YubiKeys for Entra ID passwordless lifecycle management

Copyright

Trademarks

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. Yubico shall have no liability for any error or damages of any kind resulting from the use of this document.

The Yubico Software referenced in this document is licensed to you under the terms and conditions accompanying the software or as otherwise agreed between you or the company that you are representing.

Contact Information

Yubico Inc
5201 Great America Pkwy #122
Santa Clara, CA 95054
USA
yubi.co/contact

Version History

Version	Date	Changes
1.0	May 20, 2025	• Initial Release

Copyright 2
Trademarks 2
Disclaimer 2
Contact Information 2
Version History 2
Introduction 4
Onboarding 4
Passkey Registration Methods 4
Attestation and Key Restrictions 5
Authentication Scenarios 5
Offboarding 6
User-Retained Model 6
Reclaimed Model 6
Key Management 6
PIN Management 6
Lost or Stolen YubiKeys 7
Recommended Actions 7
Recovery 7
Forgotten PIN 7
Locked YubiKey (PIN Retry Limit Reached) 7
Key Resets 8
Re-registration 8
Usage Monitoring and Auditing 9
Sign-in Logs 9
Audit Logs 9
Delete a passkey (FIDO2) 10
Option 1: Deleting a Passkey (FIDO2) via the Entra Admin Center 10
Option 2: Deleting a Passkey (FIDO2) via PowerShell 11
Confirm passkey (FIDO2) deletion 12
Option 1: Using the Entra Admin Center 12
Option 2: Using Microsoft Graph PowerShell 13

Introduction

Effective lifecycle management ensures that YubiKeys remain usable, secure, and accounted for throughout a user's time with the organization. This section outlines best practices and operational considerations for managing YubiKeys after initial deployment.

Onboarding

Enterprises may adopt different registration and identity binding models when onboarding new users. YubiKeys support secure, phishing-resistant authentication through passkeys (FIDO2), and Entra ID provides multiple options for enabling users to register their YubiKeys during onboarding.

Passkey Registration Methods

Entra ID enforces an identity verification step during registration to prevent unauthorized credential creation.

Self-Service Registration

To complete passkey registration via self-service, users must first verify their identity. Entra ID supports the following verification methods:

- Microsoft Entra Multifactor Authentication (MFA)

  Any enabled and registered MFA method can fulfill this requirement. This includes using the YubiKey 5 Series as an OATH-TOTP credential with the Yubico Authenticator App or using a PIV smart card credential on a YubiKey.

- Temporary Access Pass (TAP)

  A time-limited, configurable passcode that can be used for one-time or multi-use sign-ins. TAP enables secure onboarding without ever requiring a password and satisfies the MFA requirement for passkey registration.

There are two modes for self-service registering passkeys (FIDO2) with YubiKeys

- **Managed Mode (Self-Service Security Info Portal)**

  Users register their passkey by navigating to the Microsoft Entra Security Info portal, where they can add or remove authentication methods. This model supports phased rollouts and minimal IT intervention.

  Refer to the **User Enablement Guide** for the setup and user experience for Managed Mode registration.

- **Interrupt Mode (In-line Registration)**

  Users are prompted to register a YubiKey passkey during sign-in or while updating their security info. This mode is typically enforced through Conditional Access when accessing a resource protected by an authentication strength requiring passkeys.

  Refer to the **Enforce YubiKeys for Entra ID Sign-in guide** for the setup and user experience for Interrupt Mode.

Once verified, users can register a passkey on their YubiKey, which can become their primary sign-in method removing the need for passwords or additional MFA during future sign-ins.

Pre-registration with the Yubico Enrollment Suite

Yubico offers the **Yubico Enrollment Suite** to delegate passkey registration through both IT-administered and automated workflows. This includes tools for pre-registering YubiKeys on behalf of users, as well as a service that delivers factory-programmed YubiKeys ready for immediate use.

- **YubiEnroll** is a client application that enables IT to pre-enroll YubiKeys on behalf of users.
- **Yubico FIDO Pre-reg** is a service that delivers factory-enrolled YubiKeys, pre-programmed with passkeys (FIDO2 credentials), for immediate use by new or existing users using YubiEnterprise Delivery.

Attestation and Key Restrictions

Attestation is a mechanism by which a YubiKey cryptographically proves its make and model at the time of registration. Microsoft Entra ID uses this attestation statement to validate that the device being registered is genuine and originates from a trusted manufacturer.

While Entra ID does not require attestation by default, enabling it allows organizations to collect and verify device metadata during registration.

This capability works in conjunction with key restrictions, a policy setting that allows administrators to enforce which FIDO2 security keys are allowed for registration, based on their AAGUID (Authenticator Attestation GUID). The AAGUID is specific to the make and model of the device and is consistent across YubiKeys that share the same capabilities and firmware. For organizations with compliance-driven requirements, Yubico offers a service to program YubiKeys with a custom AAGUID, enabling enforcement of customer-specific policies and key management.

By combining attestation and key restrictions, organizations can ensure that only approved YubiKeys are used for registration.

Yubico strongly recommends enabling attestation and configuring key restrictions to ensure that only genuine, enterprise-approved YubiKeys are used. This helps organizations meet internal policy requirements, mitigate supply chain risks, and support compliance with industry regulations that require verification of hardware authenticity and source.

Refer to the **Admin Deployment Guide** for instructions on enabling attestation and enforcing key restrictions.

Authentication Scenarios

After successful registration, users can authenticate using passkeys stored on their YubiKey for both browser-based and desktop both local and remote sign-ins.

- **Browser and Application Sign-in**

  Sign-in occurs using a supported browser or client app (e.g., Microsoft Edge or Outlook) to access services protected by Entra ID such as Office 365.

- **Windows Sign-in**

  Users can sign into Windows directly from the sign in screen or lock screen using their YubiKey. Once signed in, Single Sign-On (SSO) can be automatically provided to all Entra-protected resources when using Microsoft apps and browsers.

- **Remote Desktop Sign-in**

  Users can sign-in remotely using a supported Remote Desktop connection client for pre-session and in-session authentication scenarios.

Refer to the **User Enablement Guide** for sign-in experiences for local session scenarios and the **Remote Desktop Guide** for remote session sign-in scenarios.

Offboarding

YubiKey offboarding procedures ensure secure deprovisioning while supporting operational flexibility. Organizations may adopt one of two models when handling YubiKeys during user separation: **user-retained** or **reclaimed for reuse**.

To complete the offboarding process:

- **Delete the user's passkey (FIDO2 credential)** from their profile in Microsoft Entra using different methods including:
  - Entra Admin Center
  - PowerShell
  - YubiEnroll or other tooling leveraging Microsoft Graph API

User-Retained Model

In this model, the departing employee is permitted to retain the YubiKey. This approach is often favored in large-scale deployments due to the low cost of hardware and the reduced overhead of repurposing devices.

Note: There is no residual risk as long as the FIDO2 credential (passkey) associated with the user is deleted from Microsoft Entra ID. Without a registered credential, the YubiKey cannot be used to access any corporate resource.

Reclaimed Model

If the organization opts to reclaim the YubiKey for future use, the device should be reset to its factory state to remove any stored credentials.

Important: Resetting the YubiKey deletes all credentials, including any used for personal accounts. There is currently no way to restrict enterprise-issued YubiKeys from being used with non-enterprise services.

Resetting may occur:

- During the offboarding process
- Immediately prior to reissuance as part of your provisioning workflow

Key Management

Effective key management ensures users can securely maintain access while preserving the integrity of the YubiKey. This section covers PIN changes, device resets, and biometric fingerprint management for YubiKey Bio models.

PIN Management

All YubiKeys used for passkey (FIDO2) authentication require a PIN. This includes the YubiKey 5 Series, Security Key Series and YubiKey Bio models. The PIN serves as the local user verification method and plays a critical role in securing authentication.

- **Initial Configuration:**

  A PIN must be set before the YubiKey can be used for FIDO2 authentication. This can occur:

  - **As part of the passkey registration process**, when the user registers their YubiKey in Entra ID.
  - **Independently**, prior to registration, using supported tools. This is particularly relevant during provisioning workflows or when preparing a YubiKey Bio for fingerprint enrollment, which cannot proceed without a PIN.

- **Biometric Enrollment (YubiKey Bio):**

  For YubiKey Bio models, a PIN is required to enroll fingerprints and serves as the fallback verification method if fingerprint authentication is unavailable or fails.

- **PIN Changes:**

  Users may change their PIN at any time. This action requires user verification and requires knowledge of the existing PIN.

  A PIN change may be enforced on first sign-in if the key has been pre-registered by the administrator using the YubiEnroll utility or by the Yubico FIDO Pre-reg service.

Refer to the **User Enablement Guide** for instructions on PIN management.

Lost or Stolen YubiKeys

Even though a YubiKey is protected by a PIN or biometric verification, if a YubiKey has been lost or stolen the credential remains registered in Entra ID. To prevent unauthorized access, the associated passkey must be promptly removed from the user's profile.

Recommended Actions

In the event a YubiKey is reported lost or stolen:

- **Delete the passkey (FIDO2 credential)** associated with the device from the user's profile in Microsoft Entra ID as soon as possible. This can be performed using:
  - The Entra Admin Center
  - PowerShell
  - YubiEnroll or other tooling leveraging Microsoft Graph API
  - Via Self-service in the Security Info Portal
- **(Optional)** Revoke all active sessions for the user via the Entra Admin Center or PowerShell to enforce reauthentication.

  Yubico recommends that users register at least one backup YubiKey to ensure uninterrupted access in the event of loss or theft. Users have the ability to self-serve deletion of their lost YubiKey via the Security Info portal.

Recovery

Recovery is a critical component of YubiKey lifecycle management. It ensures that users can regain access to their accounts in the event of forgotten PINs, lost devices, or other disruptions. A well-defined recovery process minimizes downtime while maintaining security assurance. The following subsections outline key recovery scenarios and the procedures required to restore access.

Forgotten PIN

There is no recovery mechanism for a forgotten PIN on a YubiKey. If the PIN is forgotten, the only option is to perform a **key reset**. This is a destructive and irreversible action that deletes all stored passkeys and restores the device to its factory state.

After the reset, the user must re-register a new passkey for each service that was previously associated with the device.

Locked YubiKey (PIN Retry Limit Reached)

If a user enters the incorrect PIN too many times (8), the YubiKey will lock as a security measure against brute-force attacks. Once locked, the device cannot be unlocked and must be reset. This reset deletes all stored credentials and returns the key to its factory state. As with a forgotten PIN, the user must re-register their passkey after the reset is complete.

Key Resets

A YubiKey reset permanently deletes all stored FIDO2 credentials and biometric data (if applicable).

Key resets have the following characteristics:

- **No privilege restriction:** Any user with physical access to the key can reset it.
- **Must be performed locally:** Resets cannot be performed remotely.
- **Requires physical interaction:** The process involves reinserting the key and confirming with a touch gesture.
- **Irreversible:** Once initiated, the reset cannot be undone.

Refer to the **User Enablement Guide** for instructions on performing a key reset.

Re-registration

After a YubiKey has been reset or replaced, users must re-register a new passkey for each account or service previously associated with the device. This process re-establishes the binding between the YubiKey and the user's identity in Microsoft Entra ID.

There are two common recovery paths:

- Using a Backup YubiKey (recommended):

  If the user has a registered backup key, they can authenticate using the backup and re-register a new primary key via the self-service Security Info portal. This is the recommended approach for minimizing downtime and preserving productivity.

- Without a Backup Key:

  If no backup key is available, administrators may issue a Temporary Access Pass (TAP) to bootstrap the registration of a new YubiKey. TAP provides time-limited access and satisfies the multifactor authentication requirement for registering a passkey.

  If other passkey solutions are available, such as Windows Hello for Business, platform credentials on macOS, or passkeys stored in the Microsoft Authenticator app, these authenticators can be used to register a new passkey on a reset or replaced YubiKey.

Yubico strongly recommends **registering at least one backup key** per user to support seamless recovery, maintaining an elevated security posture and consistent user experience.

Usage Monitoring and Auditing

Monitoring how YubiKeys are used across the organization helps maintain a secure authentication environment, ensure compliance, and proactively identify issues. Microsoft Entra ID provides robust logging and auditing capabilities that administrators can use to track authentication activity and key lifecycle events.

Sign-in Logs

The Microsoft Entra sign-in logs capture each authentication event, including the method used. This allows administrators to:

- Confirm that users are signing in with passkeys (FIDO2 security keys)
- Detect fallback to less secure methods (e.g., password + SMS) if permitted
- Identify inconsistent behavior or potential misuse

Audit Logs

The Microsoft Entra audit logs record changes to user authentication methods, such as:

- Registration or deletion of passkeys
- TAP issuance events (if used)

This is useful for:

- Verifying successful onboarding or offboarding
- Confirming that lost or stolen keys have been revoked

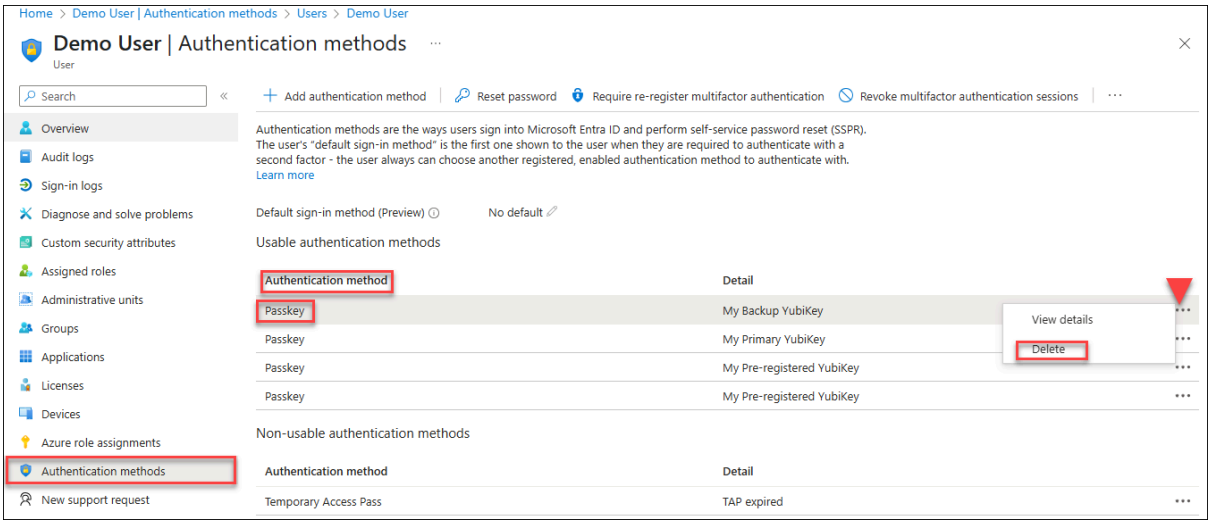
Delete a passkey (FIDO2)

Deleting a passkey is an unrecoverable action and requires the appropriate role based on the role of the target user. For self-service deletion refer to the **User Enablement Guide.**

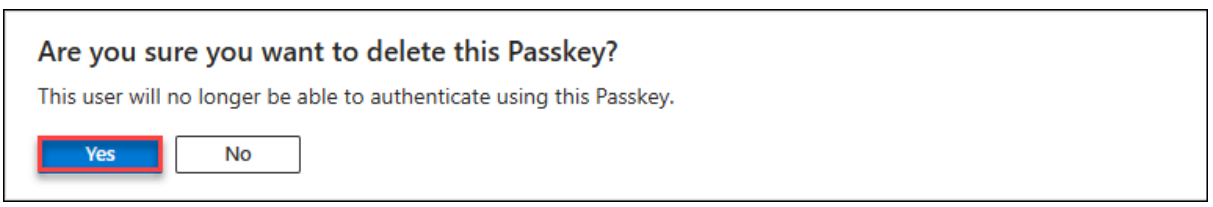
Action	Target User	Required Role(s)
Delete passkey	Standard user	-Authentication Administrator -Privileged Authentication Administrator
Delete passkey	Privileged user	-Privileged Authentication Administrator -Global Administrator

Option 1: Deleting a Passkey (FIDO2) via the Entra Admin Center

1. Navigate to the **Entra Admin Center**: https://entra.microsoft.com
2. Sign-in with an account that holds at the appropriate role based on the table above
3. In the left menu pane, expand the **Identity** blade and select **Users**
    a. Select the intended user, then select **Authentication methods** in the menu in the left menu pane:
    b. Under **Authentication method**, click on the **ellipsis** towards the right of the menu for the passkey you intend to delete
    c. Select **Delete**



    d. Select **Yes** to delete the passkey



    e. The passkey will be deleted and can no longer be used to sign-in

Option 2: Deleting a Passkey (FIDO2) via PowerShell

1. Launch **PowerShell** as an **administrator** and run the following commands modifying input values as needed:
2. List the user's FIDO2 authentication methods:

```
Get-MgUserAuthenticationFido2Method -UserId user@domain.com
```

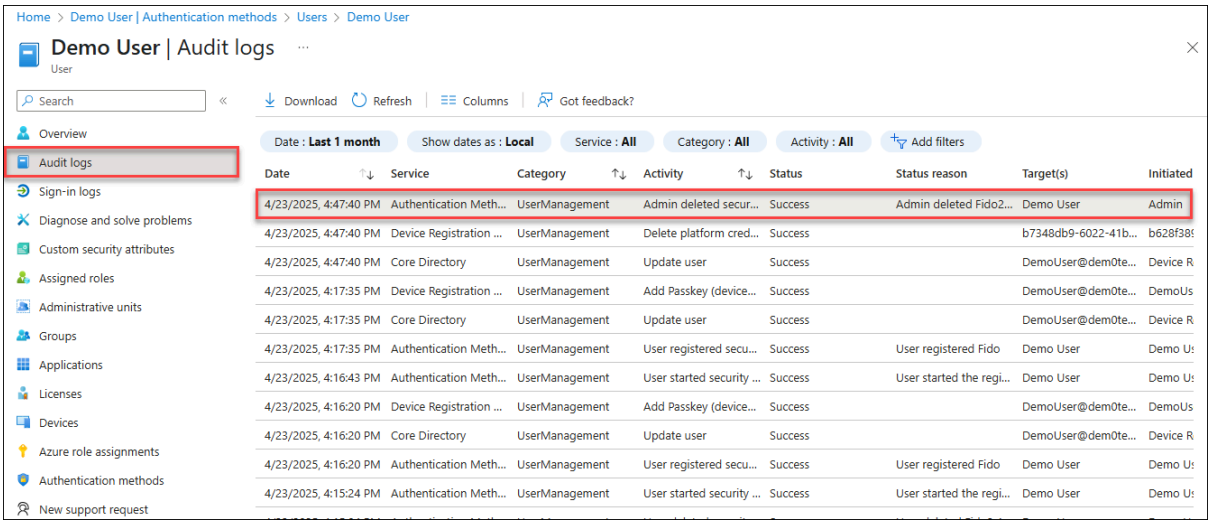
3. Delete a specific passkey (FIDO2) using its Id:

```
Remove-MgUserAuthenticationFido2Method -UserId user@domain.com
-Fido2AuthenticationMethodId <method-id>
```

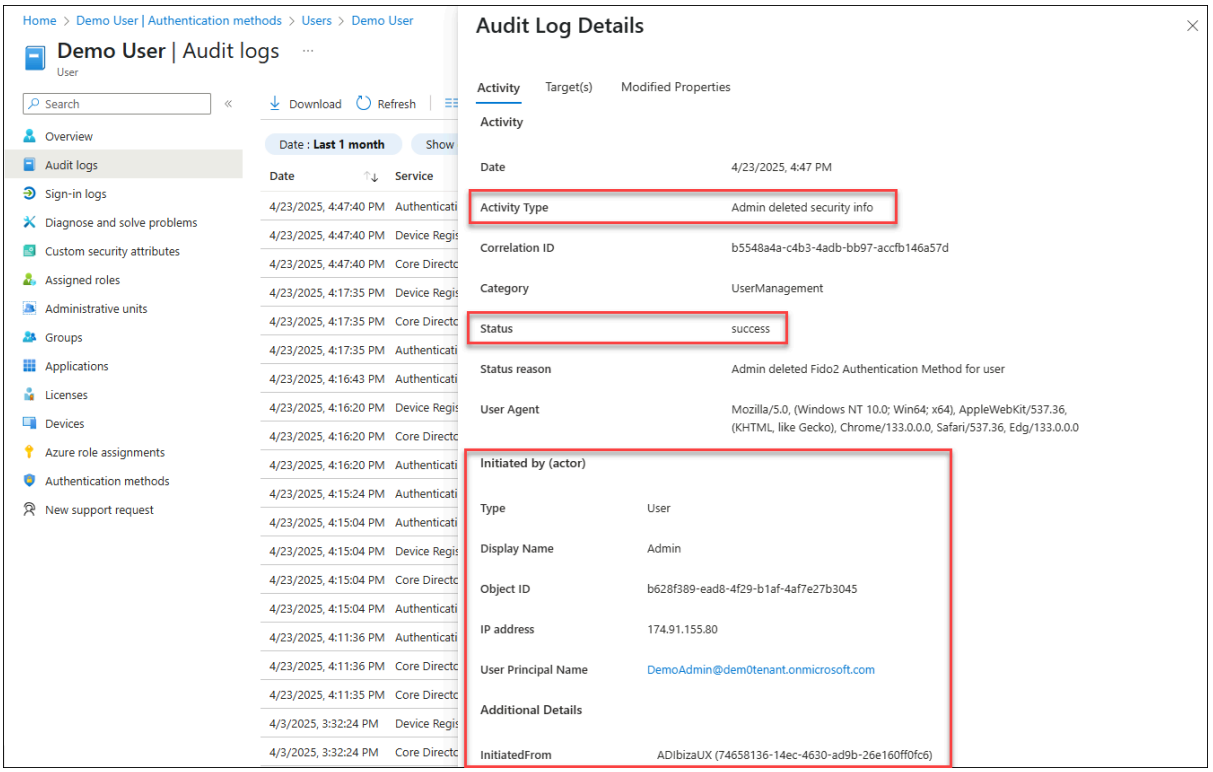
Confirm passkey (FIDO2) deletion

Option 1: Using the Entra Admin Center

1. From the user properties section in the left pane, select **Audit logs**
2. Select the event that corresponds to the deletion



3. Confirm the passkey has been successfully deleted under the **Activity** tab



Option 2: Using Microsoft Graph PowerShell

4. Launch **PowerShell** as an **administrator** and run the following commands modifying input values as needed:

```
# PowerShell Script: Verify if a FIDO2 passkey was deleted via
Entra ID audit logs

# Connect to Microsoft Graph with AuditLog.Read.All permissions
Connect-MgGraph -Scopes "AuditLog.Read.All"

# Set target user's display name and the audit log start date
$targetDisplayName = "Demo User"
$startTime = "2025-04-01T00:00:00Z"

# Retrieve audit logs where an admin deleted security info
$events = Get-MgAuditLogDirectoryAudit `
  -Filter "activityDateTime ge $startTime and
activityDisplayName eq 'Admin deleted security info'" `
  -All

# Filter for FIDO2 deletions for the specified user
$filtered = $events |
  Where-Object {
    $_.ResultReason -eq "Admin deleted Fido2 Authentication
Method for user" -and
    ($_.TargetResources | Where-Object {
      $_.Type -eq "User" -and $_.DisplayName -eq
$targetDisplayName
    })
  }

# Display the results in a table
$filtered |
  Select-Object ActivityDateTime,
                @{Name = "DeletedBy"; Expression = {
$_.InitiatedBy.User.DisplayName }},
                @{Name = "TargetUser"; Expression = {
($_.TargetResources | Where-Object Type -eq "User").DisplayName
}},
                ResultReason |
  Format-Table -AutoSize
```

You should receive an output with the following values:

ActivityDateTime	DeletedBy	TargetUser	ResultReason
Timestamp	<user>	<user>	Admin deleted Fido2 Authentication Method for user