# FINAL PAPER

Diego Izquierdo

April 23rd, 2025

CYSE 368 /Internship

Spring 2025

TowneBank

# Table of Contents

# Interning at TowneBank

The primary reason I selected TowneBank for my CYSE 368 internship was convenience, as I am currently employed there as an IT Systems Architect. While I would have enjoyed the opportunity to do my internship at a different organization to gain additional experience, exposure and networking opportunities, my professional, academic, and personal obligations simply would not have allowed the flexibility to take on an additional commitment. I am beyond grateful that this course allows students to use their current IT employment to fulfill the cybersecurity internship requirement. For someone working full-time, taking 15 credit hours, and managing personal and family responsibilities, adding a separate internship would have been nearly impossible and back-breaking.

When I mentioned to my manager that one of the requirements for my cybersecurity degree—which TowneBank is partially funding as part of its employee benefits and perks—was to complete a cybersecurity internship, and that I had the option to do it with my current employer, he approved it without hesitation and before I could finish my sales pitch, I even had a Power Point presentation I had put together. One common issue across many organizations, including TowneBank, is that cybersecurity departments often operate in silos, with little collaboration with other teams. Having the opportunity to do my internship with my employer turned out to be a win-win for all parties involved: I get to fulfill a graduation requirement, and our engineering department has the opportunity to build stronger relationships with the cybersecurity team.

My manager and I sat down during one of our one-on-one meetings and analyzed my skillet and the operational gaps in cybersecurity operations and identified four areas that we could target with the internship. The areas we identified are listed below

1. Enhance security posture for private and public cloud environments: Microsoft Azure provides a variety of tools and assessments to identify misconfigurations in an Azure environment. Some of these tools are freely available to the public and some require a paid engagement with Microsoft and third-party vendors. We performed these assessments against our Azure environment and identified several configurations that needed to be addressed.
2. Develop policies and procedures for securing Azure storage accounts and Key Vaults: The Azure storage accounts, and key Vaults fell under Ronco's catchy phrase of "set it and forget it." Unfortunately, these cloud artifacts require some upkeep, particularly the regular rotation of keys and secrets. Once we identified the objects that needed to have the keys and secrets rotated, we decided to implement a policy that required that the keys and secrets are rotated on a regular basis. The next step was to identify the usage and functionality of these artifacts so we could minimize the impact of rotating the keys and secrets. This is still an ongoing effort due to the large amount of storage accounts and key vaults.
3. Implement IAM solution for Azure Cloud: The main purpose of this objective was to identify cloud-based applications that are using local accounts or legacy methods and migrate them to SSO using

SAML or oAuth. We successfully migrated several applications to SSO while also establishing configuration standards.

4. Learn procedures and best practices for managing Cisco Firewalls: The main purpose of this objective was for me to gain hands-on experience with Cisco Firewalls. I did not get a chance to do much with this due to other priorities, but at least I now have access to the firewalls so I can learn at my own pace.

# TowneBank History

**[1]** TowneBank started in 1999 in two-car garage in Portsmouth, VA, based on the idea of strong relationships, excellent service, and the most respected and experienced bankers in each market we serve. Building on the idea of serving others and enriching lives, we now provide a full range of banking and other financial services. Dedicated to a culture of caring, we value each employee and member by embracing their diverse talents, perspectives, and experiences.

Today, TowneBank operates 51 banking offices throughout Hampton Roads, Central Virginia, and Northeastern and Central North Carolina. As a practical expression of our mission, TowneBank is intentional about being a local leader in each community—actively promoting social, cultural, and economic growth.

TowneBank works with each member toward your ultimate success. A key to this is experienced local bankers providing high-level expertise and personal attention, empowered to make local decisions.

TowneBank's total assets and deposits were valued at $17.25 billion as of December 31, 2024. This represents an increase of 2.45% compared to $16.84 billion reported at the end of 2023. The recent acquisitions of Village Bank in the Richmond area and Old Point National Bank [2] out of Hampton VA is expected to further bolster TowneBank’s total assets to $20 billion.

Before working at TowneBank, I worked at a local non-profit for the majority of my career. After 14 years of working there, I decided it was time to pursue other challenges and opportunities, and I decided to apply for the Senior Systems Engineer position. I had always kept my eye for job openings at TowneBank as I had heard nothing but good things about the team and the company and this time, I happened to notice the job opening through a LinkedIn post. I think it was a week after I applied for the job that I received a screening call from TowneBank’s HR department. I had thought the call had gone well and I was expecting a call back from HR to schedule an appointment with the hiring manager, but after two of not hearing back from HR, I decided to reach out and see if there were any updates. Still crickets, at this point I had figured that they went in a different direction, and I was out consideration for the job, The next Monday, I had traveled to San Francisco to attend VMworld and this is when I got a call back from HR and they wanted to schedule a meeting with the hiring manager for Wednesday. I thought about declining the meeting since I was at a conference and I was under the impression that they had gone in a different direction, but I decided to go for it. The problem is that they scheduled the call for 8 AM EST, but I was on the west coast, so it was 5 AM for me. I figured I wouldn’t lose anything and waking up early for an interview wasn’t such a bad thing. The call with the hiring manager went extremely well to the point that he scheduled me for a follow-up meeting on Friday, while I was still on the west coast. The third interview went well too, and I received an offer within the hour. After a week of salary and benefits, and starting date negotiations, I agreed to the job offer and submitted my 30-day resignation notice. Almost 3 years later, I am still at TowneBank and enjoying every minute of it. Working at TowneBank has been a very rewarding career both personally and professionally.

In terms of the internship, there was not any fanfare about it nor any type of introduction since I was a current employee. I most seemingly transitioned into the role by splitting my regular working hours and the cybersecurity internship, which sometimes blended with my normal duties, so it worked out.

# TowneBank Management

As a former U.S. Marine, I deeply understand and cherish the importance of being surrounded by strong, capable, confident leaders. One of the primary reasons I made the difficult decision to leave my previous job after fourteen years was due to a shift in leadership where I felt like the new management team was no longer aligned with my personal values or professional standards. Leaders and managers in all organizations have a direct impact on morale, productivity, and the overall work environment, and when that alignment is lost, it's difficult to thrive or care about your job.

Here are the core qualities I look for in a leader:

- **Leading by Example:** This is the most important quality I look for. I want my leaders to set the tone through their actions and not just their words. It's important for me that my leader's behavior models the expectations of their subordinated.
- **Clear Communication:** Effective leaders must be able to communicate their vision and objectives with clarity. They should keep their team informed of any changes. Poor communication leads to confusion and sets individuals and teams up for failure.
- **Trust and Empowerment:** I strongly dislike micromanagement. A great leader gives direction on what needs to be done but trusts their team to determine how to do it. Empowering team members shows confidence in their abilities and encourages growth and creativity.
- **Accountability and Recognition:** Leaders should hold their people accountable for their actions while also giving credit where it's due. Recognition should be fair, consistent, and unbiased.

Fortunately, since joining TowneBank, I have had the privilege of working with an outstanding leadership team. They have gone beyond my personal and professional expectations. I truly believe that I am supported by a management team that places trust in my abilities and skills, provide guidance and feedback when needed, challenges me to grow as an engineer and architect, and ensures I have the resources required to succeed and thrive. It's refreshing and motivating to work in an environment where strong leadership is the working standard.

# Duties and Responsibilities

within the scope of cybersecurity internship, most of my duties and responsibilities were around developing IAM policies and procedures and securing our cloud environment.

Due to confidentiality requirements, I cannot discuss any of the cloud hardening procedures or tasks, but I can discuss the SSO responsibilities.

**Single Sign-On**

I believe a significant portion of my cybersecurity internship was dedicated to working with Single Sign-On (SSO). The scope of the SSO involvement spanned from developing procedures, implementing new SSO integrations, troubleshooting authentication issues, and maintaining the entire lifecycle of SSO-enabled applications.

Single Sign-On, or SSO, is a critical framework for ensuring secure and efficient access within any organization of any size. It enables centralized control over user authentication and authorization to both cloud-based and on-prem applications and resources for end-users, SSO offers a seamless experience by reducing or even eliminating the need to repeatedly type usernames and passwords for different applications. This not only enhances productivity but also minimizes the risk associated with password fatigue and decreases the number of support tickets related to password resets. SSO allows users to access multiple systems with a single set of credentials—typically their corporate login—which makes it easier for security teams to enforce company-wide policies such as multi-factor authentication (MFA), session control, and access auditing. In essence, SSO strengthens an organization's security posture while significantly improving user convenience and operational efficiency.

In one instance, we identified a third-party cloud application with over 700 users and found out that we were receiving password reset or account lockout tickets daily so by moving this application to SSO, we were able to eliminate this unnecessary burden on the service desk.

Some of my other duties included performing security and hardening assessments utilizing some of the tools provided by Azure. Several of these tools are included in your cloud subscription and require minimal effort, all while providing a plethora of information and feedback on how to protect your cloud environment against malicious threats. The two assessments I performed during my internship were the Microsoft Azure well-architected review and the Azure Landing Zone review. The Microsoft Azure well-architected assessment is based on the Microsoft Azure well-architected framework, which focuses on five key pillars for success: Cost optimization, Operational Excellence, Performance efficiency, Reliability, and Security. The assessment is a self-guided questionnaire based on the best practices and industry standards of the aforementioned pillars. The questionnaire does require some operating knowledge of the cloud environment and some read-only access, but it took me less than 30 minutes to complete. Once all the questions were answered, it created a baseline of the environment and provided a series of recommendations of things to fix based on criticality and ratings.

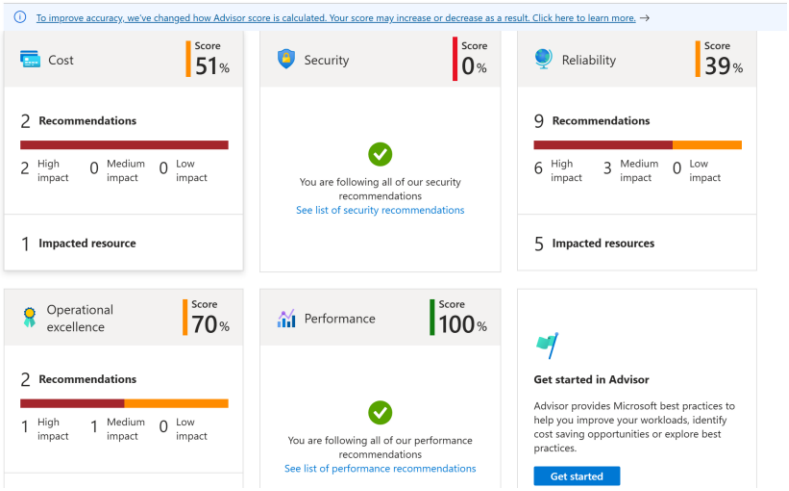



*Figure 1- Azure advisor report.*

# Cybersecurity Knowledge and Skills

I started working in the Information Technology field in 2008 after leaving active duty. Even before that, I always had an affinity for computer systems and technology. I got my first computer when I was 11years old and it was a black-and-white Windows 3.1 system, and I remember reading all the manuals to figure out how everything worked. I even tried to reverse engineer some of the components and “attempted” to hack the operating system… shh, we do not talk about that anymore.

I have been very fortunate in my IT career, as I've been exposed to various areas and technologies within the field, which has allowed me to gain a significant amount of knowledge and experience that has helped in building a successful career. My first real job after the military was as a cabling technician handling copper and fiber cabling. It was a great starting point in my IT journey because it introduced me to one of the most critical layers of any environment, the physical layer.

After about six months of running cables through ceilings and under data center floors, I told myself that, while I loved the job, I didn’t want to do it forever. So, I started looking for ways to upskill my career. I enrolled in school to pursue a degree in computer science, which was bust and also began reaching out to senior peers to learn new skills, which allowed me to learn about Cisco switches and a little bit of routing.

In late 2009, while serving in the Marine Corps Reserves, I received orders to deploy to Iraq, which forced me to pause my IT career for a little bit. Once I fulfilled my military duties in 2011, I picked up right where I left off, but this time, in a new role. I began working as a Network Operations Technician or NOC, which gave me exposure to almost every vertical within the IT department in our organization. I managed backups, antivirus solutions, server patching and upgrades, system monitoring, and even physical access control for data centers and IT closets, one of our duties as NOC technicians was to challenge anybody trying to access the Data Centers so we kept a couple nerf guns to show people we meant business, halacha good times.

Working in the NOC was a very rewarding experience, as it gave me the chance to wear multiple hats and learn about a wide range of technologies. Around 2013, a position opened up for a Systems Administrator. I decided to throw my name in the hat. The role involved supporting the web farm in the DMZ and required knowledge I didn't yet have. However, with a proven track record and a winning attitude, I landed the job and moved onto to the next phase of my career.

Due to the environment being in the DMZ, this role pushed me to learn about load balancing protocols, hardening Windows OS components, network routing, DMZ networks, firewall configurations, IIS setup, VMware, and even some Linux since we supported a small but critical website running on Red Hat with a MySQL backend. I learned a lot in this role and had the opportunity to further sharpen my skills and gain invaluable experience.

Fast forward to late 2014, and my mentor decided it was time for him to move on to greener pastures, so his position became available. I decided to once again take a leap of faith and apply for his role. This role was more focused on virtualization, storage platforms, e-mail, and the early beginnings of cloud computing. I ended up moving up to the role of senior systems engineer and right away, I was tasked with planning a migration from Exchange on-prem to Office365. I remember thinking I had made it to the big leagues because my office had a window. The next year or so I would spend my time learning more about offic365 and this new thing called Azure.

I believe it was around 2015 when the concept of cloud computing started becoming more mainstream, and the industry began experiencing a shift, not just in technologies, but in people as well. The rise of cloud computing

led to the convergence of traditional IT roles into a new role: the cloud engineer/architect. This new breed of IT professionals were now expected to take on a cross-functional role, requiring operational knowledge in storage, virtualization, networking, automation, scripting, and most importantly security. The days of being just a "network" or "storage" engineer were starting to phase out. I remember my manager telling me that if I wanted to stay relevant in IT, I need to have a well-rounded experience in areas like security and automation. I listened to her advice and started to expand my skills into PowerShell, Python and Cybersecurity.

Cloud technologies effectively, and unintentionally, extended the boundaries of the private data centers into the public cloud. The rapid adoption of IaaS and PaaS meant that developers, project managers, and IT engineers could easily deploy cloud resources by swiping a credit card often without proper governance or oversight.

n the very early days of using Azure, I remember one time I was reviewing some logs and noticed some RDP/SQL traffic to an unknown virtual machine. After digging into it, I discovered a developer had deployed a VM in their personal Azure subscription and installed our enterprise VPN client on it to create a pseudo IPSEC tunnel to query a SQL database hosted in our data center, but he had also opened RDP to it so he could connect to the VM in the cloud from his home and then office. I was not even mad; I was impressed he figured it out. But we did end up shutting it down due to the security risk, and we implemented controls to prevent it from happening again.

There were several incidents like that, and even today, we still hear about small and large organizations accidentally exposing cloud resources to the internet due to a misconfigured firewall rule or missing authentication protocols.

The dynamic and constantly evolving nature of the cloud means that security can no longer be the responsibility of a single team. Everyone in IT must adopt a security-first mindset and embrace fundamental cybersecurity principles like zero-trust access, role-based access control, and the principle of least privilege.

## Preparing for the Internship

As you can read from my last rambling, I came into this cybersecurity program with extensive experience in IT from working in the industry for many years across multiple platforms, but I think the biggest eye-opener has been how easy systems can be compromised and the importance of patching your systems. As part of the curriculum for my degree, I am taking CYSE 301 Cybersecurity Techniques and Operations this semester and one of the things we did in the class was exploit a system with an unpatched vulnerability. It was astonishing how easy it was to hack a remote system that had not been patched and that gave me a much deeper appreciation about the importance of patching. This does not mean that I never cared about patching, I always understood the importance of it, but having the opportunity to exploit vulnerability gave me a renewed sense of urgency when it comes to patching. I think it was the first week of April when we received a notification about new VMware vulnerabilities, and I quickly shifted my workload to patch those vulnerabilities. I also took that opportunity to develop a framework or standard to patch future vulnerabilities:

- Vulnerabilities with a CVSS score of 7 or higher with known exploit should be patched within two weeks of initial disclosure.
- Vulnerabilities with a CVSS score of 7 or higher without known exploits should be patched within 45 days of initial disclosure.
- Vulnerabilities with a CVSS score of 6.9 or lower should be patched within 180 days of initial disclosure.

# Learning objectives

When I first started the internship, I thought that 150 hours would have been plenty of time to complete the majority of the goals, but we severely underestimated the effort needed to complete some of these tasks. Someone smarter than me once told that whatever hours you estimate for an IT project, multiple that number by 3 and that's a more realistic estimate.

| Objective | Fulfillment rate | Comments |
|---|---|---|
| Implement IAM solution for Azure Cloud | 70% | Configured and migrated five SaaS applications for SSO with minimal user disruption. Two applications are in the staging process. The most important aspect of this is that we developed a standard for configuring SSO and now we are trying more people using the standard |
| Enhance security posture for private and public cloud environments | 50% | Performed several changes and modifications to harden our cloud infrastructure, but some of the remaining work is on old due to change management and compliance approval |
| Develop policies and procedures for securing Azure storage accounts and Key Vaults | 30% | Developed the initial standards and procedures to lock down Azure Storage accounts by implementing Azure private endpoints. Successfully converted the first Storage account from public facing to private facing by deploying a private endpoint. |
| Learn procedures and best practices for managing Cisco Firewalls | 5% | This was the most neglected objective. Didn't much traction with this due to scheduling conflicts. I was able to get credentials to log in to the firewalls to look around and familiarize myself with the console, but did not get to spend much time with the network going over features, design, etc. |

# Motivations

I've always been driven by curiosity and a passion for learning new technologies, so I would say that my biggest motivation for pursuing the internship was the opportunity to explore new areas and expand my technical knowledge. As I mentioned earlier, modern IT engineers and architects need to have a well-rounded understanding of the many facets of IT systems, and this internship was a great way to gain hands-on experience across those areas. Another major motivator for me was graduating. I started school back in 2009, and I am now almost at the finish line, so it is a great feeling to know that I am almost done for now. I think I may take a break after I graduate and then maybe look into pursuing a masters.

# Discouragements

I think the biggest discouragement was the sheer amount of work that needed to be done and how little time there was to get everything taken care of. That being said, I see this as a positive thing, especially for someone preparing to graduate and enter a competitive job market. There are plenty of job opportunities out there for cybersecurity specialists, especially for those willing to put in the effort and a bit of sweat equity.

# Challenges

In all honesty, I did not have any challenges outside the time management issues I mentioned in the previous section. I think being able to incorporate this internship with my regular IT job was a huge benefit as I do not think I would have had the bandwidth to be able to meet the demands of working full, attending school and juggling time with family.

# Recommendations

My recommendation for anybody wanting to pursue an internship is to find a way to stand out, in a good way of course. This is a very competitive market for internships and jobs, so you need to find to stand out, you need to go beyond what's expected of you, you know, go the extra mile.

As I mentioned before, everybody in IT needs to have a well-rounded experience in multiple. Never fall into the trap that because you work in X you do not need to learn about Y.

These are some self-paced free resources I have put together for helping people learn new skills in the cloud and gain that extra edge.

**Microsoft 365 Virtual Training Day: Fundamentals**

https://msevents.microsoft.com/event?id=2377277003&wt.mc_id=eventscatalog

**Microsoft Azure Virtual Training Day: Fundamentals**

https://msevents.microsoft.com/event?id=1862411159&wt.mc_id=eventscatalog

**Microsoft Security Virtual Training Day: Security, Compliance, and Identity Fundamentals**

https://mktoevents.com/Microsoft+Event/461764/157-GQE-382?wt.mc_id=eventscatalog

**Azure self-paced training**

**Azure fundamentals**

https://learn.microsoft.com/en-us/training/courses/az-900t00#course-syllabus

**Microsoft 365 Fundamentals**

https://learn.microsoft.com/en-us/training/courses/ms-900t01

**Microsoft Security, Compliance, and Identity Fundamentals**

https://learn.microsoft.com/en-us/training/courses/sc-900t00

**AWS**
https://explore.skillbuilder.aws/learn/course/external/view/elearning/134/aws-cloud-practitioner-essentials

**Cybersecurity training**
https://www.isc2.org/Certifications/CC?filter=featured&searchRoot=A82B5ABE5FF04271998AE8A4B5D7DEFD

In addition to completing some of those training modules, these are some of the non-technical suggestions.

- **Be hungry:** Hungry for knowledge and learning new skills and concepts. Always be on the lookout for new responsibilities and opportunities.
- **Be Smart:** know how to deal with people and the intricacies of group dynamics,
- **Be Humble:** Be confident, but not arrogant. Don't like you are the smartest person in the room.
- Use the internship as a networking tool. Take advantage of any social events, anything to meet new people and make ne connections.

# Final conclusions

This internship has been a very rewarding and enlightening experience for me. Being able to put into practical use the cybersecurity concepts I have learned in school has provided immeasurable value from an educational and professional perspective. I have been in IT for over a decade and my exposure to cybersecurity has been limited, but this internship afforded me the opportunity to experience the day-to-day joys and challenges of working in cyber.

# Citations

**[1] TowneBank's website. (n.d.). *About Towne*. TowneBank. https://www.townebank.com/about-towne/**

**[2] https://www.globenewswire.com/news-release/2025/04/03/3054985/10357/en/TowneBank-and-Old-Point-Financial-Corporation-Announce-Agreement-to-Merge.html**

# Appendices

## Appendix A – Single Sign-On implementation procedure

Azure SSO: Configure Single Sign-On Applications through Azure Active Directory

Necessity Assessment

Requests for Single Sign-on implementations should always be accompanied by Jira a request or project that includes the necessary information to complete the SSO implementation. The required information to complete the SSO implementation is listed in the technical requirements section of this document. The steps outlined in this document are only for SAML 2.0 compatible SaaS applications, additional SSO protocols like WS-FED, or OpenID are not supported.

Pre-requisites

Technical requirements

Azure Active Directory Subscription with a P1 entitlement

At least one of the Azure AD roles needs to be assigned to the user configuring the SSO integration.

Application administrator

Cloud administrator application

Global Administrator

Basic understanding of SAML 2.0 implementations and terminology. See the additional information section below.

If configuring SSO integration with an external SaaS application, you will need to have a technical point of contact with the vendor to configure and test the SSO integration on their side. The vendor or technical contact will also need to provide a SAML metadata file and/or the following information about their SSO implementation.

Identifier (entity ID)

Reply URL (ACS URL)

Sing on URL (this is optional and only necessary if SP initiated SSO is supported)

Relay state (Optional)

Logout URL (Optional)

Required SAML claims and format.

Azure AD supports automatic user provisioning for SaaS applications using SCIM. If the SaaS Application supports SCIM user provisioning, the vendor or technical contact will need to provide the following information:

SCIM Token

SCIM URL

Active Directory Groups

We should always use Active Directory groups to grant users access to the SSO resource. When creating an Active Directory group to access the SSO resource, use the following format:

| **Group Name** | **Description** | **Members** |
|---|---|---|
| *"Name of the SaaS Application"* User SSO Access Management. | Grants Members SSO access to *"Name of the SaaS Application"* SaaS through the Corporate SSO Portal | Add Users as necessary. **Do not use nested groups to add users.** |

SSL Certificates

A self-signed certificate will be generated by Azure AD to sign and decrypt SAML tokens. These certificates have a maximum lifespan of 3 years and need to be renewed prior to their expiration. Refer to this document for guidance on renewing these SSL certificates Azure SSO : Update SAML signing Certificates for Azure AD Single Sign-on applications

Change Plan Steps

Azure has published a collection of over 3,000 SaaS Applications that been pre-configured and pre-validated to work with Azure AD for SSO, see Azure AD Marketplace https://azuremarketplace.microsoft.com/en-us/marketplace/apps/category/azure-active-directory-apps. If a SaaS application is not listed in the marketplace, we still have the option to add a non-marketplace application to our Azure AD environment. Follow the steps below to add a gallery and/or a non-gallery Application to Azure AD

Add Gallery application to Azure AD.

Sign in to the Azure portal using your corporate account.

On the left navigation pane, select the **Azure Active Directory** service.

Navigate to **Enterprise Applications** and then select **All Applications**.

To add new application, select **new application**.

In the Browse Azure AD gallery section, you can browse the gallery for the application you want to add, or search for the application by entering its name in the search box, then select the application from the results. If the application is not available in the marketplace, proceed to the next section to add a non-gallery application.

Review the details of the application and ensure it supports SAML-Based SSO, and the application is using a simple, but recognizable name; Application names are limited to 93 characters. If SAML SSO is not supported, do not add the gallery application and instead proceed to adding a non-gallery application in the next section.

Click create to add the gallery application.

Wait for the application to be added to the Azure AD tenant and then proceed to the “Configure user sign-in properties for Enterprise Application” section below.

Add non-gallery application to Azure AD.

Sign in to the Azure portal using your corporate account.

On the left navigation pane, select the **Azure Active Directory** service.

Navigate to **Enterprise Applications** and then select **All Applications**.

To add a new application, Click Create your own application.

In the Create your own application section, type the name of the application you want to add. The name of the application should be simple, but recognizable. The name can be changed post-deployment.

Select the “integrate any other application you don’t find in the gallery” option.

Wait for the application to be added to the Azure AD tenant and then proceed to the “Configure user sign-in properties for Enterprise Application” section below.

Configure user sign-in properties for Enterprise Application

Configuring the user sign-in experience for Enterprise application applies to both gallery and non-gallery applications.

In the Azure AD portal, select Enterprise applications. Then find and select the application you want to configure.

In the Manage section, select Properties to open the Properties pane for editing.

Set the following options to determine how users who are assigned to the application can sign in.

**Enabled for users to sign-in?** Yes (if set to not, users will not be able to use the SSO integration with this SaaS App, even if they are assigned to it.)

**Name:** This is the name of the application the users will see on their Myapps portal. This can be changed at anytime.

**Logo:** This is the application logo that users see on the Access Panel, in the Office 365 application launcher, and when admins view this application in the application gallery. Custom logos must be exactly 215x215 px in size and be in the PNG format. This logo can be changed at anytime after deployment.

**User Assignment required: Yes** (If this option is set to no, then any users who navigate to the MyApps portal or application URL directly will be granted access)

**Visible to users: Yes** (if set to no, users will not see the application on their Myapps portal. They will still be able to use it by navigating to a direct URL or by performing Service Provider(SP) initiated SSO

**Notes :** Use this field to capture the change ticket number associated with deploying this enterprise application

When you're finished, select Save.

Configure SAML-based single sign-on for gallery applications

Azure AD gallery applications will be deployed using the necessary SAML settings. However, it is recommended to confirm these settings with the vendor as the requirements may have changed since the application was added to the gallery.

Sign in to the Azure portal as a cloud application admin, or an application admin for your Azure AD tenant.

Navigate to Azure Active Directory > Enterprise applications and search for the application

Under the **Manage** section, select **Single sign-on** and confirm the pre-configured SAML settings per the vendor's requirements.

Configure SAML-based single sign-on for non-gallery applications

Sign in to the Azure portal as a cloud application admin, or an application admin for your Azure AD tenant.

Navigate to Azure Active Directory > Enterprise applications and search for the application

Under the **Manage** section, select **Single sign-on**.

Select **SAML**. The **Set up Single Sign-On with SAML - Preview** page appears.

If the vendor provided a SAML metadata file, upload the file by clicking on the "upload metadata file" option. if no metadata file has been provided, proceed to step 6-7 to edit the basic SAML configuration option.

Select the Edit icon (a pencil) in the upper-right corner of the Basic SAML Configuration section.

Configure the following settings with the information provided by the application vendor

| **Configuration setting** | **Value** | **Comment** |
|---|---|---|
| **Identifier (Entity ID)** | Provided by the vendor | Uniquely identifies the application. Azure AD sends the identifier to the application as the Audience parameter of the SAML token. The application is expected to validate it.<br>This value also appears as the Entity ID in any SAML metadata provided by the application. You can find this value as the Issuer element in the AuthnRequest (SAML request) sent by the application. |
| **Reply URL** | Provided by the vendor | Specifies where the application expects to receive the SAML token. |
| **Sign-on URL** | Provided by the vendor (Optional) | This URL contains the sign-in page for this application that will perform the service provider-initiated single sign-on. Leave it blank if you want to perform identity provider initiated single sign-on. |
| **Relay State** | Provided by the vendor (Optional) | A SAML RelayState parameter can be provided. The RelayState instructs the application where to redirect users after authentication is completed, and the value is typically a URL or URL path that takes users to a specific location within the application. |
| **Logout URL** | Provided by the vendor (Optional) | This URL is used to send the SAML Logout response back to the application. |

Click Save to complete the basic SAML configuration

Configure SAML User attributes and claims

When a user authenticates to the application, Azure AD issues the application a SAML token with information (or claims) about the user that uniquely identifies them. By default, this information includes the user's username, email address, first name, and last name. Some applications may require specific claim values and/or name formats and those requirements should be provided by the application vendor.

In the User Attributes and Claims section, select the Edit icon (a pencil) in the upper-right corner.

Verify the **Unique User Identifier (Name ID)** attributes. Unless the vendor requests something else, the format should be set to "Email address" and the source attribute should be set to user.mail

Add the following claims by clicking "Add new claim"

| **Name** | **namespace** | **source** | **source attribute** |
|---|---|---|---|
| email | blank | attribute | user.mail |

Click save. The new claim appears in the table

Additional claims can be added as requested from the vendor.

Configure SAML signing certificate

Azure AD uses a certificate to sign the SAML tokens it sends to the application. You need this certificate to set up the trust between Azure AD and the application.

Go to the SAML Signing Certificate section.

Select the Edit icon (a pencil) in the upper-right corner of the SAML Signing Certificate section.

Verify that the signing option is set to Sign SAML assertion and the signing algorithm is set to SHA-256

Change the notification email address to **confidential**

click save to complete the signing certificate configuration

Complete SSO configuration

The Set up <applicationName> section lists the values that need to be configured in the application so it will use Azure AD as a SAML identity provider. The required values vary according to the application

Scroll down to the Set up <applicationName> section.

Copy the value from each row in this section and provide them to the application vendor to configure the SSO on their side. These values are unique for each application. You may also provide them with the App Federation Metadata Url located in the SAML signing certificate section
Login URL :
Azure AD Identifier:
Logout URL:

Click Save to complete the SSO configuration

Add users to the application

In the left navigation menu, select Users and groups

Select the Add user button.

On the Add Assignment pane, select Users and groups.

Select the user or group you want to assign to the application, or start typing the name of the AD group created in the previous section. Individual user accounts maybe be added for B2B users or online accounts online.

Testing and Troubleshooting Plan

Log in to https://myapps.microsoft.com

Click on the tile of the newly created SSO application

Confirm that you are able to log in to SaaS application.

If unable to log in, Install the Chrome Extension “SAML-Tracer” to troubleshoot the SAML messages between Azure AD and the SaaS application

Configure SCIM provisioning (Optional)

Sign in to the Azure portal and select Enterprise Applications, select All applications, then select the application that needs to be configured

Select the Provisioning tab.

Set the Provisioning Mode to Automatic.

Under the Admin Credentials section, input the Tenant URL and Secret Token provided by the vendor. Click Test Connection to ensure Azure AD can connect to the SaaS application. If the connection fails, ensure the account has Admin permissions and try again

In the Notification Email field, enter the email address ssoadmin@confidential - Send an email notification when a failure occurs.

Click Save

Under the Mappings section, select Synchronize Azure Active Directory Users to SaaS application

Review the user attributes that are synchronized from Azure AD to the Saas application in the Attribute Mapping section. Select the Save button to commit any changes.

Under the Mappings section, select Synchronize Azure Active Directory Groups to SaaS application.

Review the group attributes that are synchronized from Azure AD to SaaS application in the Attribute Mapping section. The attributes selected as Matching properties are used to match the groups in SaaS application for update operations. Select the Save button to commit any changes.

To enable the Azure AD provisioning service, change the Provisioning Status to On in the Settings section.

Define the users and/or groups that you would like to provision to the SaaS application by choosing the desired values in Scope in the Settings section

When you are ready to provision, click Save.

Once you've configured provisioning, use the following resources to monitor your deployment:

Use the provisioning logs to determine which users have been provisioned successfully or unsuccessfully

Check the progress bar to see the status of the provisioning cycle and how close it is to completion

Additional Considerations

If the saaS application does not support SCIM or automatic user provisioning, the application owner will need to create the user accounts or enable automatic provisioning in the SaaS application before the SSO users can log in.

Create a local user account in the SaaS application. This should only be used as backdoor entrance in case Azure AD SSO services are unavailable.

Additional information

**Common Terminology**

Assertion Consumer Service (ACS) : The service provider's endpoint (URL) that is responsible for receiving and parsing a SAML assertion

Identity Provider (IdP) : The authority that verifies and asserts a user's identity and access to a requested resource. In our environment, Azure AD is our IdP

IdP initiated SSO : This refers to users being able to use a portal within the IdP to log in to an applicat

Claims

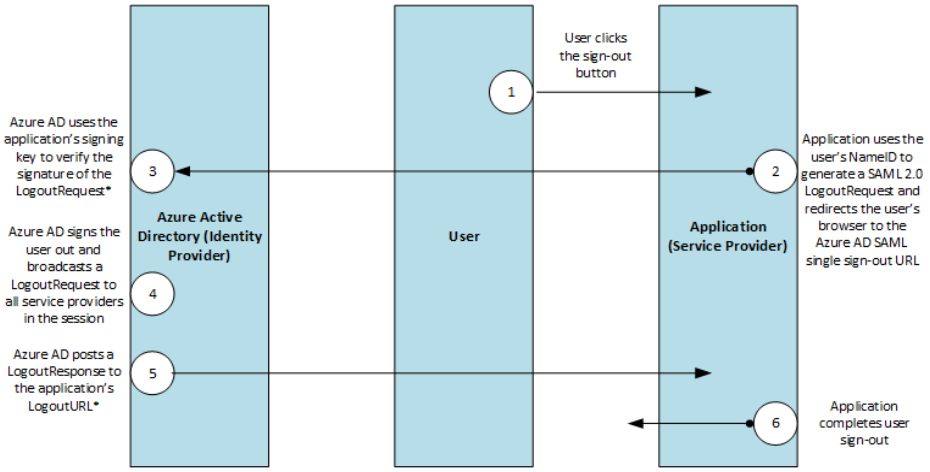
Metadata : A set of information or parameters supplied by the IdP to the SP, and/or vice versa, in xml format

NameID : An attribute within the assertion that is used to specify the username

SAML : Acronym for Security Assertion Markup Language. SAML is an identity federation standard that enables single sign-on. It is an XML-based standard for exchanging authentication and authorization data between a Service Provider (providing a service to the user) and an Identity Provider (providing user identity verification for the Service Provider).

SAML Assertion : Data provided by the IdP that supplies information about the SSO user to the service provider.

Service Provider (SP) : The hosted resource or service that the user intends to access, such as Box, Workday, Salesforce, a custom application, etc




* Azure AD gets the signing key and LogoutURL of the application from the application metadata

Configuration Items

CIs being changed

Azure AD

Risk Assessment

Misconfigured SSO settings may prevent users for logging in to the application

validate the SSO settings in accordance with the vendor requirements.

Backout Plan

Remove the Azure AD enterprise application

Change Requirements

NOTE: This process will not automatically generate tickets. Once your schedule is approved, highlight the steps below to create tickets for the TOC for these pre-change requirements

Alerts need to be held for the identified CIs

Server snapshots required

SQL Database backup required

Communication Plan

| | |
|---|---|
| To: | <SaaS application owner> |
| Pre Change Message | The purpose of this message is to inform you about the upcoming change to integrate the <name of the application> application with our Azure Active Directory for Single Sign-on. We do not expect any issues with this implementation, but we expect the logon experience to change after the SSO has been implemented. <application name> users will need to access the application through the SSO portal located here https://myapps.microsoft.com |
| Post Change Message | The purpose of this message is to inform you that the SSO implementation for <name of the application> has been completed. Users will now need to access the “name of the App” by using the Corporate SSO portal located here https://myapps.microsoft.com |
| Reschedule Message | The purpose of this message is to inform you that the SSO implementation for <name of the application> has been rescheduled. We will follow up when a new date has been confirmed. |
| Backout Message | The purpose of this message is to inform you that the SSO implementation for <name of the application> has been backed out due to unforeseen issues. |

## Appendix B – Azure Privileged Identity management presentation


# Azure Privileged Identity Management (PIM)

## ENGINEERING TEAM

Diego Izquierdo– Senior Systems Engineer

03/12/2025

## What is Azure Privileged Identity Management(PIM) ?

Azure PIM is a feature of EntraID that provides just-in-time access to privileged roles and groups.

## Why are we doing Azure PIM?

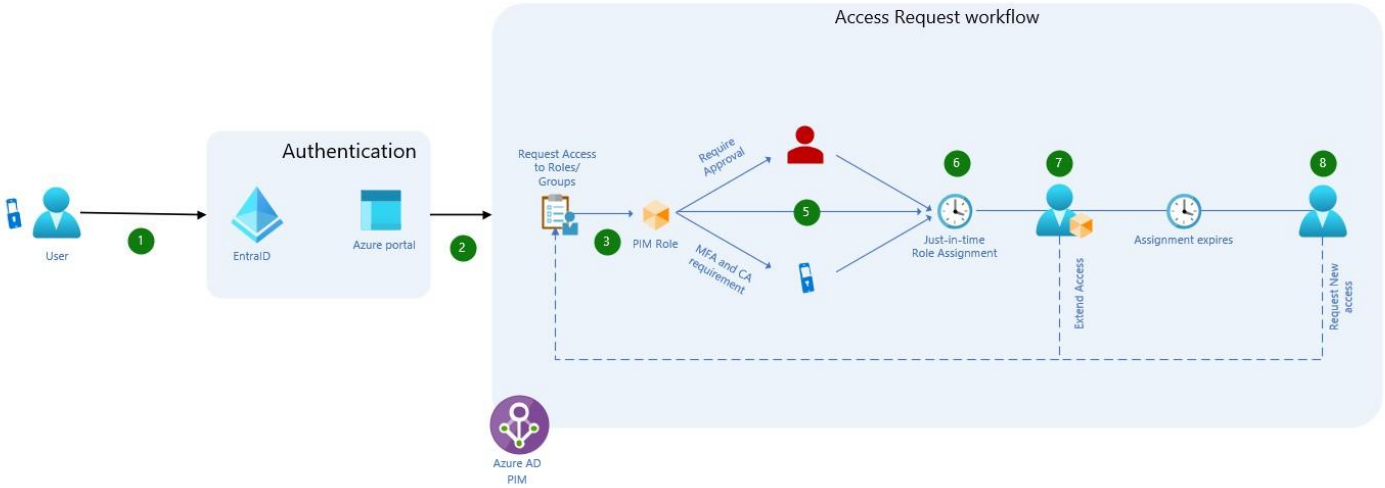
To reduce our risk exposure by minimizing the amount of users with privileged roles.

# What is Azure PIM?

# Objectives

- Per Microsoft's best practices, we will aim to reduce the number of global administrators from 10 users to 5
- Designate 2 break glass accounts as permanent global administrators
- Reduce the number of users with privileged roles.
- Review current role assignments and use principle o least privilege access.


Pimp
My Account
PIMPED


Access Request workflow
Authentication
User
EntraID
Azure portal
Request Access to Roles/ Groups
PIM Role
Require Approval
MFA and CA requirement
Just-in-time Role Assignment
Assignment expires
Extend Access
Request New access
Azure AD PIM

- Requester cannot approve own requests
- We can use users or groups as designated approvers.
- It can only be used for cloud groups
- Can assignments be extended? Yes

# Questions?