A Trusted Federated System to Share Granular Data Among Disparate Database Resources

Joanna F. DeFranco, David F. Ferraiolo, Rick Kuhn, Joshua Roberts

Sharing data between different organizations is a challenge primarily due to database management systems (DBMSs) being different types that impose different schemas to represent and retrieve data. In addition, maintaining security and privacy is a concern. The authors leverage two proven National Institute of Standards and Technology (NIST) tools to address this challenge: Next Generation Database Access Control (NDAC) and data block matrix.

While access to unstructured data in support of big data analytics captures a great deal of attention, it remains a budding industry. However, since structured data is eminently searchable with human-generated queries and algorithms, database management systems (DBMSs) have and will remain a primary means for storing, managing, manipulating, and retrieving data for the foreseeable future.

To meet this demand, there exists a wide variety of database products to choose from, each providing a powerful means for extracting, creating, and updating data in accordance with the broadly recognized Structured Query Language (SQL) standard. SQL queries comprise four basic types of operations – Select, Insert, Update, and Delete – that respectively read, create, write, and delete data in tables.

Given the sensitivity of much of the data that resides in DBMSs, the ability to control access to this data is widely recognized as a fundamental security requirement. Today's approach of imposing policy for accessing this data is by no means standardized and requires separate configurations of an often-complicated amalgamation of mechanisms, including those that are custom-implemented in applications and are specific to DBMS products.

Information stored in DBMSs is not only important to host organizations but is often of tremendous value to other *relying parties* (RPs) outside of an organization's network. This situation may pertain to the sharing of medical information related to patient care in collaboration with multiple providers and in discovery of new therapeutics; the prevention, monitoring, and investigation of terrorism-related incidents; and, in general, the improvement of collaboration among alliances in pursuit of new strategies, findings, and observations.

To put things into perspective, consider the vast amount of data associated with just a single person with the onset of insulin-dependent diabetes at a young age. There is no cure for this disease, and it needs to be treated and managed for the patient's lifetime. The patient requires continuous insulin therapy to survive. Additionally, multiple other specialist providers are required to proactively diagnose and treat possible complications of this disease. For treatment and management of the disease, a provider requires access to a patient's entire medical history (with the consent of the patient). This will reduce many possible instances of duplication, all of which come at a high price when providers are changed or added. Every patient encounter should ensure a 360° view regardless of when, where, or how the patient is being treated.

The patient must also be offered the ability to selectively describe who may or may not access portions of their protected health information. That is, a patient must be able to keep sensitive, unrelated medical information private. Further complicating the situation, regulatory requirements pertaining to consent and restricted access often vary from state to state.

There exist three fundamental approaches to sharing database resources: (1) direct transfer between RPs, (2) centralized resource storage accessible by all RPs, and (3) remote RP access to local data. Unfortunately, data sharing among RPs is made difficult for two primary reasons. First, data with different formats and different record schemas that are located in different DBMSs are difficult to transfer, consume, and interpret correctly among RPs. Second, DBMS data can often only be accessed by authorized users in accordance with the policies of the originating RP. Authorized access to transmitted data is hard to verify, given the need for access rules to follow, and granting local access to unknown users with unknown credentials to databases with varying authorization mechanisms is precarious at best.

Opportunity

To solve the database sharing problem, NIST has developed an infrastructure solution, in its early implementation, that is founded on the Attribute Based Access Control (ABAC) model and allows policy-preserving access to resources stored in the databases of a federation of RPs. This solution is based on the integration of two proven NIST technologies: Next Generation Database Access Control (NDAC) and data block matrix. NDAC provides a database and application-agnostic approach for controlled access to DBMS data by application users. The scope of NDAC's policy enforcement is based on its implementation of Next Generation Access Control (NGAC), an American National Standards Institute/InterNational Committee for Information Technology Standards (ANSI/INCITS) ABAC standard. In keeping with the ABAC model, capabilities to access DBMS resources through NDAC are dictated by the attributes assigned to its users.

Attributes are established as the means for allowing access to shared resources across the federation. The data is stored in a data block matrix—a new type of distributed ledger that provides a means for storing, managing, and sharing attributes—with the integrity protection of a blockchain but with the ability to edit or delete data. It is the responsibility of each RP to assign users in their organization to attributes. Consequently, those attribute assignments are written to the block matrix. However, it is important to recognize that an RP's assignments are not for the benefit of accessing resources in their own organization but rather to potentially grant their users access to the resources of other RPs.

To allow the sharing of resources across the federation, each RP would need to apply the same set of attributes when configuring their unique access policies. To ensure a global naming convention, there must exist a catalog of user attributes that are specific to the federation. For instance, this catalog could include the standardized nomenclature of roles and responsibilities used in the healthcare industry (e.g., Systematized Nomenclature of Medicine Clinical Terms: SNOMED CT) for allowing access to medical records (SNOMED International, 2020). To enable a remote user's access to local resources, the user would be onboarded into the local NDAC system via assignment to their block matrix-validated attributes.

Establishing privileges for RPs to read from and write to the block matrix provides the basis for trust in the federation, a notion that is consistent with other privileged blockchain technologies. In this case, trust is further bolstered by using NIST's NGAC reference implementation to impose administrative policy as to who and under what authority can create or delete user-to-attribute assignments within each RP. The source of these rules should come from a mutually agreed upon governance policy.

The integration is, in effect, an overlay of a system within systems devoted to the sharing of resources. In other words, the integration is completely transparent to the otherwise normal business operations of the RPs. The solution is presented as a federated variation on approaches that permit remote user access to local DBMS resources through a globally recognized catalog of attributes. In the following sections, the technologies will be described, and medical data sharing will be the use case to describe a proof of concept. Note, however, that this federated solution is viable for a wide variety of other use cases.

System Concepts and Technologies

NGAC

NGAC is an ABAC authorization framework (Ferraiolo et. al., 2016) that can express access privileges as combinations of access control policies. These policies may pertain to access rights related to resources, such as read and write, and administrative access rights that pertain to the management of NGAC's access control data, such as creating and deleting user-to-attribute assignments. It serves this integration in two vastly different ways. First, it is a narrow, stand-alone product for the expression and enforcement of policies over the administration of user attributes made available to the federation. Second, it provides a basis for NDAC in the expression and enforcement of a wide variety of controls in accessing and protecting resources stored in DBMSs.

For this system, access control policy is divided into two categories that have been used since the early days of computer security: *discretionary access control* (DAC) and *mandatory access control* (MAC). DAC is a policy that permits system users to read and write and allow or disallow other users' access to resources that are placed under their control. For instance, a user may be provided with access rights to read their entire medical record and write to specific fields. Additionally, a patient may grant a doctor read access to their medical record under DAC. These capabilities are provided through the delegation of narrow administrative access rights by an administrator to the DAC user. Although not exclusively, granting discretionary access is typically based on the name attribute of the granter, name attribute of the grantee, and name of the resource of concern. For instance, access could be granted to Bob to read Mary's (a patient) medical-record-x. Delegating administrative capabilities to the granter involves a variety of administrative access rights. To conveniently address the delegation of complex administrative access rights, NGAC recognizes parameterized administrative routines, such as create DAC-user(user name, resource name).

In contrast to DAC, MAC policies unavoidably impose non-discretionary rules on users when accessing resources. For instance, under a regulatory policy, a RP might restrict access to sensitive portions of medical records to users that are either doctors or nurse practitioners. The MAC policies often pertain to the popular role-based access control (RBAC) model. However, depending on the type of data under consideration, other forms of MAC policies could readily come into play, such as those pertaining to organizational affiliation, geographic region, or institutional ranking.

What DAC and MAC policies have in common is that they refer to named subsets of users (e.g., Bob, doctor, nurse practitioner), resources (e.g., Record-12), and fields of records (e.g., treatments and diagnosis) that can be respectively modeled as user and object attributes. NGAC refers to a user or resource membership in these attributes as “attribute assignment,” often denoted by an arrow (i.e., “→”). For instance, u1→Bob and u1→Doctor specify the user u1 is assigned to Bob and Doctor. Another important notion of access control is *policy combinations*. For instance, for a user u1 to access a sensitive portion of a medical record, the user must comply with both the DAC policy, pertaining to data ownership (e.g., u1→Bob), and one or more MAC policies, which are requirements based on other relevant attributes (e.g., u1→Doctor or u1→Nurse Practitioner). Finally, access policies may pertain to prohibitions regardless of policy combinations. For instance, when granting access to their medical record, a patient may wish to restrict access to their substance abuse data. What all of these notions of policy have in common is that they can be expressed as NGAC access control data and enforced through NGAC’s functional components.

An important NGAC virtue is its efficiency. Thanks to linear time algorithms, NGAC can compute access control decisions and conduct policy reviews quickly. These policy reviews can be of many types, including the determination of the set of resources for which a user has access or the minimal set of attributes needed to access a given resource.

NDAC

NDAC is a system that leverages NGAC (Ferraiolo et. al., 2017). In support of integration, NDAC controls access to DBMS resources as a middleware that sits between remote RP applications and local RP DBMSs, providing a standardized means of protecting databases down to the field level. NDAC uses NGAC policy elements and relations as a means of expressing policies. Among these are *object attributes*, which can be perceived as containers that group and characterize resources in diverse ways.

The NDAC process for expressing policy begins with an existing DBMS schema with columns and tables that are converted into NGAC-corresponding object attributes and assignments. Given that rows are also object containers, existing rows are automatically converted as well.

NGAC relations are further configured in formulating policy in terms of the created object attributes using NGAC’s administrative Application Programming Interface (API). Note, that these configurations only need to apply to resources that are of mutual interest to the federation. The configuration is stored as NGAC Access Control Data. Additionally, NDAC includes an *Access Manager* for trapping SQL queries from applications and a *Translator* for converting SQL queries to permitted SQL queries. Regarding this use case, a user may issue a query to fetch an entire record, and NDAC will translate (re-write) the query to restrict access to fields that are accessible by the user in accordance with regulatory requirements and patient consent expressed in terms of NGAC's configurations of DAC and MAC policy combinations and prohibitions (see Figure 1.).

Translation is achieved through policy reviews that compute the rows, columns, and tables that a user is authorized to access. With respect to read and write, reviews allow for the translation of Select and Update. With respect to administrative access rights (i.e., create and deletion object attributes), NDAC translates Insert and Delete queries.

Figure 1 shows the placement of NDAC's Access Manager, Translator, and NGAC Access Control Data in an authorization flow that involves an application and a target database.

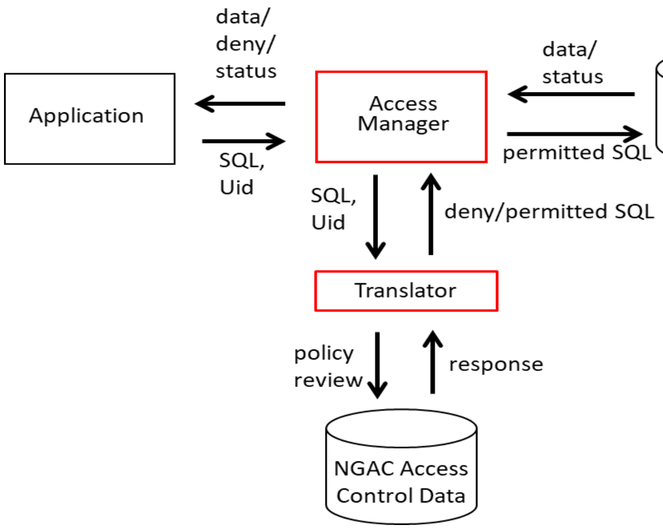


Figure 1. Placement of NDAC (in red) with respect to existing components

NDAC's authorization flow is as follows:

(1) The SQL statement from a user of the Application is intercepted by the Access Manager and sent to the Translator.

(2) The Translator converts the SQL statement from the user into an NGAC policy review.

(3) Using the results of the review, the Translator converts the user's SQL statement into either an access DENY or a permitted SQL statement that is sent back to the Access Manager.

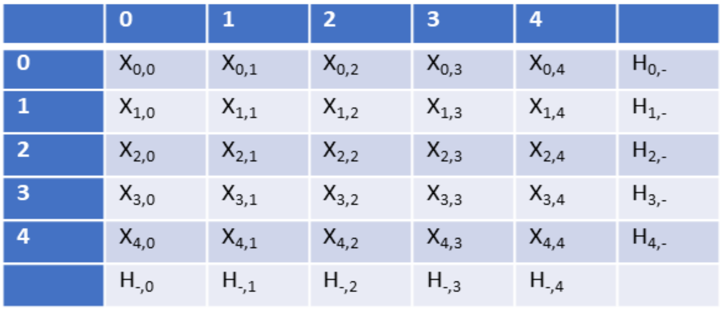
(4) In the case of a DENY, the Access Manager forwards the indication to the application user. Otherwise, the Access Manager submits the Permitted SQL statement to the Database.

(5) In the case of a Select operation, data is sent back to the Access Manager and forwarded to the Application and user.

Data Block Matrix

The integration makes use of a centralized *data block matrix* for storing and sharing user-to-attribute assignments. When a user is assigned to an attribute in the catalog, that assignment is also reflected in the block matrix. Subsequent changes to those assignments would also be reflected. A data block matrix is a distributed ledger technology (DLT) data structure that provides integrity protection like that provided by blockchains (i.e., trust) and allows for the controlled deletion or modification of data blocks in a permissioned system.

The block matrix is appealing to this integration due to the need and frequency for RPs to delete and modify attribute assignments. The block matrix resolves this problem with the goal of providing a more flexible design tool for distributed systems. The structure, shown in Figure 2, uses an array of blocks with hash values for each row and column (e.g., $H_{0,-}$ is the hash value for row zero). When a cell (e.g., $X_{2,1}$) is deleted (i.e., written over with zeros) or modified, only the hash value for that column and the hash value for that row are altered (e.g., $H_{2,-}$ and $H_{-,1}$). The integrity of the other blocks and hash values of their rows and columns are not affected. This is possible because no two consecutive blocks appear in the same row or column (Kuhn, 2019).



	0	1	2	3	4	
0	$X_{0,0}$	$X_{0,1}$	$X_{0,2}$	$X_{0,3}$	$X_{0,4}$	$H_{0,-}$
1	$X_{1,0}$	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$H_{1,-}$
2	$X_{2,0}$	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	$H_{2,-}$
3	$X_{3,0}$	$X_{3,1}$	$X_{3,2}$	$X_{3,3}$	$X_{3,4}$	$H_{3,-}$
4	$X_{4,0}$	$X_{4,1}$	$X_{4,2}$	$X_{4,3}$	$X_{4,4}$	$H_{4,-}$
	$H_{-,0}$	$H_{-,1}$	$H_{-,2}$	$H_{-,3}$	$H_{-,4}$	

Figure 2. Data block matrix

Within a permissioned system, establishing privileges for RPs to read from and write to the block matrix provides the basis for trust in the federation. By using NIST's NGAC reference implementation, administrative policy is imposed (e.g., who is authorized to create or delete user-to-attribute assignments within each RP). A mutually agreed upon governance policy might stipulate that only a member of the HR department can create or delete attribute assignments. However, in reality, these rules will be more granular and specific to the federation.

For a remote user to access local resources, two types of policies will likely come into play: DAC and MAC. Regarding MAC, each RP would be responsible for the configuration of MAC policies for protecting of resources in accordance with their own non-discretionary policies. Under such a configuration, users that might be assigned to roles such as "doctor" or "nurse practitioner" would be given rights to access portions of medical records under, for example, an RBAC policy (a type of MAC). The source and naming conventions for these roles come from the catalog and are also instantiated in the block matrix. Although users with assignments to MAC-related attributes stored in the block matrix, such as "doctor," reflect access rights to federation-wide resources, these access rights alone are not enough to access resources. To do so, the DAC policy, often referred to as a "need-to-know" policy, must also be satisfied. In effect, to gain access to local resources, either the RP or some external entity with control over those resource needs to establish consent. In support of a resource owner exercising their discretionary capabilities, the block matrix stores a user-to-username assignment for each MAC user in the federation.

Onboarding Remote Users

Central to the data sharing use case is consent for a healthcare provider to access a patient's medical record with potential prohibitions. Figure 3 shows a series of events that begins with a patient issuing consent with prohibitions regarding their medical record stored in DBMS-x of RP-x and consequently ends with a user u1 that is a Doctor in RP-y receiving access rights to portions of the patient's medical record. This involves the patient portal. The patient portal runs on behalf of a user (e.g., the patient) with both NDAC administrative privileges and capabilities to read block matrix information. The portal can be thought of as a user-friendly application for accomplishing tasks through the execution of those privileges. NDAC administrative privileges both grant a provider's discretionary access to the medical record and impose prohibitions. However, the provider may not be known to the NDAC system that protects the medical record. In this case, onboarding is a process of creating UIDs and user attribute assignments in an NDAC system to enable provider access.

As further shown in Figure 3, the patient portal serves as an intermediary between the block matrix and NDAC-x for querying the block matrix and onboard the provider. Figure 3 includes a cell of the block matrix to exemplify the process.

In this regard, the patient portal queries the block matrix for information, such as using the provider's proper name (e.g., John Smith) and RP (e.g., RP-y) in the search. Once the proper cell is identified, the patient portal establishes the uid (u1) in NDAC-x, assigns the uid to the Name attribute (i.e., John Smith), configures access rights for accessing the patient's record under the DAC policy, and establishes any specified prohibitions regarding u1.

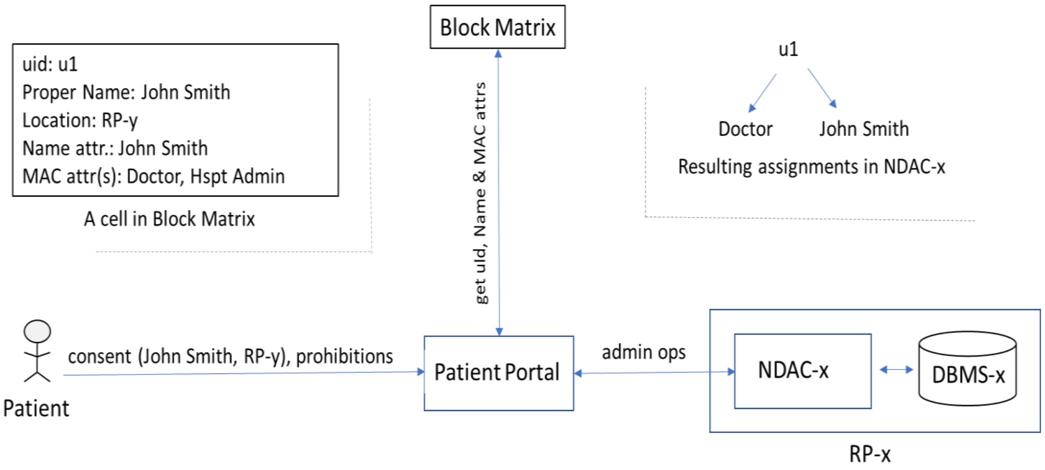


Figure 3. Sequence of events permitting policy-preserving access to database resources stored in RP-x by a user in RP-y

Recall that in order to access a patient record, the user (in this case, u1) needs to possess access rights under both DAC and any MAC policies. Also recall that MAC configurations in terms of MAC attributes are already in place in each NDAC system. However, the provider's MAC attribute has yet to be assigned. To create appropriate MAC attribute assignments, the patient portal instructs the NDAC system to conduct a policy review to determine the minimum set of attributes necessary to read the patient's medical record. Assuming the policy review identifies "doctor" as the MAC attribute, the patient portal queries the block matrix for the doctor MAC attribute associated with user u1. If found, the patient portal assign's u1 to the doctor attribute in the NDAC system. Note that u1's cell in the block matrix also includes the attribute "Hspt Admin" but pertains to access rights outside of the scope of the patient's consent.

Provisioning FIM, Block Matrix, and NDAC

Federated identity management (FIM) systems allow various applications in various enterprises to share user identities. The integration necessitates this need for establishing and authenticating user identities and establishing sessions across its applications and services. In a FIM, the onus of authenticating user credentials (e.g., via a password) is on an identity provider (idp), not the applications. So, when a user attempts to log into a specific application, the RP then communicates with the idp to authenticate the user. Several off-the-shelf solutions are available to meet this need.

This integration involves two types of RP services for establishing user identities and credentials along with the creation of attribute-associated information, as shown in Figure 4. Recall that each RP creates users and assigns attributes in accordance with an NGAC-enforced governance policy. In this capacity NGAC serves as an intermediary for managing attributes that are stored in the block matrix. The left side of Figure 4 indicates a user assigned to HR to meet these responsibilities on the part of RP-x. This effect involves an agent for the establishment of a uid (the user) in the FIM system and the management of user (uid)-attribute assignments in the block matrix.

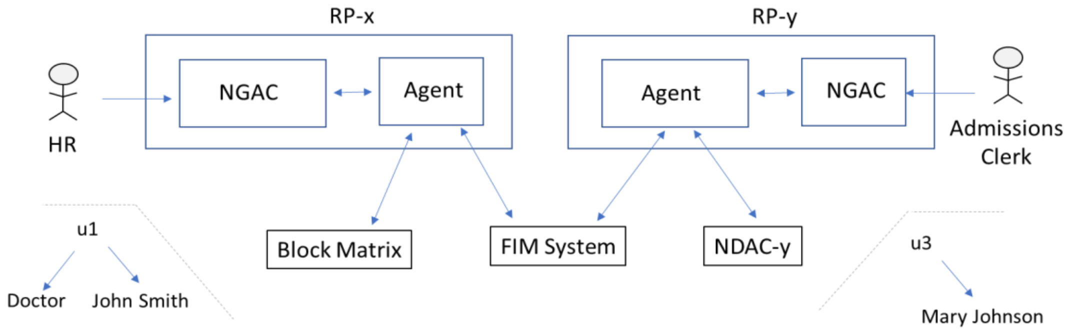


Figure 4. Establishing uids in FIM System, attributes in the Block Matrix, and DAC users in NDAC

When a new resource is created in the database of a RP of mutual interest to the federation, some user needs the NDAC capability to grant discretionary access to the resource. Under this use case, for example, that user would be a patient user with capabilities to grant access to their medical record. In the context of this use case, the right side of Figure 4 shows an NGAC user assigned to the role "Admissions Clerk" in RP-y. Although not shown, Admissions Clerk is associated with administrative capabilities to create users, name attributes, and assign those users to those attributes. As such, the Admissions Clerk may create u3 assigned to Mary Johnson. Through the assistance of an agent, the uid of the user is established in the FIM system, and the user-to-attribute assignment—along with the record name of the patient—is used to establish DAC capabilities in the NDAC system in RP-y. In this implementation, this would be achieved through an administrative routine (i.e., create DAC-user(user name, resource name)).

Federated Sharing of DBMS Resources

Assuming population of the block matrix, provisioning of the FIM system, and MAC configurations of RP NDAC systems, Figure 5 depicts an operational flow to enable consent-driven sharing of resources stored in heterogeneous DBMSs belonging to different RPs.

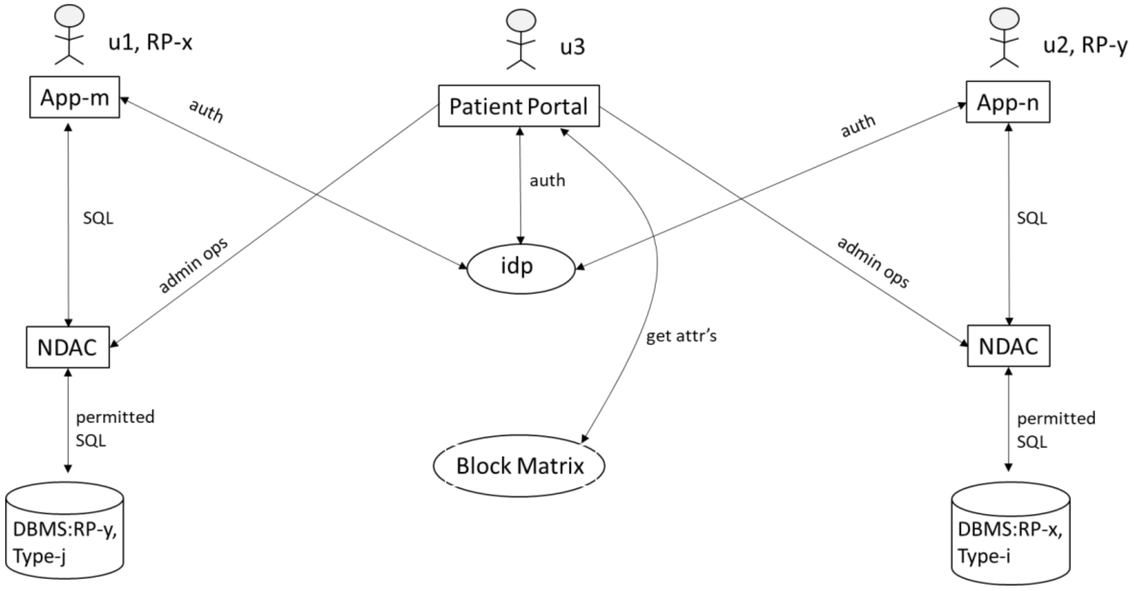


Figure 5. Operational Sharing of DBMS Resources

Figure 5 includes two users (u1 and u2) of different RPs, presumably healthcare providers, interacting with arbitrary database applications for accessing each other's DBMS resources. These DBMSs may be of different types and deploy different schemas. Also included is a user u3 (presumably a patient) interacting with the patient portal for issuing consent with potential prohibitions. Prior to performing these actions, each user authenticates their identities via the identity provider (idp). In collaboration with the block matrix, the patient portal—on behalf of u3—issues NGAC administrative operations to an NDAC system regarding the u3's medical record. Consequently, policy preserving (DAC, MAC, Prohibitions) access to u3's medical record is established for a specific healthcare provider. Once a user (healthcare provider) has been onboarded into an NDAC system of a RP, that user may issue SQL queries to the RP's NDAC system in accessing a medical record under the RP's consent, prohibitions, and mandated regulatory requirements.

Conclusion

The ability to share database resources is desirable for how it improves collaboration among alliances and enables the discovery of new strategies, findings, and observations. However, challenges persist regarding interoperability in the exchange of resources between DBMSs and the preservation of local access policies. This article presents an integration of two existing technologies that effectively solve these problems in a federation of RPs founded on an ABAC model. The integration is non-intrusive. It requires no changes to a RP's DBMSs, methods of authenticating users or authorizing access to local resources. This ease of deployment and its efficiency make this new infrastructure solution practical for large, real-world applications.

References

Ferraiolo, D., Chandramouli, R., Hu, V., Kuhn, R., “A Comparison of Attribute Based Access Control (ABAC) Standards for Data Service Applications,” NIST Special Publication 800-178, October 2016.

Ferraiolo, D., Gavrila, S., Katwala, G., Roberts, J., “Imposing Fine-grain Next Generation Access Control over Database Queries,” ABAC’17, Scottsdale, AZ, 2017
DOI: http://dx.doi.org/10.1145/3041048.3041050.

Kuhn, R., Yaga, D., & Voas, J. (2019). Rethinking distributed ledger technology. *Computer*, *52*(2), 68-72.

SNOMED International, “SNOMED CT Release File Specifications,” SNOMED CT Release File Specifications - Release File Specification (ihtsdotools.org), February 6, 2020.