Privacy Technologies for Controlled Information Sharing in Coalition Operations

Karen Myers[*], Tim Ellis[*], Tancrède Lepoint[*], Ronald A. Moore[^], David Archer[#], Grit Denker[*], Steve Lu[†], Stephen Magill[#], Rafail Ostrovsky[†]

[*]SRI International
Menlo Park, CA
{<firstname>.<lastname>}@sri.com

[^]Pacific Science & Engineering Group
San Diego, CA
ronmoore@pacific-science.com

[#]Galois, Inc.
Portland, OR
{dwa,stephen}@galois.com

[†]Stealth Software Technologies, Inc.
Los Angeles, CA
{rafail,steve}@stealthsoftwareinc.com

Abstract

Information sharing among coalition partners must balance the benefits that can accrue from improved coordination with the risks of releasing information that ideally would be kept private. We consider how advanced privacy technologies can enable improved information sharing among coalition partners by both providing increased control over how information is used or released, and enabling principled characterizations of the impact of individual and cumulative sharing activities. We describe this work in the context of a humanitarian aid and disaster relief scenario, showing how the technologies can enable significantly increased and informed sharing.

1. Introduction

Information sharing is a major challenge for coalition operations. Coalitions can range in composition from single-nation, inter-service or inter-agency teams to large, multi-national groups augmented with non-governmental organizations (NGOs) and corporations. Members can range from close allies to infrequent collaborators to adversaries or competitors. Furthermore, these relationships can change abruptly, underscoring the need for flexibility and adaptiveness.

Effective coordination with partners can require the intentional release of information that ideally would be held private, given the anticipated benefits that can result. However, information security mechanisms developed for the military have been designed to impede rather than facilitate sharing, due to concerns over unintended consequences of information releases. The

cryptography community has made significant strides in recent years in developing advanced technologies that can be leveraged to safeguard privacy (for example, see the description in [Archer et al., 2016]). This paper describes an exploration into how these types of technologies can be employed to enable informed and controlled information sharing within coalitions.

To ground our work, we have been considering a use case rooted in humanitarian aid and disaster relief (HADR). We chose to focus on HADR for several reasons. First, it is representative of real-world, multi-nation coordination tasks that happen on a regular basis. Second, it encompasses privacy concerns at multiple levels: individuals, intra-organization, and inter-organization, with organizations spanning nation-based, commercial, and NGOs. Third, it supports a range of challenging privacy problems, including access to both structured and unstructured data, and multi-party coordination tasks that require sequences of information exchanges and joint computations.

At the heart of our approach is a platform called PRIME (Privacy-preserving Information Mediation for Enterprises). PRIME provides privacy management by integrating a set of privacy controls, comprised of security mechanisms and policy setting capabilities for data owners, as well as various analysis tools for measurement and prediction of information leakage. Beginning with requests from authorized users, PRIME manages request processing using permitted data and services to provide a response that satisfies the requester's needs while remaining in compliance with the privacy requirements of data owners. Many tasks within coalitions involve ongoing, temporally extended coordination. For this reason, our approach adopts a *process-oriented* perspective, performing selection and configuration of workflows for responding to information and coordination requests while taking into account privacy implications for their execution. Figure 1 provides a graphical depiction of this concept.

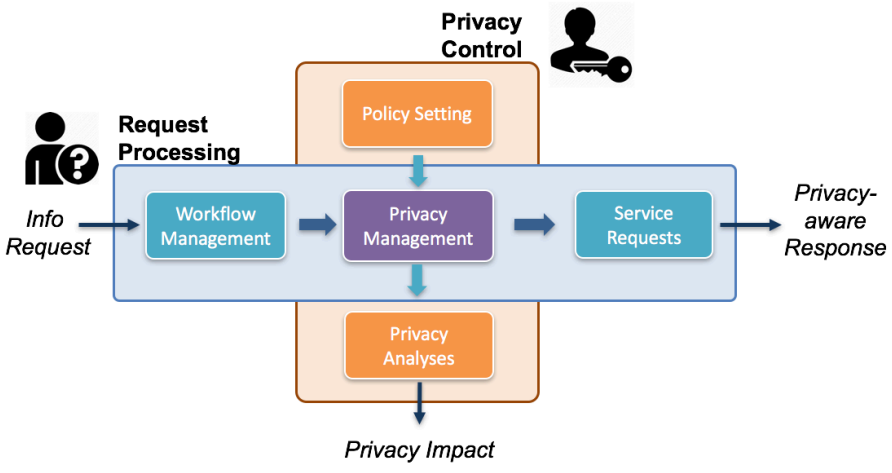


Figure 1. PRIME concept: privacy-aware request processing

The privacy technologies used currently within PRIME include *searchable encryption*, *secure multi-party computation*, *function secret sharing*, and *differential privacy*. To complement those technologies, PRIME also leverages information-theoretic characterizations of what is being revealed through data releases, thus enabling informed decisions regarding the implications of sharing.

Much attention in the security community and the media is focused on adversaries that access data systems without authorization, for example by stealing credentials or exploiting software vulnerabilities. In this work, we focus on adversaries that access information solely through authorized channels. We note that such adversaries may still exploit that legitimate access to obtain information that an owner would prefer to keep private, for example by performing more data accesses than typically expected, or by inferring connections between data that are not explicitly related.

The remainder of the paper is organized as follows. Section 2 summarizes our HADR use case. Section 3 presents the core our PRIME information mediation platform, covering policy, workflow management, and services. Section 4 describes the privacy technologies that incorporated into PRIME to date and provides examples of their use. Section 5 describes directions for future work. Section 6 presents our conclusions.

2. Use Case: Humanitarian Aid/Disaster Relief (HADR)

Within our HADR scenario, a typhoon has caused extensive damage across a set of countries in the Pacific and relief (food, medicine, water, fuel, shelter, security, etc.) is needed in a number of communities. Adding to the complexity of the situation is the outbreak of a deadly and highly infectious disease that begins working its way through the populace.

The use case focuses on three fictional nations (Cebu, Bohol, Siquijor) that have sustained significant damage. To enable the use of state-of-the-art mapping and visualization capabilities, we elected to ground these fictional nations in real-world geographic entities, namely islands in the Philippines (see Figure 2). Each nation has five communities, which are marked by push-pins in the map.

HADR activities are being organized at multiple levels. Response Coordinators have been defined for each community and nation; there is also an over-arching International Response Coordinator to address cross-nation issues. There are ships from multiple nations in the general vicinity that could potentially provide resources to assist the impacted areas. An ad hoc coalition forms among nations (both aid providers and aid recipients) with varying degrees of amity/hostility/trust towards each other in order to distribute resources and to support necessary evacuations. Coalition members, while eager to coordinate on the relief effort, must take steps to ensure that information sharing is deliberate and conducted in a manner consistent with their organizational policies on information sharing.

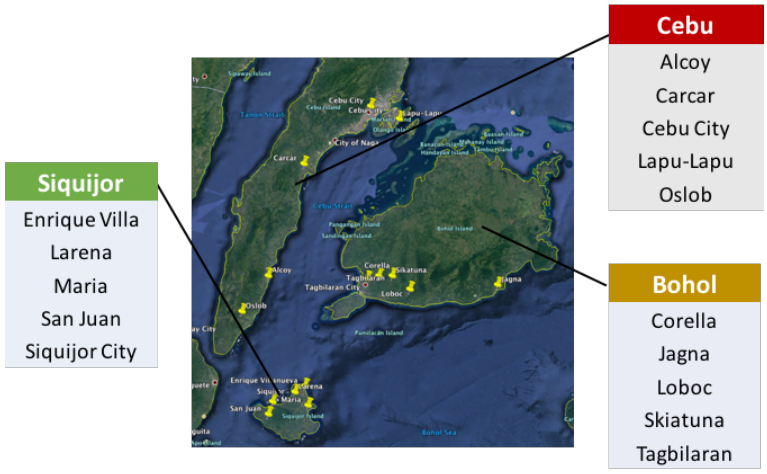
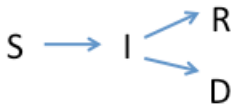


Figure 2. HADR scenario setting: three fictitious island nations, each with five communities

To simulate the pandemic outbreak, we used the well-known Susceptible, Infectious, Removed, (SIR) compartmental model of disease progression [Kermack & McKendrick, 1927], augmented to support a Deceased compartment (i.e., an SIRD model);

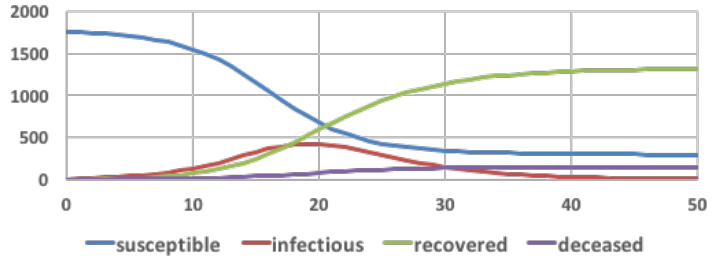
- *S(t):* # individuals not yet infected at time *t*
- *I(t):* # individuals currently infected at time *t*
- *R(t):* # individuals recovered at time *t*
- *D(t):* # individuals deceased at time *t*

Possible state progressions for an individual are summarized below:



The standard model computes aggregate SIRD totals; we enhanced the model to track disease state for individuals and to support inter-community transmission. Figure 3 shows the equations used in the simulation along with a sample progression of the disease for a given set of model parameters.

Within this overall HADR use case, our focus to date has been on the three detailed scenario threads summarized below.



- $\frac{\partial S}{\partial t}(t) = -\beta S(t) I(t)$
- $\frac{\partial I}{\partial t}(t) = (\beta S(t) - k - d) I(t)$
- $\frac{\partial R}{\partial t}(t) = k(1-d) I(t)$
- $\frac{\partial D}{\partial t}(t) = kdI(t)$

- $N = 5000$: population size
- $\beta * N = 0.5$: transmission rate
- $k = 0.2$: removal rate
- $d = 0.1$: mortality rate
- Initial infection rate: 10% in Maria, Siquijor

Figure 4. Model used to generate pandemic data, with graph of a sample progression for one community

- *Privacy-aware COP:* The operational objective in this thread is to provide a continuously updating common operational picture (COP) to the coalition members for ships in the area of responsibility. The privacy challenge is to provide information that will facilitate situational awareness and coordination without revealing information about ship positions, trajectories, and capabilities to parties that should not receive it. In particular, different nations will receive different views of the COP, based on controls imposed by the individual data owners.
- *Pandemic:* The operational objective in this thread is to predict the progression of a major disease outbreak through the impacted communities and to take steps to counter it. This thread introduces the challenge of protecting personally identifying information (PII) within medical records of individuals in the impacted communities while providing access sufficient to enable accurate characterization of the disease and its spread. A second privacy challenge relates to protecting information about certain aspects of the disease itself, to avoid inducing panic that could lead to mass migration and increased transmission among communities.
- *Aid Distribution:* This thread focuses on allocating and distributing resources (food, water, medicine) from coalition ships in the area to provide relief to hard-hit communities. Allocation and distribution planning require knowledge of ship positions, capabilities, and content, as well as of transportation and logistical capabilities in areas to which aid will be delivered. Each of these elements has contextually dependent privacy implications.

3. PRIME Platform

3.1 Overall Design

PRIME has been designed as an information mediator that provides access to data in accord with privacy restrictions imposed by data owners (see Figure 1). The PRIME platform leverages two proven, core technologies:

- A service-oriented architecture (SOA), called SIMON (Smart Integration Manager Ontologically Networked), that provides industry-standard identity management, policy enforcement, and micro-service capability integration. SIMON has been used to build and deploy a number of U.S. government systems for multi-nation information sharing.[1]
- An adaptive agent platform, called Lumen, that is used as a high-level workflow engine for processing requests within the system. Lumen is a hardened implementation of the SPARK framework [Morley & Myers, 2004], which has been operationally deployed to support adaptive task execution within the U.S. Army's Command Post of the Future [Myers et al., 2011].

Requests are processed initially in SIMON, making use of its native authentication and logging capabilities in a pre-processing phase. Acceptable requests (as determined by identity management policies) are then forwarded to Lumen, which applies workflow models to respond appropriately. These responses can involve posting service requests back to SIMON for retrieving and processing data, or invoking various privacy technologies. Before making these service requests, the system consults a *policy reasoning engine*, built on an ontology framework called Sunflower [Ford et al., 2016], to determine appropriate controls on query-related data accesses. The Lumen workflow orchestration assembles the results and returns them to the user's display, via the SIMON framework services. Throughout, logging is performed to track all accesses and transformations to data, providing the means to support continuous awareness of what information has been released, to whom, and for what purpose.

The computational environment in which the system operates include a range of data and processing capabilities, wrapped as services within the SIMON SOA, that are leveraged during workflow execution. Most interesting here are the privacy-enhanced data and services, which are used to protect information. Section 4 elaborates further on those technologies.

[1] For example, SIMON was used to build the Cooperative Situational Information Integration (CSII) system for US Southern Command (US SOUTHCOM). CSII integrates partner nations with the U.S. into a regional, web-based, unclassified, network-centric information sharing system that spans air, maritime, and land domains.

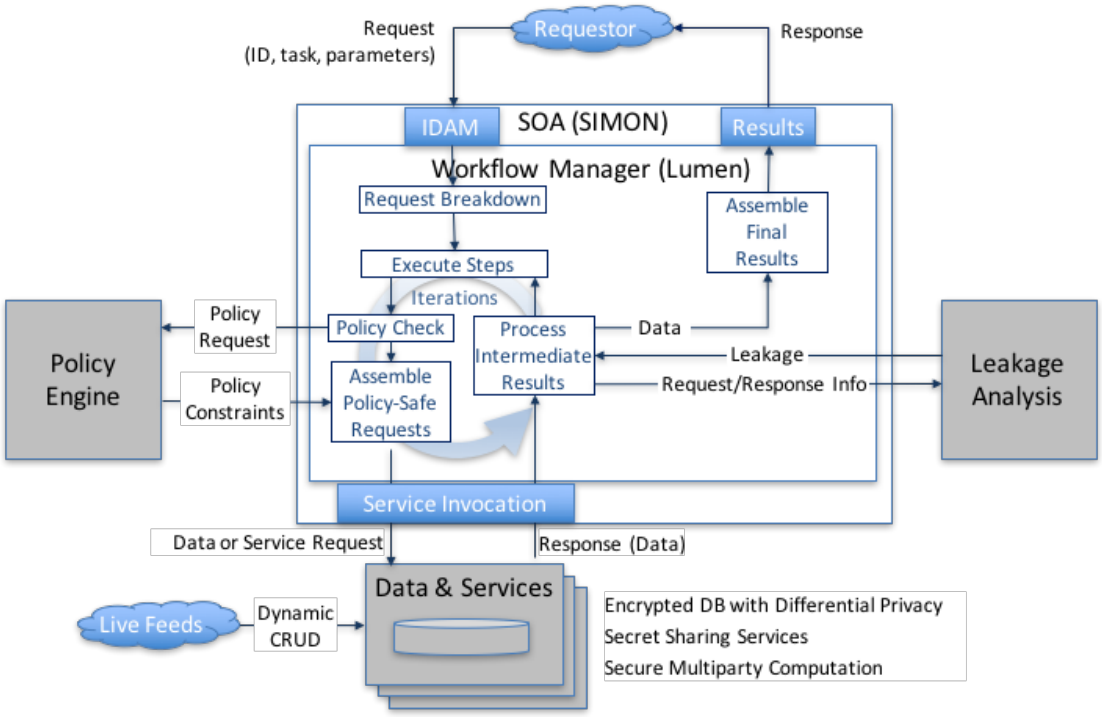


Figure 5. PRIME platform architecture

3.2 Workflow Management

The workflow manager in the PRIME platform responds to user requests in a manner that is consistent with privacy policies defined by data owners. In particular, a given request may require a sequence of information retrieval and processing steps, each of which must be performed in a manner permitted by the policies. Thus, while the policy reasoning engine provides the capability for representing and reasoning with policies, the workflow manager is responsible for policy enforcement.

Workflows are used for two purposes in the current system. One is to support processing and coordination tasks that necessarily involve multiple data accesses and computations. For example, in response to a user's request for berth allocations, the PRIME workflow manager orchestrates predefined queries and computations to perform the allocation.

The second use of workflows is to compensate for technical shortfalls in query support within the encrypted database technology, which currently supports a restricted subset of SQL.[2] Relative to our HADR use cases, current gaps relate primarily to advanced/aggregate query capabilities. For such gaps, workflows decompose the complex queries into simpler ones that fit within the

[2] We anticipate less need for compensation of this type in the future, given the rapid advances being made in encrypted database technologies.

capabilities of the current query language, invoke those simpler queries, and then aggregate the results for presentation to the user. In other words, the workflow manager effectively performs the database operations not supported currently in the encrypted database. This approach has the potential to leak information from intermediate results. One possible way to address this leakage (to be explored in future work) is to compile this workflow into code that can be migrated to the client; if intermediate results are sent encrypted, then the potential for additional leakage is greatly reduced.

3.3 Policies

Data owners can define policies that limit access to structured data based on characteristics of the requester, request history, and request details. Policies are enforced through a query rewriting mechanism (described in the next section) that guarantees all information releases from encrypted databases are sanctioned by relevant data owners. Currently, policies are limited to controlling query access to structured data and web services. In future work, they will be extended to control access to a broader range of data types and information services.

The details of the policy language are beyond the scope of this paper. At its core, however, the representation allows expressive specification of constraints for accessing specific pieces of data as well as aggregate information (e.g., counts and averages), leveraging an underlying ontology of classes and relations. The policy representation also contains two constructs that are particularly important for privacy controls within coalition settings. One construct is an *override mechanism*, which enables one policy to take precedence over another in the event that they conflict. The second is an explicit linkage to organizational structures, which can provide the basis for defining overrides. Within the HADR use case, for example, nation-level policies are set to override community-level policies. Together, these constructs enable increased modularity of representation for policies within hierarchically structured organizations, compared to having to explicitly embed override conditions within policies for lower-level organizations. Overall, the policy representation is much richer than that of entitlement mechanisms in standard information systems, providing the flexibility to express the kinds of complex privacy restrictions necessitated in a coalition setting.

One interesting property of the policy reasoner is that it can generate *residual constraints* that serve as conditions for accessing requested information. In particular, policies do not gate access on a yes/no basis. For example, a policy may allow a Response Coordinator to access demographic information for people in a community but only for people older than thirteen. In the event that a request is made by the Response Coordinator for demographic information, he would be returned only the records for appropriately aged individuals.

Figure 5 illustrates the application of policies within the HADR use case, showing different views of information depending on the policies in force for different information requestors. Here, pie-charts depict percentages of the population in the different SIRD compartments. As shown, the International Response Coordinator is allowed to see nation-level views of the SIRD data (left); the Cebu City Coordinator is allowed to see a community-level view for its own community (middle); and the Bohol Nation Coordinator can see the community-level view for Bohol communities but only nation-level views for other communities (right).

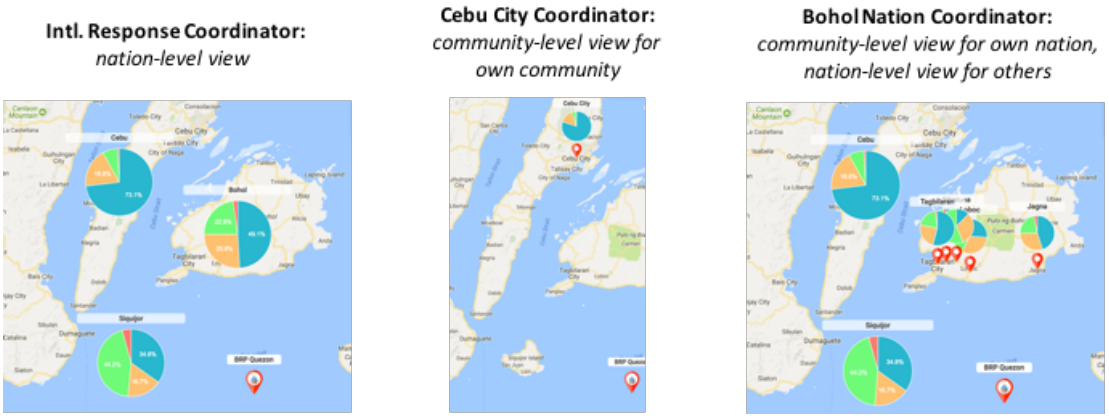


Figure 6. Policy-differentiated information access, with different content and resolution based on community, national, and international roles

3.4 Policy enforcement

As noted above, the policy reasoner stores policies and makes decisions about what policies apply (and how) to a particular request. Actual enforcement of the policies, however, is done within the workflow manager.

Our current approach to policy enforcement centers on the notion of *policy-safe queries* (depicted in Figure 6). A given query is made policy safe by reformulating it to ensure that all information accesses are allowed given current policies. The simplest way to make a query policy safe is by reducing it to a 'null' query. However, our goal is to maximize the exchange of information while remaining compliant with policies. To this end, we developed an approach for automatically rewriting SQL queries to provide maximal access. With this approach, an initial query gets mapped to a collection of derived queries, some with additional WHERE clauses (to limit access to records) and some with SELECT clauses removed (to prevent access to data that should not be revealed to that user). The results of executing this modified set of queries are then merged to provide the overall policy safe response to the query. The additional constraints for the WHERE clauses and the set of SELECT clauses to be removed are generated by the policy reasoning engine.

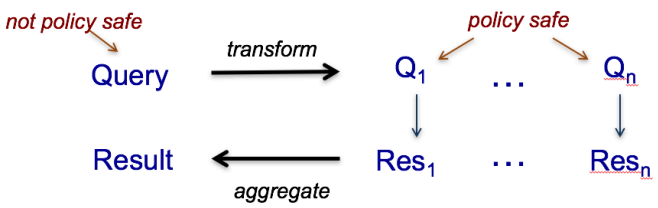


Figure 7. Policy-safe query reformulation

Figure 7 shows an example of this policy safe reformulation for a particular SQL query that seeks to retrieve information (name, gender, birthdate, nation) for evacuees that have tested positive for an emerging virus, ZV1. Extant policies from different nations limit what information that requestor is allowed about their citizens: while Japan places no restrictions, the U.S. disallows access to birthdates and New Zealand disallows access to gender.

Consider first the case of Japanese evacuees. Because there are no restrictions in this case, the query can be made mostly as is, simply adding a constraint to the WHERE clause limiting the scope to evacuees from Japan (step 1). In step 2, separate queries are made for the countries that imposed limits on access. For records from New Zealand, the query is modified to exclude gender from the SELECT clause; for records from the U.S., the query is modified to exclude birthdate. In step 3, these partial results are then aggregated by the workflow manager into a composite result to be returned to the requestor.

Query: retrieve info for ZV1-positive evacuees

```
SELECT name, gender, birthdate, nation
FROM evacuee
WHERE ZV1Positive = TRUE;
```

Policies
- US: no access to birthdate
- NZ: no access to gender
- Japan: no restrictions

Policy-safe response:

1. Collect results for countries with no restrictions

```
SELECT name, gender, birthdate, nation FROM evacuee
  JOIN community on community.id = evacuee.community_id
  JOIN organization on organization.id = community.organization_id
WHERE ZV1Positive = TRUE AND organization_name IN ('Japan')
```

2. Collect filtered results for countries with restrictions

```
SELECT name, ~~gender~~, birthdate, nation FROM evacuee
  JOIN community on community.id = evacuee.community_id
  JOIN organization on organization.id = community.organization_id
WHERE ZV1Positive = TRUE AND organization_name = 'NZ'
```

```
SELECT name, gender, ~~birthdate~~, nation FROM evacuee
  JOIN community on community.id = evacuee.community_id
  JOIN organization on organization.id = community.organization_id
WHERE ZV1Positive = TRUE AND organization_name = 'US'
```

3. Aggregate results

Figure 8. Example of policy-safe query reformulation, with modifications highlighted in red.

The workflow manager also provides a *query relaxation* capability that can enable responses with lower fidelity than what was originally requested in response to policy restrictions, thus adhering to privacy requirements while releasing the maximally allowed amount of information. For example, if policies prohibit SIRD queries at the community level, the system will retry the query at the nation level. This type of relaxation is possible in cases where there is a natural generalization to a lower-fidelity characterization of the requested data.

3.5 Common Data Model

Following our initial platform development, we identified the need for a general data model that abstracts from particular data storage and processing representations to facilitate independent advances in design of the data processing, policy reasoning, and data storage specific schemas. A general model was needed that would both capture the ontologies of our scenarios and support automated mappings to the languages or schemas used for policy reasoning, request processing, and data storage. This model would also capture the meta-properties of the associated data and policies, including owner, time of creation, and their relationships. Such a model would greatly facilitate extension of the PRIME platform to other enterprise settings, which typically would have pre-existing data models.

To this end, we developed a *common data model* (CDM), using the Web Ontology Language (OWL) standard [Patel-Schneider et al., 2004], to represent relevant object classes, their properties, and relationships. The CDM is used to capture the information and relations needed to map data request representations to any independent data sources. It is also used as the underlying ontology for the policy representations. In this way, the CDM provides the semantic glue to connect the various components in the system, enabling metadata about information and services in the system to be captured in one place and then distributed to the modules that need it. It also isolates the policy engine and interface development from the specifics of individual data stores. Our first implementation handles SQL access to secure databases. Future developments will add mappings for RESTful data web services and unstructured file systems.

We developed automation tools to leverage the CDM representation for workflow and database accesses, as well as integration with the policy reasoner, enabling a common ontology to be used for data requests, processing, policy reasoning, and data accesses. This metadata explicitly captures the correlation intention of requested data (how requested data elements are to be connected in a single request) so that the appropriate set operations (e.g., JOINs in SQL) can be determined automatically from the mapping information associated with the CDM.

Figure 8 shows a subset of the HADR pandemic thread ontology and associated schema. The CDM-based classes and properties on the left are captured explicitly in OWL. The associated SQL schema, on the right, includes typical relational normalizations. The CDM intentionally abstracts the objects and instances within a particular domain, such as *person* or *nation*, as well as the hierarchy of their relationships such as *DiseaseStatus* as a specialization of *MedicalStatus*, which is in turn a specialization of *MedicalInformation*. While these abstractions are useful for building an ontology for policy representations, a typical normalized database schema is quite different in both structure and naming conventions. These differences were intentionally captured in our data models to explore the challenges in providing privacy-preserving technology for

existing enterprise systems. The needed mapping information is captured directly in the CDM model syntax and is then used to convey what information is being requested for determination of applicable privacy policies and request construction, as well as how to map this ontology to the SQL schema stored in the secure database for data access.

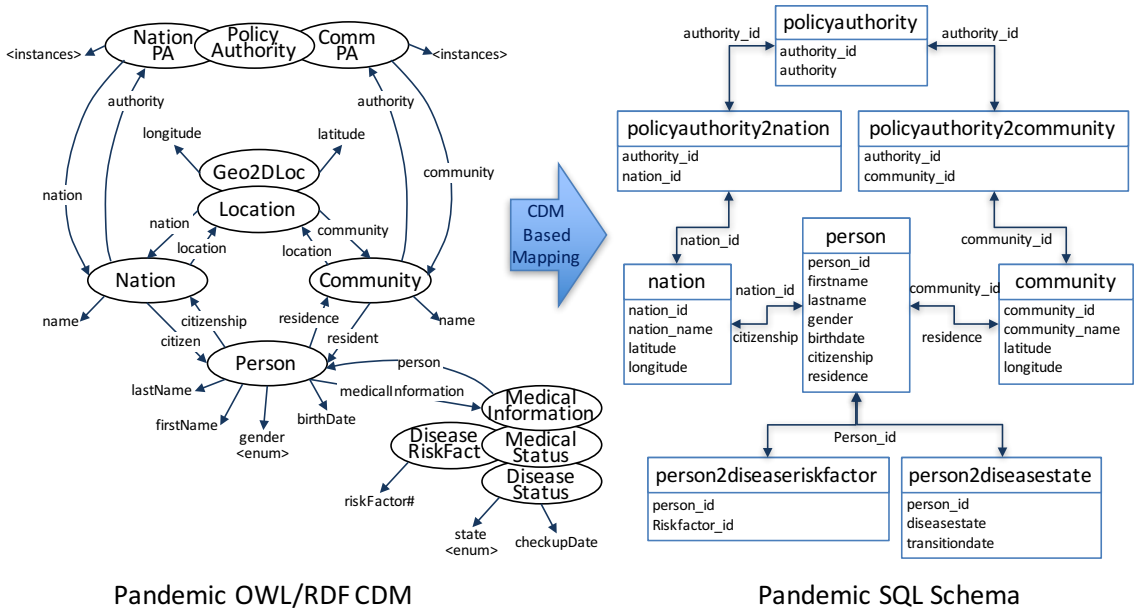


Figure 9. Pandemic subset of the HADR CDM with associated database schema

Figure 9 shows a sample mapping for the pandemic thread from the requested data elements represented in the OWL/RDF representation used by the CDM to the associated SQL query. The syntax of this CDM request contains the data elements requested using abbreviated RDF namespaces (e.g., prime#, medical#), as well as any user or policy-safe constraints that must be applied. In this example, the information requested is all persons' first and last names, gender, resident community, and current disease state. The constraints include a user-defined filter for only those persons whose last checkup is before 10 April, 2017, and policy-based restriction that limits the request to data owned by the person's community policy authority. The CDM-based mapping metadata is used by the automation tools within the PRIME platform to generate corresponding SQL queries.

```
[(prime#Person,prime#lastName)]
[(prime#Person,prime#firstName)]
[(prime#Person,prime#gender)]
[(prime#Person,prime#residence),(prime#Community.prime#name)]
[(prime#Person,prime#medicalInformation),(prime#DiseaseStatus,medical#state)]
(LTE([(prime#Person,prime#medicalInformation),(prime#DiseaseStatus,medical#checkupDate)],'2017-04-10'))
(EQ([(prime#Person,prime#residence),(prime#Community,  prime#authority)],'CebuNationPA'))
```

Map

```
SELECT person.lastname, person.firstname, person.gender, community.community_name,
person2diseasestate.diseasestate
FROM person, community, person2diseasestate, policyauthority2community, policyauthority
WHERE person.residence = community.community_id
  AND person2diseasestate.person_id = person.person_id
  AND person2diseasestate.transitiondate <= '2017-04-10'
  AND community.community_id = policyauthority2community.community_id
  AND policyauthority.authority_id = policyauthority2community.authority_id
  AND policyauthority.authority = 'CebuNationPA';
```

Figure 10. Sample (abbreviated) CDM property chains and constraints auto-mapped to SQL

4. Application of Privacy Technologies

In this section, we discuss the various cryptographic and reasoning technologies used in PRIME to achieve privacy objectives for the HADR use case, focusing on their utility within the coalition setting and current limitations in their use.

4.1 Secure Multiparty Computation for Resource Allocation

Secure multi-party computation (MPC) enables agreed-upon computations to be performed on data supplied by multiple stake-holders without revealing any more information than the output of the function itself. In particular, the stake-holders provide encrypted versions of their inputs to the computation and derivative information from intermediate steps of the computation remain secret [Yao, 1982; Goldreich et al., 1987; Chaum et al., 1988; Ben-Or et al., 1988;]. One recent example of an application of MPC technology is for performing a secure probability analysis of satellite collision [Hemenway et al., 2016].

One use of MPC within PRIME is to support resource allocation. As an example, consider the following aid distribution task. There are k communities each of which requires M_c amount of a particular resource (e.g., food, medicine), and p resource providers each of which can supply R_i amount of the resource. The resource allocation task requires finding a set of feasible assignments

$$\begin{aligned} &\text{Provider 1:} && M^1_1, \ldots, M^1_k \\ &\text{Provider 2:} && M^2_1, \ldots, M^2_k \\ &\quad \ldots \\ &\text{Provider } j\text{:} && M^p_1, \ldots, M^p_k \end{aligned}$$

such that

- for each provider $1 \leq i \leq p$: $M^i_1 + \ldots + M^i_k \leq R_i$
- for each community $1 \leq c \leq k$: $M^1_c + \ldots + M^p_c \geq M_c$

In particular, we require a given resource provider to completely service the request from an individual community; however, the provider can service multiple communities provided it has sufficient resources to satisfy their aggregate need. We also assume that provision of resources has an associated cost, which could be the cost of delivering the requested resources, or simply the motivation for servicing a particular request. Costs are considered private, with each participant seeking to protect this information from other coalition members.

We explored a range of different MPC algorithms to address this problem, considering options for optimal usage (relative to the provided cost models) and fair allocation that seeks to balance the costs evenly between the resource providers. Even within an MPC setting, care must be taken to avoid unintentional leakage of information. Consider an approach that seeks to optimize the allocation relative to the stated cost model by always selecting the lowest cost bid. In the case where there are two resource providers, each assignment of a resource provider to a requester reveals the relative costs for each of the two parties (i.e., the lower-cost bid always gets assigned). For this reason, we chose to mask this cost information by selecting the lower-cost bid only *P* percent of the time, for some selected threshold *P*. Through this probabilistic selection, the involved parties cannot be certain whether the assigned nation was in fact the lower bid for a particular request.

4.2 Searchable Encryption Meets Secure Multiparty Computation

A *searchable encryption* scheme securely encrypts data in a way that preserves one or more properties of interest, such as relative order or equality [Song et al., 2000; Boneh et al., 2004; Curtmola et al., 2006; Bösch et al., 2014]. PRIME provides a privacy-preserving relational database functionality implemented using the complementary technologies of secure multi-party computation (described above) and searchable encryption.

Unfortunately, while such encryption schemes may enable fast data access, they typically allow information leakage that may be observed by adversaries. In contrast, secure multi-party computation typically leaks very little information but is often several orders of magnitude slower than computation in the clear. In the Jana privacy-preserving database used in PRIME, relational queries written in SQL are answered in part by normal queries over such searchable encryptions, and in part by operations executed using a secure multi-party computation engine [Damgard et al., 2012]. By combining the two, and by allowing for each attribute in each database relation to be encrypted in one of several ways, Jana supports bespoke trade-offs between information leakage and query performance. Some configurations of Jana may be practically limited to 5,000 or so records in often-accessed relations. Other configurations may provide practical performance for much larger relations. Jana provides a tool for studying such trade-offs in practical use cases, something not previously reported in the secure computation literature.

4.3 Secret sharing

Secret sharing [Shamir, 1979] is a secure multiparty computation technique in which shares of some secret value are distributed to a group of participants by a trusted party. The original secret can be reconstructed only when a threshold number of shares are recombined. *Function secret sharing* (FSS) [Boyle et al., 2015] extends that concept to the computation of a function. Shares of a function are distributed to multiple participants in such a way as to enable them to each compute part of the overall function given a sufficient number of shares (otherwise nothing is revealed), and the result of the function on the secret inputs is additively recovered at the end of the computation without revealing anything about the inputs.

We use FSS to compute multi-level aggregation of SIRD population data, in accord with policies. In particular, the numerical SIRD pandemic population counts are stored over multiple function secret sharing services, along with representative attributes for functional processing. Our PRIME system, using the same workflow and policy decision mechanisms described above, accesses the policy-safe level of aggregations of these statistics (i.e., either nation-wide aggregation or community-wide aggregation) on a specific date. In this way, the FSS services, potentially kept by members of a non-trusting community, cannot reveal anything about the shared data without a minimum number of participants answering the policy-safe request.

4.4 Differential Privacy

Differential privacy [Dwork, 2006] enables statistical queries (in particular, aggregations) over a database of values, while minimizing the chances of identifying any individual within its records. Differential privacy is achieved through the use of principled noise injection for the results of aggregate queries to obscure the presence or absence of individuals within a database, with the amount of noise added linked to a specification of the degree of privacy that is to be maintained.

As a concrete example, consider an epidemiologist working to track and predict the disease spread. Policies are in place that enable him to access gender, birthdate, and disease status from the collected medical data as part of this process; however, to protect personal privacy, he is not allowed access to names. Consider if the epidemiologist is allowed access to fully accurate counts of people in the various SIRD categories. Suppose an update to the data is made to record a new entry for an infected individual. Queries immediately prior to and after the update would show that the new entry increased the Infected count by one, hence the addition must be infected.

Although the epidemiologist does not have direct access to the identify of this person, prior work has established that birthdate, gender, and zipcode are sufficient to identify individuals in 85% of cases [Sweeney, 2000]. As such, the epidemiologist could now likely determine the SIRD status as an indirect result. If instead the epidemiologist can access only differentially private counts of the number of infected people, the difference between the pre- and post-update count queries would not reveal the infected status of the new entry. Importantly, though, the differentially private counts can still provide useful statistical information for epidemiological modeling purposes [Ellis et al., 2017].

Differential privacy can also be combined with the multiparty computation technologies described above. As such, these enhancements have been included by the developers of the encrypted database and the FSS services in their technologies to further protect data privacy data.

4.5 Quantitative Leakage Analysis

To complement the above technologies for privacy preservation, we are exploring the use of technologies that can characterize what secondary information is being revealed through data releases. Awareness of this "leakage" is important for enabling informed decisions regarding the implications of sharing.

As noted above, all information releases and processing in the PRIME system are logged. This information feeds a real-time *quantitative information flow* analysis [Mardziel et al., 2013] that maintains an information-theoretic model of how accumulated releases decrease the receiver's uncertainty regarding some target value that ideally would be kept private (such as a resource capacity). In particular, the model quantifies this leakage in terms of the number of bits of information that have been revealed.

To make this concrete, consider the task of allocating *B* berths in aggregate from nearby ships to transport seriously injured people from one of the impacted communities that has been particularly hard hit by the typhoon. The berths need to be available before some deadline *T*. Nations with ships in the area are willing to help out but do not want to reveal critical information about their ships (e.g., number of available/filled berths, position, travel speed).

The workflow required to complete the allocation can require multiple requests to the ship owners regarding their capabilities. One approach (which we call *separate*) first determines which ships can arrive by the deadline. For those ships, a form of binary search is then performed that establishes lower and upper bounds on each ship's berth capacity. To start, each ship is asked whether it can provide the required *B* berths. If one answers positively, then it can be selected and the task is done. If not, the initial bounds are [0*, B*] and the ships are then asked whether they can provide *B/2* berths. If two of them can, the allocation is done. Otherwise, the bounds are adjusted: for a ship answering affirmatively, the lower bound is updated to *B/2* while for ships answering negatively, the upper bound is reset to *B/2*. This process continues until a set of ships is determined whose sum of lower bounds exceeds the required number of berths. A variant algorithm (*combine*) merges the deadline and capacity queries. In this case, a negative response could mean either that the ship is too far away or that it lacks the requisite capacity, so can reduce the amount of information leakage.

These algorithmic variations (*separate*, *combined*) can lead to different allocations and leakage. Figure 10 illustrates this point for two separate berth allocation tasks (i.e., initiated for different communities), showing overall allocations along with leakage from the perspective of the USNS ships. In the case on top, no USNS ships are required and therefore drop out of the allocation process early with low leakage. In the case on the bottom, the capacity of the USNS is needed, incurring greater leakage as the binary search process refines the bounds on available capacity.

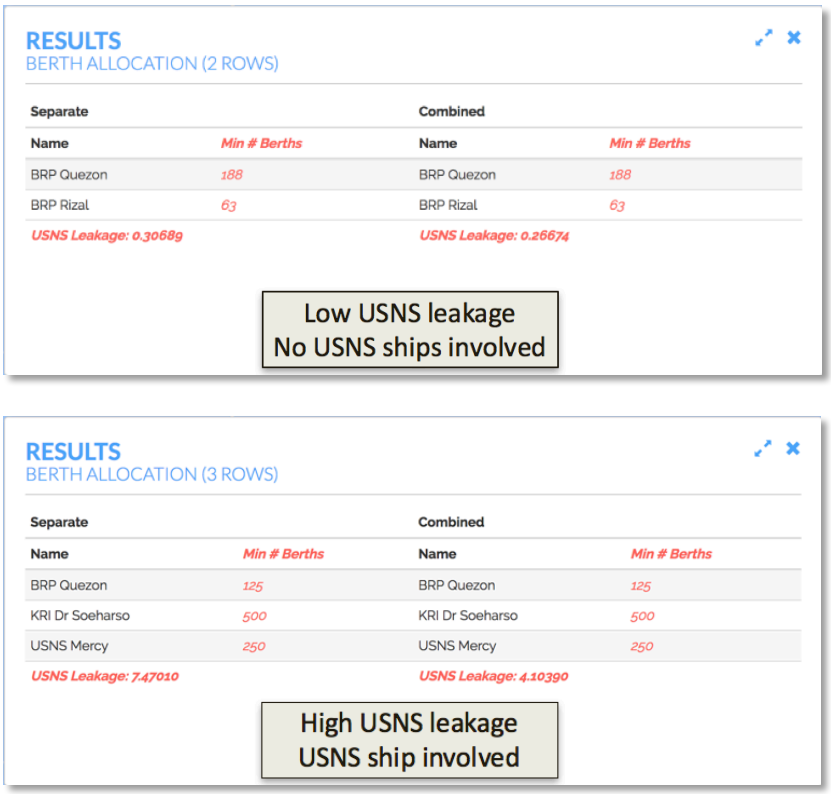


Figure 11. Comparison of leakage (in bits) for the combined and separate allocation algorithms

5. Future Challenges

Many issues remain in the area of technological support for privacy in coalition settings. Here, we describe three that we are addressing in our ongoing research.

- *Unstructured, streaming data*: As noted above, much of the information that is used in operational settings is unstructured (text, imagery, acoustic, PowerPoint, etc.) and continuously changing. In contrast, most of the work on privacy has focused on protecting information stored in structured databases. Streaming, unstructured data introduces significant challenges for developing mechanisms to minimize unnecessary sharing and to understanding the implications of information releases.
- *Privacy vs utility*: In this paper, we have focused primarily on mechanisms for safeguarding privacy. We are also exploring the tradeoff between increasing privacy versus decreased utility. For example, the work in [Ellis et al., 2017] examines the impact of different degrees of differential privacy on the accuracy of disease models that can be estimated by an epidemiologist.

- *Organizational policies*: Our work to date on policies has focused on controlling access to data. However, policies will also need to be formulated to control other aspects of the information systems. For example, policies should be defined to characterize who is authorized to write policies, or for what sorts of encryption levels are required to protect certain types of data.

6. Conclusions

This paper summarizes an approach to applying state-of-the-art privacy technologies to enable increased and informed information sharing within coalitions. Given the focus on extended collaborations within coalitions, we adopted a process-oriented perspective, performing selection and configuration of workflows for responding to information and coordination requests while taking into account privacy implications for their execution.

Appendix A: Privacy Technologies

The table below summarizes the technologies being used within PRIME and the organization that is providing them.

Organization	Technology
Galois	Searchable encryption Differential privacy Secure multiparty computation Quantitative leakage analysis
Pacific Science and Engineering Group	User workflow and policy modeling
SRI International	Policy representation and reasoning Workflow orchestration
Stealth Software Technologies	Function secret sharing Secure multiparty computation

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA), Air Force Research Laboratory (AFRL), and Space and Naval Warfare Systems Center, Pacific (SSC Pacific) under Contracts No. N66001-15-C-4065, N66001-15-C-4069, N66001-15-C-4070, N66001-15-C-4071, FA8750-16-C-0011, and FA8750-16-C-0022. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA, the Department of Defense, AFRL, or SSC Pacific.

The authors thank their colleagues who have contributed to the concepts and software described in this paper: Khang Duong, Thomas Garvey, Joshua Gould, Thomas Lee, and Laura Tam (SRI International) for their contributions to the PRIME platform; Linda Briesemeister, Samuel Wood, and Daniel Elenius (SRI International) for their contributions to the HADR CDM, policies, and the policy decision engine; Paul Bunn and Yuval Ishai (Stealth) for their work on MPC algorithms and their application to resource allocation; Jonathan Daugherty, Thomas DuBuisson, and Tristan Ravitch (Galois) for their contributions to the Jana privacy-preserving relational database technology; Jose Calderon (Galois) for his contributions to the quantitative leakage analysis capability; and Mark St. John (Pacific Science and Engineering) and Dusko Pavlovic for discussions related to user workflow and policy management.

References

Archer, D. S., Bogdanov, D., Pinkas, B., and Pullonen, P. (2016). Maturity and performance of programmable secure computation. *IEEE Security and Privacy*, 14(5), 48–56.

Ben-Or, M., Goldwasser, S., and Wigderson, A. (1988). Completeness theorems for non-cryptographic fault-tolerant distributed computing. *Proc. of the 20th Annual ACM Symposium on Theory of Computing*, 1–10.

Bösch, C., Hartel, P., Jonker, W., and Peter, A. (2014). A survey of provably secure searchable encryption. *ACM Computing Surveys*, 47(2), Article 18.

Boneh, D., Di Crescenzo, G., Ostrovsky, R., and Persiano, G. (2004). Public key encryption with keyword search. *Proc. of EUROCRYPT*, LNCS 3027, 506–522.

Boyle E., Gilboa N., and Ishai Y. (2015). Function secret sharing. In: Oswald E., Fischlin M. (eds.) *Advances in Cryptology* - EUROCRYPT 2015. Lecture Notes in Computer Science, vol 9057. Springer, Berlin, Heidelberg.

Chaum, D., Crepeau, C., Damgård, I. (1988). Multiparty unconditionally secure protocols. *Proc. of the 20th Annual ACM Symposium on Theory of Computing*, 11–19.

Curtmola, R., Garay, J., Kamara, S., and Ostrovsky, R. (2006). Searchable symmetric encryption: improved definitions and efficient constructions. *Proc. ACM Conference on Computer and Communications Security*, 79–88.

Damgard, I., Keller, M., Larraia, E., Pastro, V., Scholl, P., and Smart, N. (2012). Practical covertly secure MPC for dishonest majority – or: breaking the SPDZ limits. Cryptology ePrint Archive: Report 2012/642.

Dwork, C. (2006). Differential Privacy. *Proc. of the 33rd Intl. Colloquium on Automata, Languages, and Programming*, Vol. 4052. Springer, 1–12.

Ellis, T., Lee, T., Lepoint, T. and Myers, K. (2017). On the usability of differentially private SIRD data during a pandemic outbreak. *Proc. of the 13th Symposium on Usable Privacy and Security* (Poster Track).

Ford, R., Denker, G., Elenius, D., Moore, W., and Abi-Lahoud, E. (2016) Automating financial regulatory compliance using ontology + rules and Sunflower. *Proc. of the 12th Intl. Conference on Semantic Systems*, Leipzig, Germany, 113–120.

Goldreich, O., Micali, S., and Wigderson, A. (1987). How to play ANY mental game. *Proc. of the 19th Annual ACM Symposium on Theory of Computing*, 218–229.

Hemenway, B., Lu, S., Ostrovsky, R., and Welser IV, W. (2016). High-precision secure computation of satellite collision probabilities. *Proc. of the Intl. Conf. on Security and Cryptography for Networks*, 169–187.

Kermack, W.O. and McKendrick, A.G. (1927). A contribution to the mathematical theory of epidemics. *Proc. of the Royal Society A: Mathematical, Physical and Engineering Sciences.* 115 (772): 700.

Mardziel, P., Magill, S., Hicks, M. and Srivatsa, M. (2013). Dynamic Enforcement of Knowledge-based Security Policies using Probabilistic Abstract Interpretation. *Journal of Computer Security*, 21:463–532.

Morley, D. and Myers, K. (2004). The SPARK agent framework. *Proc. of the 3rd Intl. Joint Conf. Autonomous Agents Multi Agent Systems.*

Myers, K., Kolojejchick, J., Angiolillo, C., Cummings, T., Garvey, T., Gervasio, M., Haines, W., Jones, C., Knittel, J., Morley, D., Ommert, W., and Potter, S. (2011). Learning by demonstration technology for military planning and decision making: a deployment story. *Proc. of the 23rd Conf. on Innovative Applications of AI*.

Patel-Schneider, P., Hayes, P. and Horrocks, I. (2004). OWL web ontology language semantics and abstract syntax. W3C Recommendation.

Shamir, A. (1979). How to share a secret. *Communications of the ACM,* 22 (11): 612–613.

Song, D., Wagner, D.A., and Perrig, A. (2000). Practical techniques for searches on encrypted data. *IEEE Symposium on Security and Privacy*, 44–55.

Sweeney, L. (2000). Simple demographics often identify people uniquely. Carnegie Mellon University Working Paper. http://dataprivacylab.org/projects/identifiability/paper1.pdf

Yao, A.C. (1982). Protocols for secure computations. *Proc. of the 23rd Annual Symposium on Foundations of Computer Science*, 160–164.