Difference between authentication and authorization pdf

Authorization and authentication. Difference between authentication authorization and accounting. Difference between authentication authorization.

When you're starting out in web development, you'll likely hear the terms authentication and authorization all the time. And it doesn't help that they're both usually abbreviated 'auth', so it's very easy to get the two confused.In this article, you will learn:The differences between authentication and authorizationHow each of these processes workExamples of authorization and authentication in everyday life.Ok, let's get started.What is Authentication?Authentication is the the process of verifying the credentials a user provides with those stored in a system to prove the user is who they say they are. If the credentials match, then you grant access. If not, you deny it.Methods of AuthenticationSingle Factor authentication:This is often used as the authentication process for lower risk systems. You only need a single factor to authenticate, with the most common being a password, so it's more vulnerable to phishing attacks and key loggers. In addition to this, a recent article by DataProt showed that 78% of Gen-Z people utilize the same password for multiple services. This means that if an attacker gained access to one user account, they have a high probability of gaining access to others by simply using the same password.2-Factor Authentication:This method is more secure, as it comprises two factors of authentication – typically something you know, for example username and password , plus something you have / own, for example a phone SMS or a security token. For 2-factor authentication, you would enter a one-time SMS password sent to your device, or perhaps a linked authenticator app code and provide an ever-changing access code.As you can imagine, this is a lot more secure than simply entering a password, or a single authentication credential. You would need to know the login credentials, as well as have access to the physical device for the second part.2-factor authentication has become very common amongst online services in recent years, and with many large companies it is the default authentication method. Many require that you setup 2-factor auth in order to even utilize the service.Multi-Factor Authentication:Going one step further to make your authentication process even more secure is having 3 or more factors. This form of authentication usually works on the premise of:something you know (username + password or a username + security question and answer)something you have (mobile phone sms, authenticator app, USB key)something you are (like a fingerprint / face recognition)For these reasons, multi-factor authentication offers the most protection, as you would need to compromise multiple factors, and these factors are a lot more difficult to "hack" or replicate. The downside to this method of authentication, and the reason it's not utilized in many average systems, is it can be cumbersome to setup and maintain. So the data / system you're protecting really has to justify the need for such security.This question comes up at many security architecture meetings, and the answer is "it depends".It is not unusual for companies to combine various authentication methods to increase security based on the nature of application.For example, take a banking app. It contains very sensitive information, and could have a huge financial and reputational impacts should it be obtained by the wrong person.

Authentication Vs Authorization

Consider authentication and authorization all about the attacker and less about the victim

Authentication
Authentication is the process of asserting the identity of a user before granting access into a system. In simple terms, it means verifying users by confirming who they say they are.

Authorization
Authorization refers to validating the roles, permissions, and privileges assigned to a specific user. It is performed after authentication to grant or deny access rights to users for certain resources.

Authentication
are you who you say you are?

Authorization
are you allowed to do this action?

Difference Between Authentication And Authorization

Both authentication and authorization confirm the identity of users and are often used interchangeably. But in reality, they perform different functions.

Authentication	Authorization
Verifies user identities.	Validates access permissions.
Verifies users to affirm if they are who they say they are.	Confirms whether users have permission to access certain resources.
Determines via. factors like username passwords, retina scan, facial recognition, etc. to identify users.	Validates users' permissions and privileges to access resources through pre-specified rules.
Performed before authorization.	Performed after authentication.
Data is transmitted through Token IDs.	Data is transmitted through access tokens.
Example: Employees are required to authenticate themselves before they can access organizational emails.	**Example:** After successful authentication, employees' are only allowed to access certain functions based on their roles.

AuthN And AuthZ Techniques

Understanding Authentication and Authorization Within The Organizational Environment

Password-Based Authentication

A user first creates an account by providing the necessary details, such as email address and password, and then accesses the account using the details.

Passwordless Authentication

Using this type of authentication, a user can either log in through a magic link or through an OTP delivered via email or text message.

Social Authentication

This method uses existing credentials from social networking platforms such as Facebook, Twitter, Google, Microsoft, etc. to identity users. The user need not fill out any registration form.

Multi-Factor Authentication

Two-factor or multi-factor authentication may be used to include an extra security layer as a step-up and flexible authentication.

API Authentication

API authentication is the process of certifying user identity attempting to access services on the server. Some of the most popular authentication APIs include:

- Basic HTTP Authentication
- Core API Authentication
- OAuthentication

Barcode Authentication

This approach involves logging into computers or facilities without manual typing by scanning a barcode. Web applications make extensive use of it to authenticate users and provide access.

Biometric Authentication

It includes the use of distinctive biological features of the individual to validate identity. The user's biometric data is captured and stored in the database which is then compared to confirm user authentication.

HTTP Authorization

This technique is used in both authentication and authorization. A user simply enters a username and password to prove their authentication. Since the HTTP header itself is leveraged, this method does not include cookies, session IDs, or login pages.

API keys

This method is also used in both authentication and authorization. When the user tries to obtain authorized access to a system during registration, an API key is generated. Henceforth, it is paired with a hidden token and sent along with forwarding requests. When the user wants to re-enter the program, their unique key is used to validate the identity.

HMAC Authorization

Hash-Based Message Authentication Code (HMAC): Most APIs allow users to sign in to an API key to use the API. The API key is a long string that you usually include either in the URL or header of the request. The API key acts primarily as a means of identifying the person calling the API. This method is used in both AuthN and AuthZ.

OAuth 2.0 Authorization

OAuth allows the API to authenticate and access the requested system or resource. OAuth 2.0 is one of the most secure methods of API authentication and supports both authentication and authorization.

JWT Authorization

JSON Web Token (JWT) is an open standard for securely transmitting data between parties. It is another secure method of identification that supports both authentication and authorization. JWT is commonly used for authorization and can be signed using a secret or a public/private key pair.

SAML Authorization

Security Assurance Markup Language (SAML) is an authentication and authorization system based on XML between two entities: a service provider and an Identity Provider. SAML is a standard Single Sign-On format (SSO) where authentication information is exchanged through XML documents that are digitally signed.

OpenID Authorization

OpenID Connect is an authentication layer on top of OAuth 2.0, a framework for authorization. It allows clients to verify the end-user identity based on an Authorization Server's authentication, as well as to obtain interoperable and REST-like basic profile information about the end-user.

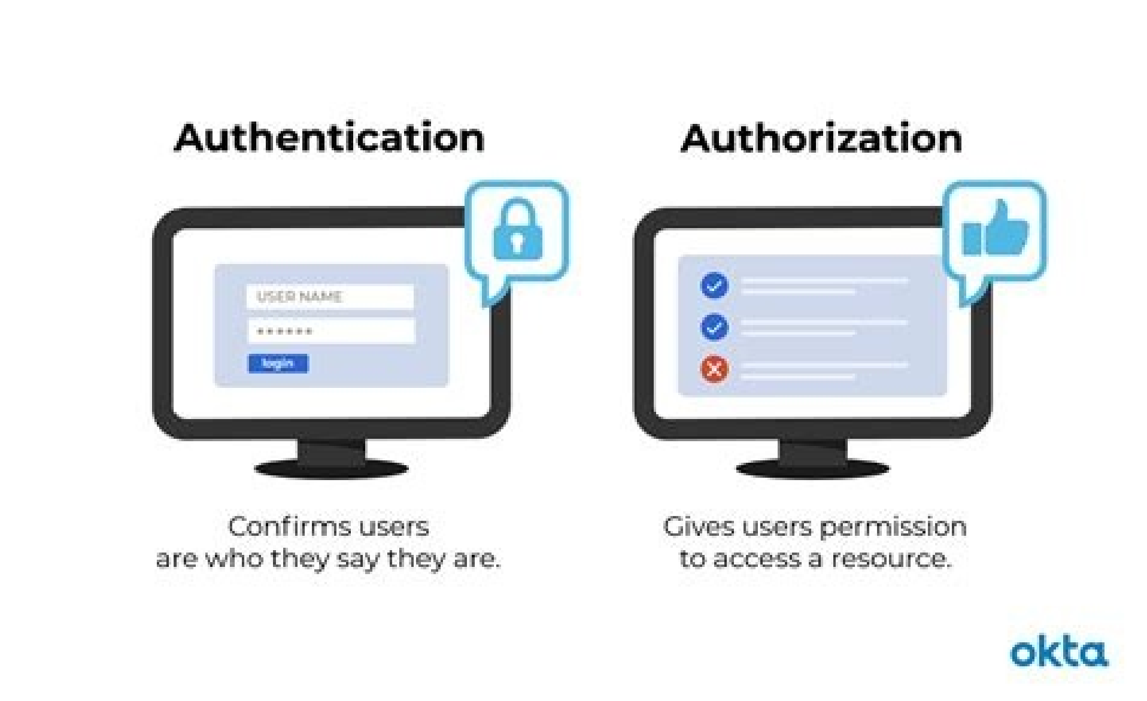
ABOUT LOGINRADIUS

LoginRadius empowers businesses to deliver a delightful customer experience without compromising security. Using our customer identity platform, companies can offer a streamlined login process while protecting digital accounts and complying with data privacy regulations.

© LoginRadius Inc | www.loginradius.com

The bank may combine personal questions to be answered, along with a customer number and complex password.On the other hand, for a social media site, you might only require a username and password, which is then checked and verified before allowing access.It's all about the level of risk involved and what information someone can access once they're in the application. This helps determine the level of authentication you need.If you or your team underestimates the level of authentication your app needs, you could be prosecuted for not securing the data within your system adequately. So companies employee security specialists to advise on best practices and appropriate solutions.How

Does Authentication Work in the Real World?Let's take an example of a social media account. You choose your favorite social media site (which is hosted on a server).


Authentication
Authorization
USER NAME
login
Confirms users are who they say they are.
Gives users permission to access a resource.
okta

The server will ask you to provide credentials to access the site via a sign in page. Here you would type in your username and password that you used when creating the account.Image showing the authentication processThese details are then sent to the server, and the authentication process begins. The details you provided are verified and checked in the server's database, and if they match the details on record you are authenticated.

Then you're provided with a form of identification data, for example a cookie or Json Web Token (JWT token).Success! You have accessed the site and are given entry.You can learn more about JWT tokens in another FreeCodeCamp article by Beau Carnes here.Next, let's look at authorization.Authorization, is the process of verifying that you're allowed to access an area of an application or perform specific actions, based on certain criteria and conditions put in place by the application. You may also hear it called access control or privilege control.Authorization can either grant or deny permission to carry out tasks, or access areas of an application.Let's look at an example:We've gained access to the social media site, but what we're allowed to do there depends on what we're authorized to to do.If we try to access someone's profile that we're not friends with (they've not accepted our connection request), we're not authorized to view their profile. This means that we are denied permission to view their shared posts.Image of authorization flowThere are many ways you can implement authorization depending on the frameworks you are using.Within the .NET framework, for example, you could use role-based access control, or claims-based access control.Role-based access control is centered around the ideology that each user within your system is assigned a role.

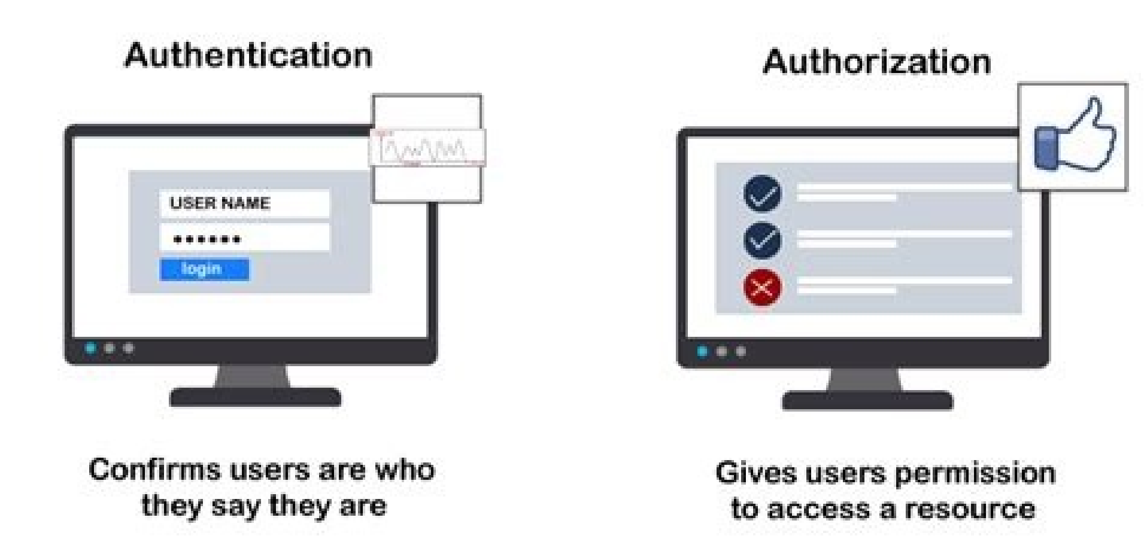
These roles have predefined permissions associated with them. Being granted a role means that user will automatically inherit all these permissions. The roles are assigned at time of user creation and setup.The endpoint or site simply then checks if the current logged-in user has the role of Admin when attempting to access the admin area.The downside to this approach is that sometimes users are granted too many permissions that they don't need or shouldn't have.For example, giving a user the role of Admin may mean they would have been givenAdvanced Create, Edit, Delete, and View user privileges.

Whereas, you may want to only give them View and Basic Create permissions.Claims-based access control can allow for finer tuning of a specific user's permissions. The application can either check that the claim simply exists on a user, or whether a particular value is assigned to the claim.As an example, a claim called CreateUser could be given to a user, and this is checked when creating a user. Or you could assign a value of Advanced to the same claim, and then have different actions and user interface available depending whether the value was Advanced or Basic.So now that we have a better understanding of the terms, let's look at a scenario you may be familiar with that involves both processes.At a dinner party with an exclusive guest list, each guest is given a nickname and a secret password.Upon arrival, a security guard asks you for your nickname and secret password. They then authenticate your credentials against the list they have. If your credentials match, you are handed an envelope showing you've been allowed in.Once inside you are allowed to access the party and public areas of the venue as these require no authorization (everyone has the permission to enjoy the party). However, you then want to visit the VIP area.As you approach, another security personnel asks to open your envelope (your permissions and roles). They take a look but unfortunately you do not have the VIP role, and therefore are not authorized to access.Put as simply as possible, authentication verifies the identity of a user or service allowing access, whereas authorization determines what they can do once they're in.As you can see, although authentication and authorization are very different, each plays an integral part in the security and integrity of the application or system.These processes go hand in hand, and without one the other is kind of meaningless.

If you can gain access to the Admin area, but do whatever you want once in there, it could lead to big problems.On the other hand, you can't authorize individuals without knowing who they are! Which is why authentication always comes before authorization.Closing ThoughtsI hope this has been insightful and you now have a clearer understanding of the differences between Authorization and Authentication, and how to use them.Remember:Authenticate = Verifies the identity of a user or process.Authorize = Determines if the user / system has permission to use a resource or carry out an action.Feel free to get in touch via Twitter if you wish to discuss this article in more detail @gweaths. If you read this far, tweet to the author to show them you care. Learn to code for free.

freeCodeCamp's open source curriculum has helped more than 40,000 people get jobs as developers. Get started Obwohl Authentication und Authorization häufig synonym verwendet werden, handelt es sich dabei um unterschiedliche Prozesse, die ein Unternehmen vor Cyberangriffen schützen sollen. Angesichts der zunehmenden Häufigkeit und des Ausmaßes von Datenschutzverletzungen sind Authentifizierung und Autorisierung die erste Linie der Verteidigung, mit der verhindert werden soll, dass vertrauliche Daten in die falschen Hände geraten. Daher sollten solide Methoden für die Authentifizierung und Autorisierung ein wichtiger Bestandteil der allgemeinen Sicherheitsstrategie jedes Unternehmens sein. Was ist Authentifizierung, was ist Autorisierung, und worin liegt der Unterschied? Kurz gesagt: Bei der Authentifizierung wird überprüft, wer eine Person ist, während bei der Autorisierung überprüft wird, auf welche spezifischen Anwendungen, Dateien und Daten ein Benutzer Zugriff hat. Ein Beispiel aus dem Leben macht es deutlich: Eine Fluggesellschaft muss festlegen, welche Personen an Bord kommen dürfen. Dabei besteht der erste Schritt darin, die Identität eines Passagiers zu bestätigen, um sicherzustellen, dass er derjenige ist, der er vorgibt zu sein. Sobald die Identität eines Fluggastes festgestellt wurde, besteht der zweite Schritt in der Überprüfung von Sonderleistungen, zu denen der Fluggast Zugang hat, sei es ein Flug in der ersten Klasse oder ein Besuch in der VIP-Lounge. In der digitalen Welt erfüllen Authentication und Authorization dieselben Ziele. Mit der Authentifizierung wird überprüft, ob die Benutzer wirklich die sind, für die sie sich ausgeben. Sobald dies bestätigt ist, wird die Autorisierung verwendet, um dem Benutzer die Erlaubnis zu erteilen, auf verschiedene Ebenen von Informationen zuzugreifen und bestimmte Funktionen auszuführen, abhängig von den Regeln, die für verschiedene Arten von Benutzern festgelegt wurden. AuthenticationAuthorizationMit der Authentifizierung wird überprüft, wer der Benutzer ist.Die Autorisierung bestimmt, auf welche Ressourcen ein Benutzer zugreifen kann.Die Authentifizierung erfolgt über Passwörter, Einmal-PINs, biometrische Informationen und anderen, vom Benutzer bereitgestellten oder eingegebenen Informationen.Authorization erfolgt über Einstellungen, die von der Organisation implementiert und gepflegt werden.Authentication ist der erste Schritt in einem angemessenen Identitäts- und Zugriffsmanagementprozess.Die Autorisierung findet immer nach der Authentifizierung statt.Die Authentifizierung ist für den Benutzer sichtbar und kann teilweise von ihm geändert werden.Die Autorisierung ist für den Benutzer nicht sichtbar und kann von ihm nicht geändert werden.Beispiel: Sobald die Zugriffsberechtigung erteilt ist, können Mitarbeiter und HR-Manager auf der Grundlage der vom Unternehmen festgelegten Berechtigungen auf verschiedene Datenebenen zugreifen. Gängige Authentication-Methoden Während die Benutzeridentität in der Vergangenheit mit einer Kombination aus Benutzername und Passwort überprüft wurde, stützen sich die heutigen Authentifizierungsmethoden in der Regel auf drei Arten von Informationen: Was Sie wissen: In der Regel handelt es sich dabei um ein Passwort. Es kann sich aber auch um die Antwort auf eine Sicherheitsfrage oder eine einmalige PIN handeln, die dem Benutzer nur für eine Sitzung oder Transaktion Zugang gewährt. Was Sie besitzen: Dabei kann es sich um ein mobiles Gerät oder eine App, ein Sicherheits-Token oder eine digitale ID-Karte handeln.Was Sie sind: Hier handelt es sich um biometrische Daten wie Fingerabdrücke, Netzhautscans oder die Gesichtserkennung.

Häufig werden diese Arten von Informationen mit mehreren Authentifizierungsebenen kombiniert. Zum Beispiel kann ein Benutzer aufgefordert werden, einen Benutzernamen und ein Passwort einzugeben, um einen Online-Einkauf abzuschließen. Sobald dies bestätigt ist, kann als zweite Sicherheitsebene eine einmalige PIN an das Mobiltelefon des Benutzers gesendet werden. Durch die Kombination mehrerer Authentification-Methoden mit konsistenten Authentifizierungsprotokollen können Unternehmen sowohl die Sicherheit unterstützen als auch die Kompatibilität zwischen Systemen gewährleisten. Sobald ein Benutzer authentifiziert ist, werden Autorisierungskontrollen angewandt, um sicherzustellen, dass die Benutzer auf die benötigten Daten zugreifen und bestimmte Funktionen ausführen können. Dazu gehören z. B. das Hinzufügen oder Löschen von Informationen, basierend auf den von der Organisation erteilten Berechtigungen. Diese Berechtigungen können auf der Anwendungs-, Betriebssystem- oder Infrastrukturebene zugewiesen werden. Zwei gängige Autorisierungstechniken sind: Rollenbasierte Zugriffskontrolle (RBAC): Bei dieser Authorization-Methode erhalten Benutzer je nach ihrer Rolle innerhalb des Unternehmens Zugriff auf Informationen. So können beispielsweise alle Mitarbeiter eines Unternehmens ihre persönlichen Daten wie Gehalt, Urlaubszeiten und Altersvorsorgedaten einsehen, aber nicht ändern. HR-Manager können jedoch Zugriff auf alle HR-Informationen der Mitarbeiter sowie die Berechtigung erhalten, Daten hinzuzufügen, zu löschen und zu ändern. Durch die Zuweisung von Berechtigungen entsprechend der Rolle jeder Person können Unternehmen sicherstellen, dass jeder Benutzer produktiv arbeiten kann, während gleichzeitig der Zugriff auf sensible Daten eingeschränkt ist.Attributbasierte Zugriffskontrolle (ABAC): ABAC vergibt Benutzerberechtigungen auf einer granulareren Ebene als RBAC. Dafür wird eine Reihe von Attributen genutzt. Dazu können Benutzerattribute wie der Name des Benutzers, seine Rolle, Organisation, ID und seine Sicherheitsfreigabe gehören. Umgebungsattribute, wie z. B. der Zeitpunkt des Zugriffs, der Standort der Daten und die aktuelle Bedrohungslage im Unternehmen, können ebenfalls verwendet werden. Auch Ressourcenattribute wie der Eigentümer der Ressource, der Dateiname und der Grad der Datensensibilität können einbezogen werden. ABAC ist ein komplexerer Autorisierungsprozess als RBAC, der darauf ausgelegt ist, den Zugriff weiter einzuschränken. Anstatt z.


Authentication
USER NAME
login
Authorization
Confirms users are who they say they are
Gives users permission to access a resource

B.

allen Personalleitern in einem Unternehmen generell die Änderung von Personaldaten zu gestatten, kann der Zugriff auf bestimmte geografische Standorte oder Tageszeiten beschränkt werden, um enge Sicherheitsgrenzen einzuhalten. Eine solide Sicherheitsstrategie erfordert den Schutz der eigenen Ressourcen, sowohl durch Authentifizierung als auch durch Autorisierung.

Mit einer starken Authentication- und Authorization-Strategie können Unternehmen konsequent überprüfen, wer jeder Benutzer ist und worauf er Zugriff hat, um nicht autorisierte Aktivitäten zu verhindern, die eine ernsthafte Bedrohung darstellen. Unternehmen können ihre Produktivität maximieren und gleichzeitig ihre Sicherheit erhöhen, indem sie sicherstellen, dass sich alle Benutzer ordnungsgemäß identifizieren und nur Zugriff auf die wirklich benötigten Ressourcen erhalten. Das ist extrem wichtig in einer Zeit, in der Datenschutzverletzungen die Einnahmen eines Unternehmens stark beeinträchtigen und die Reputation enorm beschädigen können. Sehen Sie hier, wie SailPoint integriert mit den richtigen Anbietern von Authentication-Lösungen arbeitet. Authentication and authorization are two vital information security processes that administrators use to protect systems and information. Authentication verifies the identity of a user or service, and authorization determines their access rights. Although the two terms sound alike, they play separate but equally essential roles in securing applications and data. Understanding the difference is crucial. Combined, they determine the security of a system.

You cannot have a secure solution unless you have configured both authentication and authorization correctly.

Authentication (AuthN) is a process that verifies that someone or something is who they say they are. Technology systems typically use some form of authentication to secure access to an application or its data. For example, when you need to access an online site or service, you usually have to enter your username and password.

Then, behind the scenes, it compares the username and password you entered with a record it has on its database. If the information you submitted matches, the system assumes you are a valid user and grants you access. System authentication in this example presumes that only you would know the correct username and password. It, therefore, authenticates you by using the principle of something only you would know.

The purpose of authentication is to verify that someone or something is who or what they claim to be. There are many forms of authentication. For example, the art world has processes and institutions that confirm a painting or sculpture is the work of a particular artist. Likewise, governments use different authentication techniques to protect their currency from counterfeiting. Typically, authentication protects items of value, and in the information age, it protects systems and data. Systems can use several mechanisms to authenticate a user.

Typically, to verify your identity, authentication processes use: - something you know - something you have - or something you are Passwords and security questions are two authentication factors that fall under the something-you-know category. As only you would know your password or the answer to a particular set of security questions, systems use this assumption to grant you access. Another common type of authentication factor uses something you have. Physical devices such as USB security tokens and mobile phones fall under this category. For example, when you access a system, and it sends you a One Time Pin (OTP) via SMS or an app, it can verify your identity because it is your device.The last type of authentication factor uses something you are. Biometric authentication mechanisms fall under this category.

Since individual physical characteristics such as fingerprints are unique, verifying individuals by using these factors is a secure authentication mechanism. People often use the terms access control and authorization interchangeably. Although many authorization policies form part of access control, access control is a component of authorization. Access control uses the authorization process to either grant or deny access to systems or data. In other words, authorization defines policies on what a user or service may access. Access control enforces these policies. If we compare authentication and access control, the comparison between authentication and authorization still applies. Authentication verifies the user's identity, and access control uses this identity to grant or deny access. While often used interchangeably, authentication and authorization represent fundamentally different functions. In this article, we compare and contrast the two to show how they protect applications in complementary ways.In simple terms, authentication is the process of verifying who a user is, while authorization is the process of verifying what they have access to. Comparing these processes to a real-world example, when you go through security in an airport, you show your ID to authenticate your identity. Then, when you arrive at the gate, you present your boarding pass to the flight attendant, so they can authorize you to board your flight and allow access to the plane.Here's a quick overview of the differences between authentication and authorization: Authentication Authorization Determines whether users are who they claim to be Determines what users can and cannot access Challenges the user to validate credentials (for example, through passwords, answers to security questions, or facial recognition) Verifies whether access is allowed through policies and rules Usually done before authorization Usually done after successful authentication Generally, transmits info through an ID Token Generally, transmits info through an Access Token Generally governed by the OpenID Connect (OIDC) protocol Generally governed by the OAuth 2.0 framework Example: Employees in a company are required to authenticate through the network before accessing their company email Example: After an employee successfully authenticates, the system determines what information the employees are allowed to access In short, access to a resource is protected by both authentication and authorization. If you can't prove your identity, you won't be allowed into a resource. And even if you can prove your identity, if you are not authorized for that resource, you will still be denied access.