Introduction to Identity - Part 2: Access Management

By Pamela Dingle

Table of Contents

Abstract

Who are you, and what are you allowed to do? In digital systems, these questions are the domain of Identity and Access Management (IAM). Access management systems provide the mechanisms for deciding who is who, and to evaluate and enforce decisions about who should get access to what. Part 2 of the introduction to the IDPro Body of Knowledge explores the big picture of access management from a historical perspective. You can expect a little advice, a lot of context, and an experience-based overview of what we do in access management and why our contributions matter.

Terminology

- Ceremonies - predictable interactions that users can infrequently navigate in a well-watched place
- Delegated authorization framework - an access control framework that decouples authentication from authorization, allowing the password to stay local and protected.[i]
- Federated Identity - the means of linking a person's electronic identity and attributes, stored across multiple distinct identity management systems.[ii]
- Least privilege - also known as the Principle of Least Privilege; a resource, such as a user, must only be able to access the resources (e.g., applications, data) that are necessary for it to function.[iii]
- Trust federation - a trust framework between multiple entities with the purpose of leveraging identity and access management information in a controlled fashion.
- Zero trust - From NIST Draft Special Publication 800-207, "Zero trust assumes there is no implicit trust granted to assets or user accounts based solely on their physical or network location (i.e., local area networks versus the internet)"[iv]

Introduction

What is access management, and why is it so exciting? There is something thrilling and urgent about the moment a decision is made, a gate is lifted, and a precious resource is made available to a stranger. Did we make the right person productive, or did we make a risky mistake? Good access management depends on good identity data; it also requires policies that represent corporate rules, an accurate understanding of current environmental and contextual factors, and tools that can enforce according to a defined risk tolerance. A lot of preparation and consideration goes into the run-time decisions that are made every day and that operate with all kinds of granularity at infrastructure, middleware, and application layers.

If you are an experienced identity professional, you have watched our tools evolve - but if you are just starting, it can be valuable to hear some perspective on why things are the way they are. Hold on to your hats: this introduction is not even remotely objective, but it will give you one perspective on how we got here and how the concepts discussed in later chapters have evolved into our current access management landscape.

To kick off the ride, here are a few critical realities to keep in mind in the world of access management:

Resources need stability

Company secrets, financial transactions, and personal communications are just a few examples of the precious resources that identity professionals are tasked with protecting. Resources may be exposed through application programming interfaces (APIs), web interfaces, or native mobile applications. Adding externalized access management capabilities to a single resource is relatively easy, but adding to a hundred or a thousand is exhausting. Owners of these applications rarely want to make frequent changes. After the first time, you as an identity professional try to schedule an application access management update within the change management windows of a hundred different applications, you will feel the same way.

Resources should not perform local identity management

If every resource you deploy performs its own login functions, it is nearly impossible to ensure that they follow the kinds of best practices detailed in places such as NIST 800-63B or adhere to unified corporate policies.[v] Hundreds of applications each separately attempting to store credentials, protect a login page, and secure an account recovery process present an immense attack surface and make it likely that users will reuse passwords across applications. This pattern means an attacker who guesses the password to one application has a credential that can be replayed to gain access to other applications, and you have no way to know which applications are at risk.

Humans need challenges, but not obstacles

While resources need stability and consistency, humans need empathy. We require users to interact with computer systems to show they are the proper operator of the digital account they claim to have a right to; this process should be easy for a good user and tough for an impostor. The best practice is to create “ceremonies” - predictable interactions that users can infrequently navigate in a well-watched place. While authentication is the best-known ceremony, there are many other ways in which humans are asked to interact, such as self-service registration or account recovery, notifications, or transactional approval. We want users to notice when an unusual ceremony takes place because it may alert them that fraud is happening. Ceremonies are guaranteed to change as new attacks force administrators to try additional techniques, including changes in user experience (UX), authentication factors, and risk detection. While it is important to keep the attackers out, the experience of the good users is critically important. Faced with a tough problem, humans often behave predictably, and that predictability is an attack vector in itself. If you as the administrator make your users’ lives too hard, you become the problem: Users will circumvent the controls you put in place to try to protect them.

Garbage In, Garbage Out

The most visible parts of access management are decisions made in the moment, but those decisions do not exist in a vacuum. Before any access management decision is made, someone has to set up digital rules and policies that closely approximate the business goals of the organization (see “Introduction to Project Management for IAM Projects” for more on managing an IAM project).[vi] User, group, and role context must exist, and some combination of device, network, and risk context as well. By the time a user attempts to access a given resource, all of the data that might go into an access choice should be available. Never forget: It doesn't matter how good your access management infrastructure is if decisions are based on bad input.

Now, on to the Fun Part

Identity professionals end up at the forefront of an age-old problem. We have resources to protect, users who want access, and attackers who want access as well and are really hard to distinguish from users. We need a system that is accurate, but no system will be 100% accurate, so the system must also follow the principles of zero trust, starting with least privilege. We must strongly authenticate users and leverage the environmental context to detect fraud. We must apply a single consistent policy view across a disparate landscape of resources. And we have to verify all the time that our systems are working the way we think they are.

Access Management as an Evolution

This body of knowledge will give you all sorts of data about the basic concepts that are deployed in an access management regime - but why do those mechanisms exist? They evolved in response to both business requirements and security threats. Administrators found themselves lacking in control and created best practices that made administration at scale easier and attacks at scale more difficult.

Password Proliferation Gave Us Directories

When businesses first began accumulating business programs within their private network, every new program required that user accounts be created and deleted. Every program asked each user to set a password. As businesses grew to have hundreds and thousands of programs, users hit the limit of how many usernames and passwords they could remember. Some programs let users choose their own usernames, and as a result, usernames varied wildly across programs. Many programs had wildly varying password policies. It was the wild west and from that wild west came the concept of "directories". Instead of a hundred programs separately storing usernames and passwords, applications began to call out to an external directory of users, often using LDAP (Lightweight Directory Access Protocol).[vii],[viii] Suddenly, users could use one password everywhere, and administrators didn't have to maintain thousands of applications individually. All was well... for a while.

Password Fatigue Gave Us Web Access Management

The upside to user directories and LDAP was that users only had to remember one password. The downside was that even if all applications at the time were within the same network perimeter and were all LDAP-integrated, the user was still prompted for their password every time they used a new application - over the course of a day, that was a lot of typing. The resulting innovation was a new access management technique called "Web Access Management" (WAM).[ix] With web access management, users would authenticate once with their password, and then a (usually encrypted) domain-wide session cookie would be generated that could be read by multiple applications. Instead of performing an LDAP “bind,” the application could check that the user had a valid cookie. Around the same time, other technologies to address password fatigue developed, including Kerberos.[x] These technologies finally give users some relief; a user could log in one time and access multiple applications. The concept of logging in once to access multiple apps has come to be known as ‘single sign-on’ (SSO).

Perimeter Limitations Gave Us Federation

As long as businesses were operating within their network perimeters, access management functions like Kerberos and WAM provided both convenience and security. But the Internet was opening up, and many companies wanted to begin allowing not only their employees to access resources, but also partners and customers. Businesses wanted to create trust relationships with other businesses and enable their users to access each other’s applications. This desire was met through a standard called SAML (Security Assertion Markup Language).[xi] Businesses pre-establish a trust “federation” between two domains and then request a secure introduction whenever a user attempts to access a resource. SAML and other federated identity specifications allowed businesses to retain control over the activities of their own users both in their own domains and across domains. Federated identity remains a backbone of access management, and SAML is still the gold standard for cross-domain access management.

Mobile & API Innovation Gave Us OAuth & Delegated Authorization Frameworks

Federation and SSO are what we call in the industry “user-present” scenarios. We can tell that the user is present in a federation request because the activity occurs using a browser, and browsers don’t have brains - they are ‘passive’ clients, and somebody has to be there to push the buttons and click the links. Around 2007, most business application delivery was focused on the browser - but the release of the first “smartphone” changed the game. Mobile applications could be downloaded from an app store and render data accessed from cloud APIs, just as cloud platforms were becoming popular. Suddenly an ‘active’ software client became a desirable way to talk to users.

Even as users got excited about the power of mobile applications, identity professionals ran into a problem: applications were calling APIs when users were not present, and even worse, many mobile applications wanted to consume and display data from cloud platforms that they were not affiliated to. If a mobile app wanted to access an unaffiliated cloud platform, the only answer was to ask the user for their password and then replay the password within every single API fetch. The result was something called the **password anti-pattern**: users got used to giving away their cloud platform passwords to any client that asked for it, and those clients had to cache user credentials on mobile devices so they could execute API calls in users’ absence.

SAML was not a perfect fit in a mobile context. XML parsers were not built into mobile platforms, and cryptographic requirements were heavy. The resulting access management paradigm was OAuth 1.0, a “delegated authorization framework” that could layer with federated protocols. OAuth addresses the ‘user not present’ scenario: applications ask for and receive an “access token” that does not introduce the user; instead, access tokens represent the ability to access a tightly scoped set data and services on behalf of a user.

Maybe access tokens don’t sound like such a big deal, but when you consider that you can pass access tokens to APIs instead of primary credentials, the results are significant. You prevent API endpoints from ever collecting or validating primary user credentials, thus removing multiple attack vectors around data leakage, man-in-the-middle-attacks, and rogue administrators harvesting credentials. Because the mechanism for authorizing the API is decoupled from the mechanism for authenticating users, the door opens to a world where a user could authenticate with factors other than a password without causing work for applications. Access tokens act as a stable currency that can be centrally architected and scalably deployed.

Multi-Factor Authentication (MFA) Is and Was and Will be Again

Through all of the above antics and shenanigans, password attacks were haunting identity administrators. All sorts of conventions evolved to try to keep attackers out of accounts they didn’t own: we forced people to change their passwords regularly; we forced them to set longer and more complex passwords; we designed our LDAP directories and login forms to stop responding if too many incorrect attempts were made. Despite all these attempts to mitigate the risk, almost any password a human could set and remember without help is trivially attackable. If you doubt this statement, read “Your Pa$$word doesn’t matter“ by Alex Weinert (@alex_t_weinert).[xii] Be prepared to weep.

The revelation that passwords are fundamentally flawed is not new - dating back to at least the ’70s, there has been research on how to get around the need for a human brain in the authentication process.[xiii,xiv] We developed the simple idea that passwords are “something you know,” but also described other options for validating a human’s ownership of a digital account could also include “something you have” or “something you are”. The idea is not that validating the thing you have can replace the thing you know, but rather that a combination of things you have, are, and know would require an attacker to compromise both digital and physical information. Today, the state of the art in multi-factor authentication is very sophisticated. A growing number of users protect their phone with a biometric, navigate an SMS message to confirm a transaction, or use an OTP (one-time password) to improve security without any need to understand the underlying principles.

We all know that MFA must continue to improve in usability to become ubiquitous. Specifications like FIDO2 are industry-changing for access management, not because the problem is solved - but because the problem is ***decoupled*** - FIDO2 (W3C WebAuthn and

FIDO CTAP2) has separated the problem of negotiating cryptographic keys from the problem of requiring user gestures.[xv] The cryptographic key exchange can now stay reliable, while we focus on innovation - and possibly even revolution - in user interactions.

The Best Security is Invisible Security

In addition to the visible ceremonies we put in front of those who attempt access to resources, a lot is happening beneath the surface. We increasingly rely on context to supplement active user challenges in calculating the risk of any given transaction. Adjacent areas to identity are now critical stakeholders in our attempts to prevent identity fraud - Cloud Access Security Brokers (CASBs),[xvi] Unified Endpoint Management (for example, Mobile Device Management or MDM),[xvii] and EUBA (Entity and User Behavioral Analysis)[xviii] fortify our access management regimes. Attackers have learned to defeat static access management processes, so we have evolved our defenses beyond password complexity: if you are not checking passwords against a rapidly updated set of banned strings including lists of newly known-to-be-breached passwords and augmenting this with real-time threat intelligence you are in serious trouble.

And the Moral of the story is...

That brings us to now. Identity professionals today still struggle with all of the anecdotal issues listed here, but we have tools at our disposal and conventions on how to best deploy them. The better we can get as a profession at working together to eliminate fraud, detect abuse, and guide our users towards successful interactions, the better off everyone is. Everyone before you leveraged the work of their contemporaries to take a step forward. Now you have the opportunity to take the next step.

What Will Access Management look like in the Future?

When we look back on today's world of access management, what stories will be our contribution? There will be an assessment of our success in helping users to adopt multiple factors - did we succeed? Did we miss opportunities? As long as we are timid, a huge chunk of our immediate future will be spent mitigating attacks that we already know are mostly preventable. Dragging your feet on MFA as an access management professional today is like catching up on social media when you know you have a report due (a behavior common enough to have its own name: akrasia)[xix]. After the fact, we will ask ourselves why we got in our own way, and there will likely be no good answer.

At some point, when enough administrators adopt MFA and eliminate the easy jackpots that are single-factor passwords, our industry will win this amazing prize:

A whole new wave of inventive attacks!

That may not sound so great, but it really is. Today, attackers can spend almost no money or time and still make a living from doing nothing fancier than running free phishing scripts from the Internet. A strongly authenticated world does not eliminate jackpots, but it does make the pool of criminals able to win those prizes a much more distinguished group. Attackers will move to post-authentication attacks like token theft and consent abuse. And the whole time, identity professionals and others will be making new things! Inventing better ways! Introducing resources and content that businesses want! We will embrace wearables as security devices, perform secure transactions even in hostile places, make the measure of least privilege even tighter. We will get better at tracking the promises that products make to us and better at punishing those who mess with our data. We will find a way to share private things and have true confidence that those private things will never become public. We will weather quantum meltdowns and new social networks, and it will all be a fight worth fighting.

The identity management professional who has read this far is clearly dedicated - and that is a great thing. We need the next generation of professionals to pick up the torch, question all assumptions, and push us into a future where risk is low, productivity is high, and new challenges keep our lives interesting.

Author Bio

Pamela Dingle is a long-time member of the identity management world, and is a Director, running the identity standards team at Microsoft. The identity standards team works with standards bodies like W3C, IETF, and the OpenID Foundation on specifications like OAuth 2.0, FIDO, SCIM, and OpenID Connect, and Pamela works to ensure that customers, product groups, and the industry all understand the value of standards and other identity best practice patterns. Pamela spent eight years as an identity architect and eight years in the office of the CTO at Ping Identity and is a founder of Women in Identity.

[i] Raible, Matt, “What the Heck is OAuth?” DZone Security Zone, 28 April 2018, https://dzone.com/articles/what-the-heck-is-oauth.

[ii] Wikipedia contributors, "Federated identity," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Federated_identity&oldid=949399706 (accessed June 6, 2020).

[iii] Wikipedia contributors, "Principle of least privilege," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Principle_of_least_privilege&oldid=950981064 (accessed June 6, 2020).

[iv] Rose, Scott, and Oliver Borchert, Stu Mitchell, Sean Connelly, “Zero Trust Architecture (2nd Draft),” SP 800-207 (Draft), National Institute of Standards and Technology, February 2020, https://csrc.nist.gov/publications/detail/sp/800-207/draft.

[v] Paul A. Grassi, James L. Fenton, Elaine M. Newton, Ray A. Perlner, Andrew R. Regenscheid, William E. Burr, and Justin P. Richer. 2017. Digital identity guidelines - Authentication and Lifecycle Management. Technical Report. NIST Special Publication 800-63B.

[vi] Graham Williamson and Corey Scholefield. Introduction to IAM Project Management for IAM Projects. IDPro Body of Knowledge, volume 1, issue 1, 31 March 2020. https://bok.idpro.org/article/id/25/.

[vii] Wikipedia contributors, "Lightweight Directory Access Protocol," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Lightweight_Directory_Access_Protocol&oldid=960496535 (accessed June 6, 2020).

[viii] “The Most Complete History of Directory Services You Will Ever Find,” blog, Easy Identity, 13 April 2020, https://idmdude.com/2012/04/13/the-most-complete-history-of-directory-services-you-will-ever-find/ (accessed June 12, 2020).

[ix] Wikipedia contributors, "Web access management," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Web_access_management&oldid=959341667 (accessed June 6, 2020).

[x] Wikipedia contributors, "Kerberos (protocol)," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Kerberos_(protocol)&oldid=960957884 (accessed June 6, 2020).

[xi] Wikipedia contributors, "Security Assertion Markup Language," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Security_Assertion_Markup_Language&oldid=956779073 (accessed June 6, 2020).

[xii] Weinert, Alex, "Your Pa$$word doesn't matter," Azure Active Directory Identity Blog, Microsoft Corporation, 9 July 2019, https://techcommunity.microsoft.com/t5/azure-active-directory-identity/your-pa-word-doesn-t-matter/ba-p/731984.

[xiii] Morris, Robert, and Ken Thompson. "Password security: A case history." Communications of the ACM 22.11 (1979): 594-597.

[xiv] Feldmeier D.C., Karn P.R. (1990) UNIX Password Security - Ten Years Later. In: Brassard G. (eds) Advances in Cryptology — CRYPTO' 89 Proceedings. CRYPTO 1989. Lecture Notes in Computer Science, vol 435. Springer, New York, NY

[xv] "FIDO2:WebAuthn & CTAP," FIDO Alliance, https://fidoalliance.org/fido2/.

[xvi] Wikipedia contributors, "Cloud access security broker," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Cloud_access_security_broker&oldid=949494699 (accessed June 6, 2020).

[xvii] Raam, Giridhara, "The What, Why, and How of Unified Endpoint Management," Integration Zone, DZone, 8 July 2019, https://dzone.com/articles/the-what-why-and-how-of-unified-endpoint-managemen.

[xviii] Petters, Jeff, "What is UEBA? Complete Guide to User and Entity Behavior Analytics," Inside Out Security Blog, Varonis, 29 March 2020. https://www.varonis.com/blog/user-entity-behavior-analytics-ueba/

[xix] Clear, James, "The Akrasia Effect: Why We Don't Follow Through on What We Set Out to Do and What to Do About It," excerpt from Atomic Habits, https://jamesclear.com/akrasia (accessed June 6, 2020).