IAM Reference Architecture Model

George B. Dobbs

## Abstract

This article provides a reference model to organize the presentation of technical details associated with various implementations of identity and access management (IAM) architectural concepts. The model is conceptual, as are the set of abstract components which it provides.

To move out of the conceptual realm into specifics additional articles follow, each with a focus on a specific technical use-cases. Each such use-case indicates which of the abstract components comprise a particular implementation

## Introduction

It has been said that all models are wrong but some are useful.[Wikipedia - all models] This model attempts to find a level of generality that is broadly useful. Too general, and the model becomes untethered to reality and definitely not useful. Too specific, and the model will only work in some cases.

The model has a technical slant, but it necessarily touches on some of the process, legal, and capability dimensions as well. This is intended to give the reader a set of concepts that can be applied when thinking about identity and access management.

The model is based on the idea that the management of identities and access can (mostly) be separated from their use. This concept can apply to distributed systems as well as self-contained systems. So when you see IAM working together with, say, an application it may mean that are separate physical systems or it could mean these parts are separate pieces of software running on a single system.

The main goal of this article is allow consistent discussion of more specific use-cases.

The model is a started with the ISO/IEC framing [ISO/IEC]. The UML detail was removed for simplicity and the IAM model has been extended so that authorization, governance and risk-control can be included.

Some of the ISO/IEC names have been changed to reflect more common usage. In some cases, the ISO names have been used in a way that is more expansive than their original definition.

In an attempt to adopt the most useful terminology, the model has been reviewed in conjunction with the FICAM, Internet 2, NIST SP-800-63 definitions, and NIST Zero Trust frameworks, and with the Identity Stack presented at Identiverse 2019.

The model can be used at different levels. Here are a couple of examples:

### Example 1: Distributed Systems

A modern architecture may have a web-hosted application (the RP) which relies on a cloud identity service (the Identity Provider). The RP in this case could be a customer facing application or a workforce facing application.

### Example 2: Single System

A computer’s file system (the RP) provides access control based on the user information acquired at login (the IDP). In this case both the file system and the identity management function are encapsulated in an operating system.

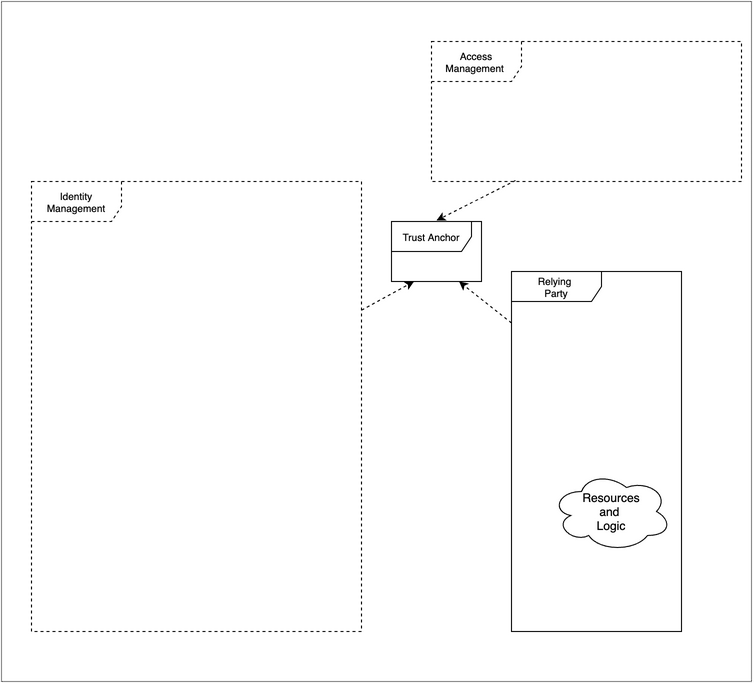
## Terminology

The terms are defined below. Those with a ✓ mark are the abstract components that comprise the model.

|  |  |
| --- | --- |
| **Item** | **Definition** |
| ✓**Access Governance (IGA)** | Access Governance provides oversight and control over access rights implemented in multiple local or shared authorization systems. These rights may be controlled in a variety of ways, starting with the existence and validity of the digital identity. Other controls include various mechanisms such as policies, the mapping of roles, permissions, and identies. The abbreviation used is for Identity Governance and Administration and is commonly used in the commercial sector. This roughly corresponds to the Access Certification section of the first-class component Governance Systems in the FICAM model. IGA is not specifically addressed in the ISO/IEC model. |
| ✓**Access Management** | This is a convenience grouping. It consists of the part of Authorization that is external to the relying party, and the Access Governance function. |
| **Assertion** | A formal message or token that conveys information about a principal, typically including a level of assurance about an authentication event and sometimes additional attribute information. Somes this is called a Security Token. |
| ✓**Attribute Provider** | Sometimes the authority for attributes is distinguished from the authority for identities. In this case the term Attribute Provider is sometimes used. It is a subset or type of an Identity Information Authority. |
| ✓**Audit Repository** | A component that stores records about all sorts of events that may be useful later to determine if a) operations are according to policy, support forensic investigations, and allow for pattern analysis. Typically it his highly controlled to prevent tampering. Audit Repository is the ISO name for this concept and is localized to the Identity Management System. In this model it is generalized to support event records from any part of the ecosystem. |
| ✓**AuthN / Assertion** | The act of determining that to a level of assurance the principal/subject is authentic. Depending on the architecture this function may also produce a security token to “assert” authentication information securely to the RP. |
| ✓**Shared or Local Authorization (AuthZ)** | Authorization is how a decision is made at run-time to allow access to a resource. We break this down into two types: shared and local. Local authorization is handled by the RP. Shared authorization is part of the access management grouping. The FICAM framework includes this as a subcomponent of the Access Management System. AuthZ is not included in the ISO or Internet 2 models. |
| **Credential** | A credential allows for authentication of an entity by binding an identity to an authenticator. |
| ✓**Credential Service Provider (CSP)** | Following NIST 800-63-3 we include both the enrollment function and credential services together under the name Credential Services Provider. |
| **Credential Services** | Credential Services issue or register the subscriber authenticators, deliver the credential for use, and subsequently manages the credentials. We include PKI information so the subscriber will include system components that need certificates and private keys. FICAM separates this into a first-class component called Credential Management System, |
| **Enforcement** | The mechanism that ensures an individual cannot perform an action or access a system when prohibited by policy. |
| **Enrollment** | Also known as Registration. Enrollment is concerned with the proofing, and lifecycle aspects of the principal (or subject). The entity that performs enrollment has sometimes been known as a Registration Authority, but we (following NIST) will use the term Credential Service Provider. |
| **Entitlement** | The artifact that allows access to a resource by a principal. This is also known as to privilege, access right, permission, or an authorization. An entitlement can be implmented in a variety of ways. |
| ✓**Identity Information Authority (IIA)** | This represents one or more data sources that are used by the IDM as the basis for the master set of principal/subject identity records. Each IIA may supply a subset of records and a subset of attributes. Sometimes the IIA is distinguished from the Identity Information Provider or IIP. We use IIA to include the service that actually provides the information as well as the root authority. This corresponds to Identity Information Source in ISO/IEC 24760-2, and Identity Sources in Internet 2. |
| ✓**Identity Management (IDM)** | A set of policies, procedures, technology, and other resources for maintaining identity information. In this model it contains information about principals/subjects including credentials. It also including other data such as meta data to enable interoperability with other components. The IDM is shown with a dotted line to indicate that it is a conceptual grouping of components, not a full fledged system in itself. |
| **Identity Provider (IDP)** | Identity Provider or IDP is a common term and its best understood the subset of Identity Management, excluding the management functions. It consists of the service interfaces: AuthN/Assertion, Service Provisioning Agent, Session Mgmt, Discovery services, and Metadata Mgmt. |
| ✓**Identity Register** | This is the data store that contains the enrolled entities, and their attributes, including credentials. See the IDM section for elaboration. The terms Directory and Attribute Store are sometimes used as a synonym. |
| ✓**Metadata Management** | Control data that allows the Identity Management System to recognize and trust the Relying Service. This corresponds to Relying Party data in the Internet 2 model. |
| ✓**Relying Party (RP)** | A component, system or application that uses the IDP to identify its users. The RP has its own resources and logic. Note that the term Relying Service is used by ISO/IEC to encompass all types of components that use identity services, including systems, sub-systems, and applications, independent of the domain or operator. We will use the more common Relying Party (or RP). This roughly corresponds to the Agency Endpoint in the FICAM model, or to Identity Consumers in the Internet2 model. |
| ✓**Risk Context (RCTX)** | Risk Context consists of additional facts that can be brought to bear to improve the overall security of the ecosystem. Internal or external events and facts can be applied to enable, limit, or terminate access. This is similar to the section Monitors and Sensors under FICAM’s Governance Systems, and to many of the inputs of the Policy Decision Point in [NIST Zero Trust]. |
| **Session** | A period of time after an authentication event when an RP grants access to resources for the principal/subject. |
| ✓**Session Management** | A coordinating function provided by an IDP to control sessions of subscribing relying parties. |
| ✓**Trust Anchor** | This component represents the legal, organizational and technical apparatus that enables trust between the Identity Management System and the Relying Parties. |

## Basic Structure of the Model

The most basic function of the identity system is to provide secure storage of the information about identities and a way for relying parties, to use that data to control access to resources. The following is the basic organization of an identity management system (IDM), supporting multiple relying services, or relying parties (RP).



Basic Component Dependencies the identity management system supports multiple relying parties. The core components of the IDM are shown. The dotted arrowed lines show dependencies.

### Identity Management

Identity Management (IDM) is a set of policies, procedures, technology, and other resources for maintaining identity information. In this model it contains information about principals/subjects including credentials. It also including other data such as meta data to enable interoperability with other components. The IDM is shown with a dotted line to indicate that it is a conceptual grouping of components, not a full fledged system in itself.

### Relying Party

The Relying Parting (RP) is a component, system or application that uses the IDM to identify its users. The RP has its own resources and logic. It comes in many forms, all of which use identity services, including systems, sub-systems, and applications, independent of the domain or operator.

### Trust Anchor

This component represents the legal, organizational and technical apparatus that enables trust between the Identity Management System and the Relying Parties. When the IDM and the RP are not in the same organization this may take on a salient aspect; when they are in the same organization the agreements may be more tacit. When the IDM and RP are both built into a single system the root of trust my be hidden in the system internals.

There are three parts to the Trust Anchor:

#### Root of Trust

There is a need for a technical root of trust. This is done through a Public Key Infrastructure (PKI). The parties agree to trust a common certificate authority which signs the certificates of all parties in the federation.

#### Trust Framework

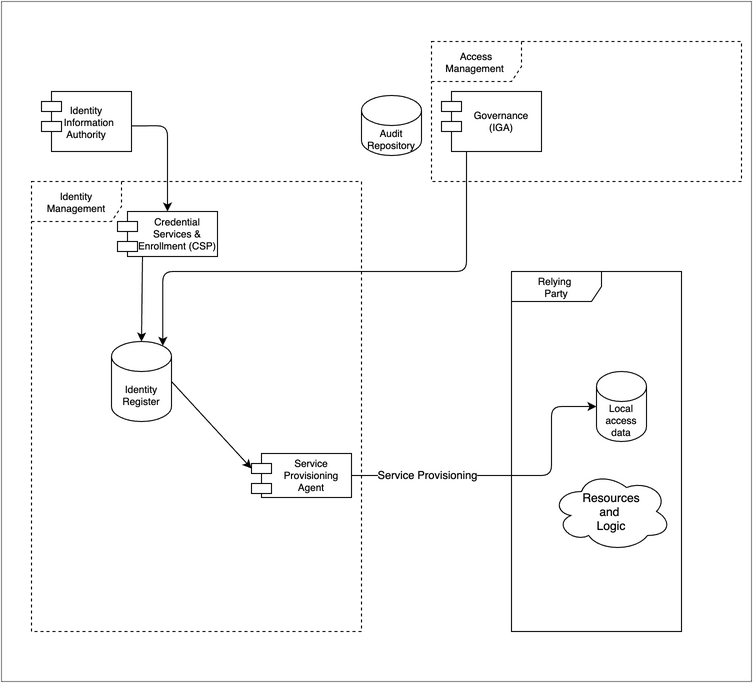
A trust framework is the set of rules and policies that govern how the federation members will operate and interact [NISTIR 8149]. In simple cases this may be a contract between two parties. In other cases it is the basis of a multilateral agreement.

#### Interoperations

To operate well, the parties of a federation establish mutual agreement upon an acceptable identity to be used between the parties in a federated relationship (for instance the level of assurance used). In addition, the definition and values of attributes of federated identities should be agreed. The parties should agree on the security/access policies of federated users between the parties in a federated relationship. For instance, are there duties to notifiy others in the event of security failures.

## Provisioning

Provisioning is a term that encompases the processes and methods that create, modify, and, eventually, delete the identity and profile information used by IT infrastructure and business applications. By these methods, records are created, or updated in the identity repository, and removed from it. Often, provisioning needs to extend to applications to support authorization decisions. The term “Onboarding” is sometimes used to refer to the sum of the initial provisioning activities, in both the identity and access aspects.



Provisioning: The Identity register receives updates from one or more external sources and administrative actions, passing the information on as needed.

### Identity Information Authorities

While, it is possible to have an identity management system that is populated without attaching to an external data service, this is typically not the case. Usually, employee or customer data needs to be imported.

Note that the authoritative sources for identity attributes transcend the HR system and may include the email system, phone system, training certification etc. In some cases, a company may have more than one HR system.

### Governance

The act of provisioning may include certain logic, best modeled as governance. In some cases the IGA system actually takes on all the provisioning duties.

### Credential Services & Enrollment

This function includes steps needed to originate and activate an identity. It is also concerned with on-going maintenance such as password reset and key rotation. This function includes administrative activities and self-serve activities.

#### Enrollment

Also sometimes known as Registration. It involves such activities as proofing, verfication or vetting, and recording sponsorship, if needed. It also is responsible for the secure delivery of credentials. Enrollment ends when a user formally receives ownership of their digital identity and assumes control and ownership of their account’s credentials.

#### Credential Services

Credential service include the creation and binding of passwords, cryptographic keys and other authenticators. It is also concerned with on-going maintenance such as password reset and key rotation. It also is in charge of revoking credentials as needed.

### Identity Register

This is the data store that contains the enrolled entities, and their attributes, including credentials. In this model we use the singular, as if it were one singular database. In practice, designs may store some attributes separately from identities. We also use this term to include the storage related to credentials, although, all or some of the credentials may be stored in their own physical repository.

Identity Registers by their nature have high availability requirements, so often at the physical level they contain multiple instances which are synchronized. The Identity Register could be implemented in several ways. Common methods include the use of general-purpose databases, optimized stores such directories i.e., a physical or a virtual directory.

Importing data does not necessarily mean making a physical copy of data, although it often does. The notion also supports the idea of virtualization - where the import of identity information is done at run-time.

### Service Provisioning Agent

Also noted is the function of propagating selected information further into the ecosystem. This typically occurs when a relying services needs additional information about the users, e.g. for the purpose of access control, or personalization. The relying system makes a copy of the identity data and that is used in the application processes. A complete solution will allow for the full lifecycle including creation, update and eventual deletion of the identity data stored locally.

#### Just in Time Provisioning

So far, the discussion of the provisioning function has been focused on “admin-time”. However, there are some cases where provisioning occurs at run time.

Not shown here, but sometimes implemented, are provisioning actions that occur on a just-in-time basis. This can happen when additional identity information is passed to a relying service in real-time to support a specific application requirement, possibly including identity attributes (See Authentication and Sessions). A similar case involves the relying service querying the identity management system in order to acquire attributes (Shown under Authorization)

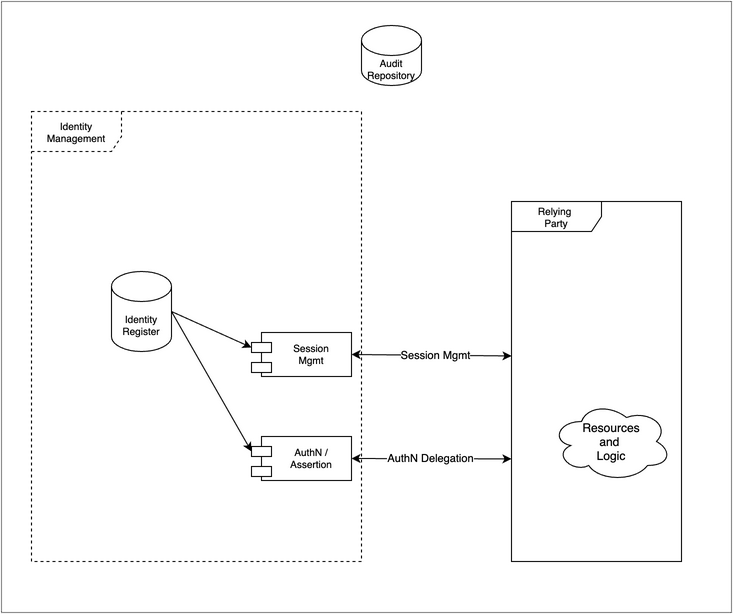
### Audit Repository

The audit repository is shown to indicate the accumulation of historical event data. To avoid clutter we assume audit information is written but don’t arrows show it.

## Authentication and Sessions

### Authentication

Authentication is the process by which a subject’s credentials are used to verify their identity. The IDP checks and verifies credentials that are presented to it. There are multiple scenarios. Typically, the Relying Party presents the credentials on behalf of the user and receives an assessment from the IDP regarding the level of certainty that the user is authentic. Often the assessment (and more information about the user) is delivered to the RP via a security token, which is protected by cryptography. There are several varities of security tokens.



Authentication and Sessions: The Identity Register supports authentication scenarios. The IDP may monitor or participate if the full session lifecycle with the Relying services.

### Sessions

A common pattern is to associate the authentication event with the start of a session. The session is mostly the concern of the relying party. However, it is sometimes desirable to keep the sessions of several relying parties in synch. For instance, logging out of one session will terminate concurrent sessions. To do this, often the IDP will act to orchestrate sessions termination. In high security environments, session management must support termination based on real-time identity data such as when a user’s entitlements have been modified.

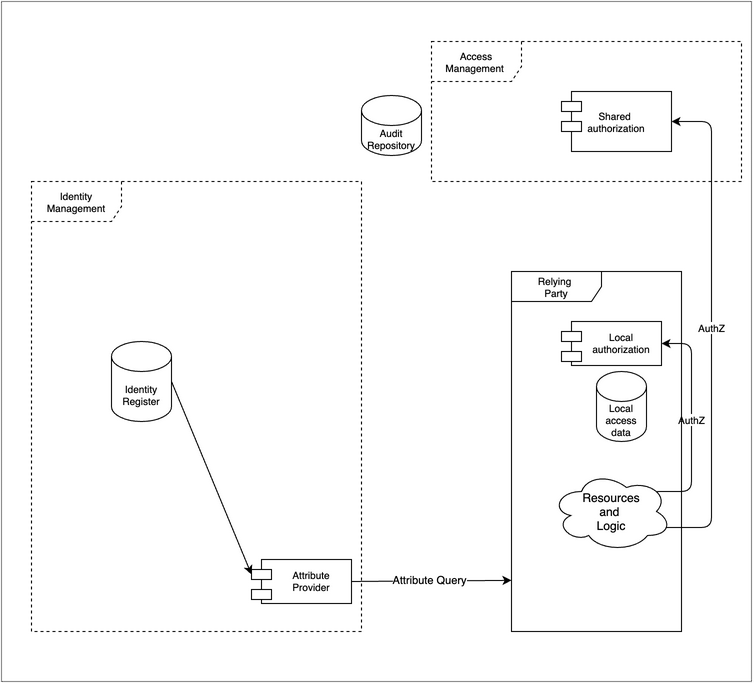
The existence of a centralized point of view about sessions, can be leveraged to support good security practices. For example, if the identity attributes of a user with an active session changes and then contravenes an access control policy the session should terminate, or if session management becomes aware of a terminated account it should end any active session that the user has. This could also occur in advanced scenarios which include external risk facts. See Risk Context below.

Sessions also support another important concept: step-up authentication. A session can keep track of the level of assurance of a particular authentication, so when a user requests access to a transaction or application requiring a higher level of identity assurance, the IDP can be prepared to determine the course of action, such as improving the certainty that the user is the right person by asking the user provide additional evidence. For example, maybe the password is good enough to review some information but to withdraw money the additional factor of of a one-time password from a phone app is required. The detection of the assurance gap and subsequent action will logically be done at the RP, but to avoid a poor user experience in multiple RP scenarios the step-up needs to be recorded in the session.

## Authorization

Authorization models are many and diverse. The diagram illustrates two approaches for authorization: local and shared.

Both approaches typically use subject attributes help determine access. These values may have been provisioned into a local store, in the Provisioning process described above. Or the values can be acquired at run-time from the Identity Management System as shown by the attribute query.



Authorization models: Some RPs perform authorization tasks internally. Sometimes authorization is a shared resource for many RPs.

#### Local Authoriziation

Many relying services perform authorization tasks internally. Often the fine-grained access control required by a protected resource makes this appealing. For instance, a financial management system may maintain a user’s entitlements to specific functionality within the application. In this scenario the application makes the authorization decision and implements (enforces) the result. This type of application may need some form of provisioning to create the appropriate user records (shown stored in the local access data). Other mechanisms exist too, such as providing the user’s role or other attributes during the signon, perhaps as a value in the security token, or through an attribute query.

#### Shared Authorization

Sometimes authorization is a shared resource for many relying services. This design can improve consistency of authorization decisions and supports organizations wishing to include advanced access decisions strategies such as those required by a "Zero Trust" access control approach. [ NIST 800-207 ]. Shared authorization systems typically have a consistent approach to policy such as a standardized policy language. In this scenario the RP asks the shared authorization function to make the decision but implements (enforces) that itself.

### Authorization Mechanisms

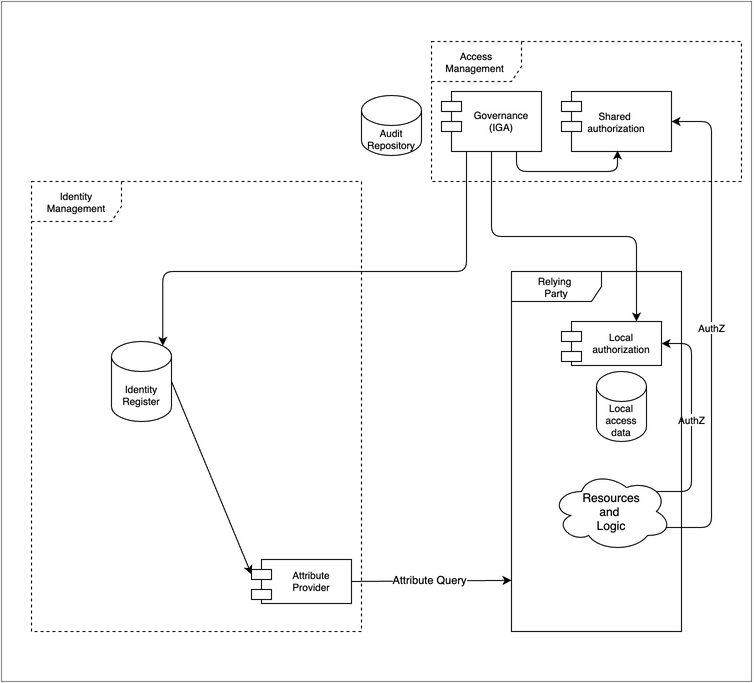
In either approach, the access rights may be established, maintained and revoked in a variety of ways, starting with the existence and validity of the digital identity. Other controls include various mechanisms such as policies, roles, permissions, and identities. Some controls rely on user attributes including group memberships or roles stored in an Identity Register.

Each mechanism relies and a particular logical data structure to implement the access control and that data structure becomes and the focus of implementers. For instance, in role based access control, there is some art involved in “Role Management”, (defining and managing a useful set of roles), since too many roles becomes difficult to manage and too few leads to users with access to things they don’t need. Similarly, in the case of policy based access control the set of policies (the Policy Rules) needs to be designed, stored and managed.

## Access governance (IGA)

Access Governance provides control over access rights implemented in multiple local or shared authorization systems. This function is often broken into the administration of these rights and the oversight needed to ensure that these rights are in good order over time.

Access Governance is required in enterprise systems focusing on management of staff (employee/contractor) entitlements. The concept can also apply to customer facing scenarios such as business to business delegated rights or business to customer scenarios where delegation such as power of attorney or other agents are required.



Access Governance provides oversight and control over access rights implemented in many Local authorization systems and, sometimes, in Shared authorization systems.

#### Control

The controls may also include methods such as procedures and workflows to ensure proper review.

Often deployed to prevent internal fraud is the “segregation of duties” control. The control defines groups of access rights that cannot be held by the same person. This is best done in a location that has visability to all the implicated access rights, i.e. the IGA system.

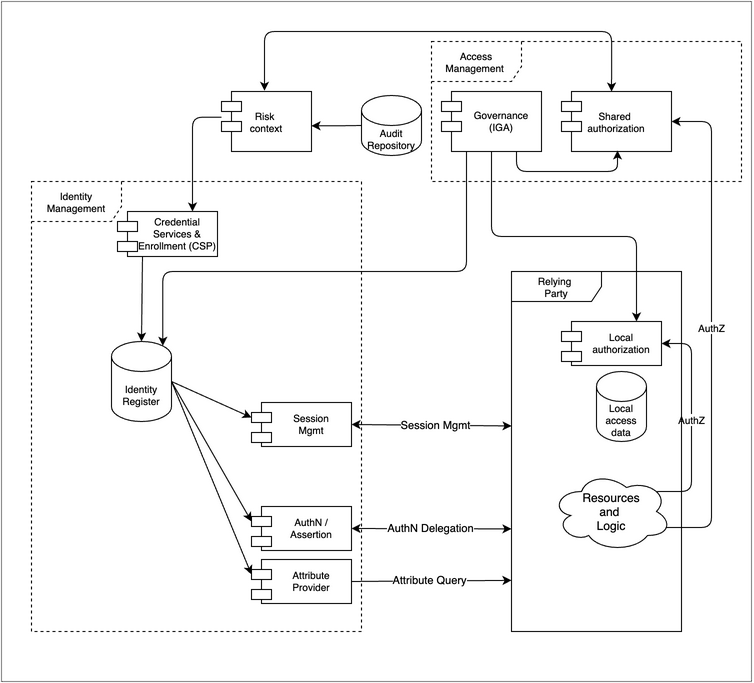
#### Oversight

Typically, governance activities review and may modify the data in one or more of the authorization components in order to effect a change in entitlements. Often organizations will have formal process to review existing entitlements and may require a responsible party to certify or attest that the entitlements are in good order. Additional tools to ensure that IAM policies are effective at enforcing their stated control include internal and external audits as well as analytic reports.

### Risk Context

Risk context information can be valuable to improve the security of the relying service. Risk can be judged based on information in the request, information about the history of the user, or assertions/evidence from third parties.

The linkage from the Audit Repository illustrates that the Risk Context may consume the local historical data about events.



Risk Context: It is possible to use risk information in authentication decisions. For instance, if a stolen password is found on the dark web, don’t allow login.

External events may be visible to the Identity Management System operator through consortia or vendor packages. In some mutual-support scenarios, it may be possible for the IDM operator to also publish events for the benefit of others, supporting other operators’ risk management requirement.

Events need to be delivered into the IDM so that they can selectively be used to modify the behavior of the authentication function. In some severe scenarios it may be desirable to attach the events to the session management function so that current sessions can be reviewed and terminated if needed. The OpenID Shared Signals and Events working group is developing standard ways to deliver these signals.

As shown in the diagram, shared authorization systems may consume risk data as well. For example, an authorization might be denied if the subject's recent activity history is outside of normal bounds, possibly indicating a compromised credential. Logically this could happen with local authorization as well, but this is not shown.

#### Example: Information in the request

##### Boundary control

An authentication or authorization decision may be influenced by specific criteria such as whether an the request is coming from a known or unknown network. A more sophisticated version of this attempts to prohibit access from, say certain countries.

#### Examples: Historical usage

##### Usage pattern match

Determine if this request is outside the normal usage patterns for a given individual. The reference to historical usage patterns allows for pattern detection and can help establish a metric for risk for a user, in general, or for a specific transaction. Such activity can be called risk profiling.

##### Land speed violation

By amending the user’s request and history with location information, it is possible identify likely compromised account due to the fact that the user can’t be in two places at one.

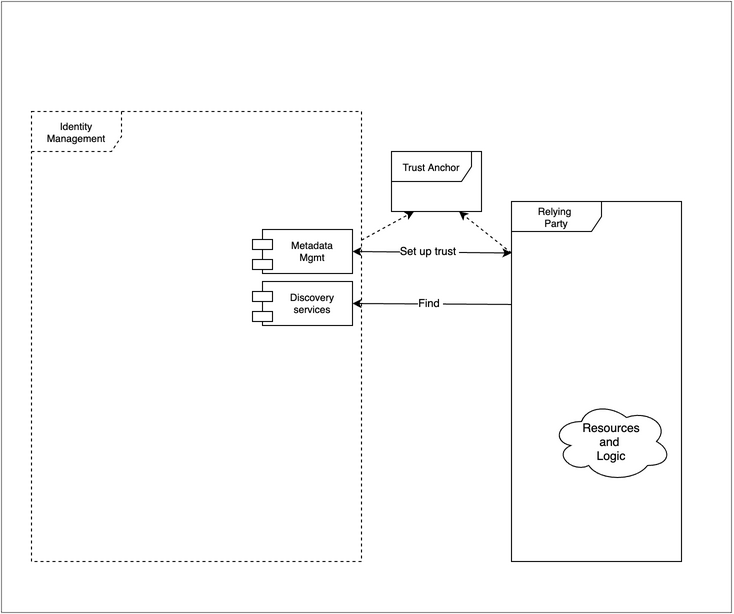
Such examples depends on signals from the local environment, but it is also possible to obtain signals from further afield.

#### Example: Third party

it is possible to determine commonly used passwords based on postings on the “dark-web”. Bad actors acquire these in the hope that users will use the same password at other sites. A counter measure is for the Identity Management System operator to require additional certainty if one of those passwords were presented.

## Metadata and Discovery

Metadata refers to control data that allows the Identity Management System and the Relying Parties to interoperate.



Metadata and discovery these two functions are involved with mutual recognition of the Identity Management System and Relying Service.

One example is the registration of public key certificates to enable mutual authentication. In some scenarios this information is shared between the parties manually. At run-time for distributed systems the technial root of trust is needed to validate the security channel (PKI)

Another example points out that configuration information is another form of metadata, OpenID Connect has a list of required, recommended, and optional values that describe a particular implementation, aimed at providing a degree of automation during setup.

The metadata may include information that limits the types of interactions and scope of the data that is exchanged. It can also contain security information to allow the counterparties to authenticate each other. For instance, public key components such as certificates with a common trust root may be used.

Discovery refers to protocols that facilitate automation. For instance OpenID Connect defines a method for relying parties to locate an end-point where a user’s identity can be verified. A Discovery service can advise where specific data can be accessed and which end-points are maintained to allow a relying party to use the identity service.

### Acknowlegements

Thanks to

Ian Glazer, Graham Williamson, and Corey Scholefeld for detailed review

Jon Lehtinen and Steve Hutchinson for some of the definitions from their unpublished Introduction to Identity Part 3 document.

## References

1. ISO/IEC 24760-2:2015(E) Figure C.1 provided the starting point. ISO/IEC 24760-1 Second edition provided improved naming and granularity (specifically breaking out CSP and Enrolment)
2. FICAM <https://playbooks.idmanagement.gov/arch/components/>
3. Internet 2 <https://playbooks.idmanagement.gov/arch/components/>
4. NIST Zero Trust <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf>
5. OpenID Connect discovery https://openid.net/specs/openid-connect-discovery-1\_0.html
6. NIST SP-800-63 definitions https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf
7. NISTIR 8149 https://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8149.pdf
8. Wikipedia - all models https://en.wikipedia.org/wiki/All\_models\_are\_wrong