Authentication and Authorization (v2)

By Mark Morowczynski (Microsoft) and Michael Epping (Microsoft)

Updated by the IDPro Body of Knowledge Committee for v2

To comment on this article, please visit our GitHub repository and submit an issue.

Table of Contents

Abstract

This article describes the fundamentals of authentication and authorization, two core components of Identity and Access Management. It also delves into federation and Identity Providers, common tools for performing authentication and authorization in an organization.

Introduction

This article describes authentication and authorization, two core components to a sound Identity and Access Management strategy. Organizations typically have multiple tools that leverage authentication and authorization, both on-premises and in the cloud. The core concepts of each are described, and common ways authentication and authorization are used are explored.

Terminology

Many of these terms have been sourced from the "Terminology in the IDPro Body of Knowledge".[i]

Term	Definition
Access Control Lists	Access Control Lists are definitions around who or what are allowed or denied access to a resource. For example, a file share may have an Access Control List that allows Marketing Department users to read and write, IT Department users to read-only, and denies all other users' access.
Attribute-Based Access Control (ABAC)	A pattern of access control system involving dynamic definitions of permissions based on information ("attributes", or "claims"), such as job code, department, or group membership.
Authentication	Authentication is the process of proving that the user with a digital identity who is requesting access is the rightful owner of that identity. Depending on the use-case, an 'identity' may represent a human or a non-human entity; may be either individual or organizational; and may be verified in the real world to a varying degree, including not at all.
Authorization	Determining a user's rights to access functionality with a computer application and the level at which that access should be granted. In most cases, an 'authority' defines and grants access, but in some cases, access is granted because of inherent rights (like patient access to his/her own medical data). Authorization is evaluating what access or rights an identity should have in an environment.
Directory	A directory is a central repository for user identities and the attributes that make up those identities. A user identity might be John Smith

	with firstName attribute as John, lastName attribute as Smith, title attribute as Director, and Department attribute as Marketing. The attributes in the directory can be used to make authorization decisions about what this user should have access to in applications.
Identification	Uniquely establish a user of a system or application.
Identity Federation	An identity federation is a group of computing or network providers that agree to operate using standard protocols and trust agreements. In a Single Sign-On (SSO) scenario, identity federation occurs when an Identity Provider (IdP) and Service Provider (SP) agree to communicate via a specific, standard protocol. The enterprise user will log into the application using their credentials from the enterprise rather than creating new, specific credentials within the application. By using one set of credentials, users need to manage only one credential, credential issues (such as password resets) can be managed in one location, and applications can rely on the appropriate enterprise systems (such as the HR system) to be the source of truth for a user's status and affiliation. Identity federations can take several forms. In academia, multilateral federations, where a trusted third party manages the metadata of multiple IdPs and SPs, are fairly common.
Identity and Access Management	Identity and Access Management (IAM) is the discipline used to ensure the correct access is defined for the correct users to the correct resources for the correct reasons.
Identity Information Authority, aka Sources of "Truth"	This represents one or more data sources 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 Internet2.
Identity Provider	An Identity Provider (IdP) performs a service that sends information about a user to an application. This information is typically held in a user store, so an identity provider will often take that information and transform it to be able to be passed to the service providers, AKA apps. The OASIS organization, which is responsible for the SAML specifications, defines an IdP as "A kind of SP that creates, maintains, and manages identity information for principals and provides principal authentication to other SPs within a federation, such as with web browser profiles."
Multi-Factor Authentication	An approach whereby a user's identity is validated to the trust level required according to a security policy for a resource being accessed using more than one factor (something you know (e.g., password),

	something you have (e.g., smartphone), something you are (e.g., fingerprint).
Relying Party	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 in the ISO/IEC standards 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). An RP roughly corresponds to the Agency Endpoint in the FICAM model or to Identity Consumers in the Internet2 model.
Role-based access control	A pattern of access control system involving sets of static, manual definitions of permissions assigned to "roles", which can be consistently and repeatably associated with users with common access needs. Role-based access control is a control scheme in which roles are granted to identities, and those roles determine what access to resources those identities should have. Basic roles might be "admin" and "read-only user" – an admin would be able to make changes to a system and a read-only user would only be able to view resources.

What is Identification

Identification is the act of determining which identity is in use or being interacted with by uniquely establishing a user of a system or application. Before an identity can be authenticated, it must be determined which identity is being used. A common way this occurs is through a user providing their username. The username is used to identify the user, while the password is used to authenticate the user.

Some types of credentials can provide both identification and authentication simultaneously, such as FIDO credentials or some biometrics.

What is Authentication?

Conceptually, authentication, sometimes abbreviated as AuthN, is the process of ensuring ownership of an account at the time the account is used to access a resource or establish a session. You complete authentication dozens of times a day and don't even realize it. When you log in to your computer with your username and password, you just did authentication. Then when you log in to check your email through a browser or an application like Outlook, you again authenticate to prove you own or are otherwise responsible for that email account. When you pick up your mobile device and use a biometric like a fingerprint or your face to unlock the device, you again complete authentication. If you go to the ATM to withdraw money, you first need to provide a card

and then a PIN. If you successfully authenticate, then the ATM can trust that you are the owner of the account. Authentication can take different forms for different resources:

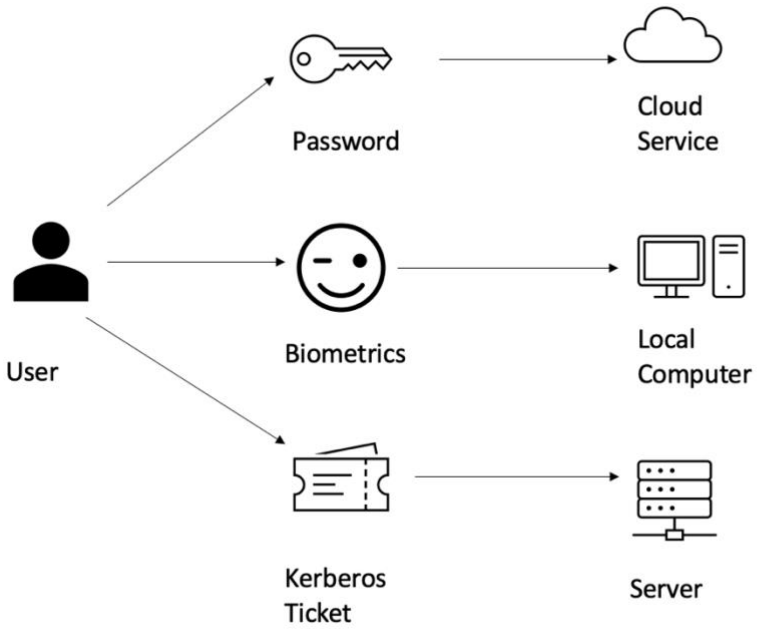

Figure 1: The user and their different authentication factors

There are many different possible authentication factors, such as memorized secrets, hardware tokens, and biometrics. These are often referred to as "something you know," "something you have," and "something you are." The most common factor is username and password. Also growing in popularity is the use of multifactor authentication methods. The most common is a text message or phone call, although these are no longer the strongest options available. There are also methods like One Time Passcode (OTP) software apps or hardware keys, where the password can be used only once and is usually valid for a limited duration. There are also different authenticator apps where a push notification is sent to the device and approved by the end-user. Physical FIDO2 security keys and biometrics like fingerprints and facial recognition are becoming more common and passkeys are rolling out to replace the standard password authentication ceremony entirely.[ii] Non-human identities also need to authenticate. Computers and services authenticate to each other using things like certificates, shared secrets (really just a password for an application), or other protocols developed for this purpose. Authentication, it's not just for people!

Authentication is often the first step when an entity wants to access a resource. We must first determine which identity is trying to access the resource and determine if it is the legitimate identity or an imposter. Then we can move on to the next step, determining what access, if any, should be granted or denied to the identity.

What is Authorization?

The next critical part of Identity and Access Management is authorization, sometimes abbreviated as AuthZ. Conceptually you can think of this as what an entity is allowed to do. Once the system or services knows who you are through authentication, you will be granted rights or permissions to do things through authorization. Authentication helps verify you are the same subject every time; authorization determines if you as the subject are allowed to access or do whatever action you are trying to do. These rights can be as simple as viewing a file (a grant permission) or denying the ability to view a file (a deny permission). You've probably experienced this when someone sent you a file or a link to a site and you received an "Access Denied" error message. You don't have the authorization to access that resource. You've also experienced this when you were able to view a file or access a site. There was just no message saying you were allowed to do it! You've probably come across hundreds if not thousands of authorization decisions a day and not even realized it (unless you get stopped, of course).

Authorization decisions can be made based on many factors. To start with a common one, if you have a specific role assigned to your account, you might have permissions in the system to add, modify, delete, or view things. This authorization architecture is commonly referred to as Role-Based Access Control (RBAC). For example, if you hold the role of administrator in a system, you might be able to manage all aspects of that system. Alternatively, if you hold the role of a reader in the system, then you may be able to view all the same things as the administrator, but you don't have the ability to make any changes.

Similarly, there are Attribute-Based Access Control (ABAC) systems where users may be granted specific rights depending on attributes on their account. For example, if you are a member of the sales organization, you would probably be a member of a sales group in your corporate directory or have a Department attribute set to "Sales". This group membership or attribute would grant you access to the sales shared network folder or a file-sharing site. But you wouldn't be able to access the engineering shared network folder or engineering site. Only those that were a member of the engineering department would be able to. These decisions are made typically by Access Control Lists (ACLs) determined by the system administrator. RBAC and ABAC are large topics unto themselves, deserving of their own articles.[iii]

Another example of authorization based on information about the user could be their job title. When a regular user logs into their HR application, they see information about themselves. How many hours they've worked, their manager, their pay stub, and information about their benefits. They are only authorized to view their own information. Their manager has a similar view about their own information but may also have additional information they can see about their employees. They can see all the hours worked for their direct employees but can't see that about other employees in the organization. Based on their title, they are only authorized to see that additional information about their direct

reports. Finally, the head of HR might expect to see a wide range of information about the company. They might expect to see total hours worked for everyone in the company, total payroll, and benefits spent. Because they hold the title of Head of HR, they are authorized to see all this information.

Authorization applies to non-human accounts as well. A service account can hold roles in most directories. It would have the same permissions as any human account with that role. Service accounts can also be members of groups. A common example of this is the service that runs the backups on Windows servers. Depending on the design, it might require membership to a high privilege group, like Backup Operators, in order to backup and restore files on the system.[iv]

At this point, the concept of authorization should be clear and may seem straightforward. Authorization grants or denies permissions to various resources for both human and non-human accounts. However, the implementation details of this can be extremely complex. In our example above, the sales team and engineering team have access to separate corporate resources. But what do we do when they need to collaborate on something? Engineering has a new product coming out, and the sales team needs to be able to sell it. Do we add the sales team to the engineering group? Should we add the engineering team to the sales group? Or do we create a NEW group called Sales-Engineering and add the sales group and the engineering group to that new group? This addition of a new group might seem like the correct solution, but what do we do when the operations group also needs to work with engineering to ensure the production of the product meets engineering standards. Operations also need to work with sales to ensure the supply chain is aligned with their sales projections. Do we create more groups for all three teams to work together? As you can see, this starts to grow and get out of hand. Having an authorization design for these types of scenarios is important before you start implementing an Identity and Access Management (IAM) solution as well as how you will handle exception cases that will arise.

Lastly, we also need to make sure we are following the concept of least privilege when it comes to authorization. Least privilege is part of a robust strategy to ensure that users and service accounts only have the minimum permissions necessary to perform their. It is easy to grant more permissions such that things will work in an effort to make the authorization process simpler, but we'll pay the price later for those decisions, often in catastrophic ways. It's also often much more difficult to remove permissions from users and non-human accounts after they have been implemented. Take the time at the start to ensure least privilege is being followed for authorization decisions. Your future self will thank you.

The Role of Identity Providers and Federation

Both authentication and authorization may occur within a single system or application or may be externalized via an identity federation. If you have an application that doesn't

reside on your corporate intranet (i.e., is a cloud-hosted service), your users will still need to authenticate.[v]

The identity provider, frequently abbreviated as IdP or IDP, handles the authentication of the user. The authentication can be via a web browser using forms-based authentication, integrated windows authentication (IWA), or an application using a web API. It's really user authentication as a service. There are common on-premises IdPs as well as cloud services that can be used as IdPs. These IdPs are commonly also doing some degree of authorization. Suppose a user is not able to authenticate to the IdP because they do not have an account or they do not have access assigned to a particular application. In that case, the IdP will not issue the user any assertion that can be used to access the application. If the user successfully authenticates, then the IdP issues assertions to the application/relying party.

Assertions, sometimes also referred to as claims, are pieces of information that are sent to the application/resource provider that, in this case, identifies the user and any additional information about the user that the application needs to function. These pieces of information are also referred to as attributes. The firstName attribute may be provided as an assertion and have values such as "John" or "Jane". The information requested and sent varies from application to application, but information such as title, manager, employee ID, etc., can be included in the assertion.

Before a user can authenticate and have information sent as an assertion to the application and access it, a federation trust needs to be set up.[vi] The setup details vary between federation protocols, but the IdP and the application will essentially exchange some information, such as the IdP public key and the application's endpoints for authentication. This information is typically in the metadata of the trust. Standards, such as FastFed, define how this metadata should be formatted to establish application and IdP trust.[vii]

Federation and IdPs allow us to control authentication and authorization for applications even outside the corporate network. These are important tools, especially in modern environments where cloud applications and services continue to proliferate. Organizations must be able to authenticate users, validate they are who they say they are, authorize them, and grant them the appropriate access based on who they are, everywhere – including on-premises and the cloud.

Conclusion

This document is a review of two core IAM concepts: authentication and authorization. These concepts are used in every organization to validate identities and grant those identities the appropriate access once they've been determined to be legitimate. Validating the legitimacy of an identity is crucial to keeping attackers out of organizations' systems. Granting the least permissions necessary to the identity is also recommended; it mitigates

the damage if and when the wrong user or a compromised account accesses or has higher-than-necessary level of privilege in a system, thus reducing the blast radius of any nefarious actions as much as possible. Federation via Identity Providers (IdPs) is a common way to perform this authentication and authorization today, as applications and services are increasingly found outside corporate networks. Authentication and authorization techniques can protect these resources and identities regardless of location.

Author Bios

Michael Epping is a Program Manager in the Azure AD Engineering team at Microsoft. He is part of the customer experience team; his role is to accelerate the adoption of cloud services across enterprise customers. Michael helps customers deploy Azure AD features and capabilities via long-term engagements that can last years, as well as working within the engineering organization as an advocate on behalf of those customers. Michael has more than nine years of experience working with customers to deploy Microsoft products like Azure AD, Intune, and Office 365.

Mark Morowczynski (@markmorow) is a Principal Program Manager on the customer success team in the Microsoft Identity division. He spends most of his time working with customers on their deployments of Azure Active Directory. Previously he was Premier Field Engineer supporting Active Directory, Active Directory Federation Services, and Windows Client performance. He was also one of the founders of the AskPFEPlat blog. He has spoken at various industry events such as Black Hat 2019, Defcon Blue Team Village, GrayHat, several BSides, Microsoft Ignite, Microsoft Inspire, Microsoft MVP Summits, The Experts Conference (TEC), The Cloud Identity Summit, SANs Security Summits, and TechMentor. He can be frequently found on Twitter as @markmorow arguing about baseball and sometimes making funny gifs.

Change Log

Date	Change
2021-09-30	V1 published
2022-12-15	V2 published; terminology section expanded (ABAC, Identification, Identity Information Authority, Relying Party); included reference to passkeys; removed information on PKI

[i] "Terminology in the IDPro Body of Knowledge," IDPro Body of Knowledge, updated 30 September 2021, https://bok.idpro.org/article/id/41/.

[ii] For more information on FIDO2, see Fido Alliance, "FIDO2: WebAuthn & CTAP – Moving the World Beyond Passwords," website, https://fidoalliance.org/fido2/ (accessed 28 September 2021); ; for more information on passkeys, see FIDO Alliance, "Passkeys," website, https://fidoalliance.org/passkeys/ (accessed 10 November 2022).

[iii] For more information, see Koot, André, "Introduction to Access Control," IDPro Body of Knowledge, 17 June 2020, https://bok.idpro.org/article/id/42/, and McKee, Mary, "Policy-Based Access Control," IDPro Body of Knowledge, 19 April 2021, https://bok.idpro.org/article/id/61/.

[iv] For more information on managing non-human accounts, see Williamson, Graham and André Koot, "Non-human Account Management," IDPro Body of Knowledge,30 October 2020, https://bok.idpro.org/article/id/52/.

[v] For more information on identity federations and sources of truth, see Lunney, Patrick, "Federation in the Enterprise," IDPro Body of Knowledge, 19 April 2021, https://bok.idpro.org/article/id/62/, and Dingle, Pam, "Introduction to Identity - Part 2: Access Management," IDPro Body of Knowledge, 17 June 2020, https://bok.idpro.org/article/id/45/.

[vi] For more information on IAM architectures, see Dobbs, G. B., (2021) "IAM Reference Architecture", *IDPro Body of Knowledge* 1(6). doi: https://doi.org/10.55621/idpro.76

[vii] OpenID Foundation, Fast Federation (FastFed) Working Group, website, https://openid.net/wg/fastfed/ (accessed 31 August 2021).