ITProMentor

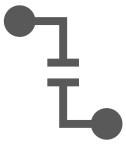
# The Microsoft 365 Security Essentials Guide

By: Alex Fields

Updated: January 2022

Verify Explicitly

Least Privilege



Assume Breach

## The Security Essentials

## Change log

**3rd Edition**: Major updates to simplify the checklist according to Microsoft's three pillars of Zero Trust, with a bonus section at the end to describe what lays beyond this starting point!

**2nd Edition**: In the second edition I rearranged the first edition's checklist of security measures and mapped them against an attack kill chain, with several items pre-breach and post-breach.

**1st Edition**: The first edition was basically just a list which came straight from Microsoft's Secure Score tool. The purpose of this first edition was to provide easy walk-throughs for the "top" recommendations from that tool.

## About the author

My name is Alex Fields; I have been both a small business employee and a small business owner for literally all of my adult life, and have been serving the SMB community as a technology adviser for over 15 years.

I started ITProMentor.com back in 2016 to document everything I had been learning about migrating and transforming customers from legacy on-premises Windows Server technology to more modern, cloud-based services such as Microsoft Office 365 and Azure Active Directory.

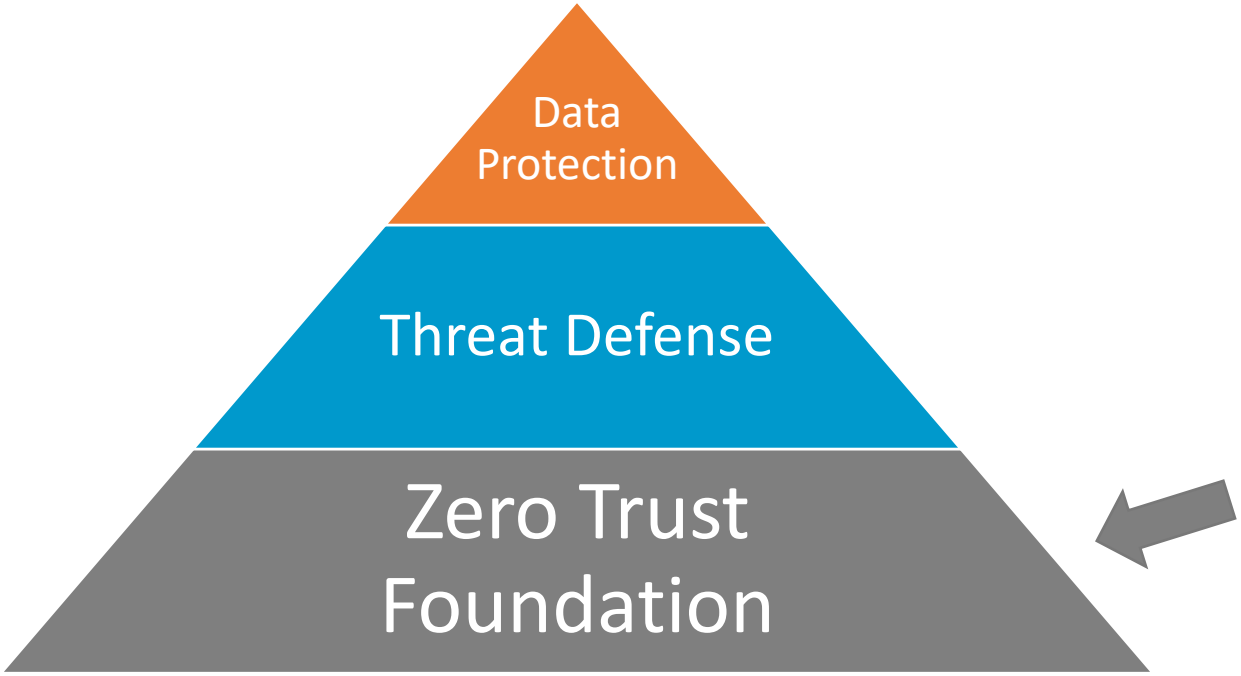
Back then, I wasn't entirely sure if anyone was ever going to find my writings, let alone find them *useful*. But at least they would be useful to me and my team! After all, how else were we supposed to keep track of all the fast-paced changes in the cloud? Now fast forward: today my blog, courses, and eBooks reach thousands of IT consultants in the SMB space all over the world. We even have a community called Square One where folks like us get together to share knowledge and experiences, and swap stories.

ITProMentor

# Table of Contents

## What are the Security Essentials?

Microsoft recommends a comprehensive approach to security, starting with a strong foundation based on Zero Trust:



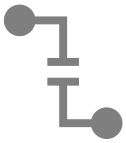

Once we have the **Zero Trust Foundation** established, we can proceed to layer additional **Threat Defense** and **Data Protection** measures on top. However, note that some of these later items may require higher-end subscription bundles; we will not cover all of the various security tools that are available in the Microsoft universe within the scope of this guide, but we will mention the essential steps to establishing your Zero Trust Foundation, regardless of subscription level.

Before we get into the detailed implementation steps, let's take a step back and describe the above framework in greater detail. Starting with the question: what do we mean by Zero Trust?

Microsoft describes Zero Trust according to three pillars: **Verify Explicitly**, **Least Privilege**, and **Assume Breach**. The core assumption with all of these is that we must not take trust for granted. For example, we extend trust only once we have met certain criteria that give us a higher level of confidence in the authenticity and integrity of our access requests.

Verify Explicitly

Least Privilege



Assume Breach

Security Essentials = Zero Trust Foundation

## Verify Explicitly

We want assurances that our users "are who they say they are." If the only thing standing between an attacker and your organization's data is a username and password, then you do not have very good assurances. There are two things that we should do right away to establish authenticity for logins as well as email messages sent from your organization:

1. Establish strong authentication for end users (i.e., MFA)
2. Implement Email authentication (i.e., SPF, DKIM, and DMARC)

## Least Privilege

When I audit tenants, I regularly find that too many people have full administrator privileges, and worse yet, those permissions often exist on "everyday" accounts which are also used for email and file sharing. This is a big cybersecurity no-no: we must limit admin privileges wherever possible. I always recommend that you:

3. Reduce & separate admin privileges wherever possible
4. Manage application consent

## Assume Breach

At some point, your defenses **will** fail. No one is bulletproof, so you have to work from the assumption that you will one day be breached. When that day comes, you want to be able to detect that the breach has occurred in a timely manner and have a plan in place to respond. Here are two things we can do with almost any subscription to help you prepare:

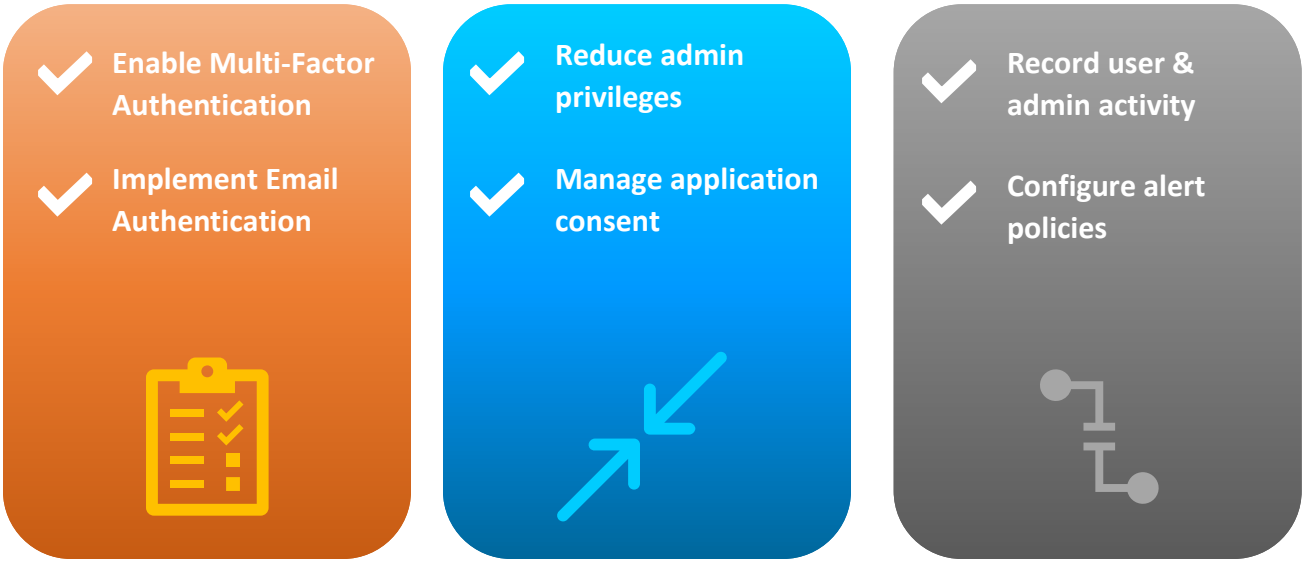
5. Record user and admin activity by turning on the Unified Audit Log
6. Configure Alert policies with a monitored email address

With premium features such as Microsoft Endpoint Manager (Intune) and Azure AD Premium, you can fortify your "Zero Trust Foundation" further via additional safeguards like Device Compliance, Privileged Identity Management, or Risk-based authentication policies. However, note that these items are beyond the scope of this guide.

***Note:*** *You can refer to my licensing page for more detail on what products and features are included within each of the major subscription bundles.*

## The Security Essentials Checklist

Here again is the Security Essentials Checklist based on the Zero Trust pillars:




In the following pages, I will describe how to accomplish these "Security Essentials." I may sometimes refer to PowerShell scripts that I use to configure new tenants quickly on my GitHub repo at: https://github.com/vanvfields/microsoft-365

To connect to services using PowerShell with support for MFA refer to these links:

- Follow these instructions to connect to Exchange Online (v2)
- Follow these instructions to connect to Azure AD (v2)

## ☐ 1. Enable Multi-Factor Authentication (MFA)

Microsoft likes to say: "Your pa$$w0rd doesn't matter, but MFA does." There are three approaches that you can take with regard to Multi-factor Authentication (MFA), which is a highly effective mitigation against common identity-based attacks (like password spray). And if you are implementing the CIS Critical Security Controls (which I recommend you do), this would help you satisfy some important Safeguards including CSC #6.3: Require MFA for Externally-Exposed Applications, and CSC #6.5: Require MFA for Administrative Access.

**First option**: You can use the "**easy button**" and let Microsoft manage security for you with the **Security defaults** feature. Presumably, the fact that you are reading this guide suggests you may not be the target market for this option. Many people find that the defaults do not provide them with enough flexibility (for example, you cannot make exceptions). On the other hand, this feature is available in every single Microsoft subscription and does not require any special or additional licensing.

**Second option**: When you are ready to take command of your own security journey, then you would disable the Security defaults and proceed to create your own **custom Conditional Access security policies**.

> ***Note:*** *Conditional Access requires specific subscription levels, such as Microsoft 365 Business Premium or Enterprise plans, Azure AD Premium plans, or Enterprise Mobility + Security bundles.*

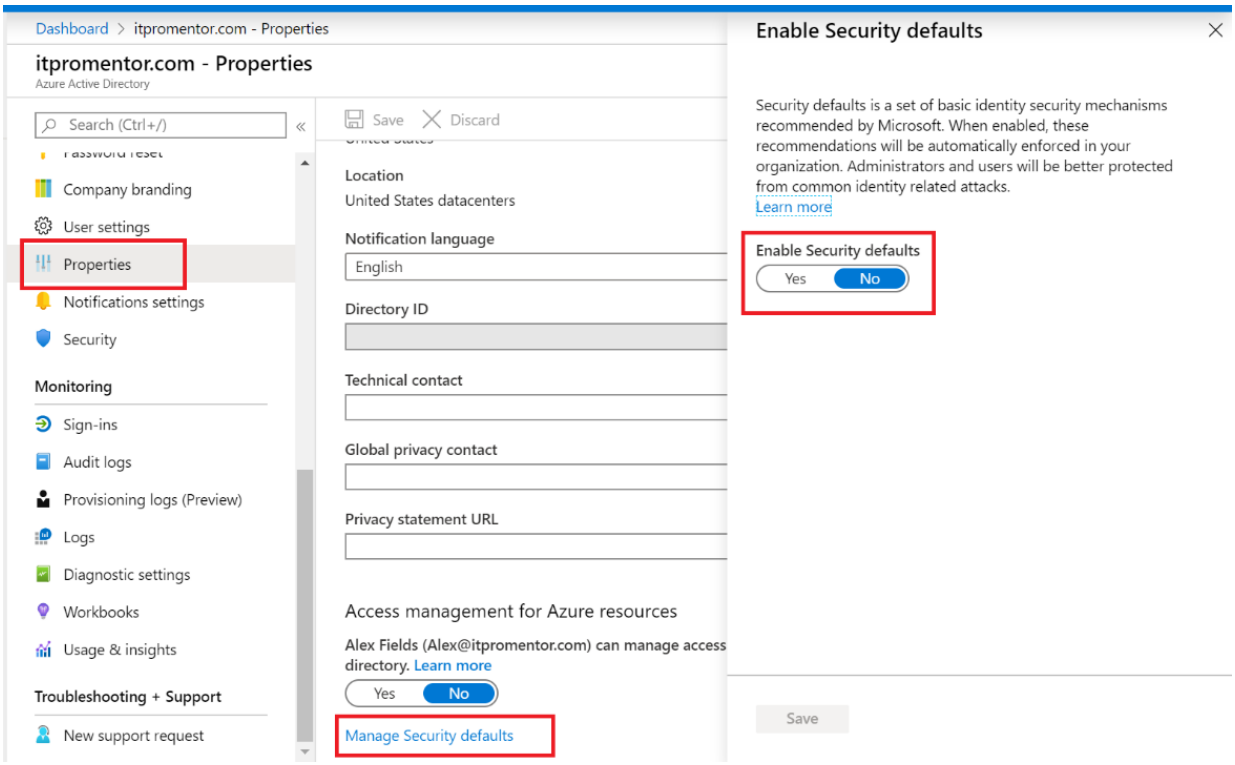
**Third option**: It is also possible to **configure stronger authentication on a per-user basis**, and this may be preferred for those customers who do not have one of the more comprehensive subscriptions but also do not want to use the Security defaults.

Whichever way you decide to get there, it is recommended to announce this change in advance at a staff meeting and also by staff-wide email. Consider flyers or other awareness raising techniques as well. Be sure to include helpful links to Microsoft support resources in your communications, e.g.:

- Set up your Microsoft 365 sign-in for multi-factor authentication
- Register and manage your security information (YouTube)

## Option 1. Use the Security Defaults feature

In the future this will be on by default in new tenants. Currently, you cannot turn this feature on with PowerShell, so you have to use the web portal. In the **Azure AD Admin center**, find the toggle for Security defaults under **Azure AD > Properties > Manage Security defaults**.




The impacts of enabling the Security Defaults are as follows:

- **Block legacy authentication**: All accounts will be prevented from using legacy apps or protocols like SMTP, POP, IMAP, etc.
- **Require MFA for admins**: all admin accounts must use MFA, no exceptions
- **MFA for standard users**: All accounts must register for MFA within 14 days, but will only be prompted or challenged for MFA if it is a “risky” logon attempt. Additionally, accounts with credentials found on the dark web will be required to change passwords
- **Require MFA for service management**: Any account signing into Azure services must use MFA whether admin or standard user

Remember: you cannot make any exceptions to the defaults. This feature must be disabled if you plan to configure your own security policies for greater control and functionality.

## Option 2. Conditional Access policies

This option requires an Azure AD Premium subscription, which you can find bundled in plans like Microsoft 365 Business Premium or Microsoft 365 Enterprise E3/E5. We will not cover the process to implement Conditional Access policies here, however I will point you to the Microsoft Docs articles describing the four policies you would use to replace Security Defaults:

- **Block legacy authentication**
- **Require MFA for Admins**
- **Require MFA for Azure management**
- **Require MFA for All users**

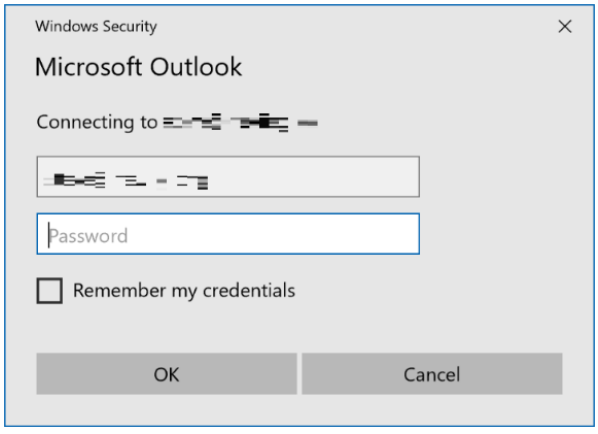
***Note:*** *You should always exclude at least one emergency access account from every conditional access policy you create. See this Microsoft Docs article for more details.*

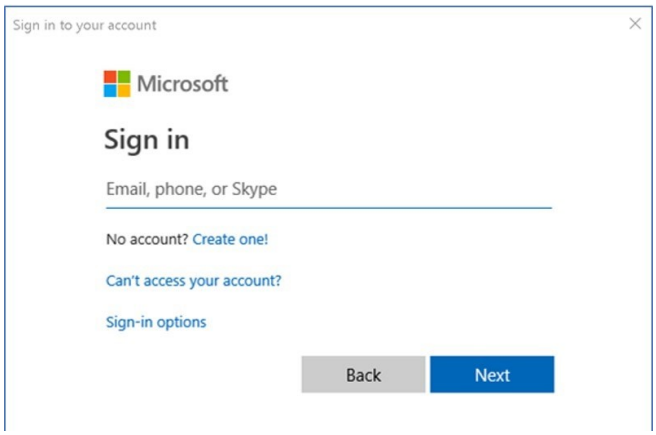
For a more comprehensive overview of Conditional Access, check out my Microsoft 365 Best Practices, which includes all of my guidance on Azure AD and Recommended Conditional Access Policies.

## Option 3. Implement strong authentication on a per-user basis

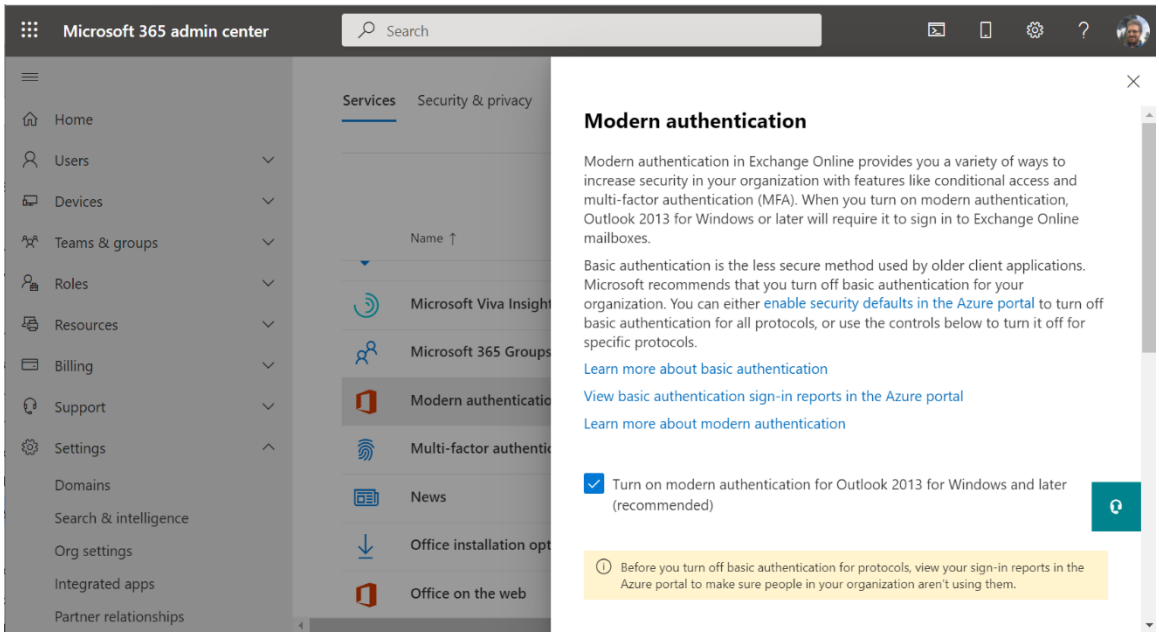
This last option works with all subscription levels. Remember that establishing stronger authentication is really two steps in one. First, you want to **block legacy authentication** protocols that do not support modern authentication and MFA. Second, you want to be sure to **enable MFA** for each and every active user account.

Modern authentication is to be distinguished from legacy (or basic) authentication. Compare prompts for legacy (left) and modern (right) below on Windows client devices:



With Modern authentication clients, credentials are never stored on the device like they are with legacy clients (notice there is no option to "*Remember my credentials*"). This makes it less vulnerable to credential capture and replay attacks that target client devices. Additionally, modern authentication supports MFA prompts, whereas basic does not, and legacy authentication is susceptible to password spray and brute force attempts as well.




New tenants should have Modern authentication enabled by default (and soon legacy authentication will be disabled by default everywhere as well). From the admin center, go to **Settings > Org settings**, and find **Modern authentication** in the list to confirm.

## Block legacy authentication

Legacy clients such as Outlook 2010 are not compatible with modern auth. Even 2013 clients aren't compatible without making a modification to the registry. But my advice is to get off of older versions and install the newer Microsoft 365 Business apps instead from the Office portal.

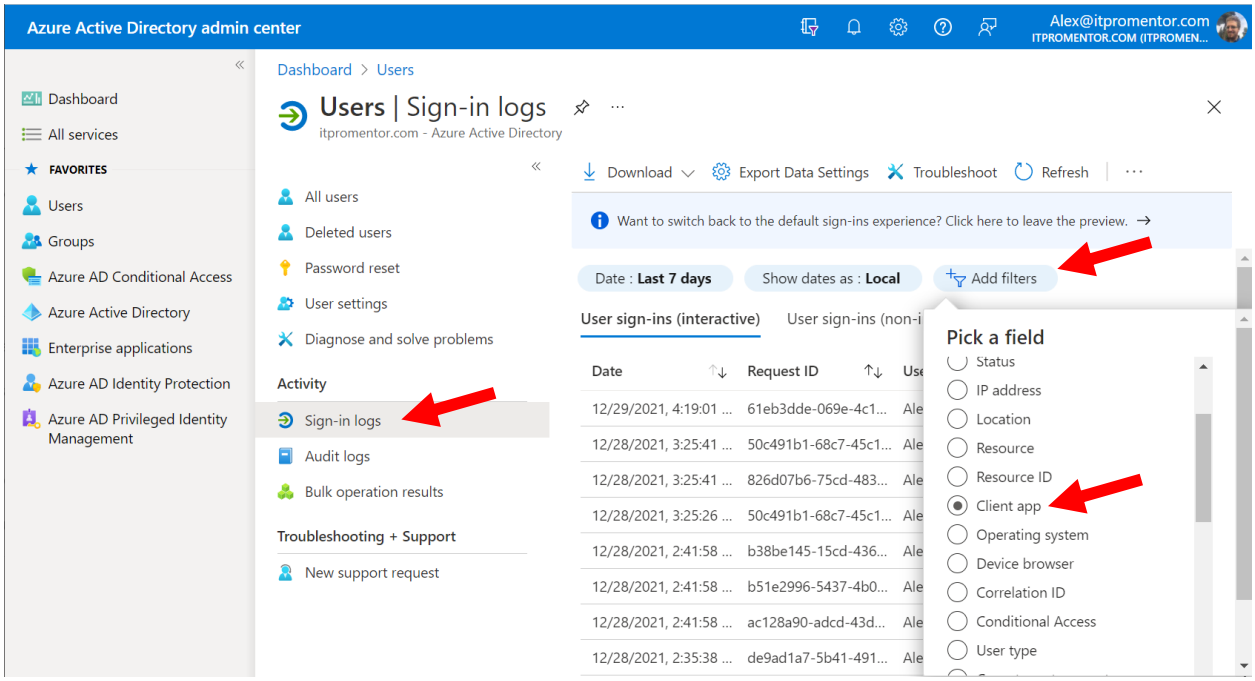
Aside from client apps, you might also find apps or devices (e.g. MFP's that scan to email) which are using legacy authentication protocols. But in October of 2020, many of these protocols will be disabled anyways, so it is best to find alternative means that support modern authentication. Talk to your vendors!

We have several ways of either limiting or completely eliminating the use of legacy auth and protocols (e.g., IMAP and POP) which do not support Modern auth:

1. Disable legacy protocols such as SMTP, POP, IMAP on mailboxes individually

2. Create an Authentication policy that blocks legacy authentication by default (recommended)

Before you proceed, it is a good practice to review the sign-in logs in the Azure AD admin center. Navigate to **Users > Sign-in logs** and then **Add filters** and choose **Client app**.

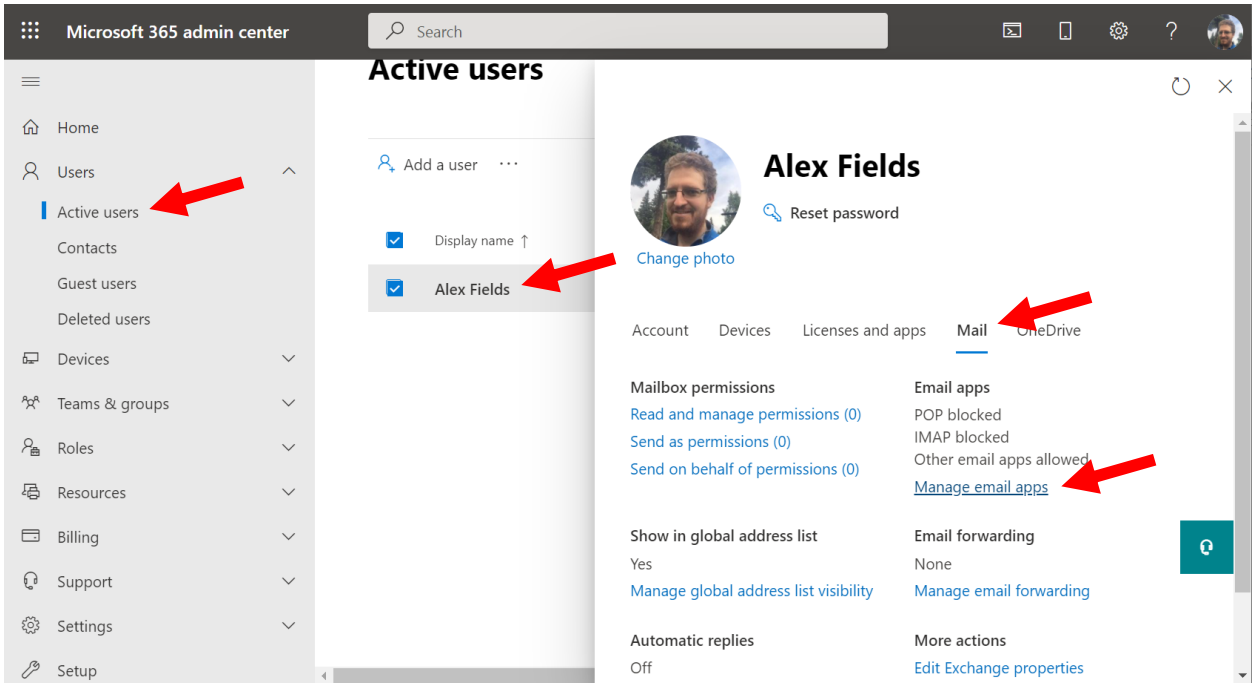



This will allow you to filter for any legacy authentication sign-ins, so you can remediate by upgrading software or making exceptions as needed. Now we need to block legacy auth for Exchange Online and SharePoint Online.

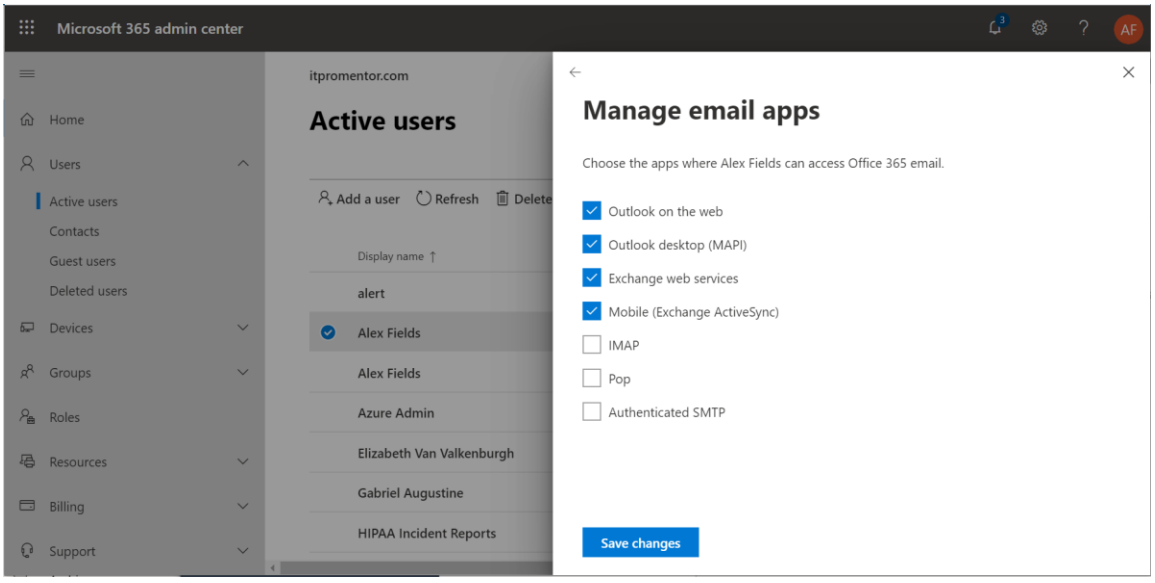
### Exchange Online Option 1: Disable legacy protocols per mailbox

This is an easy, low-risk place to start, and it is accomplished per-mailbox (rather than via policy). You should try to block the use of legacy protocols (such as SMTP, POP and IMAP) wherever possible. Attacks on these protocols are launched against your tenant daily.

From the Microsoft 365 admin center under **Users > Active users**, select a user account. Go to the **Mail** tab and select the option to **Manage email apps**.




From here it is very easy to turn off any legacy protocols that you know are not in use.




**SMTP** is by far the most targeted legacy protocol, followed by **IMAP** and then **POP**, so removing those at a bare minimum is a good idea. **Exchange ActiveSync** (EAS) is also not needed as long as end users move to the new Outlook mobile app. It is also unlikely that users would need **Exchange Web Services** (EWS), however, some applications that

integrate with Exchange Online may still need one or more of these services. This is why it is helpful to refer to the sign-in report in case exceptions need to be made (but the better thing to do is to update your apps to use modern authentication).

In PowerShell you can see all of the mailboxes at once, and the various protocols which are allowed using **Get-CASMailbox**:

```
PS C:\temp> Get-CASMailbox
Creating a new Remote PowerShell session using MFA for implicit remoting of "Get-CASMailbox" command ...

Name                                                  ActiveSyncEnabled OWAEnabled PopEnabled ImapEnabled MapiEnabled
----                                                  ----------------- ---------- ---------- ----------- ----------
EmilyB                                                True              True       True       True        True
ChristieC                                             True              True       True       True        True
Conf Room Rainier                                     True              True       True       True        True
IrvinS                                                True              True       True       True        True
HenriettaM                                            True              True       True       True        True
admin                                                 True              True       True       True        True
AllanD                                                True              True       True       True        True
JoniS                                                 True              True       True       True        True
MiriamG                                               True              True       True       True        True
PradeepG                                              True              True       True       True        True
LeeG                                                  True              True       True       True        True
JohannaL                                              True              True       True       True        True
Brian Johnson (TAILSPIN)                              True              True       True       True        True
LidiaH                                                True              True       True       True        True
DiegoS                                                True              True       True       True        True
EnricoC                                               True              True       True       True        True
IsaiahL                                               True              True       True       True        True
Conf Room Adams                                       True              True       True       True        True
Conf Room Baker                                       True              True       True       True        True
Conf Room Stevens                                     True              True       True       True        True
MeganB                                                True              True       True       True        True
Conf Room Crystal                                     True              True       True       True        True
DiscoverySearchMailbox{D919BA05-46A6-415f-80AD-7E09334BB852} True       True       True       True        True
NestorW                                               True              True       True       True        True
LynneR                                                True              True       True       True        True
AlexW                                                 True              True       True       True        True
AdeleV                                                True              True       True       True        True
PattiF                                                True              True       True       True        True
GradyA                                                True              True       True       True        True
JordanM                                               True              True       True       True        True
Conf Room Hood                                        True              True       True       True        True
DebraB                                                True              True       True       True        True
```

Or just view a single user at a time such as: **Get-CASMailbox Username**

```
PS C:\temp> Get-CASMailbox EmilyB

Name   ActiveSyncEnabled OWAEnabled PopEnabled ImapEnabled MapiEnabled SmtpClientAuthenticationDisabled
----   ----------------- ---------- ---------- ----------- ----------- ---------------------------------
EmilyB True              True       True       True        True        True
```

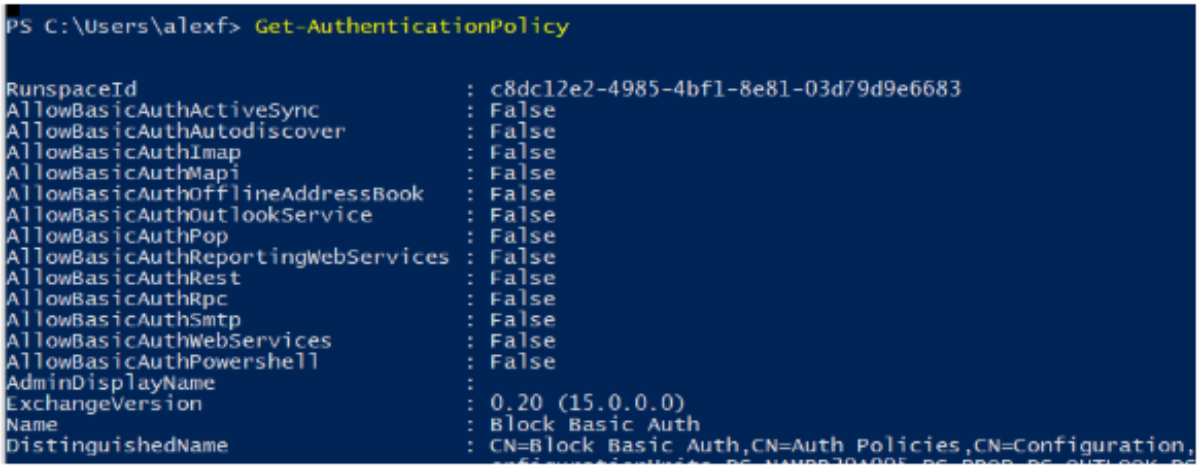
The **Advanced-TenantConfig.ps1** script contains some logic that allows you to disable these across all mailboxes at once if you want to approach it that way. However, in that case you might just want to consider implementing an Authentication policy. Which brings us to…

### Exchange Online Option 2: Block via an Authentication Policy

To completely eliminate basic authentication in Exchange Online, we simply have to create a new authentication policy with no additional parameters, and assign it as the default policy for the organization. This is accomplished in PowerShell, and you can

refer to the **Block-BasicAuth.ps1** script for more details (note that this option is included as well in the **Advanced-TenantConfig.ps1** script).

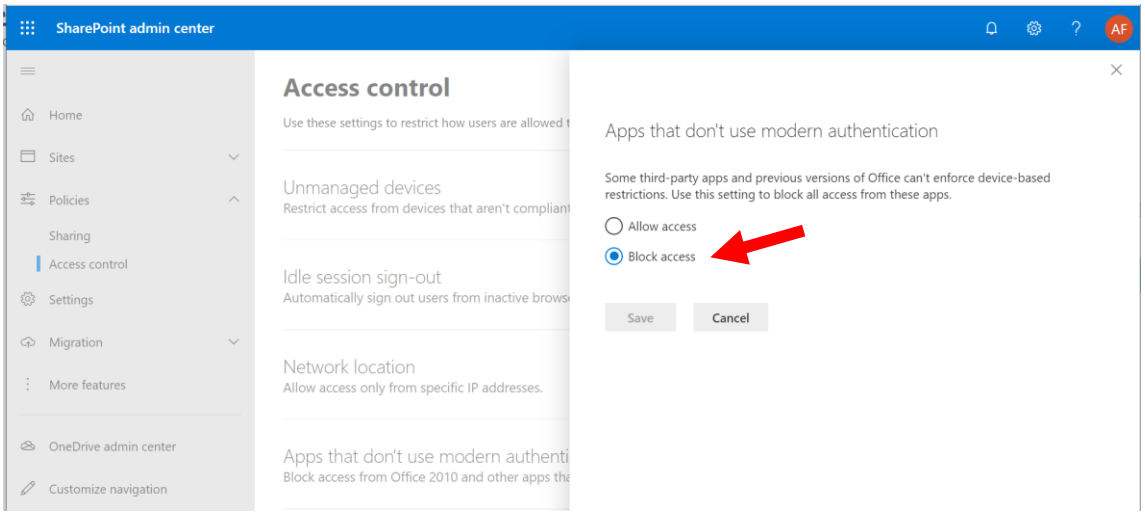
> ***Note**: If you want to see all of the protocols that are being blocked for basic/legacy auth, you can simply run:* ***Get-AuthenticationPolicy***




An example of making exceptions using this method is also detailed in the body of the script, and is commented out. Mailboxes that have a specific policy assigned will override the default that you set for the organization. The org-wide policy is applied only if there is no specific policy assigned to the mailbox.

### Block legacy authentication for SharePoint

You should also disable basic authentication for **SharePoint Online**. Navigate to the SharePoint admin center, and find **Policies > Access control > Apps that don't use modern authentication** and choose **Block access**.

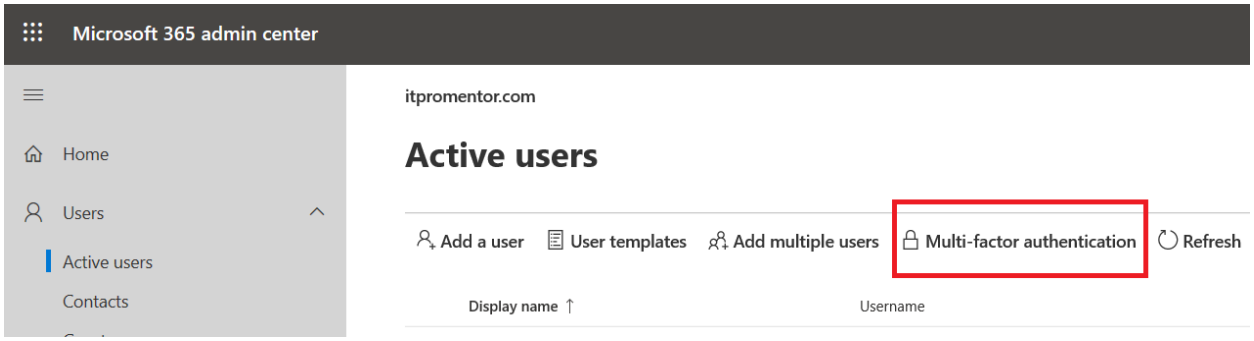



## Setup per-user MFA

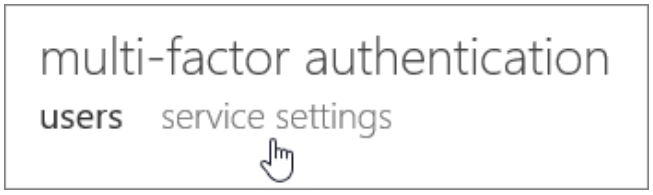
With legacy authentication disabled, you are ready to implement Multi-factor authentication (MFA) for all admin and standard user accounts alike. This method works with any Microsoft 365 or Office 365 subscription.

> ***Note**: You can find a script that will enable MFA on all licensed accounts simultaneously in my GitHub repository under Azure AD.*

Go to the Microsoft 365 Admin Center and navigate to **Users** > **Active users**. Find **Multi-factor authentication** (if you don't see it, this might also be under the "ellipses" button)

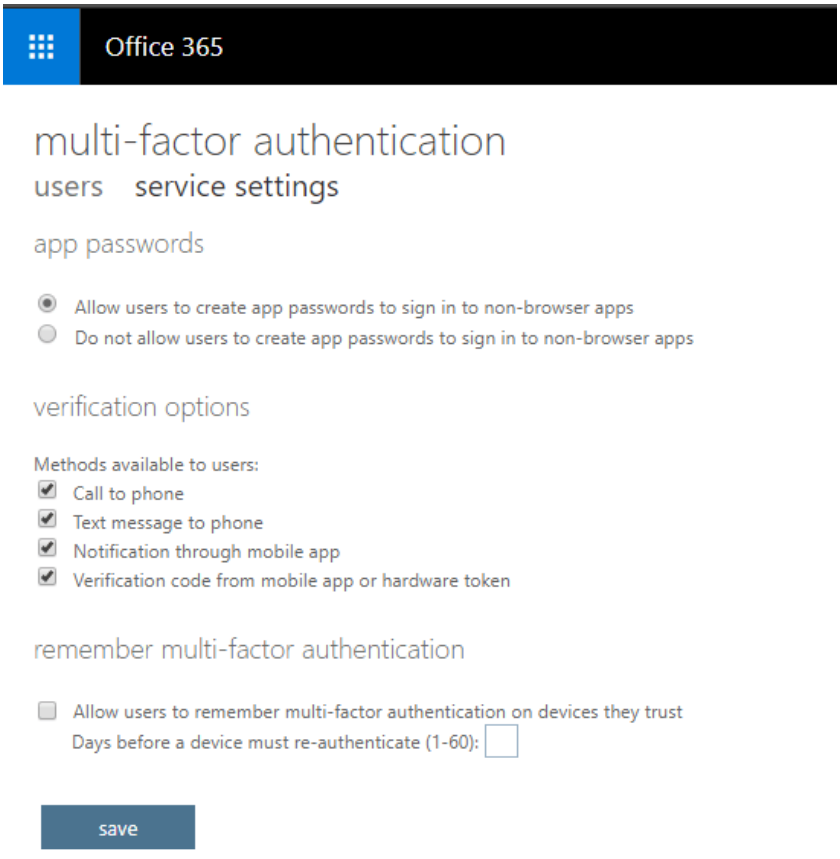



You can see your users listed here, but before you enable MFA for anyone in particular, check out the **service settings** area.

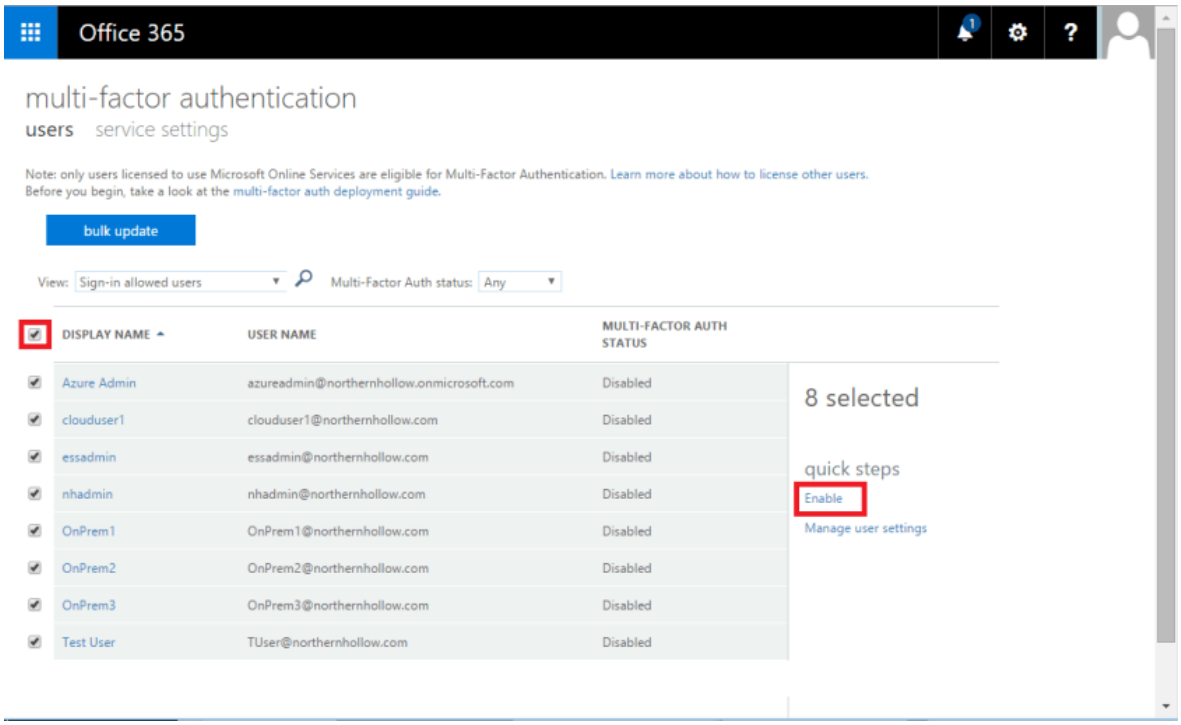



Here you can select various options surrounding the use of MFA. For example, allow certain types of MFA challenge such as phone calls, SMS, mobile app notifications, or hardware tokens. It is also where you allow or disallow users to generate app passwords (for applications that do not support a second factor prompt–e.g. older versions of Office apps, Apple Mail, etc.).

***Note**: App passwords will not work if you already disabled basic authentication, so I normally just turn them off.*



When you have access to Azure AD Premium P1 and Conditional access (standard Office 365 subscriptions will not include this however *Microsoft* 365 subscriptions will), you gain access to another option in here to exclude **trusted IPs** (e.g., corporate locations). Please note, this means the external IP addresses, not the internal IP subnets.




Back on the **users** tab, we can turn MFA on for users one by one, or several at a time. Simply select one, many (or all) of the users, and choose **Enable** on the right.

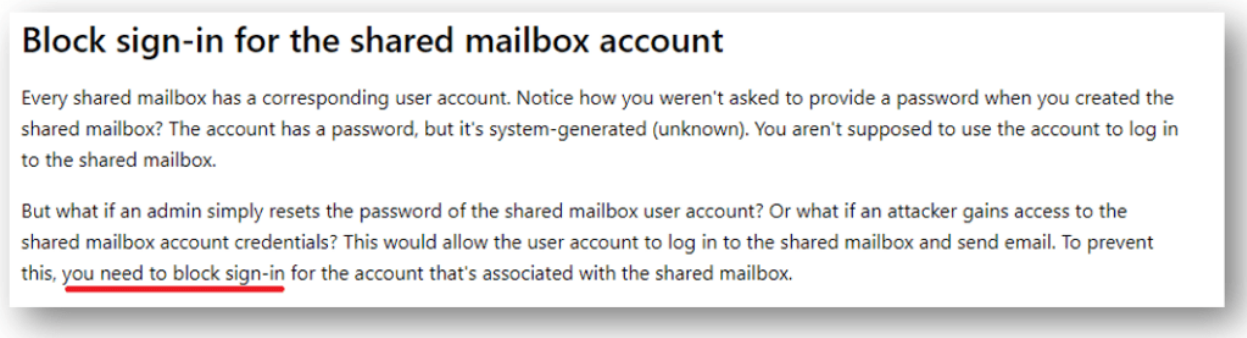
| | A | B |
|---|---|---|
| 1 | Username | MFA Status |
| 2 | chris@contoso.com | Enabled |
| 3 | ben@contoso.com | Disabled |
| 4 | kyle@contoso.com | Disabled |
| 5 | kenny@contoso.com | Enabled |
| 6 | eric@contoso.com | Enabled |

You also have the option to use the **bulk update** button at the top of this page and provide a CSV file which is formatted as depicted above.

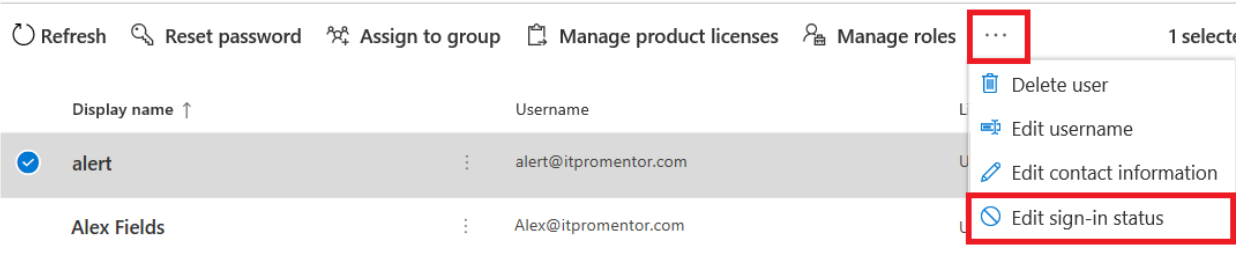
## Block sign-in for all shared mailboxes

When you implement MFA, do not overlook **Shared Mailboxes**. When companies implement MFA, they often overlook these accounts. Worse yet, shared accounts most often have very poor passwords since multiple people might be using them. But users who need access to these resources can simply be given permissions to open those mailboxes from their own account (rather than signing in with a password). Therefore, we will also look at how to disable shared mailboxes for interactive sign-in to Office 365.

Shared mailboxes (including Resource mailboxes) should not require interactive login. Rather, users who are delegated permission can access and interact with the contents of the shared mailbox. When organizations do dumb things like allow multiple users to sign into shared mailboxes on mobile devices, they are not working within the conceptual framework of a shared mailbox. So effectively, those which are enabled for interactive sign-in become real user mailboxes. But hardly anyone thinks to enable these for MFA.




Really though, you should be blocking sign-in for these accounts. Note that accounts which are synced from on-premises Active Directory would need to be disabled on-premises. In the Admin Center, select one or multiple accounts and **Edit the sign-in status** from the **ellipses**.



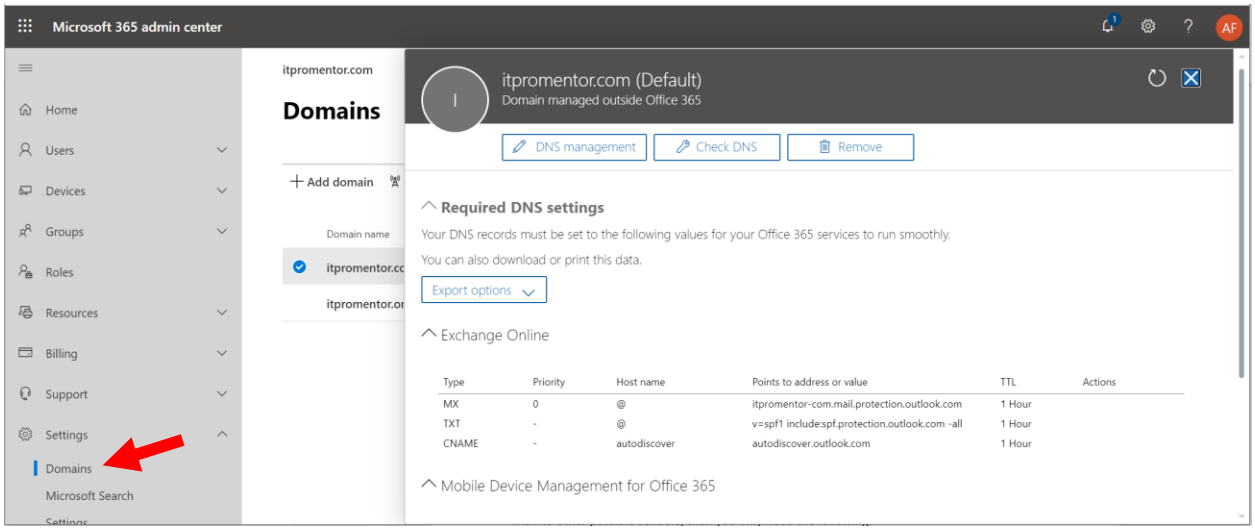

I have a script called **Disable-SharedMbxSignOn.ps1** in the GitHub repository which will find and disable sign-ins for all of the shared mailboxes simultaneously. However, note that this requires the accounts to be labeled accurately as shared mailboxes.

# ☐ 2. Implement Email Authentication

Email authentication is a means of using DNS records to validate or prove that your email is coming from a trusted source. Therefore, it is important that you also protect access to your DNS hosting provider, where these changes can be made. There are three record types in total that we need to configure.

## Sender Policy Framework (SPF)

An SPF record is a DNS "TXT" type record. It is one of the records that Microsoft has you provision when you first setup and configure mail flow to Office 365. Navigate in the Microsoft 365 admin center to **Settings > Domains**.




The function of the SPF record is to advertise to the world who is allowed to send email on behalf of your domain. When you build this TXT record, you should try to include as many "legitimate" sources of email as you can. For example, for email that is hosted at Office 365, with no other possible senders, then you only need the following:

> Host name: @ <or your domain name>
>
> TXT value: v=spf1 include:spf.protection.outlook.com -all

For third-party software such as Mail Chimp, Constant Contact, etc., you can usually find their SPF information using a quick Google search, or by contacting their support. For your own on-premises apps or scan to email devices, you may want to include an ip4 entry for your company's external IP addresses.

Let's say you had a combination of Office 365 for hosted email, Constant Contact for bulk mailing/marketing emails, and an on-premises copier/scanner internally, with your organization's external IP being 87.65.43.21. Then you would have this SPF to publish:

Host name: @ <or your domain name>

TXT value:
v=spf1 include:spf.protection.outlook.com include:spf.constantcontact.com ip4:87.65.43.21 -all

## Domain Keys Identified Mail (DKIM)

DKIM is an authentication system based on an asymmetric cryptographic key pair–a private and public key. When a message leaves Office 365, it is digitally signed with the private key. The public key is published via a DNS CNAME record, so that recipient servers can validate the signature. This proves to recipient servers that your messages really did come from the "right place."

By default, your "OnMicrosoft" domain already has DKIM configured and working. But if you are bringing a "vanity" domain name such as contoso.com (most organizations are), then you will need to setup DNS records for your domain(s), and then enable DKIM message signing in Exchange Online.

You will need to build two CNAME records per domain for DKIM. The format is:

Host name: selector1._domainkey

Points to: selector1-**CompanyDomainName-com**._domainkey.**TenantName**.onmicrosoft.com

Host name: selector2._domainkey

Points to: selector2-**CompanyDomainName-com**._domainkey.**TenantName**.onmicrosoft.com

***Note**: Your domain is separated by a hyphen instead of a period; it should match the domain as depicted in the MX record that is given to you by Office 365 (e.g.: **contoso-com**.mail.protection.outlook.com).*

*Also, the tenant's name (**TenantName**.onmicrosoft.com) can be found under **Settings > Domains** in the Microsoft 365 admin center.*

Therefore, contoso.com, whose tenant name is "contoso.onmicrosoft.com" looks like this:

**Host name**: selector1._domainkey
**Points to**: selector1-**contoso-com**._domainkey.**contoso**.onmicrosoft.com

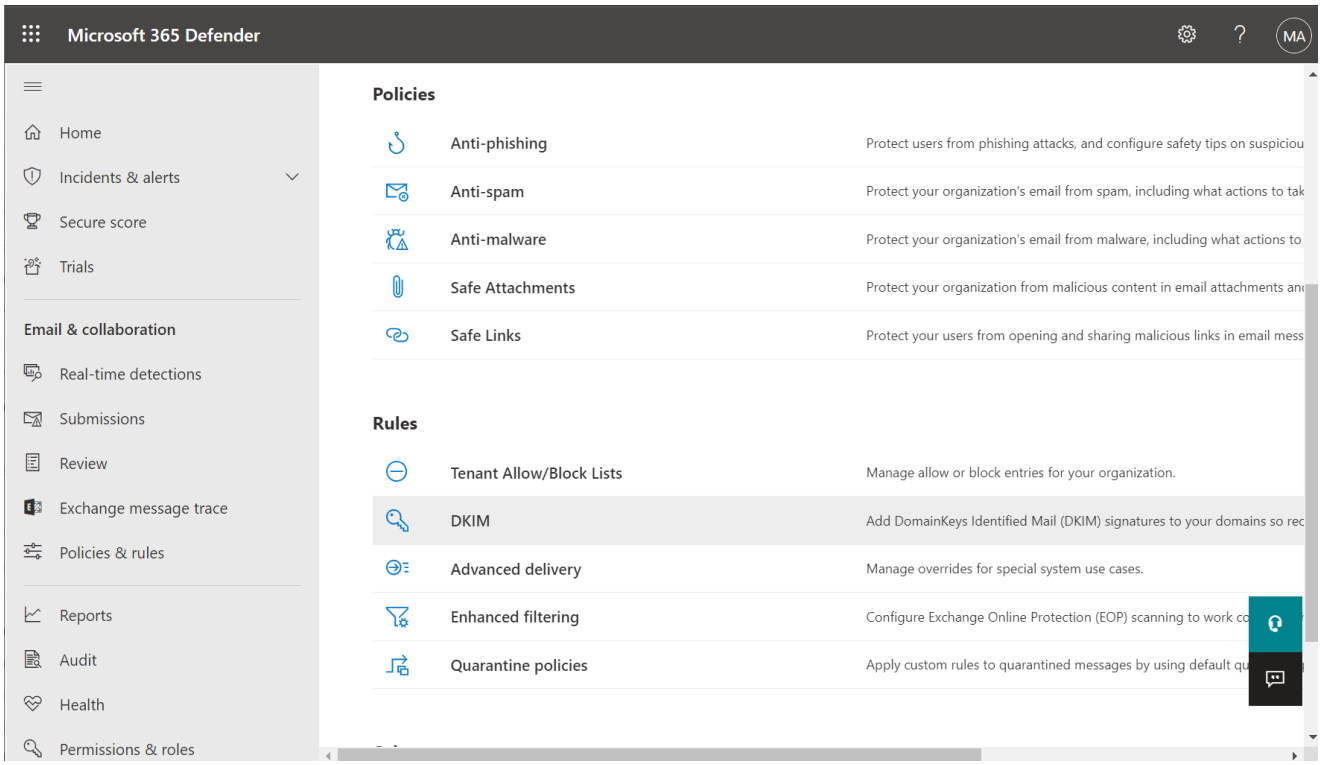
**Host name**: selector2._domainkey
**Points to**: selector2-**contoso-com**._domainkey.**contoso**.onmicrosoft.com

Another example is myfavoritecharity.org with a tenant name of charityrocks.onmicrosoft.com:

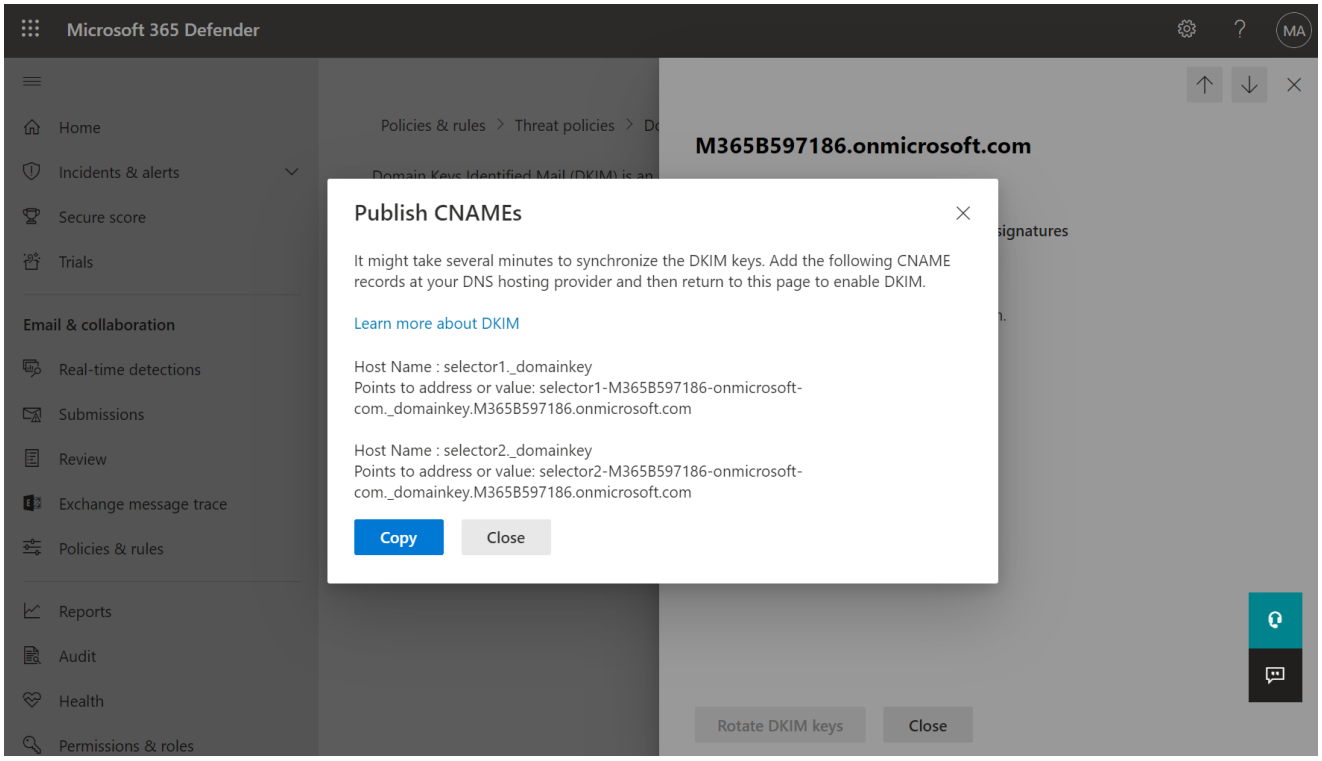
**Host name**: selector1._domainkey
**Points to**: selector1-**myfavoritecharity-org**._domainkey.**charityrocks**.onmicrosoft.com

**Host name**: selector2._domainkey
**Points to**: selector2- **myfavoritecharity-org**._domainkey.**charityrocks**.onmicrosoft.com

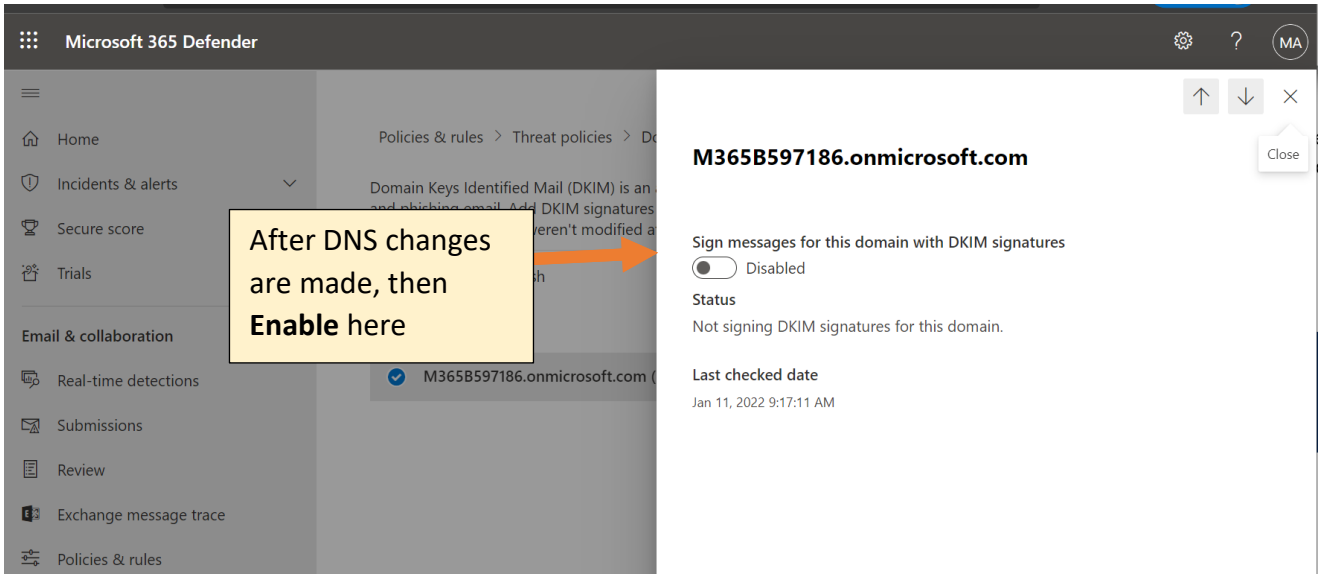
Next, in the **Security admin center**, go to **Policies & rules > Threat Policies** and scroll down under **Rules** to find **DKIM**. Pick the domain that you want to enable for DKIM signing.




In the right-hand flyout that appears, you can generate your DKIM keys for the selected domain; you must publish these records via your DNS hosting provider.




Give it at least a few minutes after publishing before you attempt to **Enable** signing. If you have not configured your DNS records, this operation will fail out, so be sure to allow enough time for DNS to propagate.



You may also see the script entitled **Setup-DKIM.ps1** for a quick way to retrieve the "points to" values.

```
PS C:\Users\alexf> Get-DkimSigningConfig itpromentor.com | fl *cname

Selector1CNAME : selector1-itpromentor-com._domainkey.itpromentor.onmicrosoft.com
Selector2CNAME : selector2-itpromentor-com._domainkey.itpromentor.onmicrosoft.com
```

***Note**: You should also work with third-party authorized senders get their DKIM information and enable signing as well.*

### Domain-based Message Authentication, Reporting & Conformance (DMARC)

DMARC is a DNS record that tells recipient servers how to treat unauthenticated messages that come from your domain, based on policy. It can also communicate where to send reports about mail from your domain.

By way of example, here is what DMARC could look like for **contoso.com**:

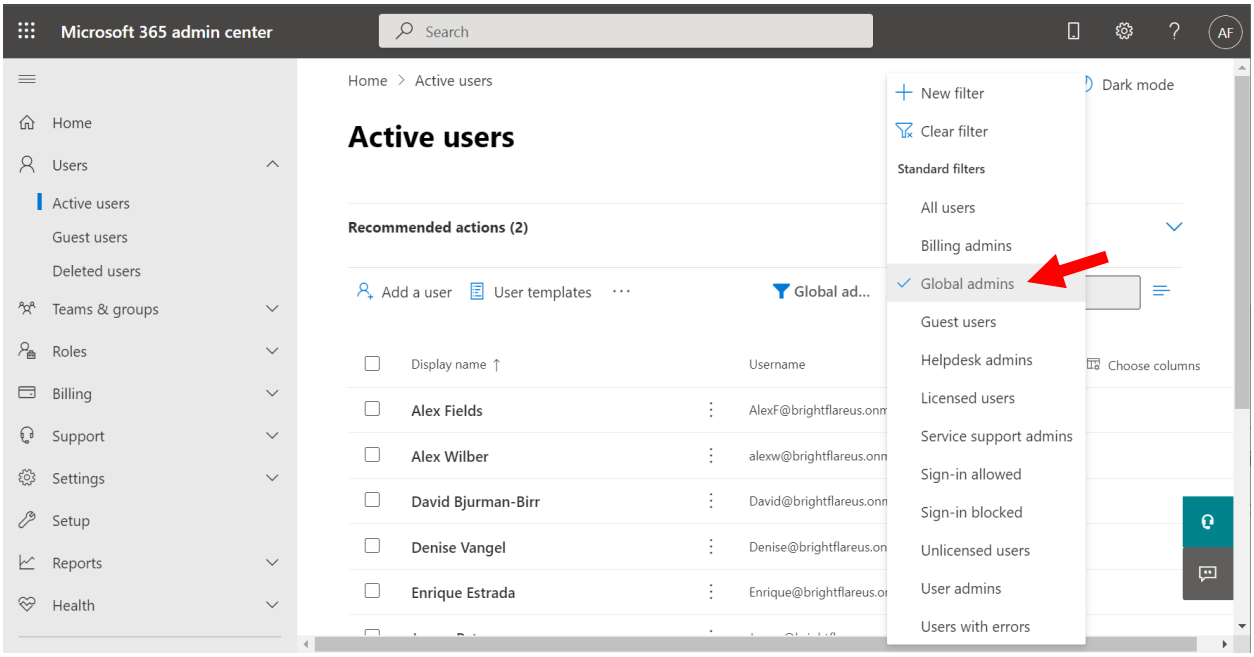
> TXT Name: _dmarc.**contoso.com**
> Value: "v=DMARC1; p=quarantine; pct=100"

However, when you are first rolling DMARC out, it is best to start with the policy set to **p=none**, because this will allow you to take time to find legitimate sources of email and update SPF and DKIM *before* moving the DMARC policy up to a setting of **quarantine**, or even **reject** (the strongest setting).

## ☐ 3. Reduce & Separate Admin Roles

Most tenants still contain administrator accounts that are really primary user accounts (in other words, accounts used for "everyday" work tasks such as email and file sharing). This kind of thing is simply not tolerated anymore, we're working in a Zero Trust framework, remember? So the first thing you need to do is find out who your Global administrators are.

From the Microsoft 365 admin center, go to **Users > Active users**. You should be able to locate a preset filter called **Global admins** which will instantly display the people who are assigned this "superuser" status in your organization.



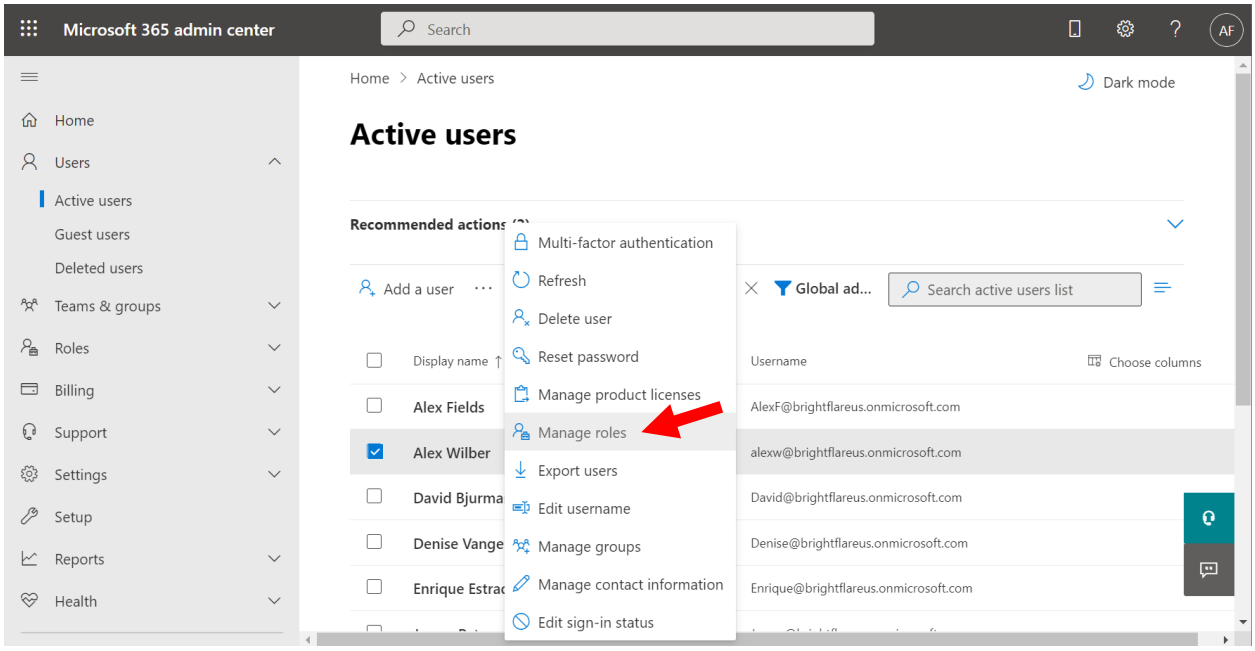

Once you know who these people are, for each one you need to ask: "*Is this privilege necessary for them, or could they get by with lesser privileges?*" For example, could they be assigned a role like ***Helpdesk admin*** or ***Billing admin***, and still do their daily job tasks?

> ***Note***: *For more information about the built-in roles in Azure AD, see this article from Microsoft Docs: Azure AD built-in roles*

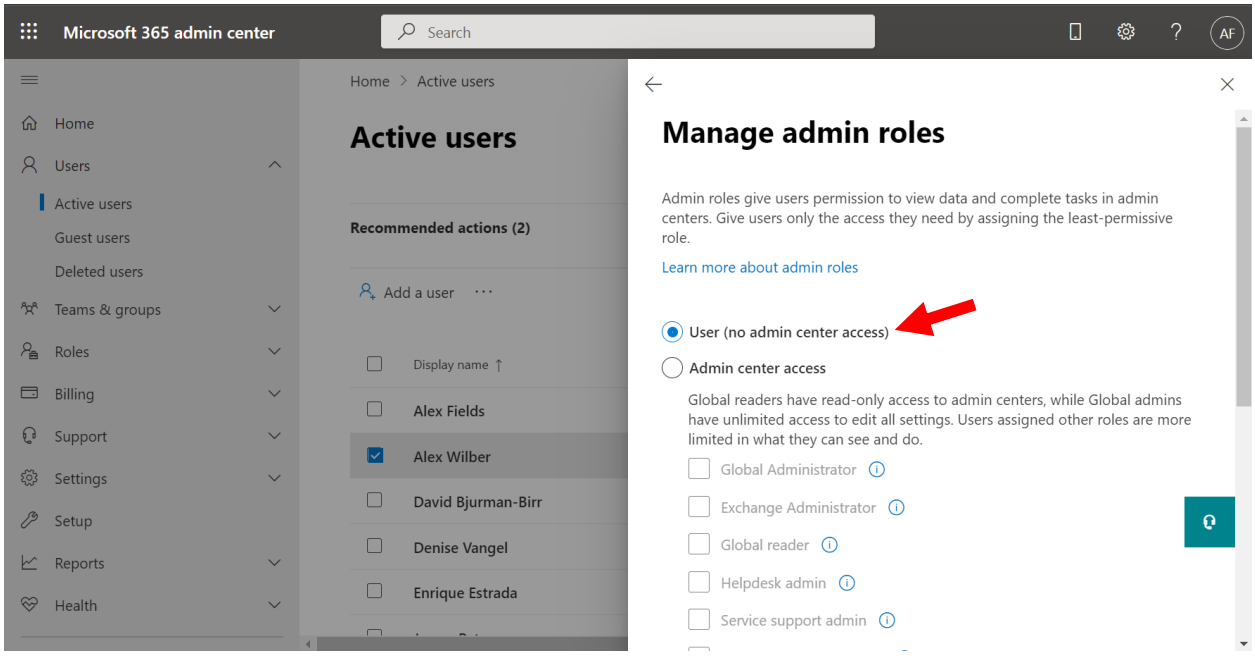
If someone does need a global administrator account, then the next thing you need to do is create a separate, dedicated account for administration activities. For example, if you have a user named Mary Smith who is already assigned the Global admin role, then you should create a second account called Mary Smith (Admin) and then assign the privileged role to it.

This secondary account does not require any kind of licensing. The user should configure this second account with a complex password, and of course, it should be enabled for MFA.

Find the primary user account again in the Microsoft 365 admin center from **Users > Active users**. Click **Manage roles**.




Choose the option **User (no admin center access)** then scroll down and click **Save changes**.




## For MSPs: A note about DAP

If you are a Managed Services Provider, then you likely have set up something called DAP (Delegated Admin Privileges) so that you can access your customer's tenants as

an administrator using your own Azure AD login. While this can be a convenient arrangement, it is hard for me to recommend it at this time. The reason being: we have no way to constrain the delegation, so basically any and every partner account can be granted *full* global administrator over every customer tenant connected to the Partner center via DAP.

In the not-to-distant future, we will have something called Granular DAP or GDAP, which will improve the experience and allow us to constrain the privileges that are delegated to our partner tenants. But until that feature arrives, it presents more of a risk than what I think is appropriate for most organizations. Which is a shame, since DAP is also a prerequisite for Microsoft's Lighthouse product, which enables a multi-tenant management view.

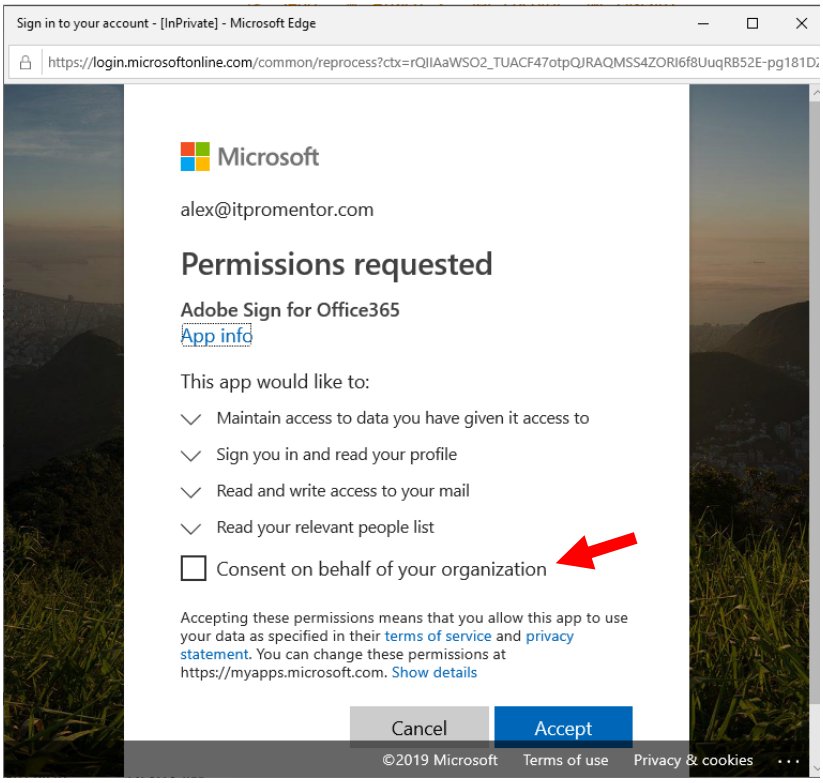
The alternative for now is to create separate accounts within the customer's organization that are used for management purposes. There are creative ways to accomplish this in a secure way including shared MFA such as virtual SMS or email-to-Teams or a shared mailbox within the organization. An even better option is to assign dedicated techs (at least a primary and secondary) to each customer, and have each person use a dedicated account. This would require you to keep track in your PSA or CRM tool who is the primary/secondary/tertiary engineers assigned to each customer.

***Note**: Be sure you update your onboarding and offboarding processes, both at the partner level and at the customer level!*

When GDAP lands, and after I get a chance to hammer on it a little bit, I will likely update this resource with some additional guidance for partners.

## ☐ 4. Manage Application Consent

Third-party add-ins and applications for Office 365 may sometimes prompt end users to consent to granting access to Office 365. Below is an example when activating an Outlook add-in called Adobe Sign (a popular app for electronic signatures):




The reason this is risky is because phishing emails may contain links to malicious apps that trigger this type of workflow, asking the user to grant permissions to their Office 365 data. And this means the attackers would not even require a username and password to get in at all (because they would have an OAuth token granted to them by the already authenticated user).

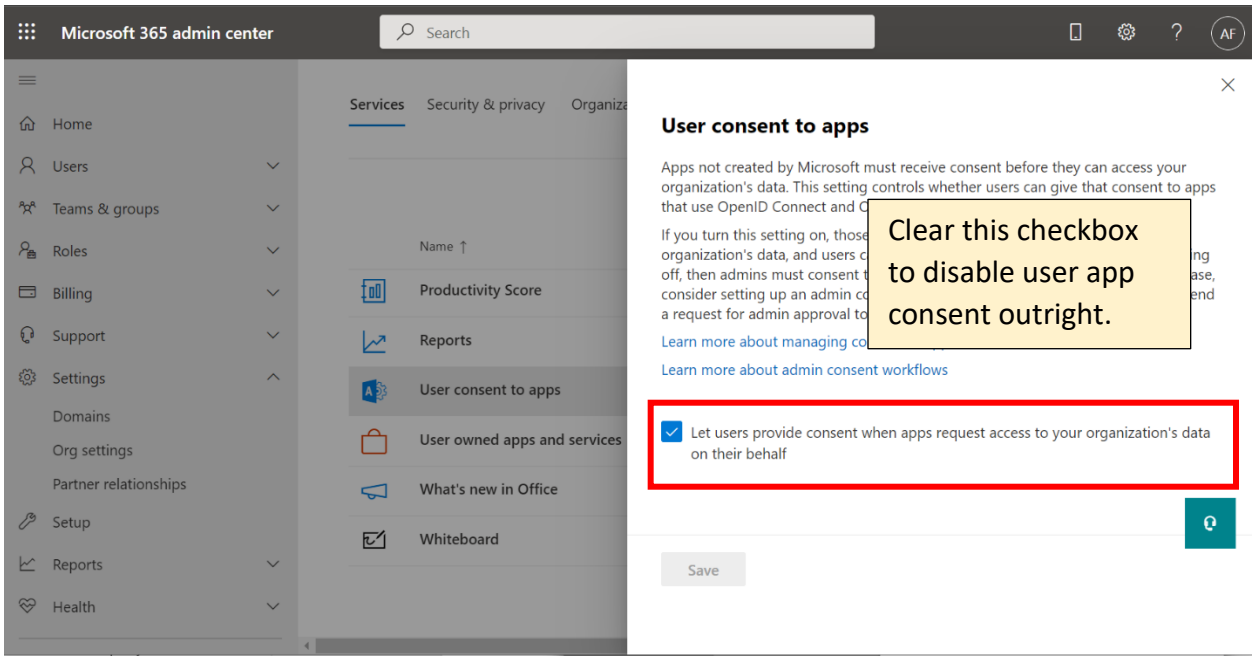
We have three options to better manage this behavior:

1. Disable consent to applications requesting permissions
2. Constrain or limit consent to low permissions from trusted publishers
3. Set up an admin consent request approval workflow

### Option 1. Disable app consent

Now it *is* possible to prevent users from being able to consent to these requests in the first place. Navigate to the Microsoft 365 admin center, find **Settings > Org settings** and

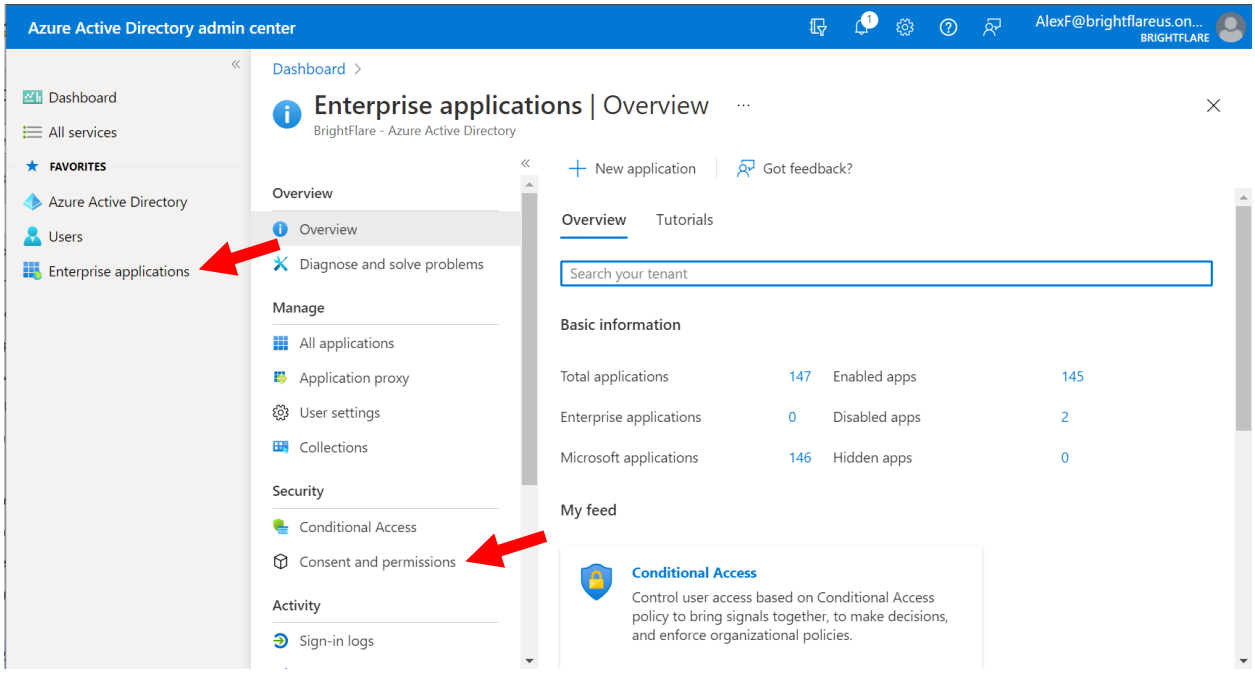
then find **User consent to apps**. Clear the checkmark box for ***Let users provide consent when apps request access to your organization's data on their behalf***. Click **Save**.



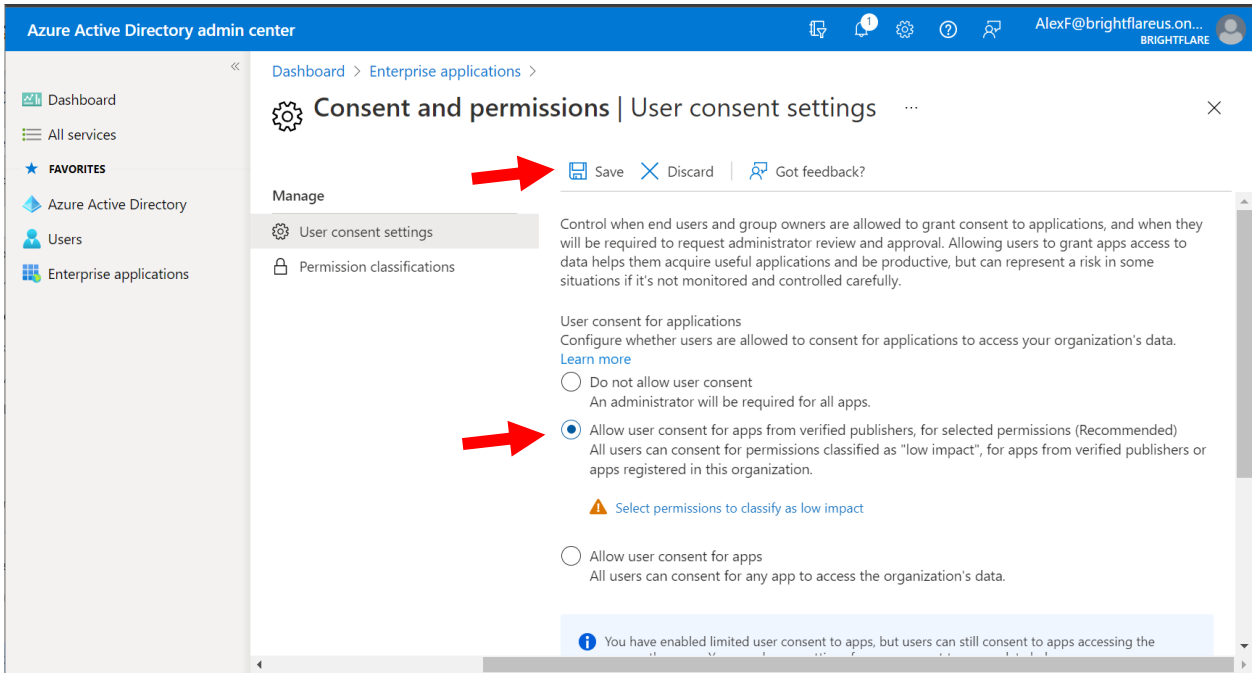

Although this is the easiest way to remove the risk, users will not be able to integrate with third-party apps in this configuration! If you still want to add applications and allow integrations, this would require an administrator to manually complete the process each time (see this article for more details). For an administrator, going through the consent request process will trigger consent for the entire organization (rather than just one user).

## Option 2. Limit app consent

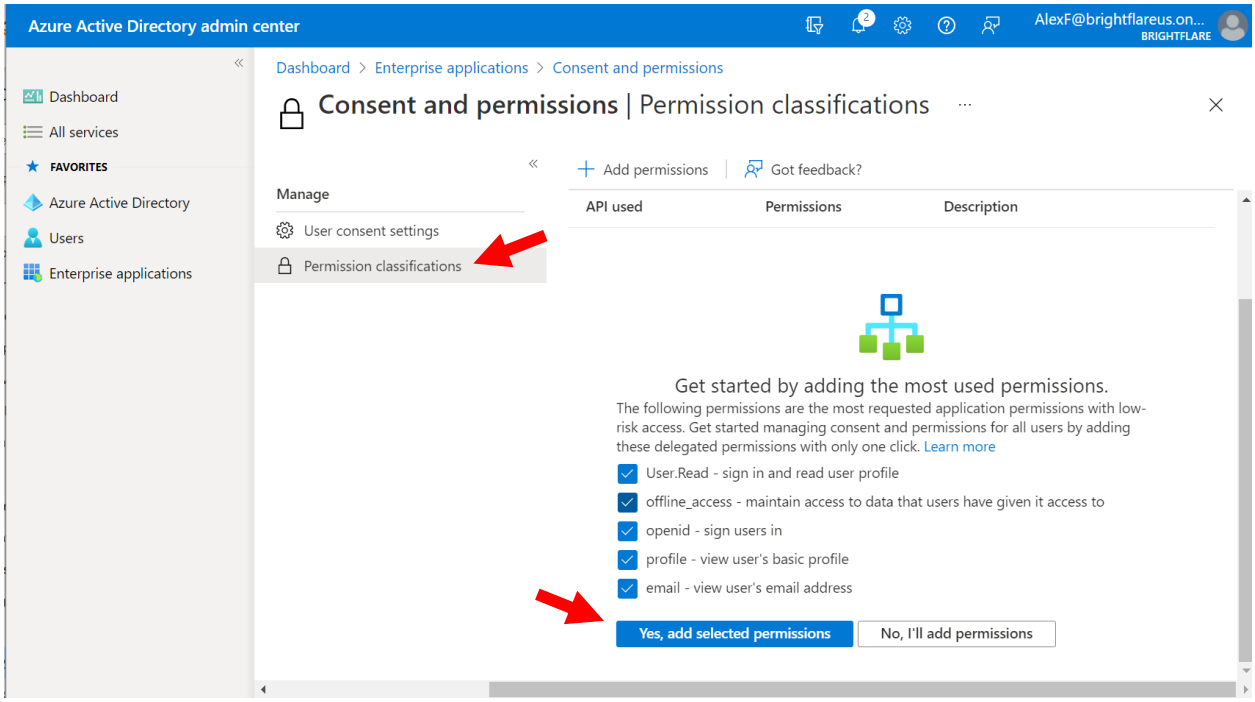
We also have the option to *limit* a user's ability to consent to application permissions requests. We can accomplish this from the Azure AD admin center; navigate to **Enterprise applications > Consent and permissions**.




To implement this option, select the middle option: "**Allow user consent for apps from verified publishers, for selected permissions (Recommended)**." Click **Save**.

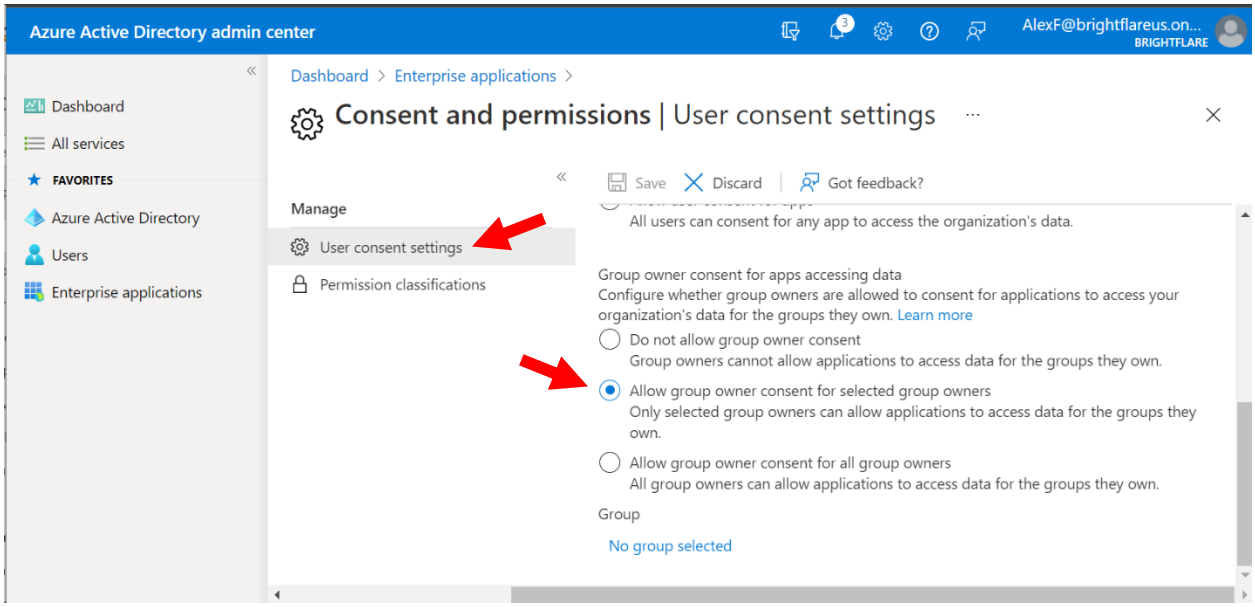



You should also visit the **Permissions classifications** page to set the permissions that are classified as "low" impact.




All you have to do for a minimum acceptable configuration is to select all the checkboxes on this page to allow the most commonly requested low-impact permissions, and then click on the button: **Yes, add selected permissions**.

You can also manage permissions requests for group owners, back on the **User consent settings** page (scroll down to find the group settings). In this case we can restrict *who* is allowed to consent to permissions requests for groups. Your options are: turn it off for everyone, turn it on for some people, or turn it on for all people.

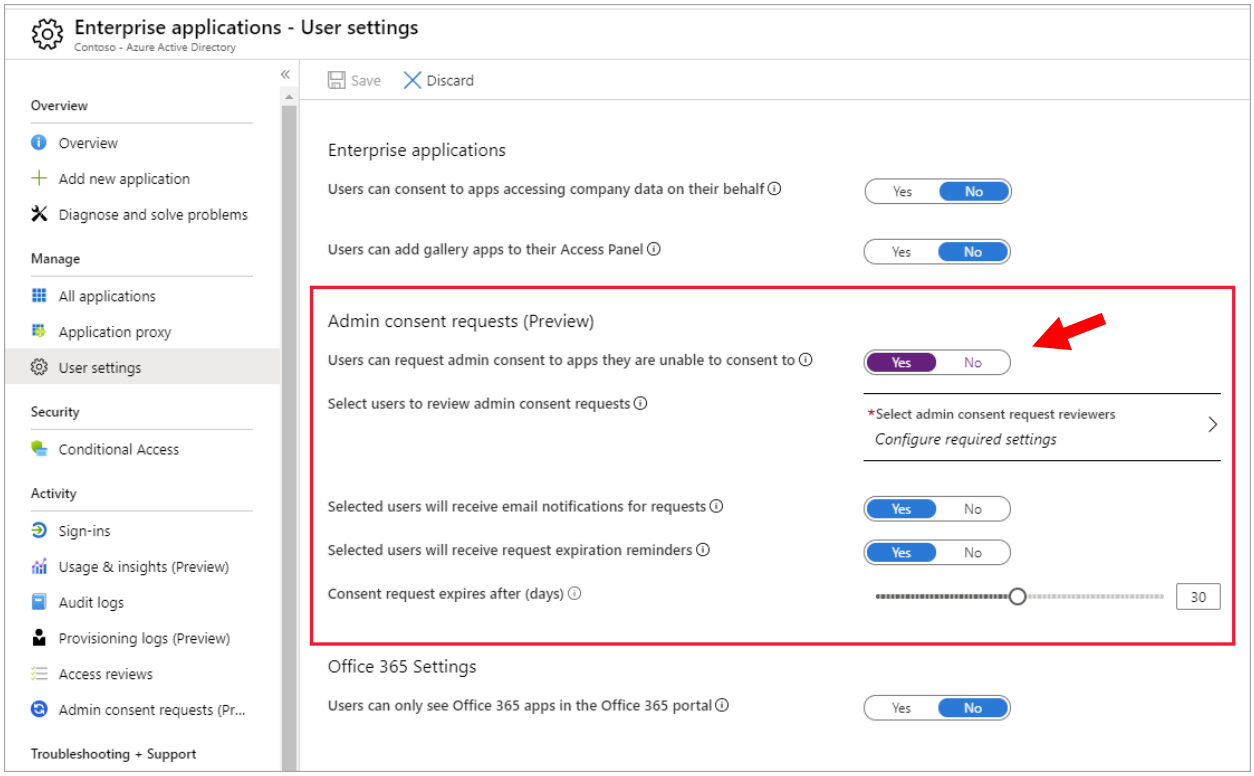



Some organizations choose to restrict who is allowed to even create Microsoft 365 Groups; if that describes your organization, then you can choose **Allow group owner consent for selected group owners**, and leverage the same security group that you use to manage group creation again here.

If your organization is taking a "closed ecosystem" approach with strong compliance boundaries and a policy that says employees should keep company data in Microsoft 365, then you would most likely select the first option, **Do not allow group owner consent**.

## Option 3. Admin consent request workflow

The last option can be combined with either of the above options. Here we enable an "approval process" where admins can review and then approve requests when users attempt to add applications. Set this up in the **Azure AD admin center** under **Enterprise Applications > User Settings > Admin consent requests**.




Using this solution, the administrator would be notified when an end user requests permissions for a new application. The designated application administrator would then need to go to the Azure AD admin portal, and consent to every "legitimate" app that users wanted to add. This is done from the Azure AD admin center under **Enterprise Applications > Admin consent requests**.

## ☐ 5. Record user and admin activity

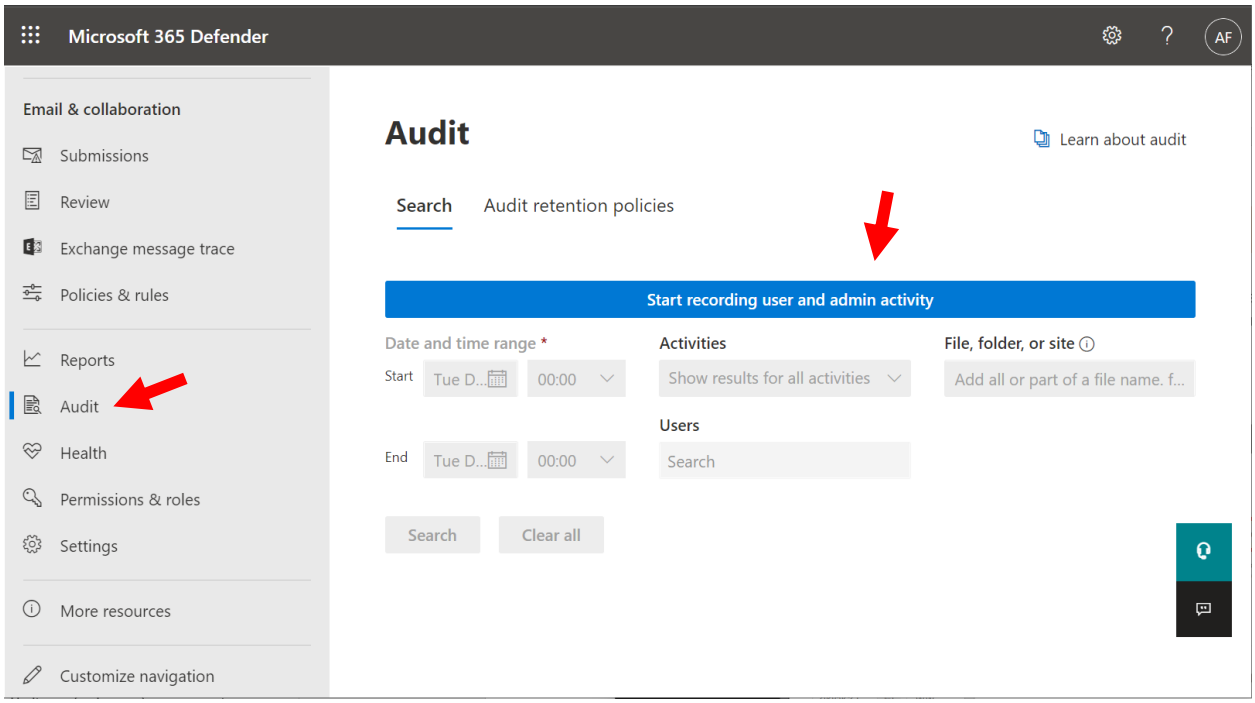
Auditing is crucial. If there ever is a breach, you want logging enabled in order to understand what happened and when. Not to mention it is usually required for compliance with various laws and regulations.

```
PS C:\temp> get-mailbox |ft identity,*audit*
Creating a new Remote PowerShell session using MFA for implicit remoting of "Get-Mailbox" command ...

Identity                                          AuditEnabled AuditLogAgeLimit AuditAdmin
--------                                          ------------ ---------------- ----------
AdeleV                                                    True 90.00:00:00      {Update, MoveToDeletedItems, SoftDelete, Har...
AlexW                                                     True 90.00:00:00      {Update, MoveToDeletedItems, SoftDelete, Har...
AllanD                                                    True 90.00:00:00      {Update, MoveToDeletedItems, SoftDelete, Har...
Brian Johnson (TAILSPIN)                                  True 90.00:00:00      {Update, MoveToDeletedItems, SoftDelete, Har...
ChristieC                                                 True 90.00:00:00      {Update, MoveToDeletedItems, SoftDelete, Har...
Conf Room Adams                                           True 90.00:00:00      {Update, MoveToDeletedItems, SoftDelete, Har...
Conf Room Baker                                           True 90.00:00:00      {Update, MoveToDeletedItems, SoftDelete, Har...
```

While auditing is enabled by default for mailboxes, the Unified Audit Log is *not* enabled by default. But having this turned on is necessary so that you can record the audit log information in one central place, and also search across and generate alerts based on that data.

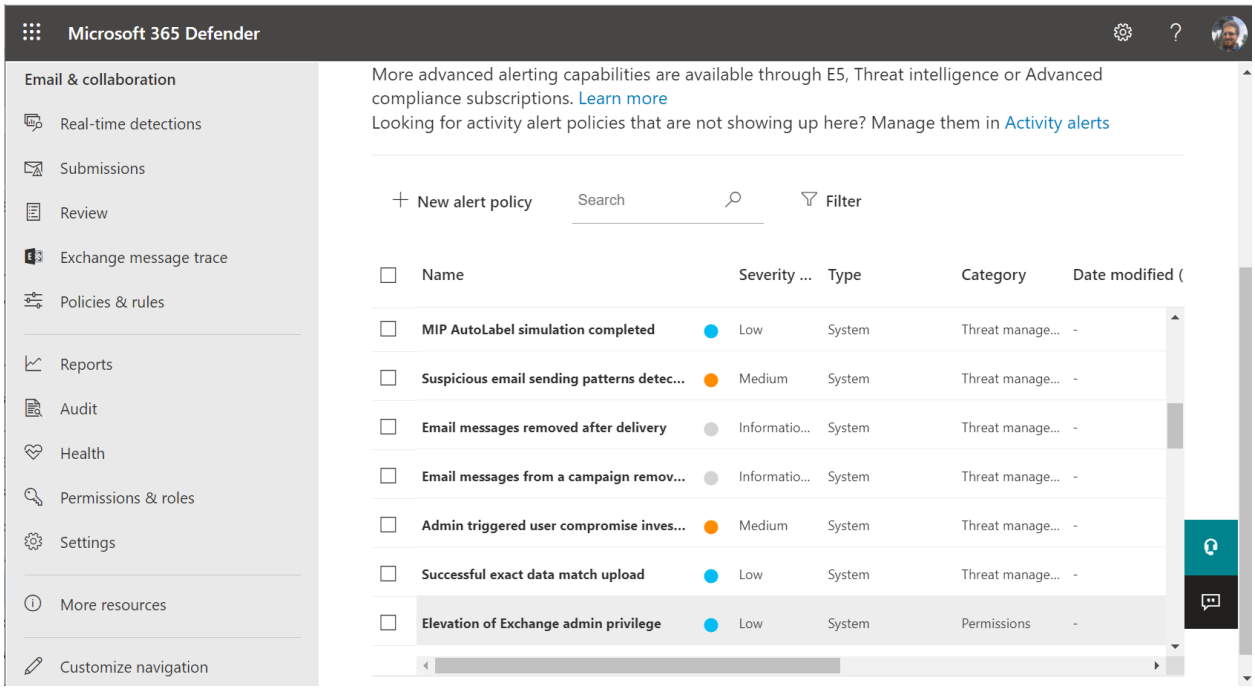
From the Security center go to **Search > Audit log search** and click **Start recording user and admin activity**. Audit data is kept for 90 days by default (you can extend this timeframe but it requires additional licensing or an E5 plan).



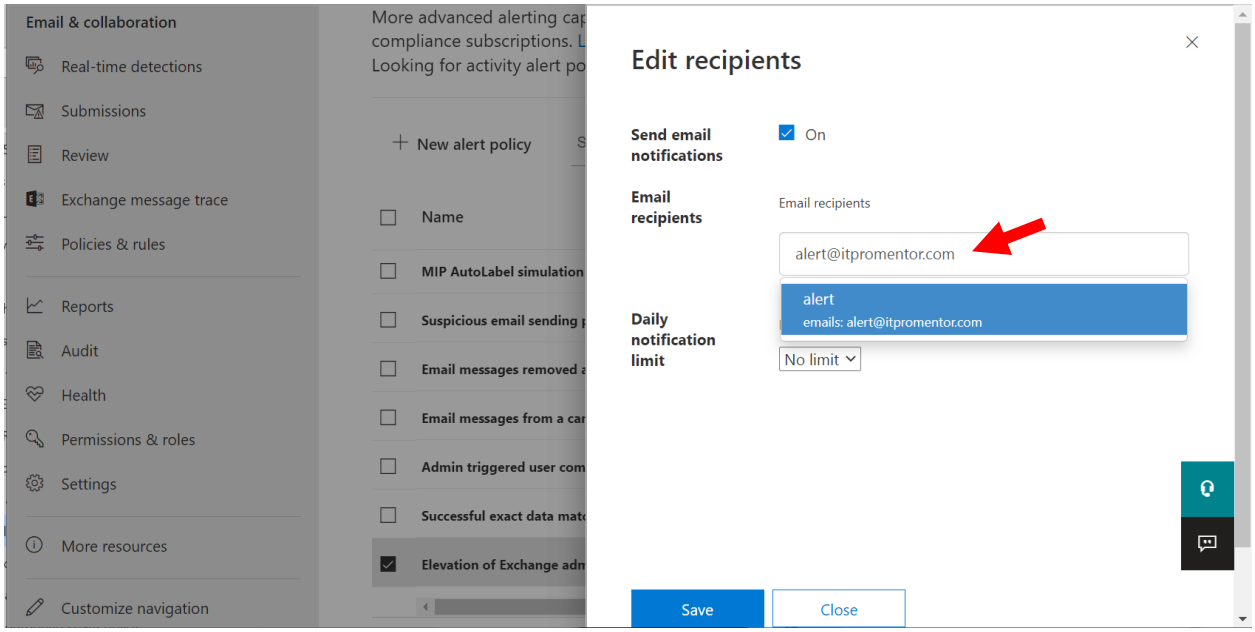

Note that it can take several hours before the data is available for searching and alert policies.

## ☐ 6. Configure Alert Policies

Alert Policies will generate email notifications when certain types of high-risk events happen in Office 365. From the Security center under **Email & collaboration** choose **Policies & rules > Alert policy**. From here, you should see at least a few basic policies which are created by default:

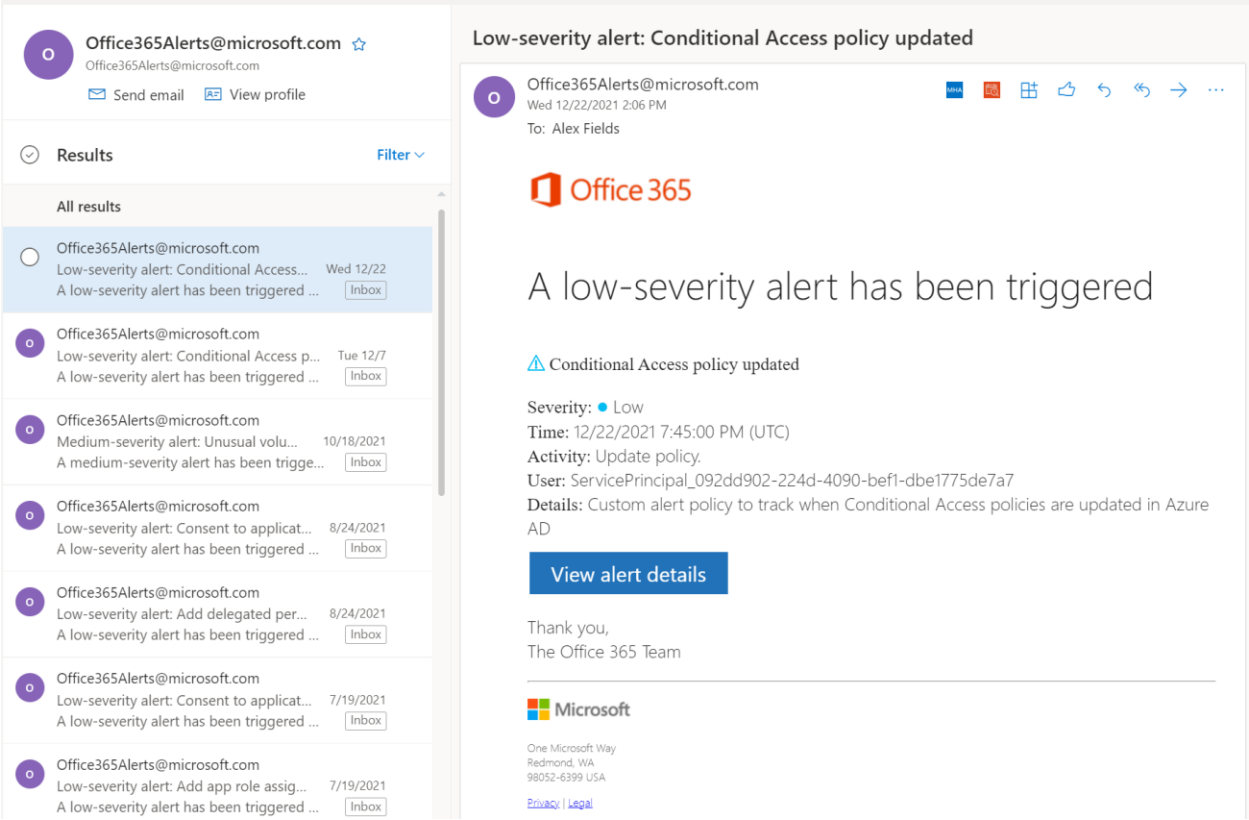



Edit the default policies now, and change the recipients to people who will actually see the alerts and be able to act on them. For example, if you are a service provider, this may be sent your ticketing system.



***Note**: If you have Microsoft Defender for Office 365 plans, then this screen will include many more default (a.k.a. System) alerts. Refer here for more detail on the default policies included with each subscription.*

When an alert is triggered, you can expect an email notification like the one pictured below.



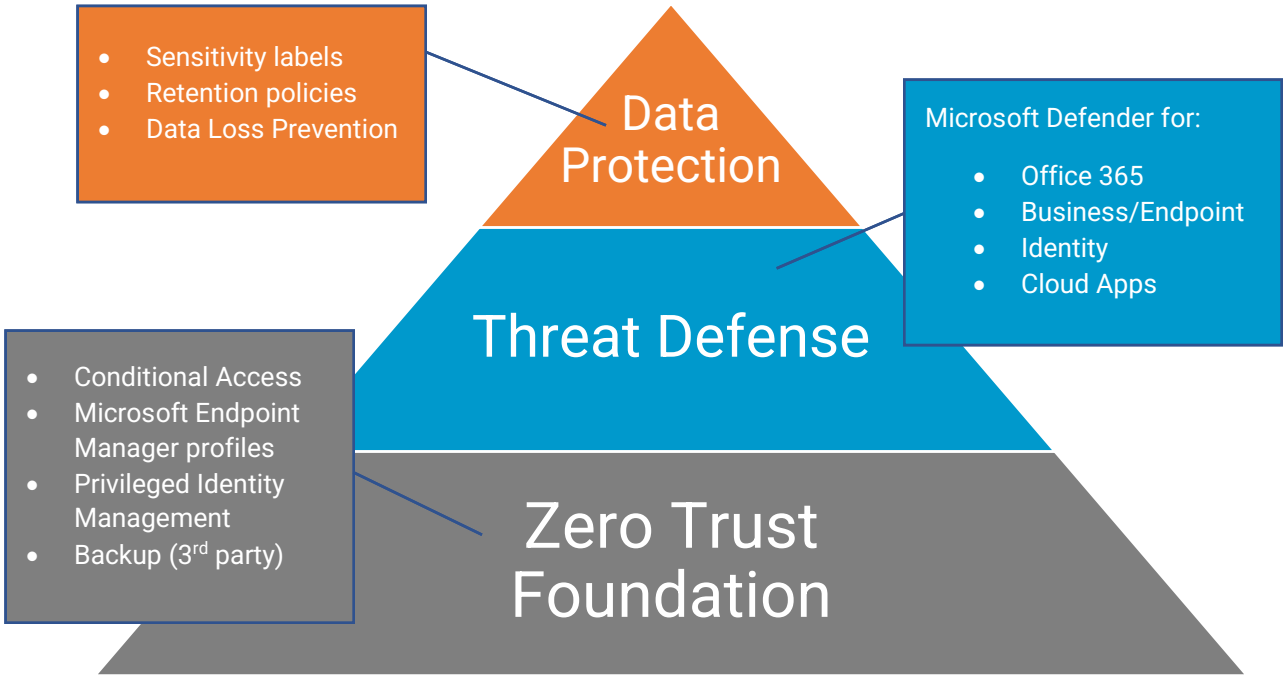

The person monitoring this alert feed will need to investigate each alert and find out whether it was an expected activity, or if it was illegitimate, whether it indicates a breach or insider risk event.

Consider configuring additional alert policies for monitoring important Azure AD activities, such as changes to Conditional Access policies and application consent requests. See this script to install several such policies that I recommend.

## ☐ 7. Bonus: Threat Defense (and beyond)

As I mentioned previously, you can further buttress your Zero Trust Foundation, but doing so may require additional products such as Azure AD Premium or Microsoft Endpoint Manager (Intune). The six items we just stepped through above are the minimum I would recommend to every tenant regardless of subscription level.

Once you have a solid foundation in place, you can proceed on to more advanced security initiatives such as **Threat defense** and **Data protection**.



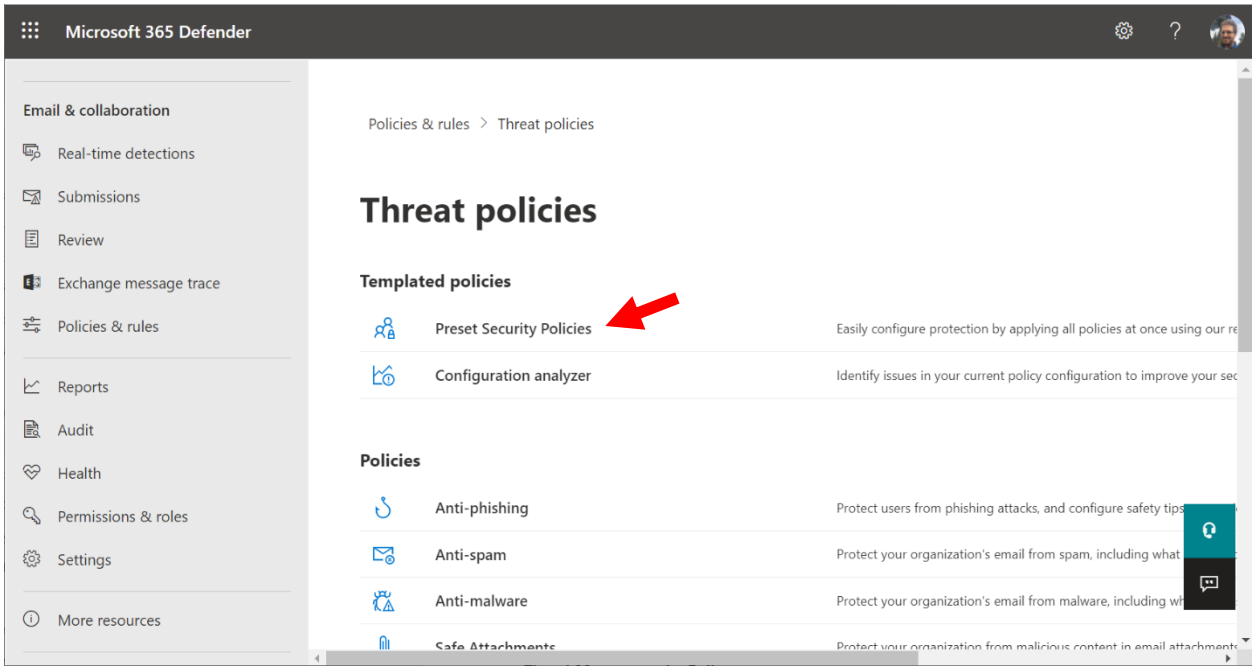

Although a comprehensive overview of this architecture is beyond the scope of this guide, in the graphic above I wanted to highlight some items that can take you further on this journey. I encourage you to check out my courses, books, and other resources at ITProMentor.com for more details on these other items.

But before we sign off, I want to share one more (BONUS!) Security Essential that takes us into **Threat defense** territory.

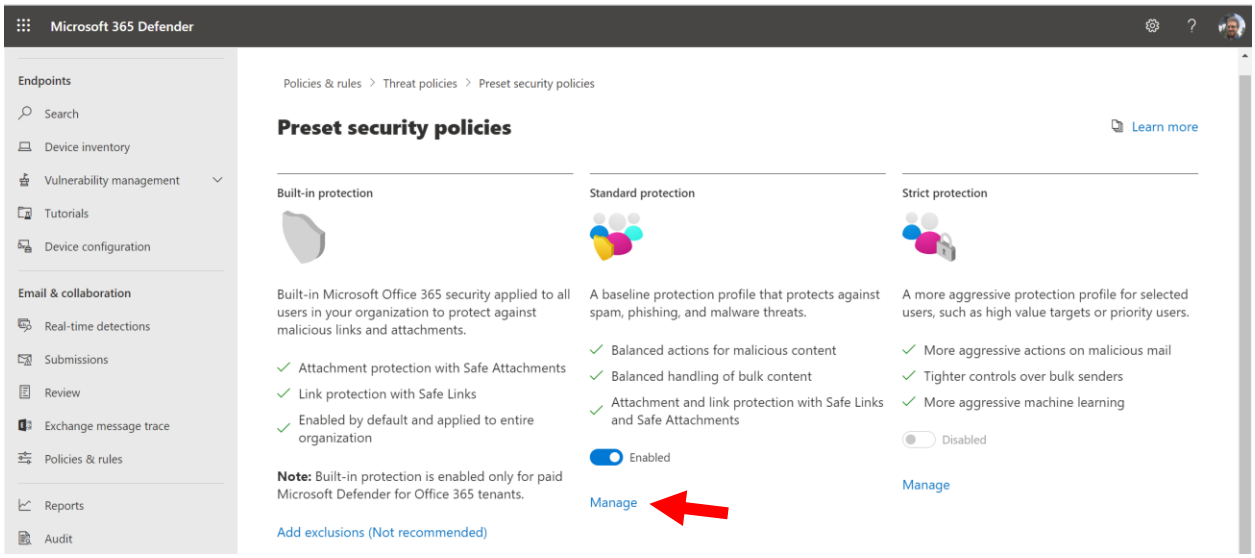
## Preset Security Policies

Most organizations have a subscription that includes some flavor of Office 365 Exchange Online for Email. In fact, this is one of the most popular email services in the world for business. As administrators, we have to manage half a dozen policies or so related to the security of Email & Collaboration. Wouldn't it be nice if Microsoft simplified this for us, with "Best Practices at the Click of a Button?"

Enter **Preset Security Policies**. These are a collection of settings that represent Microsoft's recommended configuration for both Exchange Online Protection (EOP), as well as Microsoft Defender for Office 365 (MDO)—included in some of the more comprehensive subscriptions such as Microsoft 365 Business Premium. Think of the former as your traditional anti-spam filters, and the latter as more advanced protection against targeted phishing campaigns and Zero-Day threats. Find **Threat policies** in the **Security center** under **Policies & rules**.

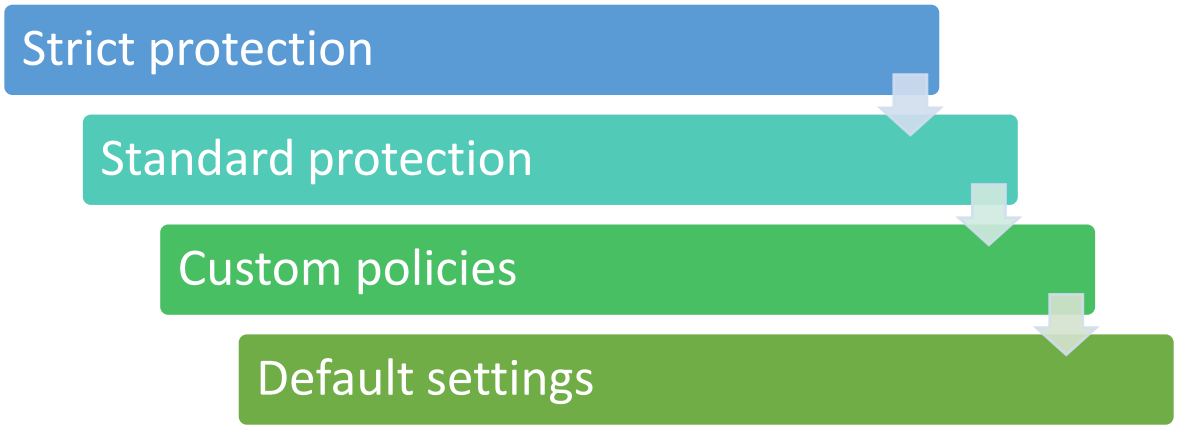



While it is still possible to manage all of your individual policies separately of course, I still recommend most small and mid-sized customers stick with the *Presets*. Especially because Microsoft may update their best practices over time. With the presets in place, you will be automatically upgraded when Microsoft adds or removes features, or alters settings based on threats that Microsoft is seeing in the wild. If you do decide to manage custom policies yourself, then please refer to this Microsoft Docs article for guidance (and revisit on a regular cadence).




***Note**: I find that the 'Strict protection' leads to more false positives, while 'Standard protection' works well for most organizations.*

Regarding policy precedence: **Strict** overrides **Standard** overrides **Custom** overrides **Default**. So, what does that mean? It means if you deploy a **Preset** policy, it will take precedence over any custom or default policies you have out there. Remember that **Strict** will always outstrip **Standard**; so, if a user falls under the scope of two policies, now you know which one wins.

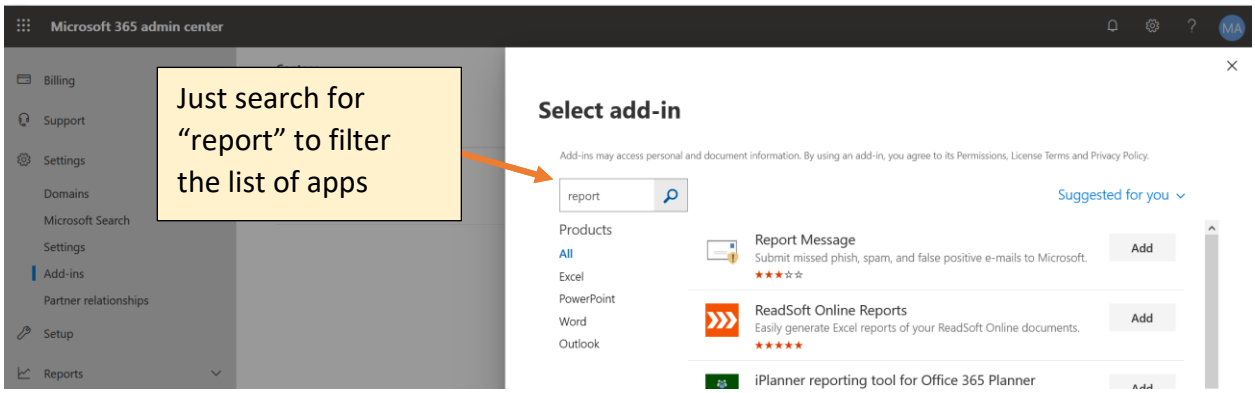



One last note: not every bad thing will be caught by these policies; some items might slip by. Therefore, you should give end users the ability to self-report email messages

that they believe are junk or phishing, by providing them with the **Report Message** add-in.

### Install the Report Message add-in for end users

From the Admin center, go to **Settings > Integrated apps** and click **Get apps**. Click **Next** then **Choose from Store**. You can find the **Report Message** add-in here and click **Add**.




Accept the terms, then choose your deploy options (***Everyone*** and ***Fixed***). Click **Deploy** to finish (but note that it can take up to 12 hours to appear for users).

## Closing comments

Many Microsoft 365 tenants out there today have not even accomplished a basic level of security yet. Therefore, on new tenants moving forward, Microsoft has started to implement certain settings like the **Security defaults** and **Built-in protection** for Exchange Online. But these are still just the tip of the iceberg. If you followed this guide through to the end, then you are doing better than approximately 80% of tenants in existence at the time of this writing. So, congratulations.

On the other hand, there is so much more to do: Device-based conditional access policies, Microsoft Endpoint Manager, Endpoint security profiles, Microsoft 365 Defender, Data protection, and more.

If you enjoyed this product, consider checking out my other publications and courses available now at ITProMentor.com/shop, or consider joining our peer group.

Thank you for reading!

Alex Fields
ITProMentor.com