Securing the Digital Gateway: A Comprehensive Review of Identity and Access Management in Modern Application Security

SHARAD SHARMA
(CISSP, Sr. Member IEEE, Alumni IIT Kanpur)

Abstract:

As Application Security controls are increasingly moving towards curbing attacks at the entry point of an application, where users attempt to access the application resources, the category of Identity Management has gained immense importance in the realm of application security. This paper gives a short snapshot of Identity Management and its role in protecting the various applications at Enterprise as well as Consumer facing applications. The paper starts with an introduction to Identity Management giving details of Digital Identities, the Identity Management (IM) Life Cycle, various Authentication & Authorization Mechanisms along with standards and protocols that are relevant for Identity Federation such as OAuth, OpenID Connect, and SAML. The paper thereafter introduces the Application Security domain along with Vulnerabilities and controls to address the same. Also, various identities management controls that can be put in place to help eliminate such vulnerabilities in an application like Principle of Least Privilege, Defense-in-Depth etc. Following this the paper elaborates on the various aspects of authentication starting from the traditional native application or user Password based authentication to step up Multifactor authentication, Biometric Authentication, Password less authentication and Adaptive Risk based Authentication. Similarly, it also covers Authorization and Access Control Models such as Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), Policy based access control (PBAC), Fine Grained Access Control along with Delegated Authorization. The paper then moves towards challenges faced in an application where an Identity and Access Management (IAM) solution is implemented, such as Scalability and Performance, User Privacy and Governance- General Data Protection Regulation (GDPR), Health Insurance Portability and Accountability Act (HIPAA), Financial integrity - Sarbanes-Oxley (SOX) requirements, Identity Federation and SSO challenges, Identity and Access Management for Identity Life Cycle such as Provisioning and Deprovisioning and Threats such as Identity Spoofing, Session Hijacking and Insider Threats. At the end the paper covers Emerging Trends and Technologies in the space of Identity Management such as Decentralized Identity, Blockchain etc., Artificial Intelligence for Anomaly Detection etc., Zero Trust Security Model and Identity as a Service (IDaaS). This paper will focus on human identities defined later and does not elaborate on the privileged or machine identities or non-human identities.

Keywords: Identity Management, Application Security, Authentication, Authorization, Access Control, Multi-Factor Authentication (MFA), Single Sign-On (SSO), Identity Federation, OAuth, OpenID Connect, SAML, Role-Based Access Control (RBAC), Attribute-Based Access Control

(ABAC), Zero Trust, Identity as a Service (IDaaS), Policy based access control (PBAC), Cybersecurity, Open Worldwide Application Security Project (OWASP), Privacy Compliance, Identity Lifecycle Management

1. Introduction

Identity management, also known as identity and access management (IAM), refers to the policies, processes, and technologies used by companies to control and manage identity and access levels to certain or all assets and also have the governance process to manage the access. It involves the management of digital identities and ensuring users have appropriate and approved access to the required resources at the right time and for the right reasons (Alsirhani et al., 2022). As illustrated in Figure 1, IAM encompasses several interconnected services that work together to provide comprehensive identity governance. Identity management includes the creation, management and deletion of digital identities and is broader than just single sign on (SSO) including identity governance, directory services, access request workflows, auditing and more. In terms of applications, an identity management system can be the access/entry point to the application and provide the information required to manage user identity, including where to store user identity attributes, who to grant access to the application and how to manage application access for one or more applications (Prajapati, 2025).

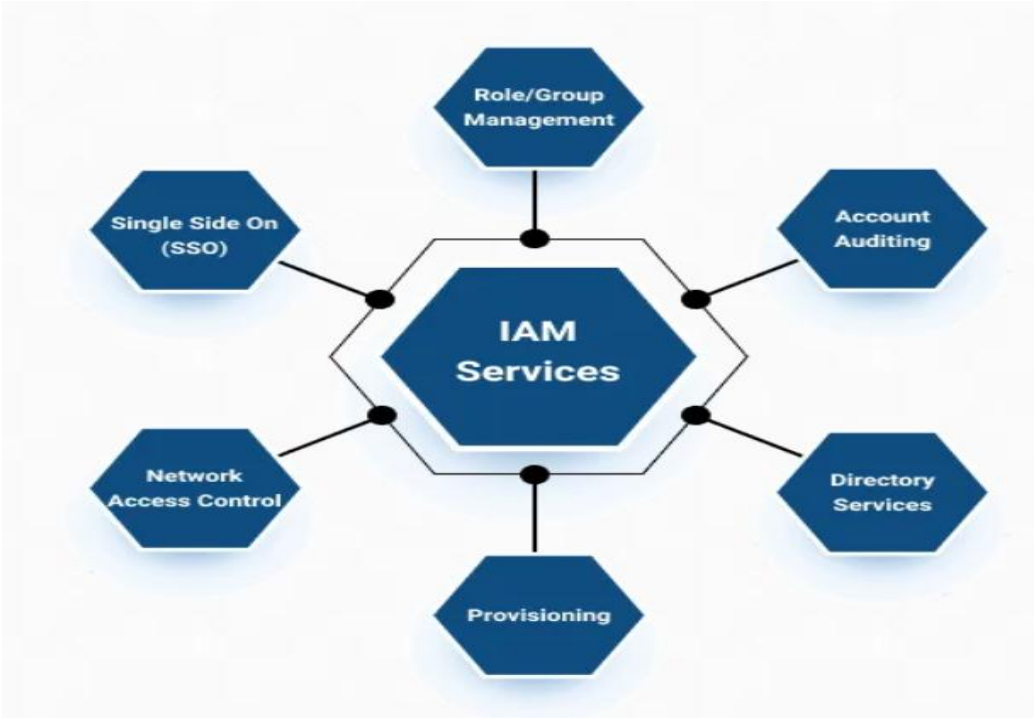

Figure 1. Identity And Access Management (IAM) Services.

Over the years the IAM landscape has expanded to also include ABAC and PBAC and access governance that includes Access Requests and user Access Reviews. Identity management and application security are fundamentally intertwined. Application security is the practice of keeping software applications secure for the entire lifecycle (defining the authorization structure, application specific RBAC, access policies); identity management is about the authentication and authorization flows that allow users to access application features and data. Identity management is built on top of application security practices and therefore a secure identity management implementation cannot exist without a secure application implementation. Conversely, identity management implementation should be considered as the first primary security control an application encounters, and the authentication and authorization checks which occur there will block all unauthorized access to the application (Liu et al., 2019). There are a number of well-known vulnerabilities to identity management that, if implemented, can lead to serious application security issues. Many of these are listed in such places as the OWASP Top 10, such as a number of the items there deal with implementations of authentication and/or session management that can be broken and thus allow users to access applications without proper identification and authorization.

There are a number of reasons why secure identity management is important. Some of the most compelling are as follows: 1. **Increased complexity**: Applications, users and access points are proliferating in vast numbers. Attackers have many more points to strike in today's IT environments, and secure, centralized and consistent identity management is necessary to counterbalance these threat vectors. 2. **Compliance**: As well as SOX, GDPR and HIPAA, numerous industry standards provide extensive guidance on access control to sensitive information or financial integrity, the need for change and access audits and the procedures in place in the event of a security breach. Failure to comply can result in heavy fines. 3. **New security boundaries**: The adoption of cloud-based applications, mobile devices and remote working has destroyed any remaining network boundaries. Today, identity has become the security boundary and must be treated and protected as such. 4. **User experience**: Users demand the highest level of user experience, and this includes easy yet secure access to applications, such as with single sign-on (SSO) and multi-factor authentication. 5. **Financial and reputational consequences**: In the event of a security breach resulting in stolen information such as data theft, ransomware attacks or compromised user accounts (account takeovers) the financial costs are rapidly increasing, as are the negative reputations of the businesses that fail to invest in identity management adequately (Nzeako and Shittu, 2024).

The rest of this paper is structured as follows. **Section 2** covers basic concepts of identity management: digital identities and identity lifecycle, authentication and authorization, identity providers, and also presents an overview of federated identity and corresponding standards and protocols. **Section 3** is an overview of application security with a presentation of common security vulnerabilities and how identity management can be used to secure the application by applying some basic security rules. With **Section 4**, this paper will continue with a presentation

of the different types of authentications: from the most basic password authentication up to multi-factor authentication, biometric authentication, password less authentication and also adaptive authentication. **Section 5** is about authorization and access control. It presents different authorization models such as role-based access control (RBAC), attribute-based access control (ABAC), Policy based access controls (PBAC), fine-grained access control, and delegated authorization. **Section 6** explains the challenges that application developers must consider when integrating an identity management solution, such as scalability, privacy regulations, identity federation, and also the identity lifecycle management and also mentions also some of the new threats that appear in this field. **Section 7** describes the new trends in the identity management field, such as decentralized identity, use cases of artificial intelligence, Zero Trust architecture and also Identity as a Service. Finally, **Section 8** provides the conclusion of this paper with a recap of the main findings and recommendations for future work.

2. Fundamentals of Identity Management

2.1 Digital Identities and Identity Lifecycle

A digital identity is a unique digital image of any entity, formed from the single or combination of attributes relating to that entity. In other words, digital identity is not a one-time phenomenon. An entity can have several attributes of identity that fit into different contexts such as an employee's identity having attributes such as job role and department and an individual's identity as customer of a video streaming service having attributes such as their viewing preferences and their subscription type (Zhu and Badr, 2018). For the purpose of this paper, non-human identities have not been included. Identities will essentially have three attributes:

- **Identifiers:** Elements consisting of numbers, letters or symbols that identify a particular identity (e.g. username, email)
- **Attributes:** Elements that state a fact about the identity (e.g. name, position, location, type of user (employee, vendor, supplier, customers etc.)
- **Credentials:** Elements that relate to proving ownership of an identity (e.g. password, token)

The identity lifecycle refers to the series of stages a digital identity goes through from its inception to its eventual removal (Lesavre, 2020; Ghazizadeh et al., 2012). Different types of users can have different types of lifecycle events. A universal identity lifecycle framework consists of five essential stages:

- **Provisioning:** The creation of an identity including the generation of a unique identifier, the establishment of credentials and the capturing of the subject's attribute profile.

- **Propagation:** The distribution of identity components to other systems or applications which require access to the identity or provide access to the identity.
- **Utilization/Usage:** The utilization of the identity within applications and by users for authentication and access control purposes.
- **Maintenance:** The maintenance of the identity by keeping track of changes in the attributes and credentials associated with a subject.
- **De-Provisioning/Removal:** The removal of an identity when it is no longer required, including the disabling or deleting of the identity

These 5 stages are also referred to as joiner, mover and leaver stages which are tagged to the different lifecycle events an identity goes through. These stages can be distinctive for different types of identities.

2.2 Authentication vs. Authorization

Authentication (AuthN) and authorization are two separate mechanisms that are essential to identity and access management (IAM). Authentication confirms that the user is who they claim to be. The identity of the user is confirmed through credentials that a user must provide that only they should know or have or are. A user can prove their identity through multiple factors, also known as authentication factors, such as something you know (e.g., a password), something you have (e.g., a token), and something you are (e.g., a fingerprint). These are typically tagged to the core principles of who you are (example biometrics), what you know (password that you know) and what you have (example physical token)

Authorization (AuthZ) is the decision that follows successful authentication. Authorization will determine whether an authenticated user is allowed to access a resource or perform an action against that resource and will do so based upon rules and policies that have been established. Authentication basically answers the question of "Who are you?" whereas authorization answers the question of "What are you allowed to do?". It's possible for a user to be successfully authenticated, but still not allowed access to a resource if authorization denies it. So, there is a relationship between the two concepts. Authentication precedes authorization (Naik and Jenkins, 2016; Zhu and Badr, 2018). An application should support both authentication and authorization safeguards.

2.3 Identity Providers and Federated Identity Systems

An Identity Provider (IdP) is the entity that creates and verifies identities and provides authentication assertions to other services. A Service Provider (SP) is an application that needs to verify that a user is who they claim to be. Instead of handling authentication internally, the SP can outsource the authentication to an IdP that it trusts. This is the basis for federated identities, which are agreements between organizations or even applications that allows them to share and use identity information for authentication and authorization purposes. Federated identities

enable users to access multiple applications and domains with a single set of credentials, often referred to as Single Sign-On (SSO). This eliminates the need for users to create separate accounts for each application or domain that they wish to access, providing a level of convenience while maintaining security. Federated identities can take multiple forms. The most common is a hub and spoke model, in which a central entity such as an IdP proxy handles all authentication requests between multiple IdPs and SPs, providing a degree of abstraction and centralized policy management. Another approach is a mesh model, in which each participant in the federation is connected to every other participant, providing flexibility but also complexity in terms of management (Lesavre, 2020; Ghazizadeh et al., 2012).

2.4 Standards and Protocols

Standardized protocols are essential to enable secure, interoperable identity management between different systems and organizations. Standardized protocols specify how identity management requests are issued, identities are shared and identities are confirmed.

- OAuth 2.0: an authorization protocol to allow a third party to access a resource on a secondary service, without needing credentials to be shared; most commonly implemented for delegated authorization; access tokens are provided to the requesting application (Naik and Jenkins, 2016; Naik and Jenkins, 2017).
- OpenID Connect (OIDC): an identity management extension to OAuth 2.0; whereas OAuth 2.0 provides authorization for resource access, OIDC extends this for user authentication; requesting applications can verify identities, obtaining basic info about the user (Naik and Jenkins, 2016; Naik and Jenkins, 2017).
- Security Assertion Markup Language (SAML): an XML-based framework to share identity management information between an identity provider (IdP) and a service provider (SP), typically used to enable single sign on (SSO) where the IdP sends an authentication assertion to the SP, confirming the user has been authenticated (Naik and Jenkins, 2016; Naik and Jenkins, 2017).

3. Application Security Overview

3.1 Common Application Vulnerabilities: The OWASP Top 10

Application security is the process of protecting software applications against vulnerabilities and attacks throughout their development and operational lifecycles. Application vulnerability assessments are typically carried out by security professionals who understand the landscape of application vulnerabilities, such as through reference to the Open Worldwide Application Security Project (OWASP) Top 10. The 2025 update to this well-known vulnerability list shows significant changes that correlate with the evolving application attack landscape, including the promotion of software supply chain failure vulnerabilities and new categories for exceptional condition handling (Fredj et al., 2021; Lala et al., 2021).

Rank	Category	Description
A01:2025	Broken Access Control	Allows attackers to bypass authorization or access resources without proper permissions.
A02:2025	Security Misconfiguration	Systems, applications, or cloud services configured incorrectly, creating vulnerabilities.
A03:2025	Software Supply Chain Failures	Compromises in the process of building, distributing, or updating software, including third-party components.
A04:2025	Cryptographic Failures	Lack of cryptography, insufficient encryption strength, or exposure of cryptographic keys.
A05:2025	Injection	Attackers insert malicious code or commands into program input fields.
A06:2025	Insecure Design	Missing or ineffective control design, including architectural and design flaws.
A07:2025	Authentication Failures	System incorrectly recognizes invalid users as legitimate or fails to properly verify identity.
A08:2025	Software or Data Integrity Failures	Failure to prevent invalid or untrusted code/data from being treated as trustworthy.
A09:2025	Logging and Alerting Failures	Insufficient logging and alerting prevent detection of attacks and hampers incident response.
A10:2025	Mishandling of Exceptional Conditions	Software fails to properly handle errors, exceptions, or unusual system states, potentially leading to security failures.

Table 1. OWASP Top 10 application security risks for 2025.

Table 1 shows the OWASP Top 10 application security risks for 2025, offering a complete view of vulnerabilities that companies should focus on for their security initiatives.

Broken access control is the most prevalent application vulnerability, affecting 3.73% of tests in a prior study. Broken access control includes issues like URL tampering that bypasses access

control, misconfigured missing access controls for APIs, and violations of the principal of least privilege. Security misconfiguration becomes the second most common vulnerability due to the increased complexity of cloud environments and the trend toward configuration-based security rather than other approaches. Software supply chain vulnerabilities appear in third place, even though it has a relatively low number of occurrences, given its high average exploit and impact scores for vulnerable instances (Khayer et al., 2025).

3.2 Role of Identity Management in Mitigating Application Threats

Identity management addresses many vulnerabilities in the OWASP Top 10. Identity-based attacks are now the #1 cause of breaches according to industry reports, so integrating fool proof identity management into your application security landscape is now a must. Identity management secures your applications against threats in several key areas (Khayer et al., 2025).

- **Authentication and Access Control Mitigation:** Identity management is a direct mitigation for authentication failures (A07:2025) since it enables multi-factor authentication that ensures even breached credentials cannot grant unauthorized access. Identity management platforms can support MFA that is phishing resistant and covers all apps, including those that use outdated authentication protocols (e.g. NTLM, Kerberos). The identity management platform can validate users with device-bound credentials and signal real-time device posture compliance before allowing access and blocking attacks that rely on stolen credentials (Kron, 2018).
- **Authorization Enforcement:** Identity management is the key to avoiding broken access control vulnerabilities (A01:2025). Role-based access control (RBAC), attribute-based access control (ABAC), Policy based Access Control (PBAC) and even fine-grained permissions in identity management programs ensure users only access what they're supposed to, eliminating the basic OWASP recommendation that access should only be granted by default for public resources (Kron, 2018).
- **Protection of Sessions and Tokens:** Identity management protocols defend against tampering (A08:2025) with secure session token management and token protection. Contemporary identity management protocols such as OAuth 2.0 and OpenID Connect protect authentication tokens from compromise, but tokens that are intercepted or replayed constitute a continuing threat (Kron, 2018).
- **Threat Detection and Response:** Identity management systems contribute to logging and alerting capabilities (A09:2025) through creating detailed logs of authentication attempts, access decisions, and privilege modifications. These logs enable a security team to identify anomalies, investigate suspicious activity, and respond to detected threats before impacting data integrity (Khayer et al., 2025).

3.3 Security Principles: Least Privilege and Defense-in-Depth

Two key vulnerability mitigation principles which support secure identity management and application security are the principle of least privilege and defense-in-depth.

3.3.1 Principle of Least Privilege

The principle of least privilege (PoLP) states that users and programs must have the least amount of access rights necessary to perform their designated functions. This principle has several key benefits to system security. It reduces potential breach damage by limiting what compromised accounts can access. It minimizes insider threats since authorized users only have minimum required access. Additionally, it supports regulatory compliance; HIPAA, SOX and GDPR encourage least privilege access controls. The principle ensures users and programs have only the minimum access rights needed for their functions. If an account is compromised, damage is limited to its granted privileges. This approach reduces insider threat risks by restricting access to only what's necessary for specific tasks. It also helps achieve compliance with regulations like HIPAA and GDPR that advocate least privilege controls (Kron, 2018).

3.3.2 Defense-in-Depth

Defense-in-depth, or layering, refers to a security strategy that eliminates single points of total compromise through multiple layers of security controls and risk accepting mitigations. The basic idea is that no single layer of security can provide 100% protection, therefore, organizations must build layers of security so that even should one layer fail, others remain effective. In the context of identity management and application security, defense-in-depth manifests across multiple dimensions (Fredj et al., 2021; Lala et al., 2021).

- **Network Layer:** Network segmentation and ACLs keep workloads isolated, and control traffic between environments to limit attacker movement.
- **Application Layer:** WAFs and RASP tools block malicious inputs in real-time, reinforcing app-level auth and auth controls.
- **Identity Layer:** MFA, risk-based conditional access, and continuous authentication keep identity claims credible throughout sessions.
- **Data Layer:** Data is encrypted at-rest (AES-256) and in-transit (TLS), including field-level encryption for especially sensitive data.
- **Monitoring Layer:** Logs & alerts from apps, identity providers, and infra flow into a single observability platform for detection & response.

The defense-in-depth approach recognizes that applications have sophisticated attackers that will try to exploit any weakness in any of the layers of protection. Instead of relying on a single, effective mechanism, multiple, complementary control mechanisms can be layered to create an overall level of protection that is not diminished, even if one of the layers of protection can be breached by an attacker seeking to maintain access to application environments over time. This approach is particularly relevant for identity management, as identity management

implementations themselves can become the focal point for attackers seeking to create persistent access to application environments (Fredj et al., 2021; Lala et al., 2021).

4. Authentication Mechanisms

Authentication mechanisms are the building blocks of identity management systems and are the primary means by which users authenticate themselves to applications. This section reviews the historical development of authentication, from passwords to biometrics, password less authentication, and risk-based adaptive authentication.

- **Password-based Authentication and Its Shortcomings:** Password-based authentication is the cheapest and easiest form of authentication to implement. Yet it has many shortcomings. It is vulnerable to brute-force attacks, replay attacks, phishing, and social engineering (Alsaleem and Alshoshan, 2021). Passwords create a cognitive load for users, encouraging weak or reused passwords (Alsaleem and Alshoshan, 2021). Password cracking also exposes massive user bases when password databases are compromised (Jiang et al., 2015). Thus, passwords are ineffective for securing critical applications.
- **Multi-Factor Authentication and Biometrics:** Multi-Factor Authentication (MFA) and Biometrics MFA increase the security of authentication by requiring multiple (independent) factors (something the user knows, has, or is), thus increasing the difficulty of factor compromising (Dasgupta et al., 2017). Typical factors that are combined with passwords include one-time codes, hardware tokens, and biometric factors such as fingerprints or face recognition. The advantages of biometrics are its uniqueness and non-transferability, which increases the strength of the authentication (Ratha et al., 2001). However, biometric systems must also be designed to mitigate privacy and template protection concerns because biometric data cannot be changed if it is compromised (Sarkar and Singh, 2020). In addition, the complexity and usability of MFA can also reduce its adoption (Ibrokhimov et al., 2019).
- **Password less and Context-Aware Authentication:** Password less authentication solutions, such as those based on Fast identity online Alliance (FIDO 2) protocols using security keys or biometric authentication, enable password-less authentication thus reducing the attack surface and improving usability (Lyastani et al., 2020). This solves many of the issues related to passwords and provides cryptographic proof of possession. However, user acceptance and recovery scenarios remain challenging (Lyastani et al., 2020). Adaptive authentication uses device, location and behavioral attributes to adapt authentication criteria in real-time thus enhancing security and usability (Alsaleem and Alshoshan, 2021).
- **Risk-Based Authentication Methods:** Risk-based authentication assesses the risk of each access attempt based on a wide range of contextual factors and behavioral patterns, in real-time. If the access attempt shows suspicious or anomalous behavior, it undergoes additional verification or is denied (Alsaleem and Alshoshan, 2021). This approach

provides a low-friction experience for trusted users while securely addressing high-risk scenarios. It also uses machine learning and analytics to detect evolving threats but requires sound modeling and respects user privacy.

These authentication mechanisms combine to provide hierarchical security protection for the application that prevents certain types of attacks that affect the weaknesses of each mechanism individually.

5. Authorization and Access Control Models

Authorization and access control models determine how users who are authenticated are allowed or denied accessing resources within applications. This section will survey the evolution from role-based traditional access control to dynamic attribute-based models, fine-grained permissions, and delegated authorization, all of which support contemporary microservice-based architecture.

5.1 Role-Based Access Control (RBAC) in Applications:

Role-Based Access Control (RBAC) is currently the most prevalent means of controlling access in enterprise applications. RBAC assumes that access rights are assigned to roles (functions) rather than users, and that users are assigned to roles based on their job functions. The abstraction that results simplify administration by reducing the number of assignments that administrators must make. In the RBAC model (defined in ANSI INCITS 359) two types of objects are defined. Roles represent the job function that a user has within an organization. Examples of roles include “manager” and “engineer”. Permissions define what actions users with certain roles are allowed to perform on certain targets. A third type of object is defined for mapping users to roles; this object is called user-role assignments. Finally, a fourth object is defined that specifies which permissions belong to each role; this object is called role-permission assignments. The organization is able to control the access rights of its users using a level of abstraction that makes sense for their business. Despite the many advantages of RBAC, traditional RBAC has limitations as an adaptive security control. Static role assignments cannot consider changes that occur in real time. This includes changes to the trustworthiness of a user, time-limited access permissions, and environmental factors. The problems of role explosion have been well documented in the literature: organizations end up creating thousands of roles in an effort to provide fine-grained access control, which reduces the simplification provided by RBAC. However, dynamic extensions to the RBAC model have recently been proposed. The Task-Role Based Access Control (T-RBAC) model, for example, considers real-time environmental factors and incorporates machine learning that predicts user trustworthiness trends. Other models continuously monitor user behavior and update their estimates of user trustworthiness (Das et al., 2018; Bhatt et al., 2016).

Thus, while traditional RBAC does provide an acceptable level of security, these extended models do maintain security while providing the functionality required for adaptive security controls in cloud computing environments. Even more exciting developments include hybrid

RBAC models that combine blockchain with RBAC for transaction security in intelligent manufacturing systems; these models show that the role-based model is continuing to evolve to meet the security requirements of modern systems.

5.2 Attribute-Based Access Control (ABAC) and Policy-Based Controls

Attribute-Based Access Control (ABAC) or Policy based Access Controls (PBAC) extends RBAC in that access decisions consider the attributes of subjects, resources, actions, and environmental conditions; unlike RBAC, which is based on roles assigned to users in advance, ABAC evaluates the policy conditions of access control policies against the values of the attribute appearing in the access request. PBAC evaluates the policy conditions of access control policies against the enterprise access requests requirements in the access request. The attribute-based or Policy based model are more flexible than RBAC for the complex environment of distributed systems, where decisions about access often depend on additional factors. The National Institute of Standards and Technology (NIST) defines ABAC as an access control method that evaluates subject requests to perform actions on objects and grants or denies access based on the attributes of the subject, object, environment, and policy conditions expressed in terms of these attributes. The main components of ABAC involve subject attributes (e.g., roles, department, clearance level), object attributes (e.g., classification level, owner, creation date), action attributes (e.g., read, write, delete), environmental attributes (e.g., location, threat level), and policy conditions that specify the access control rules for combinations of these attributes. A variety of current research projects aim to automate ABAC policy management to eliminate the administrative burden of managing an increasing number of access control policies. One direction involves using algorithms to create structured ABAC policies from access request logs, with a categorization of principals and resources based on values of their attributes, and rule-based permissions for categories of principals for resources. These types of algorithms also generate positive and negative categories and create new policies or update existing policies with new requirements (Karimi et al., 2021; Zhang et al., 2005; Iyer and Masoumzadeh, 2018; Das et al., 2018).

The organizational move from RBAC to ABAC is more complex. Research has proposed migration paths that extend static infrastructure access management (IAM) products to a policy management framework for dynamic ABAC policies that includes attribute management and policy management to distribute access control permissions in an application-independent manner. Migration to the ABAC policy model is more necessary in complex environments like the cloud and IoT, where traditional centralized access control models lack scalability for large-scale environments with diverse devices and technologies, with issues like high authentication latency, single points of failure, and delayed updates. Several hybrid access control models that combine aspects of ABAC and RBAC have also been introduced. In high-performance, distributed IoT systems, researchers have proposed models using blockchain and edge computing that combine hybrid access control with ABAC and RBAC, maintaining a role-based structure

while providing fine-grained policy enforcement with smart contracts on consortium blockchains (Lazouski et al., 2010).

5.3 Fine-Grained Access Control and Dynamic Permissions

Fine-grained access control enables organizations to manage resource access based on multiple dynamic conditions at a level of detail that aligns with the complexities of the real world. While coarse-grained access control methods rely on a single condition (e.g., a user's role), fine-grained access control methods define permissions based on a combination of conditions, such as job functions, time frames, group memberships, and organizational hierarchies. The importance of fine-grained access control has grown with the emergence of cloud environments, where diverse data types must be accessed based on diverse criteria. Fine-grained authorization limits access to data in a manner that balances security needs with the coexistence of data that requires varied levels of access. Dynamically defined permissions enable context-dependent access control, such as permissions that expire after a specified time frame. In systems inspired by Zanzibar, relationship expiration also supports time-bound permissions, automatically removing expired relationships to boost performance. Use cases for fine-grained access control methods include limiting access to specific data or operations, controlling remote access based on geolocation and time, and managing third-party access. Organizations can use time-based access control to permit access only during working hours and provide temporary read-only access to third-party vendors. Fine-grained access control methods are far more scalable than coarse-grained methods, however, which are easier to set up and more suitable for smaller organizations. Fine-grained access control, however, requires significantly more expertise and, if misconfigured, may create performance issues or security vulnerabilities (Servos and Osborn, 2017; Bhatt et al., 2016).

5.4 Delegated and Federated Authorization

Delegated and federated authorization address the complexity of controlling access in distributed environments and across organizational boundaries. Delegated authorization permits users to grant restricted access to third-party applications without sharing credentials, and organizations can establish cross-domain trusting relationships. Delegated authorization allows users to grant third-party applications access to their resources without sharing credentials. OAuth 2.0 is the most prevalent authorization framework. It specifies how an application gets an access token with limited permissions for what the application can do (Bhatt et al., 2016).

Federated authorization extends delegated authorization across organizational boundaries, allowing users in one organization to access resources in another organization, as long as they share trusted identity information. Federated identity management establishes trusting relationships where identity providers authenticate users in their home organization and send trusted attribute information to service providers in other organizations instead of duplicating identity management effort and compromising security with SAML, OpenID Connect, etc. Research has been conducted into attribute transfer in federated identity management to facilitate authorization decisions based on attributes gathered from multiple sources.

In an IoT environment, decentralized identifiers and zero-knowledge proofs have been suggested for privacy-preserving authentication and, where required, cross-domain interoperability. Hybrid authorization frameworks combine multiple authorization models to meet complex set of requirements. In real-time chat applications, authorization decisions must be made in real-time as they traverse layers of access control, requiring role/attribute-based permissions that must be updated dynamically across all applicable sessions without interrupting the chat sessions. Policy engineering challenges have led to research into formal verification capabilities. In Kubernetes environments, formal methods can express RBAC and ABAC policies in first-order logic and use SMT solvers to identify potential policy conflicts that may provide a path for privilege escalation (Karimi et al., 2021; Iyer and Masoumzadeh, 2018).

6. Identity Management Challenges in Applications

In applications, identity management faces key challenges related to scalability, privacy compliance, federation, lifecycle management and security threats.

- **Challenges Related to Scalability and Performance:** Identity management systems can handle a certain level of traffic. However, with the growth of applications and the complexity of their infrastructures, the existing identity management systems need to scale to handle a high volume of authentication and authorization requests without impacting their performance. Scalable identity management systems can be achieved using blockchain-based identity management systems, particularly those built on top of a decentralized protocol such as OAuth 2.0, which provides a scalable solution to decentralize identity verification while maintaining the security controls required in healthcare (Sutradhar et al., 2023). However, the challenge of implementing an identity management system that is scalable relates to achieving a suitable balance between throughput, latency and resource requirements, which can be particularly difficult to achieve in applications with a high-performance requirement such as healthcare.
- **User Privacy and Compliance (GDPR, HIPAA):** Various compliance challenges around user privacy with regulations like GDPR and HIPAA which create challenges regarding user consent, cross-jurisdictional data transfer, data protection, and ensuring auditability (Alhasan, 2025; Singhal, 2024; Barbaria et al., 2025). Potential solutions include leveraging dynamic consent models and blockchain-enabled compliance verification processes.
- **Management of Identity Federation and Single Sign-On (SSO):** Identity federation allows users to authenticate once across different independent systems with a single set of credentials, enhancing usability and helping to mitigate password fatigue issues. Yet identity federation is also more complex in terms of establishing trust between domains and ensuring interoperability between a wide variety of different domains. Single Sign-On (SSO) solutions require thorough management of authentication tokens and session credentials to prevent token theft or replay attacks (Sutradhar et al., 2023).

- **Identity Lifecycle Management Including Provisioning and Deprovisioning:** Identity management across the identity lifecycle (creation, update, deletion) keeps users secure over time. Proper provisioning ensures users are properly accommodated. Deprovisioning ensures no unauthorized access for users whose roles have changed or who have left the company. Automated lifecycle processes that can be integrated with policies and audit logs reduce errors and insider threats (Alhasan, 2025).
- **Identity Spoofing, Session Hijacking and Insider Attacks:** Identity spoofing allows an attacker to pose as a valid user. Session hijacking steals valid session IDs to hijack an authenticated session. Insider threats include authorized users either maliciously or unintentionally exploiting access. Defensive strategies include multiple factor authentication, monitoring, anomaly detection based on AI/ML and access governance (Folorunso et al., 2024; Mbah and Evelyn, 2024).

In summary, addressing these challenges will require scalable, privacy-compliant frameworks that incorporate secure federation protocols, comprehensive lifecycle mitigations, and advanced threat detection capabilities for the development of effective identity management solutions.

7. Emerging Trends and Technology

Emerging trends in identity management enhance the security, privacy, and usability of digital services.

- **Decentralized Identity and Blockchain-Based Identity Management:** Decentralized Identity (DID) uses blockchain to create self-sovereign identities that do not depend on centralized identity providers. This enables end users to control access to their identity information while ensuring immutable and transparent auditing of access granted through distributed ledgers. Blockchain can verify access to credentials without leaking any information about those credentials, promoting end-user privacy and fault tolerance by removing single points of failure. Did-based services are already emerging in transportation and healthcare, offering a roadmap to expand this access control ecosystem in a decentralized manner without intermediating parties. Challenges remain around usability, regulatory compliance, and integration with other services (Grüner et al., 2019; Singla et al., 2022; Stockburger et al., 2021).
- **Artificial Intelligence for Anomaly Detection in Identity Usage:** AI enhances identity management security through behavior analysis and anomaly detection that signals signs of fraud or unauthorized access. Machine learning algorithms learn normal usage patterns dynamically to identify anomalies in real-time, enabling preemptive action against threats. The addition of AI and blockchain improves the trustworthiness of these systems by linking on-chain identity data to off-chain behavioral data, increasing the reliability of risk assessments performed. The hybrid system exhibits far superior scalability and accuracy to static rule-based systems (Martinez et al., 2024).

- **Zero Trust Architecture Impact on Identity Management:** Zero Trust Architecture (ZTA) is based on the principle of implicit trust of neither internal nor external entities. Identity management in a ZTA implementation continuously validates all users and devices that attempt to access resources, with permission granted based on risk assessed by the access request. Integration of a ZTA with an identity and access management solution enhances the security capabilities of micro-segmentation, least privilege, and real-time threat detection, reducing the surface of the environment and improving protection against insider threats. Trust assessments can be validated in a decentralized manner with blockchain-assisted trust metrics (Pokhrel et al., 2024; Bouras et al., 2021).
- **Identity as a Service (IDaaS) and Cloud Identity Management:** IDaaS provides a scalable cloud identity management platform that supports authentication, authorization, and lifecycle management of identities across complex, geographically distributed environments comprising mixed and matched technologies. This service simplifies identity management and reduces the burden of implementing advanced identity lifecycle and federated authentication capabilities like Single Sign-On (SSO). IDaaS platforms increasingly offer privacy-oriented management capabilities and blockchain integration to decentralize identity management and enhance security without compromising flexibility. Organizations that leverage IDaaS for identity management can avoid the considerable operational and compliance challenges that cloud-based identity management products represent (Gruner et al., 2018; Lee, 2018). IDaaS, when combined with other emerging identity management solutions, enables identity management ecosystems that are significantly more secure, relevant, user-friendly, and interoperable, capable of adapting to the changing threat landscape and complex environments these ecosystems support.

8. Conclusion

Identity management is a key enabler of application security, ensuring that access to sensitive resources is constrained by authorized users and devices. Identity management avoids the risks of unauthorized access, identity theft, and insider threats, which are among the most pressing challenges in the modern application environment. The challenges in identity management include scalability in large-scale and federated environments, privacy regulations (e.g., GDPR, HIPAA), complexity in SSO and Federation, identity lifecycle management issues (timely provisioning and de-provisioning), and sophisticated threats (e.g., identity spoofing, session hijacking). Strategies for mitigating these challenges include the use of decentralized (e.g., blockchain) identity management approaches that provide enhanced user control and trust, the use of AI-based anomaly detection approaches, the use of Zero Trust Architecture that enforces continuous access validation with least privilege access, and the adoption of IDaaS (Identity as a Service) designs that provide flexibility in scalability management while preserving user privacy and security. Future research will focus on overcoming scalability and integration challenges in decentralized blockchain identity management approaches, on enhancing the accuracy and

explainability of AI-based anomaly detection systems, on developing seamless integration approaches for Zero Trust Architecture in evolving application architectures, and on developing effective privacy-preserving mechanisms for cloud-based identity management. Technological advancements will bring self-sovereign identities into widespread use, integrate blockchain and AI-based systems to enhance trust, and develop adaptive identity management systems that address evolving insider threats and novel attack patterns (Khayer et al., 2025; Diro et al., 2024).

9. References

1) Prajapati, V. (2025). Role of Identity and Access Management in Zero Trust Architecture for Cloud Security: Challenges and Solutions. International Journal of Advanced Research in Science, Communication and Technology, 6–18. https://doi.org/10.48175/ijarsct-23902
2) Alsirhani, A., Ezz, M., & Mohamed Mostafa, A. (2022). Advanced Authentication Mechanisms for Identity and Access Management in Cloud Computing. Computer Systems Science and Engineering, 43(3), 967–984. https://doi.org/10.32604/csse.2022.024854
3) Nzeako, G., & Shittu, R. (2024). Leveraging AI for enhanced identity and access management in cloud-based systems to advance user authentication and access control. World Journal of Advanced Research and Reviews, 24(3), 1661–1674. https://doi.org/10.30574/wjarr.2024.24.3.3501
4) Liu, Y., Sun, G., & Schuckers, S. (2019, June 1). Enabling Secure and Privacy Preserving Identity Management via Smart Contract. https://doi.org/10.1109/cns.2019.8802771
5) Lesavre, L. (2020). A Taxonomic Approach to Understanding Emerging Blockchain Identity Management Systems. National Institute Of Standards Technology. https://doi.org/10.6028/nist.cswp.01142020
6) Zhu, X., & Badr, Y. (2018). Identity Management Systems for the Internet of Things: A Survey Towards Blockchain Solutions. Sensors, 18(12), 4215. https://doi.org/10.3390/s18124215
7) Naik, N., & Jenkins, P. (2016, March 1). A Secure Mobile Cloud Identity: Criteria for Effective Identity and Access Management Standards. https://doi.org/10.1109/mobilecloud.2016.22
8) Naik, N., & Jenkins, P. (2017). Securing digital identities in the cloud by selecting an opposite Federated Identity Management from SAML, OAuth and OpenID Connect. 163–174. https://doi.org/10.1109/rcis.2017.7956534
9) Fredj, O. B., Cheikhrouhou, O., Krichen, M., Hamam, H., & Derhab, A. (2021). An OWASP Top Ten Driven Survey on Web Application Protection Methods (pp. 235–252). Springer. https://doi.org/10.1007/978-3-030-68887-5_14
10) Lala, S. K., Kumar, A., & T, S. (2021, May 6). Secure Web development using OWASPGuidelines. https://doi.org/10.1109/iciccs51141.2021.9432179

11) Khayer, B., Mirzaei, S., Alavizadeh, H., & Salehi Shahraki, A. (2025). Blockchain for Secure IoT: A Review of Identity Management, Access Control, and Trust Mechanisms. IoT, 6(4), 65. https://doi.org/10.3390/iot6040065
12) Kron, E. (2018). Effective foundational security principles. Cyber Security: A Peer-Reviewed Journal, 1(4), 343. https://doi.org/10.69554/efpq5846
13) Jiang, Q., Wei, F., Fu, S., Ma, J., Li, G., & Alelaiwi, A. (2015). Robust extended chaotic maps-based three-factor authentication scheme preserving biometric template privacy. Nonlinear Dynamics, 83(4), 2085–2101. https://doi.org/10.1007/s11071-015-2467-5
14) Alsaleem, B. O., & Alshoshan, A. I. (2021). Multi-Factor Authentication to Systems Login. 1–4. https://doi.org/10.1109/nccc49330.2021.9428806
15) Ratha, N. K., Connell, J. H., & Bolle, R. M. (2001). Enhancing security and privacy in biometrics-based authentication systems. IBM Systems Journal, 40(3), 614–634. https://doi.org/10.1147/sj.403.0614
16) Sarkar, A., & Singh, B. K. (2020). A review on performance, security and various biometric template protection schemes for biometric authentication systems. Multimedia Tools and Applications, 79(37–38), 27721–27776. https://doi.org/10.1007/s11042-020-09197-7
17) Dasgupta, D., Roy, A., & Nag, A. (2017). Multi-Factor Authentication (pp. 185–233). Springer. https://doi.org/10.1007/978-3-319-58808-7_5
18) Ibrokhimov, S., Hui, K. L., Abdulhakim Al-Absi, A., Lee, H. J., & Sain, M. (2019). Multi-Factor Authentication in Cyber Physical System: A State of Art Survey. 279–284. https://doi.org/10.23919/icact.2019.8701960
19) Ghorbani Lyastani, S., Schilling, M., Neumayr, M., Backes, M., & Bugiel, S. (2020). Is FIDO2 the Kingslayer of User Authentication? A Comparative Usability Study of FIDO2 Password less Authentication. 268–285. https://doi.org/10.1109/sp40000.2020.00047
20) Lazouski, A., Martinelli, F., & Mori, P. (2010). Usage control in computer security: A survey. Computer Science Review, 4(2), 81–99. https://doi.org/10.1016/j.cosrev.2010.02.002
21) Servos, D., & Osborn, S. L. (2017). Current Research and Open Problems in Attribute-Based Access Control. ACM Computing Surveys, 49(4), 1–45. https://doi.org/10.1145/3007204
22) Bhatt, S., Patwa, F., & Sandhu, R. (2016, November 1). An Attribute-Based Access Control Extension for OpenStack and Its Enforcement Utilizing the Policy Machine. https://doi.org/10.1109/cic.2016.019
23) Iyer, P., & Masoumzadeh, A. (2018). Mining Positive and Negative Attribute-Based Access Control Policy Rules. 161–172. https://doi.org/10.1145/3205977.3205988
24) Das, S., Mitra, B., Atluri, V., Vaidya, J., & Sural, S. (2018). Policy Engineering in RBAC and ABAC (pp. 24–54). Springer. https://doi.org/10.1007/978-3-030-04834-1_2
25) Karimi, L., Aldairi, M., Joshi, J., & Abdelhakim, M. (2021). An Automatic Attribute-Based Access Control Policy Extraction from Access Logs. IEEE Transactions on

Dependable and Secure Computing, 19(4), 2304–2317. https://doi.org/10.1109/tdsc.2021.3054331

26) Zhang, X., Li, Y., & Nalla, D. (2005). An attribute-based access matrix model. 359–363. https://doi.org/10.1145/1066677.1066760

27) Folorunso, A., Adewumi, T., Adewa, A., Okonkwo, R., & Olawumi, T. (2024). Impact of AI on cybersecurity and security compliance. Global Journal of Engineering and Technology Advances, 21(1), 167–184. https://doi.org/10.30574/gjeta.2024.21.1.0193

28) Singhal, S. (2024). Data Privacy, Compliance, and Security Including AI ML (pp. 111–126). Igi Global. https://doi.org/10.4018/979-8-3693-2909-2.ch009

29) Barbaria, S., Jemai, A., Ceylan, H. İ., Muntean, R. I., Dergaa, I., & Boussi Rahmouni, H. (2025). Advancing Compliance with HIPAA and GDPR in Healthcare: A Blockchain-Based Strategy for Secure Data Exchange in Clinical Research Involving Private Health Information. Healthcare, 13(20), 2594. https://doi.org/10.3390/healthcare13202594

30) Mbah, G., & Evelyn, A. (2024). AI-powered cybersecurity: Strategic approaches to mitigate risk and safeguard data privacy. World Journal of Advanced Research and Reviews, 24(3), 310–327. https://doi.org/10.30574/wjarr.2024.24.3.3695

31) Alhasan, T. K. (2025). Managing legal risks in health information exchanges: A comprehensive approach to privacy, consent, and liability. Journal of Healthcare Risk Management : The Journal of the American Society for Healthcare Risk Management, 44(4), 12–24. https://doi.org/10.1002/jhrm.70002

32) Sutradhar, S., Karforma, S., Bose, R., Roy, S., Djebali, S., & Bhattacharyya, D. (2023). Enhancing identity and access management using Hyperledger Fabric and OAuth 2.0: A block-chain-based approach for security and scalability for healthcare industry. Internet of Things and Cyber-Physical Systems, 4, 49–67. https://doi.org/10.1016/j.iotcps.2023.07.004

33) Martinez, D., Magdalena, L., & Savitri, A. N. (2024). AI and Blockchain Integration: Enhancing Security and Transparency in Financial Transactions. International Transactions on Artificial Intelligence (ITALIC), 3(1), 11–20. https://doi.org/10.33050/italic.v3i1.651

34) Grüner, A., Mühle, A., Gayvoronskaya, T., & Meinel, C. (2019). A Comparative Analysis of Trust Requirements in Decentralized Identity Management (pp. 200–213). Springer. https://doi.org/10.1007/978-3-030-15032-7_18

35) Stockburger, L., Kokosioulis, G., Mukkamala, A., Mukkamala, R. R., & Avital, M. (2021). Blockchain-enabled decentralized identity management: The case of self-sovereign identity in public transportation. Blockchain: Research and Applications, 2(2), 100014. https://doi.org/10.1016/j.bcra.2021.100014

36) Bouras, M. A., Lu, Q., Dhelim, S., & Ning, H. (2021). A Lightweight Blockchain-Based IoT Identity Management Approach. Future Internet, 13(2), 24. https://doi.org/10.3390/fi13020024

37) Gruner, A., Muhle, A., Gayvoronskaya, T., & Meinel, C. (2018, July 1). A Quantifiable Trust Model for Blockchain-Based Identity Management. https://doi.org/10.1109/cybermatics_2018.2018.00250

38) Lee, J.-H. (2018). BIDaaS: Blockchain Based ID As a Service. IEEE Access, 6, 2274–2278. https://doi.org/10.1109/access.2017.2782733

39) Singla, A., Gupta, N., Aeron, P., Jain, A., Sharma, D., & Bharadwaj, S. S. (2022). Decentralized Identity Management Using Blockchain. Journal of Global Information Management, 31(2), 1–24. https://doi.org/10.4018/jgim.315283

40) Pokhrel, S. R., Yang, L., Rajasegarar, S., & Li, G. (2024). Robust Zero Trust Architecture: Joint Blockchain based Federated learning and Anomaly Detection based Framework. 48, 7–12. https://doi.org/10.1145/3672200.3673878

41) Diro, A., Zhou, L., Saini, A., Kaisar, S., & Hiep, P. C. (2024). Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities. Journal of Information Security and Applications, 80, 103678. https://doi.org/10.1016/j.jisa.2023.103678