ISSN: 2583-9888
DOI: doi.org/10.51219/JAIMLD/Naveen-muppa/75

URF PUBLISHERS
connect with research world

Journal of Artificial Intelligence, Machine Learning and Data Science

https://urfpublishers.com/journal/artificial-intelligence

Vol: 2 & Iss: 1

Original Article

Enterprise Data Integration Architecture

Naveen Muppa*

10494 Red Stone Dr Collierville, Tennessee, USA

Citation: Naveen Muppa. Enterprise Data Integration Architecture. *J Artif Intell Mach Learn & Data Sci 2024*, 2(1), 234-237. DOI: doi.org/10.51219/JAIMLD/Naveen-muppa/75

Received: 02 January , 2024; **Accepted:** 18 January, 2024; **Published:** 20 January, 2024

***Corresponding author:** Naveen Muppa, 10494 Red Stone Dr Collierville, Tennessee, USA

ABSTRACT

This paper focuses on the Functional and Technical design of the CIH architecture and all the relevant publications and subscriptions.

Keywords: The data hub is an integration architecture that helps determine effective mediation of semantics (governance and sharing) by providing a unified and efficient data exchange infrastructure to collect and connect data across applications, enterprises and ecosystems.

Below are the Key functions of the Data Hub:

- Orchestrate Data Integration across multiple sources – on-premises, Cloud through a governed publication/ subscription hub.
- Connectivity to ingest all data - Any Type, Any Scale, From Any Source
- Flexible, agile, and efficient architecture that reduces dependency on source system and eliminates the traditional point-to-point integration approach, by reuse and consistencies.
- Foundation for future Analytics and MDM capabilities: advanced analytics, machine learning, big data, etc.
- Centralized monitoring and alerting for improved governance.
- Technical Capabilities:
- Data Integration Tools (Bulk/Batch, Real time)
- Application Integration (API)
- Common Data Models
- Data Governance Controls
- Persistent Storage
- Technology Components:
- Informatica Cloud Data and App Integration Platform (iPaaS)
- Informatica Cloud Integration Hub (CIH)
- Data Store in Azure (Blob, Relational/NoSQL)

1. Introduction

Identity and access management combines processes, technologies, and policies to manage digital identities and specify how digital identities are used to access resources.

Identity and access management initiatives tend to be more complex than most other IT projects, simply because of the number and diversity of identity stores, protocols, encryption mechanisms, policies, and governing bodies that need to work together. A comprehensive strategy can significantly reduce the effort required to manage digital identities in a large network by implementing standards, reducing the number of identity stores, establishing trust, delegating administration, and improving the user sign on experience while strengthening security.

An organizational strategy for identity and access management needs to include well-defined answers to the following questions:

- What are the benefits that identity and access management initiatives should produce?
- What challenges must each initiative overcome?
- What are the specific organizational factors that must be addressed?
- What business and technology projects and solutions are necessary to support each initiative?

Your organization needs to have a clear idea of the specific benefits that improved identity and access management initiatives should bring. Without this guiding vision, the outcome will not deliver concrete improvements and will likely lead to a more complex and cumbersome system.

When you evaluate potential benefits, do not overlook the challenges of implementing a particular technology solution. There should be a balance between the expected benefits and the size and complexity of each solution.

2. Access Management Technologies

Effective identity and access management involves several interdependent technologies and processes. These elements combine to maintain a unified view of identities in an organization and use them effectively. The main topics for discussion in identity and access management include directory services, identity life-cycle management, access management, and how applications should integrate with the infrastructure.

Note This chapter provides an overview of each of these topics and the remaining chapters examine the processes and services within each topic in much more detail. For an overview of these topics, read this chapter. For a more rigorous, technical treatment, read Chapters ٤ through ٧.

Many identity and access management technologies and solutions have evolved independently in response to specific tactical problems. Organizations, analysts, vendors, and systems integrators have increasingly come to recognize that these technologies and business problems are all interdependent, resulting in a single category that is simply described as "identity and access management.

To visualize this interdependency, Microsoft created the Identity and Access Management Framework, a graphical depiction of the services and processes involved in identity and access management.

The following figure shows the main components of the Microsoft Identity and Access Management Framework:

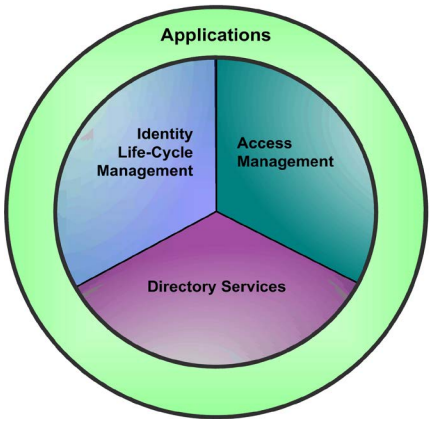

Figure 1: Main topics of the identity and access management framework.

3. Comparing Strong and Week Authentication Techniques

Authentication techniques can range from simple ones where users provide passwords directly to applications or hosts to much more complicated ones that use advanced cryptographic mechanisms to protect user credentials against potentially malicious applications and hosts.

Providing a plaintext password (that is, one that is not encrypted in any way) to an application or host is considered the weakest authentication technique because of the danger of interception of the authentication sequence. Also, if the user authenticates to a malicious host, the owner of that host has all the necessary information to act as that user anywhere on the network. If you think of a password as a secret, then it is not much of a secret if the user must tell every computer on the network what the secret is.

Stronger authentication techniques protect the authentication credentials so that the host or resource to which the user authenticates does not know what the secret is. Typically, this is done by cryptographically signing data with the secret password that is known only to the user and a trusted third party (such as an Active Directory domain controller). A computer authenticates the user by presenting the signed data to the trusted third party. The third party then compares the signature to what it knows about the user and advises the computer whether it believes the user is who they say they are. Such a mechanism helps keep passwords as true secrets. Such occurrences can significantly damage customer faith and lead to legal consequences, thereby further magnifying the.

4. Single Sign on

An important part of any authentication discussion is the concept of single sign on (SSO). SSO at the application level involves establishing a "session" between the client and server that allows the user to keep using the application without providing a password every time they take an action within the application.

The same kind of idea can be extended to a set of applications available on the network. To implement SSO amongst different applications, sessions can be established between the client, a trusted third party on the network, and various server applications and network resources. The session is represented in many implementations by a ticket or cookie which is best thought of as a substitute credential for the user. Instead of requiring the user to provide their credential during authentication, the ticket or cookie is sent to the server and accepted as proof of the user's identity.

The result is that the user only has to sign on once before using many applications - thus providing a single sign on experience.

Note Only in very unusual circumstances is it considered appropriate for an authentication mechanism to force the user to repeatedly provide authentication credentials. Applications, on the other hand, may sometimes prompt for credentials before performing a particularly sensitive operation.

5. Authorization

Authorization is the process of determining whether a digital identity is allowed to perform a requested action. Authorization

occurs after authentication, and maps attributes associated with the digital identity (such as group memberships) to access permissions on resources to identify which resources the digital identity can access.

5.1. Access control lists

Different platforms use different mechanisms for storing authorization information. The most common authorization mechanism is known as an access control list (ACL), which is a list of digital identities along with a set of actions that they may perform on the resource (also known as permissions).

Actions are typically defined relative to the type of object the ACL protects. For example, a printer might allow actions such as "print" or "delete job" while a file might allow actions such as "read" and "write".

5.2. Security groups

Operating systems that support large numbers of users typically support security groups, which constitute a special type of digital identity. Using security groups reduces the management complexity of dealing with thousands of users in a large network.

Security groups simplify management because an ACL can have a few entries specifying which groups have a specific level of access to an object. With careful group design, the ACL should be relatively static. You can easily change authorization policy for many objects at a time by manipulating the members of a group maintained by a centralized authority, such as a directory. Nesting groups within each other increases the flexibility of the group model for managing authorization.

5.3. Roles

Many applications use the term role to refer to a user classification. For example, a "Manager" role could be used to refer to all members of a security group called "Finance Managers," who as members of this group would automatically be granted the entitlements to network resources this role provides.

Roles can also be based on dynamic, run-time decisions that provide more flexibility, such as authorization in an expense report application. This application may have approval actions that only authorized users (or principals) can validate in the "Approving Manager" role. However, before granting approval to authorize an expense, the system queries the directory to determine if the submitter's "Manager" attribute matches the name of the person approving the expense. Such business-driven logic is almost impossible to configure with ACL-type mechanisms.

Roles can be defined either globally, such as by group memberships in a directory, or with application code that determines role membership based on a dynamic query. There are even combinations of both types, such as an application that defines a role called "Managers" that is locally defined to include both the global group's "HR Managers" and "Engineering Managers" roles.

There are advantages to each of these methods. A well-designed roles mechanism provides application developers with the flexibility to choose among them for the correct fit.

6. Trust

The concept of trust is becoming more important as organizations continue to share resources with their business partners. The ability to establish trust between independently administered systems is crucial for IT systems to support the required level of data exchange. Trust enables secure authentication and authorization of digital identities between autonomous information systems with less management overhead.

The mechanisms of trust are complicated because there are many tasks that must happen between independent organizations to make the authentication and subsequent authorization processes useful. The trusting organization needs to have a secure mechanism to communicate with the trusted organization. Once the trusting organization has authenticated the foreign digital identity, it must incorporate the entitlement information about that foreign account into the authorization process within the trusting organization.

6.1. Federation

A federation is a special kind of trust relationship established beyond internal network boundaries between distinct organizations. Federation enables the secure authentication and authorization of digital identities between autonomous information systems based on the principle of trust. For example, a user from company A can use information available at company B because there is a federated trust relationship between the two companies.

Federation is an attempt to remove the requirement for management of accounts in more than one place. In federation, a user from one organization can authenticate directly to a resource managed by another organization using his or her normal network account. This idea is popular because it can remove the requirement (or at least make the requirements much easier to meet) for administration of many different accounts.

Consider an organization that does business with one hundred different partners. The alternative to federation would be for the organization to use a delegated administration interface to manage accounts on one hundred different partner extranets. Through this example, it should be obvious that techniques such as delegated administration do not scale to highly connected business environments. The ability to federate digital identities reliably and securely is essential for advancing business opportunities.

6.2. Trust and federation in microsoft technologies

Microsoft Windows provides support for trust and federation through the following technologies:

- External trusts in Windows NT 4.0 and Windows Server.
- Cross-forest trusts in Windows Server.
- The Kerberos version 5 authentication protocol.
- Shadow accounts.
- PKI trusts.
- Active Directory Federation Service (ADFS) in Windows Server.

7. Summary

The following figure lists all of the processes and services in the Microsoft Identity and Access Management Framework.

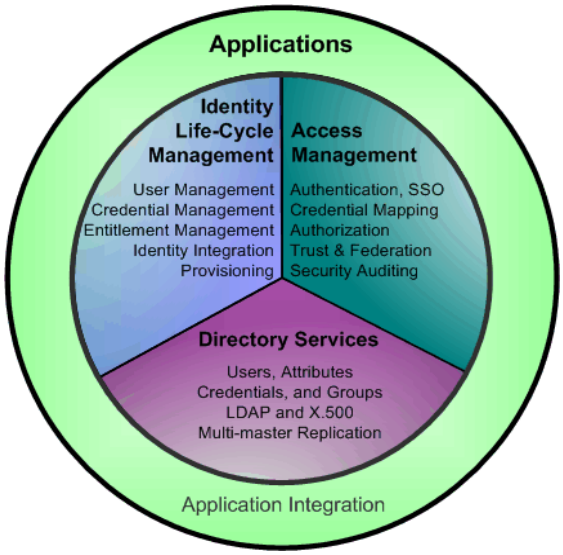

Figure 2: Processes and services in the Microsoft identity and access management framework.

8. References

1. http://go.microsoft.com/fwlink/?LinkId=14841
2. http://go.microsoft.com/fwlink/?LinkId=66833
3. http://go.microsoft.com/fwlink/?LinkId=66834
4. http://www.oasis-open.org
5. http://go.microsoft.com/fwlink/?LinkId=66835
6. http://go.microsoft.com/fwlink/?LinkId=66837
7. http://go.microsoft.com/fwlink/?LinkId=66838
8. http://go.microsoft.com/fwlink/?LinkId=66839
9. http://go.microsoft.com/fwlink/?LinkId=66840
10. http://go.microsoft.com/fwlink/?LinkId=66841
11. http://go.microsoft.com/fwlink/?LinkId=66843
12. http://go.microsoft.com/fwlink/?LinkId=66844
13. http://go.microsoft.com/fwlink/?LinkId=66846
14. http://go.microsoft.com/fwlink/?LinkId=66847
15. http://go.microsoft.com/fwlink/?LinkId=66848