

Notification is the process by which subscribers receive events that match their subscriptions. Distributed pub/sub middleware uses brokers to route events publishers to subscribers. This paradigm is appropriate for highly collaborative environments, as information is shared according to content, without concerning applications (users) of routing or addressing specifics. The producer/consumer decoupling is useful even when principals are 'known': publishers are not burdened with knowing every potential information sink.

Database Middleware

We have integrated content-based publish/subscribe functionality into a database environment (PostgreSQLPS [**?**]). The motivation is that messages typically contain information that requires storage. Coupling messaging and database functionality facilitates the management of security groups, configuration, data distribution and data replication. Common event/data-type definitions simplify persistence, data replication and rule definition. Further, the messaging substrate can benefit from database functionality such as query languages, transactions, relational integrity, query optimisation engines, and so forth.

3.3 Fluents

Fluents, as defined in Event Calculus [**?**], provide Boolean values for aspects of state. Fluent values change in reference to event instances. Fluents can be parameterised: `emergency(`*patient_id*`)` might represent whether a particular patient is in an emergency situation. In our implementation, fluent values are recorded in database tables. Fluents are used to define the circumstances for data release, and by subscribers to define their interest.

4. DATA CONTROL MIDDLEWARE

We have extended PostegreSQLPS functionality to include broker-level enforcement of data-control policy. This is an obvious point for policy integration [**?**, **?**], as the messaging system, through the storage mechanism, has access to rich representations of state (context). If messaging is controlled through the database, it facilitates storage being regulated by policy. Further, the level of indirection between applications helps to ensure policy adherence.

Data control rules are loaded into brokers to control data transmission. Rules come in two forms. Those that may react to, or affect the content of event data in transit are called *transformation rules.* Rules that are triggered on management events (e.g. clients making subscriptions to the database-messaging infrastructure) are called *client restriction rules.* Policy rules are local to each broker.

Transformations: Transformations allow more than binary access control, as information can be customised to circumstance. Transforms may enrich, degrade or produce new events that are related to the original event in some application-specific manner. This allows event-content to be customised for release as appropriate in a given situation. Since transformation functions execute inside the database, they can use system-level context, stored data, database functions and external services. A transformation rule T, is a tuple of the form: $(i_{pt}, e\_type, P, C, f(m))$. The *interaction point* (i_{pt}) is the point in the messaging process where the transformation is performed: on event publication or notification. The *e_type* refers to the type of the incoming event on which the transform occurs. P is a (potentially empty) set of contextual predicates that further refine the circumstances for the transformation, referencing event content, fluents and other aspects of state accessible to the local broker. C is a set of credential fluents that refer to user specific context: publisher or subscriber, depending upon the interaction point. Credential fluents typically reference an OASIS role, see Section 5. The function $f(m)$ takes a message m as input and returns the transformed message m'. When an event m reaches an interaction point, the broker searches for applicable transformation rules through evaluation of the rule's contextual predicates (P,C). If there is a match, the transform $f(m)$ is applied and m' moves to the next stage of the messaging process.

Client restrictions: Our model provides two types of restrictions to limit the data delivered to a subscriber:

Subscription Authorisation. Policy defines when to authorise a subscription request. A subscription authorisation rule can be represented by a tuple (e_type, C, P) where, as in previous definitions, *e_type* is the event type, C is a list of credential fluents that the requester must satisfy and P is a set of contextual predicates. A subscription request to *e_type* is authorised if the circumstances satisfy the C and P predicate sets. A change in fluent state for an authorisation rule causes a subscription request to be re-evaluated. The subscription may persist, be deactivated, or subject to modified restrictions.

Imposed conditions. Similar to subscription filters, imposed conditions also serve to filter the delivery of events, except that they are specified by policy administrators. They are imposed silently: the subscriber is unaware of any restriction. This avoids disclosure of any sensitive information encoded in the restriction policy itself. Each imposed condition consists of the tuple: (e_type, C, R) where *e_type* is the event type to which the filters apply, C the set of credential fluents, and R the set of restriction predicates. Like contextual predicates, restriction predicates can refer to event content, fluents and system context. On subscription, the applicable imposed conditions are determined for the event type and current context. The set of restriction predicates (R) for each rule is added to the set of subscription filters: for notification all must be satisfied.

Client requests—Mandatory attributes. Subscribers may specify conditions stating their preference for receiving particular information. This serves to filter the events they receive. Generally, filters are at the subscriber's discretion. We introduce *mandatory attributes* (MAs) that force

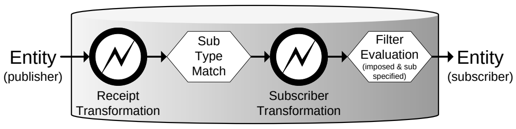

Figure 1: Broker policy enforcement points.

a subscriber to include certain attribute values as part of their subscription request. This allows evaluation of *value-based restrictions* on subscription. For example, MAs can force subscription requests for `treatment` events to include the *patient_id*. This allows the resolution of conditions upon subscription, here ensuring a relationship between the subscriber and the patient. Without MAs, enforcing a relationship becomes cumbersome: it means imposing conditions which are evaluated on *each* `treatment` event instance for *all* patients. MAs improve efficiency by reducing the number of, possibly expensive, filter evaluations. Safety is improved: policy/contextual errors denying a subscription become immediately evident. Further, evaluating constraints on subscription limits potential 'data-leaks' associated with an event type—a patient-specific subscription reduces the chance of releasing information concerning other patients.

4.1 Conflict Management

Policy rules are defined with activating conditions, therefore multiple rules may apply to a given event or request. Although appropriate in many cases, in some circumstances policy rules will *conflict*. We do not attempt to automate conflict resolution, as this may be dangerous in the complex world of healthcare data management [?]. Instead, we use the database environment to provide query mechanisms that detect situations of *potential* conflict. This allows administrators to author around conflicts, or to define resolution strategies (ordering or overriding). See [?] for details.

4.2 Policy Process

The policy evaluation process for a broker is as follows: on system-start or policy change, all event transformation (publication) policies are loaded. On a subscription request the applicable subscriber authorisation rules are determined, by matching the predicates in the rule definitions to current context: environmental state and the principal's credentials (OASIS roles). Depending on the rule, evaluation might dictate the inclusion of mandatory attributes. If no authorisation polices are applicable for the event type, the subscription request is denied. Otherwise, the relevant set of notification transformation rules are activated and any applicable conditions imposed are added to the subscription filter. Each event publication is subjected to any applicable publish transforms. The types of the resulting events are matched against the subscription types and pre-transform conditions, after which the applicable notification transformations are applied. A resulting event instance is delivered to the subscriber if it satisfies the filter predicates. The fluents used by activation rules are monitored so that policy rules are sensitive to contextual changes. This process is illustrated in Figure 1 and described with an example in §6.2.

5. CREDENTIAL MANAGEMENT

Credentials are central to our model, bringing to the messaging framework an important aspect of application context. As described, policy rules relating to subscription data are defined with predicates referencing the credentials of the subscribers. In this section, we discuss the integration of various system components.

The integration of the pub/sub and database systems gives a broker a rich representation of local state, both derived (e.g. queries) and through state transitions (events). Thus, brokers can be *active*, i.e. responsive to contextual change. While credentials represent an aspect of context, they specifically assert a characteristic about a principal. In large-scale environments, (aspects of) credential management tends to be centrally managed. Further, allocations tend to be stable: job roles change infrequently. Workgroup allocations change more often but are relatively static when compared to local environments that represent and react to many dimensions of state. As such, the separation of credential-based fluents (C) from other contextual predicates (P) allows data control rules to define the information accessible by a particular user, qualified by environmental state. This helps clarify the targets of a policy rule, the circumstances in which it applies and the system responsible for the predicate evaluation.

The combination of an established credential model with a contextually-aware data storage and distribution model facilitates enforcement of fine-grained data control policies. We use OASIS to manage the credential aspect, leaving our pub/sub database framework to handle other aspects of context. OASIS integrates into the pub/sub control model by providing the values for the credential fluents: an active OASIS role corresponds to an active credential fluent. The relationship between OASIS and the pub/sub middleware is reciprocal. The pub/sub service uses OASIS to manage user-credential allocations, while OASIS uses the pub/sub database system to be informed about event-driven contextual changes that may impact membership conditions.

5.1 Integration with NHS Credentials

Our work involves bringing health/medical context into the middleware to control information dissemination. Such mechanisms must be compatible with current and future NHS infrastructure. We integrate NHS credentials into our definitions through OASIS mechanisms.

NHS RBAC definitions. Role allocations are centrally defined by the NHS. Those holding a particular job role, perhaps qualified with an area of work, will be authorised to undertake a defined list of activities. As an event type encapsulates a specific semantic, activities may readily be associated with particular events. Role definitions are relatively static, with a bureaucratic process for any change in definition. Service providers may (only) add activities for a particular role, to better suit local practice. OASIS brings NHS role allocations into the local domain, allowing customisation to the environment. NHS roles can be represented using OASIS appointment certificates.

In an attempt to improve the manageability of credentials for local systems, the NHS reduces the degrees of freedom in its RBAC definitions by limiting staff to holding one active role at a time. This comes at the cost of expressiveness. Our approach gives local domains the flexibility to build more descriptive definitions than those provided centrally, through the use of OASIS prerequisite and parameterised roles, allowing access policy to be customised as appropriate for the local environment and its representation of contextual state.

Electronic Staff Register (ESR) and N3. The NHS provides a centrally administrated staff register, managing information including staff NHS IDs, domains of employment (provider), default role allocations, contract and leave dates, etc. The ESR acts as the definitive source for NHS staffing information, providing information on which to base rule definitions. At present, updates are sent as regular batches—evidence that staff related privilege changes at a slower rate than aspects of environmental state. The closed N3 network requires a login (smartcard and PIN).

Legitimate relationships define associations between carers and patients to ensure that staff only access records concerning patients with whom they work. Relationships are defined through workgroups, which function like patient-specific roles: e.g. a patient might be in a ward for several days, while the nurse managing that ward is employed for years. Relationships can be represented in this model as an active parameterised OASIS role. If there is no workgroup mapping (relationship) between a carer and a patient, or if a relationship ceases to exist, the role will be deprovisioned and the subscription dropped.

OASIS can encode privilege assignments in line with NHS infrastructure proposals and implementations. The current focus of the NHS is on system and process integration and request-based access. Future care is moving towards intermediate services: a highly event-driven environment. To properly manage data disclosure, it is necessary to account for dynamic aspects of environmental context, in addition to issues of user-credential management. Our coupled OASIS-pub/sub middleware gives fine-grained control over the circumstances in which data is disclosed.

6. SCENARIOS

This section demonstrates the ability of our model to represent emerging NHS practices.

6.1 Background

John is a post-operative cardiac patient being cared for in his home environment. After discharge from hospital, he is monitored through a wearable sensor system that measures aspects of well-being including heart and respiration rate, temperature and movement (acceleration, pedometer and loss of balance). Location is monitored through a GPS receiver and sensors in the home that monitor location at a room-by-room-level.

Several entities provide care services for John. Dr. Nick, his case manager, is generally responsible for John's recovery. Homecare nurses will periodically visit John's home to perform various procedures such as assessing levels of pain, mobility, and prescribing drugs as appropriate. Drugs are dispensed via the Electronic Prescribing Service (EPS). Controlled drugs (e.g. morphine) may be used in situations of acute pain. The supply of controlled drugs is monitored by an auditor [**?**]. The Accident and Emergency (A&E) Department must be notified in emergency situations.

6.2 Data Flows

The intermediate care scenario outlined in the previous section encapsulates the basic functionality concerning the homecare of a patient. Data will flow across domain boundaries as part of the homecare process. Most information concerning John moves through the Primary Care Domain (PCD)—the domain (organisation) managing John's care—which regulates the flow of information regarding John to other care providers. This is so that the main data store for the patient's information is controlled by the domain responsible for their care. This is in line with NHS notions of local control and responsibility, and because it is likely that the PCD will acquire all the data it can from the home environment.

Sensor data arrives at various intervals, based on time and context, providing summary information regarding John's state. The PCD will store this information, which Dr. Nick may query. However, he may also subscribe to be *notified* of certain events as they occur. For example, prescribing a drug might indicate a complication; emergency situations require immediate attention. Nurses receive temporary privileges when inside the home to subscribe to vital sign information, though they primarily act as publishers, performing actions relevant to various parties. All prescriptions pass through the EPS, which handles dispensing aspects and the routing of information to pharmacies. Clinical governance requires certain information to flow to auditors. An auditor may be responsible for a region, thus a single subscription might result in data from multiple domains.

Controlling enterprise event flow. A single `prescribe` event has relevance to different entities. The EPS must receive information on all prescriptions from all home environments. However, this only includes information of patient demographics, drug and prescriber information, without care record information (observations/notes). This is effected through a transformation function that converts `prescribe` events into legal prescriptions [**?**].

Auditors must monitor the supply of controlled drugs. The audit is prescriber focused: the auditor should not receive patient details unless the prescriber is under investigation. A restriction rule ensures the auditor only receives information for controlled drugs (`ControlledDrug(`*drug_id*`)`). A transformation function (`Drug_Audit_Filter()`) removes patient details from the `prescribe` events. This transform is conditional on the `Prescriber_Investigation(`*nhs_id*`)` fluent holding, which represents whether the prescriber is under investigation. If the fluent holds, the transformation does *not* execute, and the auditor receives patient specifics.

Privilege management and sensor streams. Data access privileges are dependent on context. John requests that only snapshots of his current physiological state, taken at various intervals, are delivered to carers. This is akin to routine checks by a nurse in a hospital. However, live data streams may be necessary to accurately detect trends. As such, restriction rules can effect snapshots *for subscriptions*, e.g. propagating every 50th sensor reading, while the complete data-stream may be stored but only accessible by queries in particular circumstances, e.g. if complications develop. For reasons of privacy, vital-sign events may be perturbed. An example might involve degrading location data to the values `home` or `not home`. This information assists in the interpretation of vital sign information, but does not precisely disclose John's location or movements. This is effected through a transformation rule that perturbs location information except in emergency situations (`emergency(`*patient_id*`)`) where detailed location data is required for proper interpretation and response.

The process for subscribing to sensor data is shown in Figure 2, and consists of the following steps: **(1)** Dr. Nick connects to the broker and issues a subscription request for `Vital_Sign` events, where *patient_id* = `P_144` (John's ID). Communication passing through the NHS N3 network asso-

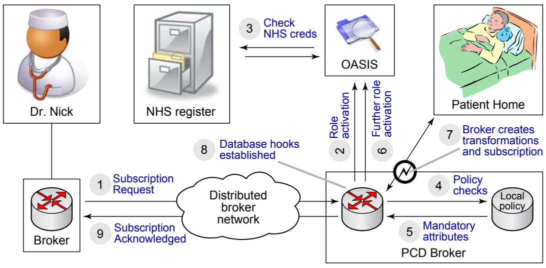

Figure 2: Steps involved in establishing a homecare sensor event subscription.

ciates a centrally-allocated employee ID with all principals: Dr. Nick is employee `NHS_6639`. **(2)** Dr. Nick's identifier is passed to the OASIS service. His identifier begins an OASIS session via an initial role activation, or identifies his existing session. **(3)** In the case of initial role activation, a prerequisite will be that the external NHS credential services indicate that Dr Nick is a `Doctor`. **(4)** The local policy cache is then searched for subscription authorisation rules through which a `Doctor` can subscribe to `Vital_Sign` events. **(5)** Policies are returned that require a mandatory attribute. In this case a *patient_id* must be included with the subscription. **(6)** The extra attribute causes a further OASIS role activation to be required: a parameterised role `Treats_Patient(NHS_6639, P_144)`. Activation of this role uses an environmental predicate to query the NHS relationship service to establish a relationship (via a workgroup). This rule also requires a `Doctor` prerequisite role. **(7)** Assuming that this parameterised OASIS role is successfully activated, and the subscription activation rule is satisfied, the broker establishes the subscription. This involves evaluating rule activation conditions to determine whether any restriction and transformation policies apply. Here there is a transformation rule to execute `scrambleLocation()` when the `not Emergency(P_144)` fluent holds. **(8)** The relevant active hook rules that interact with the messaging system are created to effect this policy. **(9)** Finally, Dr. Nick receives (positive) acknowledgment of the subscription request.

This procedure authorises Dr. Nick to subscribe to `Vital_Sign` events concerning John, through analysis of credentials and his relationship to the patient, and sets up the transformation to perturb location data on delivery of a `Vital_Signs` event in non-emergency situations. If the `Emergency(P_144)` fluent changes, the transformation rule will not apply, so events will be unperturbed and contain detailed location data. As `Treats_Patient` and `Doctor` are both membership conditions of the activation rules, any change in credential status results in subscription termination.

The separation between the local implementation specifics of a high-level NHS policy, personal credentials and the fluents within the pub/sub system, supports flexible, responsive data dissemination control. For example, nurses coming into the home environment might be outsourced carers, but upon entering the home environment they are allocated the privileges similar to those of hospital nurses. When they enter the home environment, they can request to subscribe to an event stream. The activation process is similar to the one shown above, except that the relationship with the data requires OASIS to validate the staff member's association with the outsourcing organisation to provide the `Treats_Patient(`*nhs_id*`, `*patient_id*`)` parameterised role. Given that a `Nurse` is a prerequisite role, the subscription authorisation policy involves an extra fluent check, that the nurse `IsPresent(`*nhs_id*`, P_144)`. This would give the nurse the right to subscribe to sensor stream events only when present in the home-care environment. When she leaves the home, the fluent `IsPresent` ceases to hold and her subscription is terminated.

7. CONCLUSION

This paper presents the credential-focused aspects of a middleware approach for controlling dataflow. We have focused on how event-based data-control mechanisms can integrate with the credential services of real-world, large-scale health infrastructures. We have explored the way in which credentials and dynamic state are managed in a manner appropriate for NHS applications, within a unified database and messaging middleware. Credentials, roles and fluents are defined and managed by the NHS, OASIS RBAC and our pub/sub database framework, respectively.

Our design decisions were motivated by the links between the technologies presented and NHS design documents. We have described the way that our framework implements data control through context-sensitive transformation and client restriction rules. This framework provides dynamic, flexible and highly expressive event-based data control mechanisms appropriate for the stringent access control requirements of healthcare data. We have developed much of the middleware infrastructure, and are presently evaluating the impact of a distributed broker network on data control.

8. REFERENCES

[1] Jean M. Bacon, Ken Moody, and Walt Yao. A model of OASIS Role-Based Access Control and its support for active security. *TISSEC*, 5(4):492–540, 2002.

[2] Ritu Chadha. A cautionary note about policy conflict resolution. In *MILCOM*, pages 1–8, October 2006.

[3] The Controlled Drugs (Supervision of Management and Use) Regulations 2006 (UK).

[4] Department of Health (UK). Supporting People with Long Term Conditions. A NHS and Social Care Model to support local innovation and integration, 2005.

[5] P. Eugster, P. Felber, R. Guerraoui, and A. Kermarrec. The Many Faces of Publish/Subscribe. *ACM Computing Surveys*, 35(2):114–131, 2003.

[6] R.Kowalski and M.Sergot. A logic-based calculus of events. *New Generation Computing*, 4:67–95, 1986.

[7] Royal Pharmaceutical Society of Great Britian. Medicines, Ethics and Practice: A guide for pharmacists and pharmacy technicians), July 2007.

[8] Jatinder Singh, Jean Bacon, and Ken Moody. Dynamic trust domains for secure, private, technology-assisted living. *ARES*, pages 27–34, 2007.

[9] Jatinder Singh, Luis Vargas, and Jean Bacon. A Model for Controlling Data Flow in Distributed Healthcare Environments. In *Pervasive Health*, pages 188–191, 2008.

[10] Jatinder Singh, Luis Vargas, Jean Bacon, and Ken Moody. Policy-based information sharing in publish/subscribe middleware. *POLICY*, pages 137–144, 2008.

[11] Luis Vargas, Jean Bacon, and Ken Moody. Event-Driven Database Information Sharing. In

British National Conference on Databases (BNCOD), pages 113–125, 2008.