Global Privacy Policy Technical Framework

Privacy Policy Primer

This project was supported by Grant No. 2009-DD-BX-K026, awarded by the Bureau of Justice Assistance. The Bureau of Justice Assistance is a component of the Office of Justice Programs, which also includes the Bureau of Justice Statistics, the National Institute of Justice, the Office of Juvenile Justice and Delinquency Prevention, the SMART Office, and the Office for Victims of Crime. Points of view or opinions in this document are those of the author and do not represent the official position or policies of the United States Department of Justice.

Document Purpose

This is a high-level primer for organizations looking to protect the security and privacy of the data they share with their information-sharing partners. This guide introduces the Global Privacy Policy Technical Framework, processes for defining and programming security and privacy policies, and solutions for implementing the components of the framework.

Version	Date	Description/Changes
1.0	3/17/11	Initial draft: Jim Cabral, MTG Consulting
1.1	4/8/11	Second draft: revisions by John Ruegg and Patricia Hammar
1.2	7/5/2011	Third draft: revisions by John Ruegg and Di Graski
1.3	9/6/2011	Final review and editing by John Ruegg
1.4	11/21/2011	Submission to BJA for review
1.5	11/28/2012	Publication on the Technical Privacy Training website

TABLE OF CONTENTS

Page

I. Executive Summary

The Global Privacy Policy Technical Framework enables the consistent enforcement of security and privacy policies across organizations that have formed a large information-sharing enterprise.

Implementation of the framework starts with the development of written security and privacy policies that comply with legislative, regulatory, and organizational requirements for the protection of personal and sensitive information.

The written policies can then be analyzed to extract the properties of users and resources, and the conditions and other rules that must be satisfied to permit the sharing of protected information. Encoding these policy statements into a standard language enables interoperability among organizations.

Finally, with some customization, open source and commercial solutions for federated authentication and authorization can be used to enforce the security and privacy policies and audit access.

II. Introduction

As information sharing in the public sector expands, it becomes increasingly important to find ways to use technology to help implement and enforce protections of privacy, civil liberties, and civil rights. Converting privacy policy to a form that is understandable to computers continues to be a significant problem and a high priority for government agencies. This online primer explains the steps needed to develop security, privacy, and information-handling controls for multi-agency information-sharing projects. While some technical examples are unavoidable, this primer provides definitions and examples to assist the non-technical reader.

A. Intended Audience

The intended audience of this primer includes everyone who is accountable for collecting, protecting, and appropriately sharing sensitive and private information, such as:

- **Agency executives** who are accountable under federal laws and regulations, state law, or organizational policies for protecting the security and privacy of their agencies' information.
- **IT directors** who manage and oversee the design, implementation, and operation of IT systems.
- **Enterprise architects** who develop and enforce business process and IT standards.
- **Policy analysts** who develop organizational security and privacy policies and multi-agency agreements.
- **Project managers, architects, and technologists** who manage, design, implement, or support IT.

While this primer should apply to any electronic exchanges of information between agencies, the guidance provided here will be particularly useful to agencies implementing exchanges based on the National Information Exchange Model (NIEM), the Global Federated Identity and Privilege Management (GFIPM) specification, and the Global Reference Architecture (GRA).

B. Intended Uses

This primer provides overview guidance on the steps for implementing the Global Privacy Policy Technical Framework described in the technical architecture document "Implementing Privacy Policy in Justice Information Sharing: A Technical Framework." The framework provides guidance for supporting the electronic expression of privacy policy and explains how to convert privacy policy so that it is understandable to computers and software. This primer covers five major tasks for completing an information-sharing project with privacy and sensitive information requirements:

- **Analyzing written security and privacy policies** to extract the requesting persons and organizations and information resources (the "nouns") and the security and privacy rules (the "verbs and prepositional phrases") contained in the policies. This policy analysis process includes several tasks, including:
  - » Defining the attributes and roles of requesters.
  - » Defining the attributes of information resources.
  - » Defining access, dissemination, confidentiality, retention, and other information handling rules.
- **Encoding the security and privacy rules** associated with each information resource into a programming language, such as the eXtensible Access Control Markup Language (XACML), that information systems can understand and share with other information systems.
- **Authenticating and authorizing users** to the information systems so that users can verify their identity and receive the information they are entitled to access.
- **Enforcing the security and privacy rules** by comparing the identity of the requesting user with the encoded security and privacy rules for the information being requested and taking any required actions as defined by the rules. Actions may include disclosure or non-disclosure of the information, or notification of an administrator or data owner regarding the request.
- **Selecting open source or commercial products** for performing any of the previous steps.

Technical readers who are designing, implementing, and supporting systems based on Privacy Policy Technical Framework should also consult the "Implementing Privacy Policy in Justice Information Sharing: A Technical Framework" document.

III. Defining Privacy and Security Policies

Policy, broadly construed, is a written set of rules governing the acceptable actions in a particular domain. Traditionally, procedure manuals have been the primary means for documenting policies. Procurement policy, personnel policy, records management policy, and many other policies are based on a set of organization rules and one or more local, state, tribal, federal, or international laws and regulations.

In today's information age, electronic data and documents can be rapidly exchanged among many organizations, each with its own, often conflicting, policies. The electronic sharing of personal information between organizations and the risk of inappropriate disclosure of personal information have stimulated concern about enforcing privacy laws and regulations governing disclosure of such information, including stiff financial or even civil and criminal penalties for violations of privacy policies. Personally identifiable information (PII) is broadly defined to be one or more pieces of information that, when considered together or when considered in the context of how the information is presented or gathered, are sufficient to identify a unique individual.

A. Identifying the Business Need

All public-sector organizations must comply with laws that require or restrict disclosure of information kept by government agencies about people, organizations, and their activities. Many state constitutions explicitly protect the public right to access information held by the government, and some state constitutions also protect privacy. Federal and state laws, such as the Freedom of Information Act (FOIA) and public records acts, specify when information in government records must be disclosed and under what conditions the information can be withheld. Other laws protect specific types of information – for example, medical or mental health information, education information, and information about children – or limit disclosure of information to specific groups, such as law enforcement.

Before a government agency can protect the privacy of its records, the agency should begin with a high-level inventory of the types of PII it collects and shares. Common types of PII include:

- **Behavioral** information, including anything a person does, such as engaging in hobbies and other activities.
- **Contact** information, including names, aliases, nicknames, home addresses, phone numbers, and e-mail addresses.
- **Criminal record** information, including arrests, charges, court judgments and sentences, and corrections.
- **Demographic** classifications, including race, religion, and sexual orientation.

- **Educational** background, including schools attended, degrees received, and skills acquired.
- **Employment** background, including past and current employers, employer contact information, job titles, and dates of service.
- **Financial** information, including bank accounts and balances and stock holdings.
- **Government-issued identifiers,** including tax identifiers and driver's license numbers.
- **Health** information, including past and current medical and psychological conditions, health care providers, treatments, prognosis, and DNA.
- **Juvenile/child** records, including incidents, court findings, custodies and placements, and services.
- **Location** where the person has been or currently is located. This is distinct from the person's home or work addresses.
- **Organizational** information, including membership in an organization.
- **Other identifiers**, including customer numbers, order numbers, and user identifiers.
- **Physical description** information, including images, dental records, biometrics, and scars, marks, or tattoos.
- **Political** preferences, including party affiliations.
- **Property** records, including real estate and vehicles.
- **Proprietary** information, including trade secrets.
- **Sealed** information or records expunged by a court.

In addition to collecting information about data subjects, or "subjects of record," the agency should also consider what information is collected about consumers of the agency's services (e.g., agency staff, partner agency staff, private organizations, public consumers).

The agency should also identify the clear business purpose for the collection, storage, or dissemination of the PII. If a clear business purpose cannot be identified or associated with a basis in local, state, tribal, or federal law, the agency should strongly consider whether that information should be collected or disseminated at all. Global provides Privacy Impact Assessment guidance to assist in this consideration.

B. Developing Privacy Policies

Once an agency understands the types of PII it must protect and knows the business purpose for using the data, the agency should develop privacy policies that document the rules for collecting and sharing the information where the policies do not already exist. The Global Privacy and Information Quality Working Group (GPIQWG) has developed the "Privacy Policy and Civil Liberties Policy Development Guide and Implementation Templates" to assist agencies in the development and customization of their privacy policies for their particular information-

sharing needs. The following table itemizes the major privacy policy topics addressed in the templates.

Privacy Policy Templates	
Reference	**Section Heading**
B.1.00	Statement of Purpose
B.2.00	Compliance With Laws Regarding Privacy, Civil Rights, and Civil Liberties
B.3.00	Definitions
B.4.00	Seeking and Retaining Information
B.4.10	What Information May Be Sought or Retained
B.4.20	Methods of Seeking or Receiving Information
B.4.30	Classification of Information Regarding Validity and Reliability
B.4.40	Classification of Information Regarding Limitations on Access and Disclosure
B.5.00	Information Quality
B.6.00	Collation and Analysis of Information
B.6.10	Collation and Analysis
B.6.20	Merging of Information From Different Sources
B.7.00	Sharing and Disclosure of Information
B.7.10	Sharing Information Within the Agency and With Other Justice System Partners
B.7.20	Sharing Information With Those Responsible for Public Protection, Safety, or Public Health
B.7.30	Sharing Information for Specific Purposes
B.7.40	Disclosing Information to the Public
B.7.50	Disclosing Information to the Individual About Whom Information Has Been Gathered
B.8.00	Information Retention and Destruction
B.8.10	Review of Information Regarding Retention
B.8.20	Destruction of Information
B.9.00	Accountability and Enforcement
B.9.10	Information System Transparency
B.9.20	Accountability for Activities
B.9.30	Enforcement
B.10.00	Training

Developing privacy policies applicable across multiple information systems, data resources, and organizations is complicated by differences among agencies in governance, accountability, and

organizational policies. For instance, assume a scenario in which a police department and a prosecutor's office each have their own privacy officers and policies. If a paralegal in the prosecutor's office is unaware of the police department's privacy policy and inadvertently discloses PII obtained from the police in violation of the police department's privacy policy, who should be accountable for the disclosure? The paralegal? One or both of the privacy officers?

The Privacy Guide and Templates provide sample language to help avoid these situations and to guide the response when they do occur. The templates in Section B.7, Sharing and Disclosure of Information, and the guidance in Section C, Provisions for Multi-Agency Agreement for an Information-Sharing System, will be of particular interest to agencies sharing PII with partner agencies, the public, and the subject of record.

C. Developing Security Policies

Privacy can only be enforced in a secure environment. Organizations that need to protect sensitive or private information need to ensure that they have sufficient security controls, including policies, processes, technology, and staffing resources to protect the agency's human and technical assets.

In 2004, the Global Security Working Group (GSWG) published Applying Security Practices to Justice Information Sharing to educate agency executives and managers in good, basic, foundational security practices that they can deploy within their enterprise and between multiple enterprises. The site specifically addresses the security requirements for agencies participating in four common information-sharing architectures, including a joint task force model; a centralized information repository model; a peer group model; and an interconnection services network model. In addition, the site contains background information, overviews of best practices, and guidelines for secure information sharing in 15 disciplines that are common to many information security architectures.

The first discipline, governance, begins with an assessment and classification of the level of risks or liabilities incurred if there were a breach to the confidentiality, integrity, and availability of the agency's information systems and data. These risks include compliance with laws, regulations, and rules that apply to the organization and to the information being used. For example, the Federal Bureau of Investigation (FBI) and Nlets – the International Justice & Public Safety Information Sharing Network – have baseline security requirements for agencies accessing their information systems. Based on the risk assessment, the agency should create, develop, and implement controls, including security and privacy policies, to mitigate each risk.

The Applying Security Practices Web site identifies recommended topics of security policies for each security discipline. In some cases, the site provides references to sample policies. Other sources of sample security policies and best practices for agencies include:

- SANS Institute Security Policy Templates – A guide to creating security policies and a collection of generic security policy templates that can be customized for each agency.
- US Department of Commerce National Institute of Standards and Technology (NIST) Computer Security Resource Center – A set of security guidelines and regulations intended for federal agencies but that are also useful as models for state and local governments. NIST security publications include Federal Information Processing Standards (FIPS), special publications on a variety of security topics, and periodic security bulletins on emerging threats and technologies.
- FBI Criminal Justice Information Services (CJIS) Security Policies – A set of security requirements for any agency connecting to the systems of the FBI CJIS Division. Section 5 defines policy requirements for these agencies, and APPENDIX D-1 provides a copy of the CJIS User Agreement as a sample information exchange agreement.

Once an agency has created its privacy and security policies, it might use the process described in the next section to analyze these policies and then program technology for enforcement of the policies. While some security disciplines are beyond the scope of the policy analysis process and this primer – for example, physical security and protection of stored data ("data at rest") – all aspects of information security architecture are important and should be carefully considered and implemented. Remember that the security and privacy of a system is only as strong as its weakest component.

IV. Analyzing Privacy and Security Policies

This section describes the steps required to translate written privacy, security, and information handling policies related to a group of multi-agency information exchanges into a set of programming rules and a framework for implementing and enforcing the rules for information systems. The key to this approach is to define a set of security and privacy management processes which can be implemented consistently across all organizations participating in the information-sharing community. The fundamental processes of this approach include:

- Scoping Policy Enforcement
- Defining the Privacy and Security Architecture
- Tagging Content
- Identifying Requesters
- Developing Electronic Policy Statements
- Enforcing Policies
- Auditing Access

This section describes a high-level approach for each of these processes. The next section, Implementing Privacy and Security Policies, will describe technologies and solutions useful in implementing each of these processes.

A. Scoping Policy Enforcement

The first step in analyzing privacy and security policies is to understand the constraints of the policies. That is, what information resources and users are within the scope of the information exchanges being implemented? Limiting the scope of the policy enforcement initiative, by omitting security and privacy policies not directly relevant to the information resources and users in the target information exchanges, can significantly reduce the complexity and required effort of the enforcement implementation.

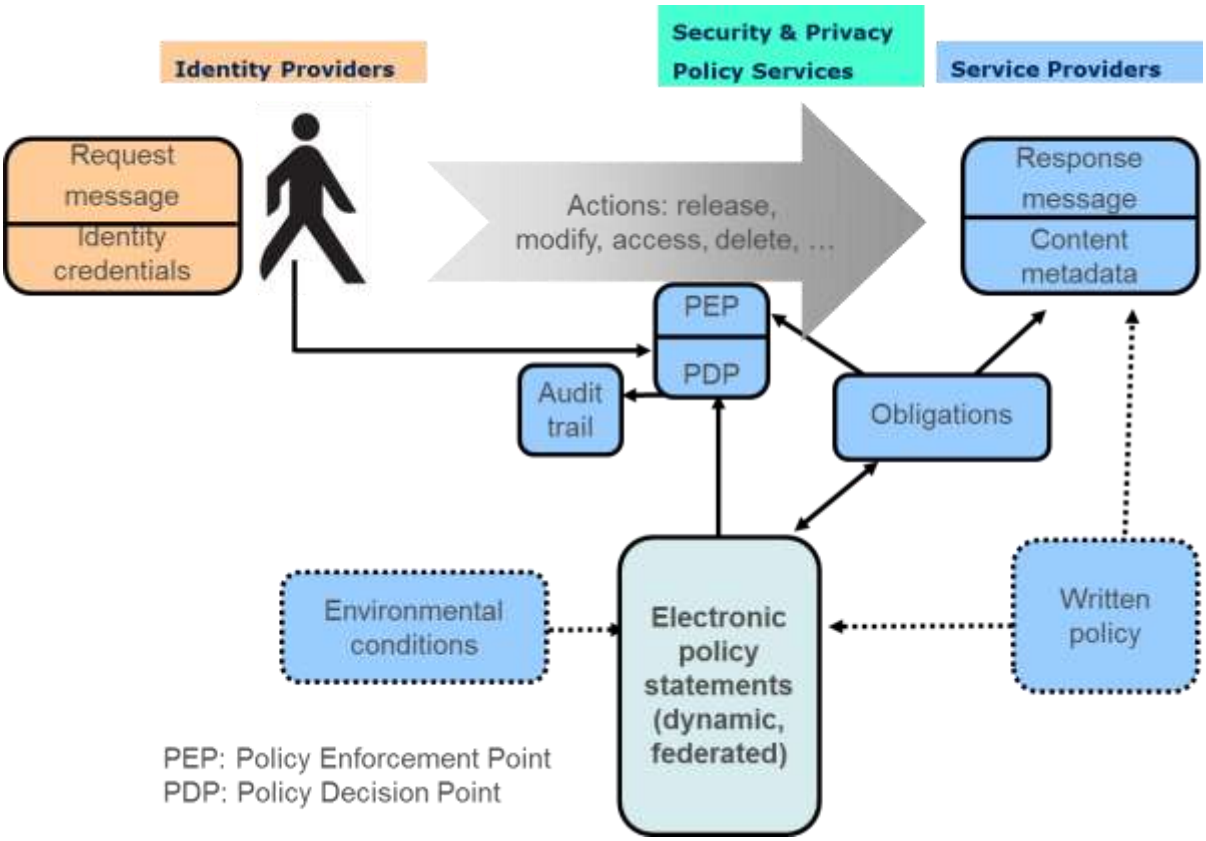
B. Defining a Privacy and Security Architecture

After defining the scope of the policy enforcement system, implementers of an information-sharing system must agree upon an interoperable architecture in which privacy and security policies will be shared and enforced by all participants in the system. This primer assumes the use of a security and privacy architecture based on the Global Privacy Policy Technical Framework.

A privacy policy enforcement framework for multi-agency information sharing must be able to:

- Express policies and applicable laws in a structured specification language.
- Convert agency-specific policy terms into a standards based vocabulary of XML processing rules and data elements (e.g., NIEM- or GFIPM-specific XML tags, attributes, or metadata).
- Verify the identity attributes and digital credentials of information providers and information requesters.
- Process the XML-based electronic policy rules.
- Implement information protection mechanisms that are consistent with monitoring compliance.

1. Global Privacy Policy Technical Framework



The Global Privacy Technical Framework, as illustrated in the above diagram, can be described with the following simple use case:

1. At the direction of a **requester** (represented by the person icon in the diagram), a software agent sends a **request message**, which includes the **identity credentials** of the **requester** with the intention to perform some **action** (e.g., read) on some **content** (e.g., a social security number) for an intended business purpose (e.g., a law enforcement investigation).
2. A Policy Enforcement Point (**PEP**) receives the request and passes it to a Policy Decision Point (**PDP**) for evaluation.
3. The **PDP** compares the **request message**, including the **identity credentials** of the **requester,** the requested **action,** and the intended purpose, against the **content metadata**, **environmental conditions,** and **written policies** encoded as **electronic policy statements.**
4. Based on **electronic policy statements** that match the **request message**, **content metadata,** and **environmental conditions**, the PDP evaluates the request and directs the PEP to permit or deny the requested **action** (e.g., read) and to perform zero or more supplemental actions (also known as **obligations**). Examples of obligations include:
    - ✓ redact the **response message**
    - ✓ notify an administrator of the information **request**
    - ✓ log the request to build an **audit log**

This use case illustrates only the simplest implementation of the framework: read-only access by a user to a small set of records in a single information system protected by one set of electronic security and privacy policies. The framework is also designed to scale to support more complex actions (e.g., writing or updating multiple records) and federated queries across many organizations, each with its own security and privacy policies.

2. Relationship to Other Privacy Management Frameworks

While the application of privacy controls to information sharing is novel, the Global Privacy Policy Technical Framework is analogous to other privacy management frameworks and standards designed for other domains. Other privacy management frameworks include the International Security Trust and Privacy Alliance (ISTPA) Privacy Management Reference Model (PMRM) and the work of the OASIS Cross-Enterprise Security and Privacy Authorization (XSPA) Technical Committee for the interoperable exchange of healthcare privacy policies, consent directives, and authorizations.

C. Tagging Content

Describing the type and attributes of information being exchanged may be required to enforce a security or privacy policy. For example, a policy may require you to notify the source agency that originally collected a piece of information whenever that information is queried. The identity of the source agency is “metadata,” information that describes additional attributes about the collected information. Without that metadata, you could not comply with the notification policy.

"PII" could be metadata to describe a piece of information such as a Person Name or Social Security Number. Metadata tags can be used to describe an entire database, a record within a database, or specific fields within a database. The tags used to describe data will vary depending on the information exchange and the policies that need to be enforced.

1. Content Metadata

Content tags are called metadata, or properties of the data, such as:

- **Business Purpose Metadata** describe the business purposes for which personally identifiable information (PII) was originally collected. This metadata can also be used to define the business purpose for a request. The primary list of business purposes comes from the Business Areas, lines of business (LoBs), and subfunctions defined in Figure 11 and Section 4.1 of the Federal Enterprise Architecture Business Reference Model, but the purpose may also be a business-specific purpose (e.g., identifying subjects, officer safety, monitoring services).
- **Data Type Category Metadata** describe the types of PII, such as contact information, medical records, or criminal records. (See the list of types of PII on pages 5 and 6.) These categories are used to distinguish groups of collected data that need to be treated differently from a privacy point of view.
- **Association Metadata** describe privacy-related associations between a subject and other persons or organizations. For instance, attorneys and healthcare providers may have access to certain private information about their clients.
- **Data Classification Metadata** describe the level of authorization required to view certain data (e.g., commercial, counter-terrorism, criminal intelligence). Privacy policies may make exceptions for certain classifications for reasons of national security or counterterrorism.
- **Data Quality Metadata** describe the information's source reliability (e.g., reliable, unreliable, unknown) and content validity (e.g., confirmed, probable, doubtful). Privacy policies may restrict collection of or access to information that is unreliable or invalid.
- **Source Metadata** describe the origin of the data, including the source, the source agency, subject, submitter, submitting agency, and the dates and times the data were gathered and submitted.

The specific process for tagging content will vary between organizations and systems, but it is generally preferable to collect this metadata at the time the information is initially collected and to assign the tags automatically whenever possible. For instance, most law enforcement information is now captured in the field using mobile devices, including handhelds and in-car mobile computers. Health care providers and human services caseworkers are rapidly moving

toward the collection of case management data on mobile devices, too. Ideally, the software and forms on the mobile devices would automatically assign the appropriate content metadata such as business purpose, data type, and source when the data are first entered into the mobile device.

2. NIEM

Content metadata must remain associated with content as it is exchanged from system to system. For instance, as an incident report is transferred from law enforcement to the prosecutor, any privacy and security-related content metadata and obligations should be conveyed from the law enforcement record management system to the prosecutor's system. Therefore, all agencies participating in the same information-sharing community must use a common vocabulary of metadata for both tagging their content and referring to the content in their electronic policy statements. The NIEM, a partnership among the U.S. Department of Justice (DOJ), the U.S. Department of Homeland Security (DHS), and the U.S. Department of Health and Human Services (HHS), has emerged as the standard XML vocabulary and development methodology for describing intergovernmental message content. It is designed to develop, disseminate, and support enterprise-wide information exchange standards and processes that can enable jurisdictions to effectively share critical information in emergencies, as well as support the day-to-day operations of agencies throughout the nation. NIEM enables information sharing, focusing on information exchanged among organizations as part of their current or intended business practices.

The NIEM defines many of the elements and attributes that would be required for tagging content in a privacy management system. For instance, the NIEM includes metadata tags for association, source, and data quality, as well as a metadata type for extending the NIEM to create additional content metadata tags.

Several projects have successfully used the NIEM to enforce security and privacy of content. For instance, a recent Program Management-Information Sharing Environment (PM-ISE) project at NIST established access rules based on a NIEM Information Exchange Package Documentation (IEPD). More information on the NIEM, including a repository of exchange specifications, is available at http://www.niem.gov.

In addition, the Justice Information Exchange Model (JIEM) 5.0 provides capabilities to associate metadata from the privacy framework with information resources, requests, and responses, including NIEM IEPD.

3. GFIPM Metadata

The Global Federated Identity and Privilege Management (GFIPM) Metadata 2.0 specification, described in detail in section D.4., also defines metadata tags for Subjects, Roles, Actions, Obligations, and Resources that leverage existing NIEM vocabulary where available.

D. Identifying Requesters

Individuals (or organizations), internal or external to the information-sharing community, must have identity credentials that can be verified in determining their rights to access or perform operations on information covered by a privacy policy.

1. Requester Metadata

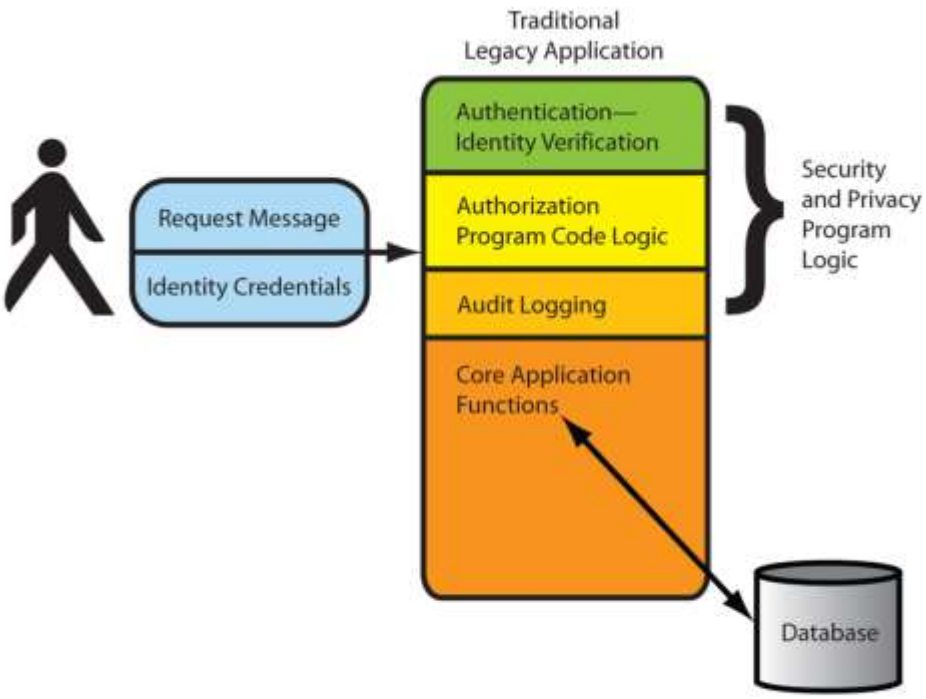
The category of metadata-defined properties about *requesters* (e.g.e.g., roles) can be used to classify and then make authorization decisions about their access to restricted data. Requesters' attributes may include:

- **Name** of the requester.
- **Organizational affiliations** of the requester, including requester's employer.
- **Contact information,** including the address, phone, or e-mail address of the requester.
- **Job title** of the requester.
- **Level of government** that employs the requester (e.g., federal, tribal, state, county, or municipal).
- **Role(s)** at the requester's employer (e.g., sworn law enforcement officer, child protection investigator, prosecutor, judge, juvenile probation officer, corrections officer).
- **Rights** and privileges of the requester (e.g., security clearance, trained on criminal history access rules including 28 CFR part 23, medical license).

The identities of requesters may be verified either through **direct authentication** to the source of the content being requested (the service provider [SP]) or through **federated authentication**.

2. Direct Authentication

In a **direct authentication** model, the requester must have an account with the SP and must provide the SP issued/certified credentials (e.g., SP issued userid/password, SP issued token, or SP issued or certified X.509 digital certificate) along with the request. Direct authentication is familiar to most users, as the login mechanism to most Web sites is a password that is registered with that site. Direct authentication requires each SP to manage accounts for every requester and requires each requester to remember his or her credentials for every site used. In addition, requesters must log in to each site separately - direct authentication does not support single sign-on (SSO). The following diagram illustrates a traditional legacy application performing direct authentication.



Stage 0—Traditional Legacy Application

3. Federated Authentication

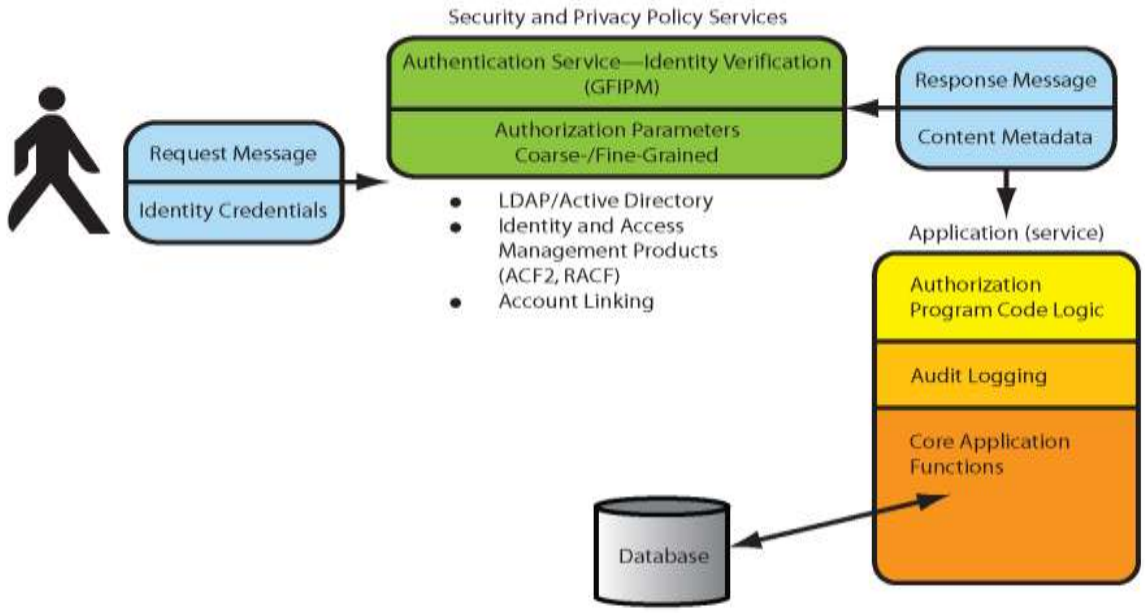
In a **federated authentication** model, the requester must have an account and log in with an identity provider (IDP), which can verify the identity of the requester and vouch for his or her identity to multiple internal and external SPs. When the requester attempts to access a SP, the IDP will automatically provide the requester's identity credentials to the SP. Federated authentication is typically implemented in an enterprise environment as a login to an Identity and Access Management (IAM) solution such as Microsoft Active Directory or a Lightweight Directory Access Protocol (LDAP) service. More detail on IAM solutions is provided in the next section of this primer.

The advantage of federated authentication to SPs is that they only need to communicate with trusted IDPs and do not need to create and manage individual requester accounts within their systems. The advantage of federated authentication to requesters is that they only need to log in once to their IDP and do not need to remember separate credentials (e.g., username/passwords) for each SP.

4. Global Federated Identity and Privilege Management

Identity federations require multi-agency governance structures and all agencies participating in a federation must use common technical standards in the implementation of IDPs and SPs.

Global Federated Identity and Privilege Management (GFIPM) is a program of the U.S. DOJ and the U.S. DHS that seeks to develop secure, scalable, and cost-effective technologies for information sharing within the criminal justice and intelligence communities. GFIPM provides guidelines for federated authentication and for expressing user identity credentials in XML. GFIPM technical standards and implementation guidance documents can be found at http://gfipm.net/.



Stage 1—Authentication moved from Legacy Application to Authentication Service

The following "Stage 1" diagram illustrates an evolution of the legacy application shown in the previous example to support federated authentication using GFIPM.

The GFIPM technical standards address many of the identity verification requirements of the Global Privacy Policy Technical Framework, including support for identity attributes that are time dependent or are calculated based on other attributes.

E. Developing Electronic Policy Statements

Security or privacy policies written in a form readable by humans must be converted into access control statements that are understood by computers. This section describes alternative technologies for implementing access control and an approach to converting security and privacy policies into electronic policy statements that are interpretable and enforceable in software.

1. Access Control Technologies

Historically, security and privacy policies have been implemented through access control lists (ACLs) and Role-Based Access Control (RBAC). Recently, these technologies have been superseded by Attribute-Based Access Control (ABAC) and Policy-Based Access Control (PBAC) that address many of the scalability and management issues of the older technologies. This section summarizes the differences between these technologies based on a recent "Survey of Access Control Methods" conducted by NIST.

ACLs

ACLs, mappings between each resource (e.g., a file) and the users and groups of users with access to that resource, are the oldest and most basic form of access control. Many modern operating systems, including Windows and UNIX variants and databases make use of ACLs at some level.

However, access control mechanisms used to protect system resources have become more complex in recent years. ACLs can also be difficult to manage in an enterprise setting where many people need to have different levels of access to many different resources. Selectively adding, deleting, and changing ACLs on individual files, or even groups of files, can be time-consuming and prone to error.

RBAC

In RBAC, access to a resource is determined based on the relationship between the requester and the organization or owner in control of the resource; in other words, the requester's role or function rather than simply their identity will determine whether access will be granted or denied. This overcomes the scalability issue of ACLs by allowing one set of access control permissions on a particular resource to be set once for all members with the same role. Most modern operating systems, including Windows 2000 and later, enterprise applications (e.g., Microsoft Exchange), and middleware (e.g., Microsoft Active Directory) include native support for RBAC. For instance, roles available in Windows operating systems and applications are the "Administrator" group, members of which have complete control over the operating system, "Power Users," who have fewer privileges than administrators, but still operate with elevated privileges, and "Users," who have limited privileges on the system.

While RBAC has many advantages over ACLs, putting individuals into categories based on roles makes it more difficult to define access controls specific to each person. It is often necessary to create more specific versions of roles or devise other mechanisms to exclude specific individuals who generally fit a particular role but should have only a subset of the full rights accorded to other members of the role. For example, an "Administrator" role might adequately describe all administrators in a small organization, but in a large organization with subsidiaries in multiple locations, there would likely be a need to segregate IT administrators

across the various locations. Otherwise, opportunities would exist for administrators in one location to have unnecessary or undesired access to particular systems in another location.

ABAC

The above example illustrates the need to differentiate individual members of a group and to selectively allow or deny access based on a granular set of attributes. ABAC was designed to fulfill this requirement. ABAC is an access control method wherein the access control decisions are made based on a set of characteristics, or attributes, associated with the requester, the environment, or the resource itself. Each attribute is a discrete, distinct field that software Policy Decision Point (PDP) can compare against a set of values to determine whether to allow or deny access. The attributes do not necessarily need to be related to each other, and in fact, the attributes that go into making a decision can come from disparate, unrelated sources. Attributes may include the date an employee was hired, the projects on which the employee works, the location where the employee is stationed, the employee's role in the organization, or any combination of the above.

ABAC enables the IDP to provide attributes necessary for the SP to audit the access and provide additional attributes to make access control decisions based on the role (RBAC) and other attributes to provide different access rights to individuals who share the same role. For example, an individual who asserts they are a Sworn Law Enforcement Officer (SLEO) and asserts they have 28 CFR part 23 training could be granted different access than another individual who is also a member of the SLEO role but does not have the 28 CFR part 23 attribute.)

A key advantage to the ABAC method is that there is no need for the requester to be known in advance to the system or resource to which access is sought. As long as the attributes that the requester supplies meet the criteria for gaining entry, access will be granted. Thus, ABAC is particularly useful for situations in which organizations or resource owners want unanticipated users to be able to gain access as long as they have attributes that meet certain criteria. This ability to determine access without the need for a predefined list of individuals that are approved for access is critical in large enterprises where people may join or leave the organization arbitrarily.

Unlike RBAC and ACLs, most operating systems do not inherently support the ABAC method. Instead, such access control is most often implemented at the application level, with an intermediary application (e.g., Policy Decision Point) that helps to mediate access between a user or application and the resource to which access is requested. For relatively simple implementations, applications often allow access based simply on attributes provided in the request. In more complicated environments, directory services infrastructure usually provide some of the attributes that go into making a decision including organizational or personal information (e.g., role).

One limitation of the ABAC method is that, in a large environment with many resources, individuals, and applications, there can be disparate naming conventions and definitions for the attributes and a variety of access control mechanisms among the organizational units. It is often necessary to harmonize access control attribute definitions across the enterprise in order to meet enterprise governance requirements.

PBAC

PBAC enables organizations to have a more uniform access control method throughout the organization than is possible with ABAC alone. Most organizations have some kind of policy and governance structure in place to ensure the successful execution of the organization's mission, to mitigate risk, and to ensure accountability and compliance with relevant law and regulations. With the institution of new regulations and legislation, such as the Health Insurance Portability and Accountability Act (HIPAA), many agencies are being forced to implement stricter policies and uniform controls across the enterprise in order to stay in compliance and support consistent policy enforcement audits. PBAC is an emerging technology that seeks to help enterprises address the need to implement standard access control rules based on policy and governance requirements.

PBAC provides harmonization and standardization of ABAC across multiple units within the enterprise and with information-sharing partners. PBAC combines attributes from the resource, the environment, and the requester with information on the particular set of circumstances under which the access request is made, and uses rule sets that specify whether the access is allowed under organizational policy for those attributes under those circumstances. In an ABAC-only method, the attributes required to gain access to a particular resource are determined on a local level and can vary greatly from one organizational unit to the next. For example, one organizational unit might determine that access to a sensitive document repository requires credentials with a username, organizational role, and password; another unit might require that the credentials necessary to access its repository also include a digital certificate issued by a trusted Certificate Authority. If documents are transferred from the latter repository to the former one, they lose the protection afforded by the digital certificates, and thus can be more easily compromised. Under the PBAC method, the organization would likely have one policy governing access to a resource and this policy would be enforced uniformly for all attempts to access the resource, no matter where the resource was housed at any given point.

Although PBAC is based on ABAC, it requires more enterprise-level infrastructure, including databases, directory services, and other middleware and management applications, since the attributes and their definitions have to be maintained across the enterprise.

Policy Authoring Language

PBAC requires a mechanism to specify policy rules in unambiguous terms. Policies must be defined in a consistent Policy Authoring Language (PAL); otherwise, there is the potential for

unintended, unauthorized access to a resource with which a particular policy is associated. The XML Access Control Markup Language (XACML), a XML-based PAL, was developed as a way to specify access control policy in a machine-readable format. XACML also supports electronic policy statements that are federated (i.e., include rules that are owned and managed by external organizations) or dynamic (i.e., change rules as a result of events or environmental condition).

One method of converting written policies into a PAL such as XACML is with an intermediary representation called a "Policy Matrix." The Policy Matrix approach, developed by Patricia K. Hammar and K. Krasnow Waterman for the U.S. DHS, has been used successfully within the U.S. DHS and the Florida Department of Law Enforcement Fusion Center to analyze and define many of the security and privacy laws and policies governing these agencies.

The Policy Matrix applies a technique to analyze written laws, regulations, and policies so they can be broken down into discrete segments that can be read by a computer. The Policy Matrix reflects the written policy rules on allowing information to be collected, retained, shared, or destroyed based on a defined set of user and data attributes in relation to those laws, regulations, and policies. The policy matrix can be used to analyze many different types of regulations with relative ease. The Policy Matrix does not interpret the law, or attempt to draw a conclusion from any given regulation; it simply generates a rule or action reflecting the law as written. This method further helps a reviewer identify problematic areas dealing with data or with policies concerning data privacy.

2. Policy Matrix Methodology

This section summarizes the Policy Matrix methodology based on an unpublished status report[1] of an implementation of the Policy Matrix approach by PKH Enterprises in the Florida Department of Law Enforcement Fusion Center.

Language is naturally composed of discrete segments – nouns, verbs, adjectives, and adverbs. A policy matrix rule is essentially a map that identifies the relevant segments and assigns them to the appropriate function in a rule. These rules are derived from statutes, laws, guidance, and statutes ("regulations") on a line-by-line basis, allowing a regulation to be examined in detail, quickly, and with context.

Through the Policy Matrix analysis, the reviewer seeks to answer the following questions:

- Which user or entity is interacting with the data?
- What actions can be performed with the data?
- What data are being processed?

[1] "Current Status of Policy Matrix Implementation," PKH Enterprises, October 1, 2010.

- What is the source of the data?
- Who is the potential receiver of the data?
- Are there general authorizations, caveats, or obligations regarding the data?
- What precedence or linkages exist among multiple regulations for the same data?

To answer these questions, the Policy Matrix provides a worksheet structure to guide the reviewer in an analysis of the policies and to facilitate searching and filtering. The worksheet is made of the following groups (in *italics*) and columns within groups:

- *Party Subject to Rule* (person or entity that is accessing, updating, or collecting the *Target Resource*)
  - » Party Subject to Rule **[Subject(s)]**
  - » Attribute of Party Subject to Rule **[Subject Information Context]**
  - » Person Context **[Subject Condition(s)]**
- *Subjects Involved in the Data Transaction* **(only if** different from the *Party Subject to Rule*)
  - » Releasing Entity **[Origin Subject(s)]**
  - » Attribute of Party Subject to Rule **[Origin Subject Information Context]**
  - » Person Context **[Origin Subject Condition(s)]**
- *Receiving Entity* **(only if** different from the *Party Subject to Rule***).**
  - » Receiving Entity **[Target Subject(s)]**
  - » Attribute of Party Subject to Rule **[Target Subject Information Context]**
  - » Person Context **[Target Subject Condition(s)]**
- *Rule Action*
  - » Share, Delete, Manage Policy, System Develop **[Action]**
- *Rule Activity*
  - » Permitted, Denied, Prohibited **[Denied/Permitted by Statute/Policy]**
- *Data Resource Subject to Rule*
  - » Data Category **[Target Resource(s)]**
  - » Special Data Category **[Other Resource Context]**
  - » Data Context **[Other Resource Conditions]**
- *Circumstances in Which the Rule Applies*
  - » Authorized Purpose **[General or Action Policy Conditions]**
  - » Rule/Action Conditions, Triggers and Obligations **[Obligations and Environments]**

- *Administrative Information*
  - » Precedence **[Federal/State/Tribal/County/City Constitution, Statute, Case Law, Executive Order, Regulation, Guidance or Policy]**
  - » References **[Document, Citation, Record Number and Date]**
  - » Linkages **[Linked Rule, Record and Reason]**
  - » Policy Matrix Editors **[Record Author and Reviewer]**

Each worksheet row allows a regulation to be read in a consistent and repeatable manner. For instance, a regulation can be read as a Policy Matrix rule statement as follows:

> *A **Subject(s)** with **Subject Information Context and Conditions** is **Denied/Permitted by Statute/Policy** to perform **Action** on data/information in **Target Resource** which (can/does) include **Target Resource Context and Conditions** disseminated from **Origin Subject(s) with Information Context and Conditions** to **Target Subject(s) with Information Context and Conditions** following these **General or Action Policy Conditions** in/and **Environment** with triggered/executed set of zero or more **Obligations.***

The Policy Matrix approach is particularly useful when working with the NIEM, GFIPM, and XACML. A Policy Matrix breaks down a regulation into metadata about the user (e.g., GFIPM metadata), the data (e.g., NIEM-conformant schemas), and the actions and dissemination (e.g., XACML rules). When appropriate, the values for certain columns in the Policy Matrix (e.g., Rule Action, Rule Activity) are limited to acceptable values defined in the Global Privacy Policy Technical Framework and the GFIPM program.

3. Defining Policy Granularity

As in all policy design decisions regarding information security, the level of risk involved in the erroneous release of information needs to guide the level of authentication and authorization rules that are appropriately developed in the electronic privacy policy. An important design consideration in developing electronic privacy policy is determining the level of granularity the policy needs to address. In general, the more granular the policy becomes, the more costly it will be to implement for both the information SP and the requester. Some policies could be so complex that they require an interim manual step for a decision-maker to evaluate whether the request will be granted or denied. The same granularity rules apply to metadata. A policy written to permit law enforcement personnel access is at a much higher level of granularity and reuse than a policy written to grant access to a detective or investigating probation officer.

Applicable levels of granularity for privacy-related authorization services are defined as follows:

Coarse-Grained Authorization refers to RBAC authorization of subjects within specific requester *categories* and granted access to coarse-grained data objects. Familiar examples include role-based access to intelligence applications, CJIS databases, unclassified documents, and incident reports. The user category gives access to all unclassified documents or database records within an application or service. These coarse-grained authorization rules have traditionally been embedded in application logic. For example:

> *All Law Enforcement Officers with a CJIS Law Enforcement ORI may read any record in the Wanted Persons Database, provided the agency logs each access and keeps the audit log for 3 years.*
>
> *Requester Metadata (Role) = Law Enforcement Officer with LE ORI*
> *Action = Read*
> *Resource Metadata (tag/label) = Wanted Persons Database*
> *Obligations = Log the query and keep audit log for 3 years*
> *Business Purpose = Criminal Investigation*

This is a coarse-grained authorization because access is controlled to a coarse-grained data object: all the records of the Wanted Persons Database.

Fine-Grained Authorization refers to ABAC or PBAC authorization of requesters within specific *categories* who are granted limited access to specific *data resource categories* of resources based on both the requester *category* and the *data resource category*, including an implied or explicit *business purpose*. For example:

> *Law Enforcement Officers with a CJIS Law Enforcement ORI may modify a record in the Wanted Persons Database only if the Officer ORI matches the Wanted Persons record creator ORI, provided the agency logs each access and keeps the audit log for 3 years.*
>
> *Requester Metadata (Role) = Law Enforcement Officer with LE ORI*
> *Action = Modify*
> *Resource Metadata (tag/label) = Wanted Persons Database Record*
> *Condition = Only if Requester ORI matches Wanted Persons record creator ORI*
> *Obligations = Log the query and keep audit log for 3 years*
> *Business Purpose = Wanted Person Apprehended*

This is a fine-grained authorization rule because the Action (Modify) is limited to specific records in the Wanted Persons Database, as opposed to any record in the Wanted Persons Database.

Custom Authorization refers to authorization based on policies that are so stringent or complex that they cannot be readily defined using a standard set of *requester categories* and

data resource categories. These policies are not reusable and therefore have little or no cost benefit savings from automation.

It is unlikely that an enterprise would be able to support fine-grained or custom authorizations without first investing in and deploying coarse-grained authorization.

4. Rule and Context Metadata

Processes and metadata for tagging content and identifying requesters were described earlier in sections C and D. Policy analyses must also include the following metadata to describe the rules and context expressed within each policy statement:

- **Actions** define what the creator or requester of information can do with the information. Privacy rules typically define whether a requester can perform the "read" action on the data. However, a framework intended to support security must also govern other types of actions, including "create," "update," and "delete."
- **Subject, Resource, and Policy Conditions** are expressions that evaluate the context of a request for data (e.g., the person-of-interest must be in detention, and the requester category must be Law Enforcement) to determine whether the information can be shared. The SEARCH JIEM includes a list of processes that represent the status of the person-of-interest at the time of the request. The JIEM also provides hundreds of conditions for describing the context of many justice information exchanges.
- **Obligations** define retention, dissemination, audit, and notification rules of the information Service Provider (SP) and the requester. Some of these obligations may be expressed as policy that is triggered when the information is accessed. Other obligations can be triggered by a timer. For example, obligations may include requirements such as "destroy this information after 60 days" or "remove all privacy restrictions after a year." Obligations can be used as a way of exporting policy rules from one organization to another.

5. Example Policy Matrix Analyses

The Policy Matrix approach has been used in the United States Department of Homeland Security to implement many of the agency's security and privacy laws, including:

- 5 USC Sections 552 (FOIA) and 552a (Privacy Act of 1974)
- 6 USC Sections 121-122, 483 (Homeland Security Act) and 485 (Intelligence Reform and Terrorism Prevention Act of 2004)
- 46 USC Section 3796h (US PATRIOT Act)
- 50 USC (United States Code) Section 401a (National Security Act)
- 28 CFR Part 23 (Criminal Intelligence Systems Operating Policies)

- NIST FIPS Publication 199: Standards for Security Categorization of Federal Information and Information Systems
- Executive Orders 13284 and 13353
- OMB Memoranda 06-16 and 07-16
- DHS Intelligence and Analysis (I&A) Enterprise Records System of Records Notices (SORN)

Policy matrix analyses of some of these laws and policies are available from PKH Enterprises.

6. XACML

XACML is an open standard that describes an architecture and XML language for policy and access control decisions. The architecture defines requirements and data flows for Policy Enforcement Points (PEPs),Policy Decision Points (PDPs), Policy Administration Points (PAPs) (where policies are written and edited), and Policy Information Points (PIPs) (which retrieve user/resource attributes from one or more attribute storage locations). The policy authoring language (PAL) is used to describe general access control requirements, and has extension points for defining new functions, data types, and combining logic. XACML 2.0 is an OASIS standard, and XACML 3.0, which adds support for delegation, is currently in development by the OASIS XACML TC. This primer focuses only on XACML 2.0 as the latest stable version of the standard.

The advantages of XACML over proprietary and application-specific PALs are that it is:

- **Standard.** XACML has been reviewed by a large community of experts and users. It will also be easy to interoperate with other applications using the same language.
- **Generic.** XACML can be used in any environment. One policy can be used by many different kinds of applications, which simplifies policy management.
- **Distributed.** XACML can refer to other policies kept in arbitrary locations. Different groups can manage multiple policies as appropriate, and XACML can combine the results from these different policies into one decision.
- **Extensible.** Profiles and extensions have been developed that hook XACML into other standards such as Security Assertion Markup Language (SAML) and LDAP.

XACML 2.0 policies can be quite expressive and consist of the following language constructs:

- Policies and PolicySets
- Targets, Rules, and Obligations
- Attributes and Functions

Policies and Policy Sets

XACML policies start with a Policy or a Policy Set. Each Policy represents a single access control policy, expressed through a set of Rules and Obligations. A Policy Set can contain or refer to Policies or other Policy Sets. XACML reconciles multiple Policies and Rules into decisions through standard or custom Combining Algorithms. For instance, the "Deny Overrides" Combining Algorithm states that if any Rule evaluates to a "Deny," or no rule evaluates to a "Permit," then the final decision is to "Deny."

Targets, Rules, and Obligations

A request Target is a set of attributes describing the Subject (the requester), the Resource, and the Action that the Subject wants permission to perform on the Resource. The request parameters are matched up with an applicable Policy Set, Policy, or Rule that is used to execute a decision regarding permitting or denying the request. Target information also provides a way to index Policies, which enables a PDP to identify quickly the policies that apply to a request.

Once a set of policies has been found that matches the target request, the rules in the policies are evaluated. Each rule may include a Condition, which is built from Functions and Attributes. If the rule has no condition, or the condition evaluates to true, then the rule's Effect (Permit or Deny) is returned. Evaluation of a condition can also result in an error (Indeterminate) or discovery that the condition does not apply to the request (Not Applicable).

Obligations are directives to the PEP on what must be carried out before or after a request is granted. If the PEP is unable to comply with the Obligations, the requested access must not be granted. In addition to fulfilling obligations for the Service Provider, the PEP may be required to notify the requester of obligations with which the requester must comply.

Attributes and Functions

Attributes are characteristics of the Subject, Resource, Action, or Environment in which the request is made (e.g., a user's name, security clearance, a file the user wants to access, or the time of day). A request sent from a PEP to a PDP is formed almost exclusively of attributes, which will be compared to values defined in a policy using functions. Functions can work on any combination of attribute values and can even operate on the output of other functions. The hierarchy of functions and attributes can be arbitrarily complex. Custom functions can also be written to provide an ever-richer language for expressing access conditions.

7. Translating Policy Matrix Rules into XACML Statements

A Policy Matrix translates very well into XACML electronic policy statements using any XML editor (e.g., Altova XMLSpy). The Policy Matrix can be used to highlight the minimum necessary information or general attributes that must be present using a PAL such as XACML.

It also defines what rules to follow and potential hierarchy of the rules for a PDP or PEP. The table below illustrates how a Policy Matrix rule can be translated into a XACML statement.

Policy Matrix Rule	XACML Statement
Party Subject to Rule (User)	
Party Subject to Rule	**Subject(s)**
Attribute of Party Subject to Rule [Subject Information Context]	**Subject(s) attributes**
Person Context [Subject Condition(s)]	**Conditions**
Rule Action	
Actions to be accomplished: Share, Delete, Manage Policy, System Develop	**Action(s), Action(s) attributes**
Data Resource Subject to Rule	
Data Category	**Resource(s)**
Special Data Category [Other Resource Context]	**Resource(s) attributes**
Data Context [Other Resource Conditions]	**Conditions**
Circumstances in Which the Rule Applies	
Authorized Purpose [General or Action Policy Conditions]	**Purpose(s)**
Rule/Action Conditions, Trigger and Obligations [Obligations and Environments]	if **[zero or more Subject(s) Action(s), Resource(s), Environment(s) attributes, or Condition(s)] are met]** with **[zero or more Obligation(s) to be performed]**
Rule Activity	
Enumerated values of Permit, Deny, Prohibited [Deny/Permit by Statute/Policy]	**Effect** = PERMIT or DENY
Administrative Information	
Precedence [Federal/State/Tribal/County/City Constitution, Statute, Case Law, Executive Order, Regulation, Guidance or Policy]	**[PolicyCombiningAlgorithm(s)], [RuleCombiningAlgorithm(s)]**
References [Document, Citation, Record Number and Date]	**[PolicyID], [RuleID]**
Linkages [Linked Rule, Record and Reason]	**[PolicyID], [RuleID]**
Policy Matrix Editors [Record Author and Reviewer]	This information does not translate to XACML

Policy reviewers and developers should note that XACML has some limitations in fully expressing a Policy Matrix. For instance, a XACML condition applies to ***all*** targets in a rule and each target can be single or many combinations of subjects, resources, actions, or environments. To prevent duplication, the Policy Matrix separates out triggering policy conditions in a category separate from the targets. In addition, XACML does not specify a way to communicate obligations back to the requester.

An example of a policy matrix translated to XACML is provided in the Policy Enforcement Use Case section of this primer.

8. Testing XACML Policies

XACML solutions and XML editors usually include tools for verifying that XACML policies comply with the XACML XML schemas. NIST also provides an Access Control Policy Tool, which includes a GUI XACML editor and tools for checking policies for compliance with XACML 2.0.

F. Enforcing Policies

After the written policies are converted to electronic policy statements, processes that enforce these policies are performed by the PDP and PEP. These are implemented as executable software/hardware modules that sit between the user and the information. Specifically, the PDP identifies policies that match the request, evaluates the attributes in the request and the environment against the matching policies, and directs the PEP to perform the following functions:

- Allow or disallow actions requested to be performed on the information, including:
  - **Disclose** the requested information in its entirety.
  - **Redact** and disclose some of the requested information according to one or more redaction types. Types of information that are often redacted include classified information, confidential sources, open cases/ongoing investigations, PII, and sealed court cases.
  - **Deny** and do not disclose any of the requested information.
- Perform other outcomes specified in the obligations, including:
  - **Log** the request messages and actions.
  - **Notify** someone of the request and action. There are a number of situations in information sharing in which requests for certain types of information must be reported to a third party. Persons to be notified may include the subject of the information, the submitter of the information, the supervisor of the subject, or an individual that has subscribed to watch the information (silent hits).

A variety of PDP and PEP solutions are offered in the vendor community, including platform vendor suites and single-focus vendor products. These products integrate with existing database or Web application software and typically provide other policy development, deployment, and management services. More detail on PDP and PEP solutions is provided in the Implementing Security and Privacy Policies section of this primer.

G. Auditing Access

Finally, implementations of the Global Privacy Policy Technical Framework must also support processes and systems to audit access. Audit logs should support the monitoring of policy compliance and identify which organization or persons have accessed particular information resources. Audit logs can also be used as a resource for identifying who needs to be notified when the status of a previously disclosed record changes (e.g., the record is sealed or expunged.)

V. Implementing Privacy and Security Policies

This section describes technologies and solutions that are useful in implementing the security and privacy management processes associated with the Global Privacy Policy Technical Framework. These technologies and solutions include:

- Defining Identity Federations
- Implementing IDPs, SPs, and PEPs
- Implementing PDPs

A. Defining Identity Federations

The first step in any implementation of the Global Privacy Policy Technical Framework is to define the information-sharing environment, including the authentication technologies and federations that will support the exchange and interoperability of identities and service requests between IDPs and SPs.

1. Selecting Authentication Architectures and Technologies

The advantages of federated authentication over direct authentication, including fewer passwords and support for single sign-on (SSO), were discussed in section III.D. However, there are multiple architectures and technologies available for implementing federated authentication. Common federated authentication architectures include:

- Shared Authentication
- Classic SSO
- Token-Based SSO
- Federated SSO

Each of these architectures is described below.

Shared Authentication

In shared authentication architecture, users provide their credentials (e.g., userid, password) to each service, and the service calls an IDP to validate the credentials. Many environments support shared authentication using LDAP, an open- and cross-platform standard for directories of organizations, groups, users, and user attributes based on the Domain Name System (DNS) and TCP/IP. LDAP is well supported by many IDPs and SPs but, by itself, does not provide a solution for SSO.

Classic SSO

In a classic SSO architecture, the user authenticates once to the IDP, and the user's credentials (e.g., userid and password) are cached by the client. To access a service, the client provides the user's credentials to the service and the service calls the IDP to validate the credentials. An example of classic SSO is the use of the NT Lan Manager (NTLM) protocols for "Integrated Windows Authentication" by many Web sites. However, Microsoft no longer recommends using NTLM in applications, due to its use of obsolete cryptographic methods.

Token-Based SSO

In token-based SSO, the user authenticates once to the IDP, which provides a token to the client. To access a service, the client provides the token to the service for authentication, and the service validates the token. An example of token-based SSO is the Kerberos protocol, which is implemented in many commercial products, including Microsoft Active Directory (AD), as well as many versions of UNIX and open-source implementations. However, while Kerberos enables SSO within a domain, Kerberos tokens are not transferrable between domains.

Federated SSO

Federated SSO uses claims-based authentication, the process of authenticating a user based on a set of claims about the user's identity contained in a trusted token. Such a token is often issued and signed by an agency IDP that was able to authenticate the user and that is trusted by the entity (SP, sometimes called relying party) doing the claims-based authentication. In federated SSO, the user authenticates once to their IDP and, to access a service, the client obtains a service-specific token from the IDP and provides the token to the SP, which validates the token.

SAML 2.0 is a XML-based standard for exchanging authentication and authorization data between security domains, that is, between an IDP (a producer of assertions) and an SP (a consumer of assertions). SAML is an open standard developed and published by the OASIS Security Services Technical Committee (SSTC). The SSTC specifies a profile to support interoperable Web-based SSO.

WS-Federation is an identity federation specification developed by a number of companies, including Microsoft. WS-Federation is part of the Web Services Security framework and defines mechanisms for allowing disparate security realms to share and broker information on identities, identity attributes, and authentication.

As discussed in the Identifying Requesters subsection, the GFIPM framework provides the justice community and partner organizations with a standards-based approach for implementing federated identity based on SAML 2.0. GFIPM defines a standard set of XML elements and attributes about a federation user's identities, privileges, and authentication. The GFIPM

Metadata specification is being used in several federations, including the National Information Exchange Federation (NIEF).

Authentication architecture alternatives, including support for SSO, are summarized in the following table:

Authentication Architecture	Standards	SSO Within Agencies	SSO Between Agencies
Shared Authentication	LDAP		
Classic SSO	NTLM	✓	
Token SSO	Kerberos 5	✓	
Federated SSO	SAML 2.0 WS-Federation 1.1 GFIPM 2.0	✓	✓

In order to support federated authentication, SSO between agencies, and the extensibility of user attributes required by the privacy policy framework, IDPs should, at a minimum, support federated SSO and the SAML 2.0 Web Browser SSO Profile. To provide the best interoperability with other governmental agencies also supporting federated identity, the IDPs would also need to support the GFIPM 2.0 metadata standard.

2. Selecting a Federation

If the scope of the information-sharing environment is to be limited to a few organizations, it may be simple to form an identity federation through memoranda of understanding and trust relationships between the IDPs of each organization. However, the governance and operation of information-sharing environments that involve more than a few agencies can become much more complicated and diffuse. Organizations in these environments should consider leveraging the following existing identity federations based on SAML, GFIPM, or both.

SAML Federations

Kantara Initiative, a Program of the IEEE-Industry Standards and Technology Organization, is a global identity, Web and developer community made up of enterprises, mobile operators, Web 2.0 SPs, eGovernment agencies, IT vendors, and consumer electronics vendors, along with developers and members of the open source, legal, and privacy communities. These individuals and organizations collaboratively address the harmonization and interoperability challenges that exist between enterprise identity systems, Web 2.0 applications and services, and Web-based initiatives. Kantara Initiative was co-formed by the DataPortability Project, the Concordia Project, Liberty Alliance, the Internet Society (ISOC), the Information Card Foundation (ICF), OpenLiberty.org and XDI.org.

The federal government has created the Federal Identity, Credential and Access Management (FICAM) program to set standards for SSO across federal agencies, including a SAML 2.0 profile. Many states and universities have also implemented IAM solutions for their enterprises, some of which support SAML.

InCommon federation serves the U.S. education and research communities, supporting a common framework for trustworthy shared management of access to online resources. InCommon enables access to a wide variety of protected resources using standards-based, SAML-compliant open source Shibboleth software to support Federated SSO.

GFIPM Federations

The NIEF is a collection of agencies in the United States that have come together to share sensitive law enforcement information. It was created in 2008 as an outgrowth of the GFIPM program and maintains a symbiotic relationship with GFIPM, leveraging existing GFIPM work products and serving as a source of real-world feedback to drive the development of new GFIPM work products. The NIEF includes participants from the Criminal Information Sharing Alliance (CISA), Pennsylvania Justice Network (JNET), Regional Information Sharing Systems (RISS), U.S. DHS, Los Angeles County Sheriff's Department, and FBI CJIS Division.

The FBI CJIS Federation is a federated identity and access management service of the FBI CJIS Division that supports interoperability with other federations including the NIEF. The CJIS Trusted Broker service supports both SAML 2.0 and GFIPM 2.0.

The CONNECT Consortium is a group of states, including Alabama, Kansas, Nebraska, and Wyoming, that are working together to solve specific information-sharing challenges by leveraging the Global Justice Information Sharing Initiative standards, including GFIPM. CONNECT publishes their lessons learned and the CONNECT artifacts (the "CONNECT Toolkit") freely.

A comparison of the GFIPM federations, including their compatibility with GFIPM work products, is provided at http://gfipm.net/federations.html. The federations with the best support for the GFIPM 2.0 metadata, including metadata to support the Global Privacy Policy Technical Framework, are the NIEF and the FBI CJIS Federation.

3. Joining a Federation

Each federation defines its own technical and policy requirements for membership. The process for joining the federation is typically described on the federation Web site. For instance, organizations interested in joining the NIEF should consult https://nief.gfipm.net/prospective.html.

B. Implementing IDPs, SPs, and PEPs

After the scope of the information-sharing environment is defined and the architecture and technologies for user authentication are selected, the next step is the implementation of federated user identities and attributes in the IDPs and SPs. IDPs and identity validation APIs for SPs are typically provided by Identity and Access Management (IAM) technology solutions that are designed to simplify enterprise user management. IAM tools act as a middleware layer between the enterprise's users and its internal and external applications. Each IAM solution provides up to three distinct functionalities:

- **SSO**. SSO provides a user with the ability to use a single login to achieve access to all applications. Since users only have to remember a single login, it can be made stronger and changed more frequently while being forgotten less often. IAM provides this facility by requiring that the primary login is to the IAM tool itself. Thereafter as a login process initiates, the IAM tool handles it in the background.
- **User Provisioning**. Automation of account provisioning gets new users up and working more quickly, eliminating non-productive time and saving money for the enterprise. Automated de-provisioning quickly and efficiently eliminates unused/dormant accounts and the security threat they represent. IAM provides these functions as it interacts with all applications and manages all user accounts.
- **User Activity Monitoring**. Reporting on user activities, particularly those involving protected data sources is an essential component of regulatory compliance. Since IAM sits between the users and the applications that manage data stores, it is aware of the nature of all transactions and communications and is able to provide consolidated reports.

1. Selecting IAM Solutions

Most implementers should avoid developing their own IDPs and identity validation services. In most scenarios, open source or commercial off-the-shelf solutions supporting SAML 2.0 can be configured to support the privacy framework.

Open Source IAM Solutions

There are a large number of open source IAM solutions. The advantages of these solutions tend to be their compliance with the SAML 2.0 specifications and their interoperability with other SAML 2.0 implementations. Some of the most common and well-supported open source IAMs that include source code for SAML IDPs and SPs include:

- Enterprise Sign On Engine (ESOE) – A Java implementation of SAML 2.0 and XACML 2.0.

- LaSSO – A C implementation of SAML 2.0 for use by the Liberty Alliance (now part of the Kantara Initiative).
- OpenAM – Java and PHP implementations of SAML 2.0 (formerly Sun OpenSSO).
- OpenIAM -
- Shibboleth – Java and C++ implementations of SAML 2.0 and a profile with extensions for use on the Internet2. Variations of Shibboleth include OpenSAML and Guanxi.
- SimpleSAMLphp – A PHP implementation of a SAML 2.0, WS-Federation and the Shibboleth profile.
- SourceID – Java, .NET, Drupal, and PHP implementations of SAML 1.1 and WS-Federation supported by Ping Identity.
- ZXID – A C implementation of SAML 2.0 and WS-Federation.

COTS IAM Solutions

There are also a large number of COTS IAM solutions. The following table summarizes some of the core functionality of common COTS solutions that support SAML and SSO:

COTS IAM Solution	SSO	User Provisioning	User Activity Monitoring
CA Identity and Access Management	✓	✓	✓
Centrify Suite	✓		
Evidian IAM Suite	✓		
IBM Tivoli Identity Manager and Access Manager	✓		
Microsoft Identity and Access Platform	✓		
Novell Identity Manager and Access Manager	✓	✓	
Oracle Access Manager and Identity Manager	✓	✓	
Ping Identity PingFederate	✓		
Symplified	✓		

If possible, an organization should leverage existing support for federated identity in their infrastructure. For instance, in a Microsoft server environment, federated identity may be supported by the Microsoft Identity and Access Platform, which includes:

- **Microsoft Active Directory Federation Services (AD FS) 2.0** – a Web SSO service and IDP that enables federated identity and claims-based authentication across organizations. AD FS supports SAML 2.0 and WS-Federation and is part of Windows Server 2008 R2 and later.
- **Windows Identity Foundation (WIF**) – a set of .NET Framework classes that applications (SPs) can use to implement claims-based identity.

2. Implementing IDPs

IDPs can be implemented and integrated with SPs using SAML 2.0 virtually "out of the box" using any of the above open source or COTS IAM solutions. However, the selected IDPs will need to be reconfigured to support the additional user attributes and metadata required by the privacy framework and GFIPM 2.0.

3. Implementing SPs and PEPs

SPs refer to the applications that are accessible in the information-sharing environment, while PEPs refer to the locations in the architecture in which privacy policies are enforced based on access decisions by a PDP. Many implementers choose to integrate the PEP functionality into the SPs directly – that is, applications will receive the request, forward the request to the PDP, and then grant or deny access based on the response from the PDP. In other cases, the PEP functionality may be implemented as a gateway or portal service through which users gain access to the SPs.

Open source and COTS IAM solutions provide source code libraries that SPs and PEPs can use to validate federated identities and otherwise integrate with IDPs. SPs/PEPs are not required to use the same IAM solutions as the IDPs. As long as both the IDPs and SPs/PEPs fully comply with the same standards (e.g., SAML 2.0), they should be able to inter-operate..

4. Testing IAM Solutions

Regardless of vendor claims regarding interoperability, implementers should test all combinations of IDPs and SPs/PEPs fully to ensure that all members of the federation can exchange federated identities and user attributes regardless of their specific IAM solutions.

C. Implementing PDPs

Once the IDPs and SPs are implemented, the next step in the implementation of the privacy framework is to migrate access control logic from the applications to shared PDP services. This enables more fine-grained authorizations and improves the consistency of access controls across the enterprise.

1. Selecting a PDP Solution

PDPs are typically implemented by open source or COTS implementations of XACML 2.0.

Open Source XACML Solutions

There are a large number of open source XACML solutions. The advantages of these solutions are typically their compliance with the XACML 2.0 specifications and their interoperability with other XACML 2.0 implementations. Some of the most common and well-supported open source solutions that include source code for XACML PDPs include:

- Sun XACML Toolkit – a Java implementation of a XACML 2.0 PDP and PEP.
- ESOE – a Java implementation of SAML 2.0 and XACML 2.0 PDP and PEP.
- XACMLight – a Java/Apache Axis2 implementation of a XACML 2.0 PDP and PAP.

COTS XACML Solutions

There are also a large number of COTS XACML solutions, typically provided as part of an IAM or user entitlement software suite. The following table summarizes the core functionality of some common COTS solutions that support XACML 2.0 PDPs:

COTS XACML Solution	PDP	PAP	PEP
Axiomatics Policy Auditor, Server and PEP	✓	✓	✓
BitKoo Keystone	✓	✓	✓
CA Embedded Entitlements Manager	✓	✓	✓
Cisco Enterprise Policy Manager	✓	✓	✓
IBM Tivoli Security Policy Manager	✓	✓	✓
Jericho Systems EnterSpace Vault	✓	✓	✓
Oracle Entitlements Server	✓	✓	✓

The preferred XACML solution for each organization will depend on the capabilities of the existing infrastructure and applications to support federated identity using SAML 2.0 and policy-based authorization and access control using XACML 2.0. For instance, the GFIPM program has implemented a demonstration integrating GFIPM authentication and XACML-based access controls at https://rhelsp.ref.gfipm.net based on the Sun XACML Toolkit.

2. Implementing PDPs

Most of the open source and COTS XACML solutions should support coarse-grained authorizations based on XACML electronic policy statements directly. However, support for fine-grained authorizations and the additional metadata for user and resource attributes, conditions, and obligations required by GFIPM and the privacy framework will likely require customizations to the base solutions.

Many implementers will choose to integrate the PEP functionality independently or as part of the SPs directly. In other cases, the PDP and PEP functionality may be integrated. SPs/PEPs are not required to use the same XACML solutions as the PDPs, as long as both the PDPs and SPs/PEPs fully comply with the same standards (e.g., XACML 2.0).

3. Testing IAM Solutions

Regardless of vendor claims regarding interoperability, implementers should test all combinations of PEPs and PDPs fully to ensure that all members of the federation can exchange federated identities and user attributes regardless of their specific IAM solutions.

VI. Policy Enforcement Use Case

This section includes excerpts from a sample implementation of the privacy framework for the Florida Department of Law Enforcement, including a policy analysis by PKH Enterprises. This use case illustrates three example conversions from written policy in the Florida Constitution and statutes to XACML statements using the Policy Matrix approach.

1. FLORIDA CONSTITUTION ART. I, SEC. 24(a): ACCESS TO PUBLIC RECORDS AND MEETINGS

Policy Statement

(a) Every person has the right to inspect or copy any public record made or received in connection with the official business of any public body, officer, or employee of the state, or persons acting on their behalf, except with respect to records exempted pursuant to this section or specifically made confidential by this Constitution. This section specifically includes the legislative, executive, and judicial branches of government and each agency or department created thereunder; counties, municipalities, and districts; and each constitutional officer, board, and commission, or entity created pursuant to law or this Constitution.

Policy Matrix Analysis

Using the Policy Matrix approach, the following attributes can be extracted from the above policy statement:

Attributes	
Subject	Every Person
Resource	Record Type: [Public Record] Record Use: [Official Business] Record Role: [Non Exempted]
Actions	Read
Conditions	Resource Conditions: Official Business (True), Non Exempted Records (True)
Rule	Rule Target = Resource: Public Record

XACML Statements

The policy can then be translated into the following XACML rule statement:

Policy	**Policy Rule Statement**
Fla. Const. art. I, Sec. 24 (a)	A **[Subject: All]** may **[Action:Read]** a **[Resource: Public Record]** for **[Purpose(s): All]** if **[Data: Conditions: Official Business [Yes] (True), Non Exempted Records [yes] (True)],** if **[Condition: Rule Target: Public Record]** and with **[Obligations: None]**. **Effect** = PERMIT.

2. FLORIDA SUNSHINE LAW SEC. 119.07(1)(a): INSPECTION AND COPYING OF RECORDS

Policy Statement

(1)(a) Every person who has custody of a public record shall permit the record to be inspected and copied by any person desiring to do so, at any reasonable time, under reasonable conditions, and under supervision by the custodian of the public records.

Policy Matrix Analysis

Using the Policy Matrix approach, the following attributes can be extracted from the above policy statement:

Attributes	
Subject	Any Person
Resource	Record Type: [Public Record] Record Use: [Official Business] Record Role: [Non Exempted] Approved/Reviewed by Custodian: [yes/no] Request can be Fulfilled: [yes/no] Request falls within "reasonable" policy: [yes/no]
Actions	Read
Conditions	Resource Conditions: Reasonable time (True), Reasonable conditions (True), Under supervision by custodian (True)
Rule	Rule Target = Resource: Public Record

XACML Statements

The policy can then be translated into the following XACML statement:

Policy	**XACML Policy Rule Statement**
Fla. Stat. Ann. Sec. 119.07 (1)(a)	A **[Subject: All]** may **[Action: Read]** a **[Resource: Public Record]** for **[Purpose: All]** if **[Resource: Conditions: Reasonable time [yes] (True), Reasonable conditions [Yes] (True), Under supervision by custodian [Yes] (true)]** if **[Condition: Rule Target: Public Record]** and with **[Obligations: None]**. **Effect** = PERMIT. *[Note "reasonable time" may be defined by policy, and would be an environmental condition]*

3. FLORIDA SUNSHINE LAW SEC. 119.071(1)(a): AGENCY ADMINISTRATION

Policy Statement

(a) Examination questions and answer sheets of examinations administered by a governmental agency for the purpose of licensure, certification, or employment are exempt from sec. 119.07(1) and sec. 24(a), Art. I of the State Constitution. A person who has taken such an examination has the right to review his or her own completed examination.

Policy Matrix Analysis

Using the Policy Matrix approach, the following attributes can be extracted from the above policy statement:

Attributes	
Subject	Subject Name
Resource	Record Subject: [Examination]
Actions	Read
Conditions	Record Owner: [Subject Name]
Rule	Rule Target = Resource: Examination

XACML Statements

The policy can then be translated into the following XACML statement:

Policy	**XACML Policy Rule Statement**
Fla. Stat. Ann. Sec. 119.071 (1)(a)	A **[Subject: Subject Name]** may **[Action: Read]** a **[Resource: Record Subject: Examination]** for **[Purpose(s): All]** if **[Condition: Record Owner: Subject Name (yes) [True])**. **Effect** = PERMIT.