Whitepaper

Okta Personal Technical Whitepaper

Updated Feb 2024

Okta Personal Key Security Features

- **Client Side Encryption** - Your data is encrypted on your local device with keys generated by your device. Okta never receives these local encryption keys. Okta, Okta Workforce Admins, and other third-parties are unable to decrypt the data received or stored on Okta's servers without these keys.
- **End-to-End Encryption (E2EE)** - Okta Personal encrypts your app data on your client so that only you can decrypt the data. Public key cryptography is used when sharing your App data with other Okta Personal Users.
- **Multi-Factor Authentication** - Sign in using multiple authentication factors such as possession + biometrics or possession + knowledge.
- **Passwordless Support** - Okta Personal supports authentication through email-based one-time passcode.
- **Seamless Vault Unlock** - Unlock your Vault with a secret key or a push notification to a Trusted Device.
- **Secure App Sharing** - Securely share your apps with other Okta Personal Users to whom you've given explicit consent. Revoke access to apps that you own at any time.
- **Scalable, Secure, and Reliable Infrastructure** - Okta Personal is deployed inside Okta's cell-based architecture. All communication with Okta servers occurs via Transport Layer Security (TLS).
- **Okta Software Development Lifecycle (SDLC)** - Okta Personal is developed in accordance with Okta's SDLC to reduce security risk.

Table of Contents

High Level Architecture **3**

Approach to Authentication **4**

Account Authentication 4

Unlocking your Vault 4

Encrypting the Vault **5**

Recovery Key 5

Derived Key 6

Vault Keys 7

App Keys 8

Sharing apps with others **10**

Inviting a recipient to a share 10

Accepting a share invitation 11

Revoking access to a shared app 12

Expired sharing invitation 12

Rejected sharing invitation 13

Other user actions **14**

Importing apps to Okta Personal 14

From other password managers 14

From Okta enterprise tenants 14

Exporting apps from Okta Personal 14

Recovering access to a vault **15**

Using Recovery Key to recover account 15

Lost Recovery Key 15

Okta Personal for Workforce **16**

What is a Personal App? 17

Linking work and personal accounts 17

Migrating apps to Okta Personal 17

Security Considerations **19**

Okta Browser Plugin 19

Permissions 19

Autofill 19

Context Syncing 19

Okta Software Development Lifecycle 20

External Reviews 20

Privacy 20

Cookies 21

Introduction

Okta Personal is Okta's first consumer identity manager. Okta Personal has a similar UI as Okta's workforce product, but they are fully separate product offerings by Okta.

Unlike Okta's workforce product, Okta Personal is owned and managed by the individual. Okta Personal leverages Client Side Encryption to perform all cryptographic operations on your local device. Data saved on Okta's servers can only be deciphered using your Recovery Key or a designated sharing key. Okta does not share your personal data without your explicit consent, and Okta does not sell your information to any third-parties.

If you have both Okta Personal and Okta work accounts, your Okta Personal data is yours, and your company's Okta Workforce Admins cannot access or view any data within your Okta Personal account.

This paper describes the design and controls that Okta uses to ensure that your data is secure, and is accessible only to you or to individuals you've given explicit consent to share your data with.

Region availability: Okta Personal is available globally, except in China and countries listed in the Okta IP Access Policy.

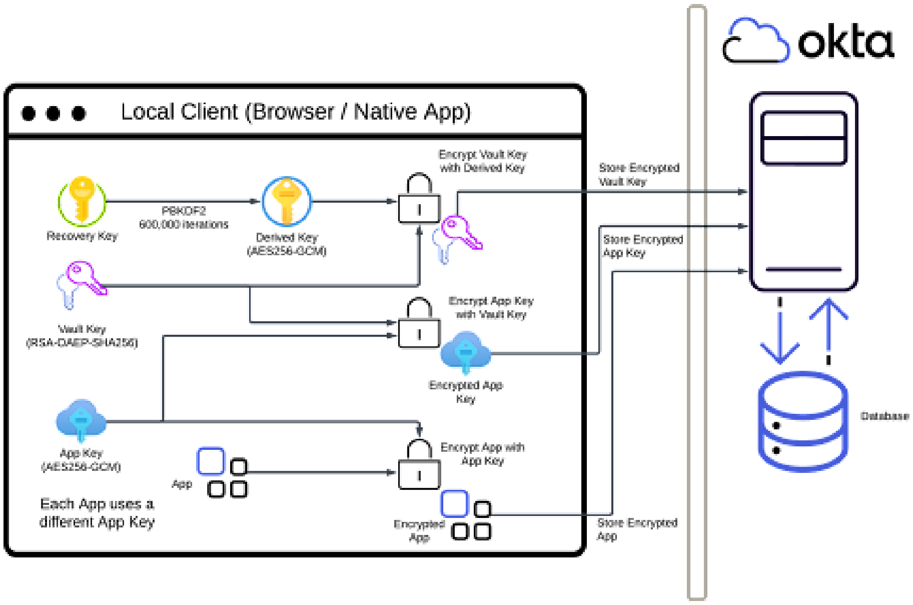
Mobile OS Support: Okta Personal currently supports iOS and Android.

High Level Architecture

Okta Personal is deployed inside of Okta's cell-based architecture. This allows Okta Personal to leverage overall Okta's compliance and hardened security posture.

Okta Personal consists of a client, browser plugin, and cloud service components. Okta Personal uses Client Side Encryption to encrypt your data prior to transmission and communication between your client and Okta's cloud services is further encrypted by TLS. The server sees only encrypted blobs of data.

- The **client** is either a mobile device or a supported web browser.[1]
- The **Okta Personal Dashboard** loaded from Okta servers is used to view, create, modify, and delete Apps. These Apps are similar to apps used by Workforce Identity Cloud.
- The **Okta Browser Plugin** supports enhanced flows such as autofill and saving Apps to your dashboard while browsing.



[1] Requires Okta Browser Plugin

Approach to Authentication

Okta Personal requires you to authenticate into your account and a separate authentication step to unlock your Vault. This prevents an attacker with access to your account from decrypting the contents of your Vault.

Account Authentication

The following methods are supported for accessing your account.

- Passwordless - A One-Time Password (OTP) is sent to your registered email. Access to this OTP demonstrates control of the specified email.
- Password-based - You authenticate using the email and password selected during registration. An additional authentication factor may be required.

Unlocking your Vault

The following methods are supported for decrypting the contents of your Vault.

- Trusted Device - A device which meets Okta's security requirements and required feature set. Trusted Devices support push based unlock.[2]
- Recovery Key - Randomly generated secret created at account creation. The Recovery Key is used to decrypt the contents of the User's Vault. See "Recovering access to a vault" section for more details.

[2] Requires iOS device running iOS 15+ with Secure Enclave and biometric enrollment or Okta Browser Plugin

Encrypting the Vault

Security is one of our top priorities. Okta Personal is designed from the ground up to maintain the confidentiality and integrity of your App data. All sensitive data is encrypted prior to transmission to Okta and Okta cannot decrypt this data. Okta uses industry-standard cryptography to accomplish this goal. Okta Personal generates and uses multiple encryption keys. The keys and their usages are described in this section.

What is End-to-End Encryption (E2EE)?

End-to-end encryption serves to decrypt data or messages on one device, send them to a recipient, and decrypt them on the receiving end. While in transit, the message cannot be read by anyone, including the server.

End-to-End Encryption (E2EE): Definition & Examples | Okta

Recovery Key

The Recovery Key is generated after account creation and consists of 24 random characters.[3] The plaintext Recovery Key is never sent to Okta. The key is stored locally by either the Okta Browser Plugin or the Okta Personal for iOS app. An encrypted version of this key is used for some onboarding flows. However, you have the ability to manually enter the key for new devices if desired.

Store your Recovery Key in a safe, accessible location in case you lose access to your devices. The Recovery Key is also used to bootstrap a new "Trusted Device".

[3] Uppercase Latin A-Z [\x41 - \x5A], and digits 0-9 [\x30 - \x39]

Derived Key

The Recovery Key is passed to a key derivation function (PBKDF2-SHA256 with a random 32 character salt and 600,000 iterations)[4] to generate a symmetric encryption key for use with AES256-GCM. The Derived Key remains on your local device and is never sent to Okta. The Derived Key is used to encrypt your private Vault Key.

How does PBKDF2 affect bruteforce attacks?

Passing the Recovery Key through PBKDF2 increases the amount of work a computer must perform and therefore reduces the number of attempts per second.

Imagine choosing a number such as 193. Now add 277 to it and multiply the result by 107. This would be one iteration of the algorithm. Repeat this another 599,999 times. This final result is the "derived key" used for encryption. The actual PBKDF2 algorithm does something similar. PBKDF2 accepts a customizable number of iterations, which can be increased in the future.

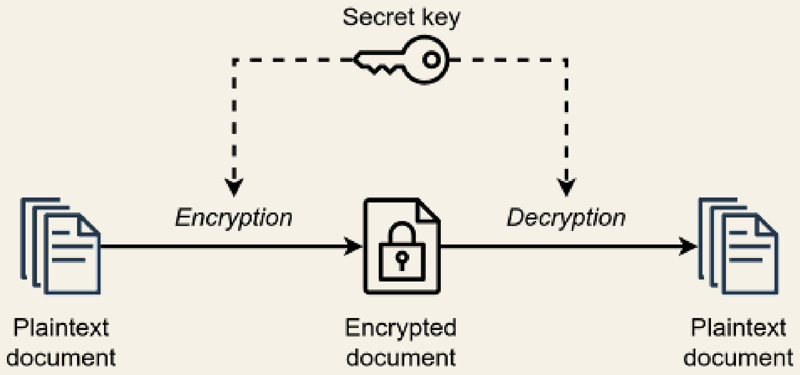
Given an input Recovery Key, it's not feasible to determine what the final Derived Key is without performing all the iterations.[5]

[4] PIN based versions of Okta Personal used 100,000 iterations instead of 600,000.

[5] Technically, an adversary could precompute all outputs for PBKDF-SHA256(SKEY, salt, 600,000, 256), but bruteforcing the AES key would be faster at that point.

What is symmetric encryption?

You can think of symmetric encryption as a keyed lockbox. Anyone with the key can lock / unlock (encrypt / decrypt) the box. The contents of the box are only viewable if you have the key. Symmetric encryption uses the same key for encryption and decryption vs asymmetric encryption which is discussed later.[6]



Vault Keys

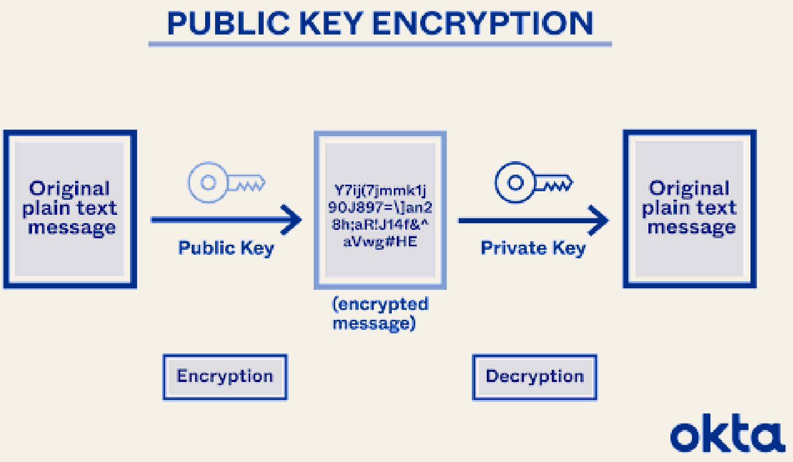
A RSA-OAEP keypair is generated during account creation to facilitate sharing. The public key and an encrypted version of the private key are stored on Okta's servers. The private key is encrypted with the Derived Key. It is not possible to decrypt the encrypted private key without access to the Derived Key. The public key is used to encrypt App Keys.

[6] https://commons.wikimedia.org/wiki/File:Simple_symmetric_encryption.png

What is RSA?

RSA is a public-key cryptosystem which uses a public and a private key. The public key may be freely shared. Anyone can encrypt data with the public key and only people with the private key can decrypt the data. See Public Key Encryption: What Is Public Cryptography? for more information.

You can think of RSA as a mail dropbox with an open slot. Anyone can place an item into the slot, but only the owner with the key can view the contents placed in the dropbox. There are some caveats with public-key cryptosystems such as how do you know who owns the key (dropbox) and what can the private key be used for aside from opening the dropbox (decryption). However, that is outside the scope of this document.


PUBLIC KEY ENCRYPTION
Original plain text message
Public Key
Y7ij(7jmmk1j 90J897=\]an2 8h;aR!J14f&^ aVwg#HE
(encrypted message)
Private Key
Original plain text message
Encryption
Decryption
okta

App Keys

These are AES-GCM keys used to encrypt / decrypt App entries. These keys are stored on Okta's Cloud Services encrypted using your public Vault Key. Your private Vault Key is required to decrypt your App Keys.

A different App Key is used for each of your Apps. This provides a seamless way to share a specific App without exposing your other Apps. Sharing involves encrypting an App Key with the recipient's public Vault Key. Only the person possessing the corresponding private Vault Key can decrypt the App Key.

How do I determine the recipient's public Vault Key?

The recipient's public Vault Key is stored on Okta's Cloud Services and associated with their email address. Okta handles email lookups and returns the corresponding public Vault Key.

Sharing apps with others

Okta Personal Users may share their Apps with other Okta Personal Users through the Sharing UI. End-to-End Encryption is used by the Sharing flows to prevent the server from eavesdropping on data sent between the Sharer and the Recipient.

What is End-to-End Encryption (E2EE)?

End-to-end encryption serves to decrypt data or messages on one device, send them to a recipient, and decrypt them on the receiving end. While in transit, the message cannot be read by anyone, including the server.

End-to-End Encryption (E2EE): Definition & Examples | Okta

Inviting a recipient to a share

The underlying process for sharing Apps involves more than a simple copy-and-paste due to the encrypted nature of the data. Your name and email are included with the share notification. The following steps occur when an Okta Personal User decides to share on of their Apps.

Actors

- App Owner / Sender - The Okta Personal User who owns an App and initiates the Share request for that App[7]
- Recipient - The Okta Personal User who will receive access to the App

[7] It's not currently possible to request access to an App. The App owner must be the one to initiate the Share request.

Process

1. The Sender's client provides the Recipient's email address and Okta's Cloud Services queries for the associated public Vault Key.
2. The Sender requests the App and the encrypted App Key.
3. Okta Personal verifies that the Sender has access to the App. An error is returned if they do not have access.
4. The Sender decrypts the App Key.
5. The Sender uses the Recipient's public Vault key to encrypt the decrypted App Key.
6. The Sender uploads the encrypted App Key to Okta Personal.
7. Okta Personal notifies the Recipient that they have a new pending share. Access to the pending share is restricted to the Recipient as an additional defense, although the contents may only be decrypted with the Recipient's private Vault Key. A share is valid for seven days.

Accepting a share invitation

The actors are the same as from "Inviting a recipient to an app".

The Recipient's name is shared with the App Owner upon acceptance.

Process

1. The Recipient accepts the share request on the site.
2. Okta Personal verifies that the Recipient has access to the share[8]. An error is returned if they do not have access.
3. The Recipient downloads the Share entry.
4. The Recipient uses their private Vault key to decrypt the Share entry. This entry contains the App Keys for the shared App.

[8] It’s possible that the Recipient hasn’t refreshed their browser and the Sender has revoked the Share in the meantime.

5. The Recipient requests the shared App.
6. Okta Personal verifies that the Recipient has access to the App. An error is returned if they do not have access.
7. The Recipient decrypts the App using the App Key from step 4.

Revoking access to a shared app

The App Owner may revoke access at any time.

Process

1. App Owner clicks "Cancel / Revoke" under the App details.
2. Okta Personal verifies that the requestor owns the App. An error is returned if the User has insufficient permissions.
3. Okta Personal deletes the Share entry and access grant for the specified Recipient. The App will no longer appear on the Recipient's dashboard

Note: Revoking a share only prevents future access to the App. The Recipient may have copied the decrypted App contents prior to revocation. Please change your password / rotate secrets if this is an issue. Okta cannot prevent a malicious Recipient from abusing App contents.

Expired sharing invitation

The current share invitation expires after seven days. If the sharing invitation expires prior to acceptance, then the Recipient will be unable to accept the Share. The App Owner will need to re-share the App.

Rejected sharing invitation

If a share invitation is rejected, then the Share notification will be removed from both the App Owner and Recipient's dashboard.

Other user actions

Importing apps to Okta Personal

From other password managers

Okta Personal supports app import from other password managers, including:

- 1Password
- Bitwarden
- Chrome
- Dashlane
- LastPass

Encryption is performed locally similar to manually adding an App. Okta never receives an unencrypted version of your passwords.

From Okta enterprise tenants

If an Okta Workforce Admin enables Okta Personal for Workforce for their enterprise, then an employee's Personal Apps can be migrated from their Okta work account to their Okta Personal account. See "Okta Personal for Workforce" section to learn more.

Exporting apps from Okta Personal

Users may export their Apps at any time. Note that the exported App data is not encrypted. Please take precautions to ensure the data is inaccessible to unauthorized users.

Recovering access to a vault

Using Recovery Key to recover account

The Recovery Key generated during account creation may be used to regain access should you lose access to your Okta Personal account and Vault. The plaintext Recovery Key is never sent to Okta, and Okta is unable to assist with the decryption of the Recovery Key.

To locate your Recovery Key in either your Okta Personal dashboard or mobile app, please see Find your Recovery Key.

Lost Recovery Key

If you lose access to your Recovery Key, Okta Personal cannot provide you with access to your account and any saved data, even with your permission. In this case, you will need to reset your account. This deletes any existing Apps in your Okta Personal Vault and you will need to re-add or re-import Apps.

If you cannot locate your Recovery Key or need to reset your account, please reach out to okta_personal@okta.com using the email address associated with your Okta Personal account.

Store your Recovery Key in a safe, accessible location in case you lose access to your devices. The Recovery Key is also used to bootstrap a new "Trusted Device".

Okta Personal for Workforce

Okta Personal for Workforce is currently available in Early Access.

Admins may enable Okta Personal for Workforce through their Okta Admin Console under **Settings > Features > Early Access.**

At General Availability, Okta Personal for Workforce will be default-on for Workforce Identity Cloud customers.

Migrating Personal Apps from Okta enterprise tenant

Okta Personal for Workforce is a set of features that integrates Okta Personal (a consumer offering) with the Okta Workforce Identity Cloud. These features allow end users to:

- Migrate personal apps from an employee's Okta work account to a free Okta Personal account
- Separate work and personal data for good security hygiene

Okta Personal for Workforce allows an organization's employees to use and switch between their Okta work and personal accounts in the same, familiar UI - while keeping the accounts separate.

Migrating Personal Apps is a multi-step process:

1. Install Okta Browser Plugin - The Okta Browser Plugin is used to display work and personal accounts. The plugin also facilitates the retrieval and transfer of Apps during migration using standard Okta APIs.
2. Linking accounts - An employee must link their Okta work account and Okta Personal account. Linking Okta accounts allows the employee to see apps on their work dashboard that are registered with a personal email address.
3. Migrating apps to Okta Personal - An employee may select the apps that they want to move. Only eligible Personal Apps may be migrated, see "What is a Personal App?". App import is a one-way process. After the import, the employee can’t restore the selected Personal Apps back to their Okta work account.

Okta Workforce Admins **cannot** see migrated Apps. These Apps will no longer exist in the employee's work account.

What is a Personal App?

Personal Apps are unmanaged Apps from the Okta App Catalog which are added by an employee after an Okta Workforce Admin enables the "Allow users to add personal app integrations" feature.[9] Admins can find a list of all Personal Apps by viewing the "User Added Applications" Pane.[10]

Not all Personal Apps may be migrated. Personal Apps with a username that matches the logged-in user or a username from the Workforce Identity Cloud domain are excluded by default. Admins may exclude additional domains from Personal app migration feature through the Okta Admin Console. See Okta Personal for Workforce for more information.

Linking work and personal accounts

Before an employee can move apps from their Okta work account, they will need an Okta Personal account and to have installed the Okta Browser Plugin.

Linking accounts enables the Personal Apps import feature and shows which apps are eligible for migration. The plugin must recognize both the employee's Okta work account and the user-created Okta Personal accounts in order to migrate apps.

Migrating apps to Okta Personal

Personal Apps import works by retrieving the selected Personal Apps from the Enterprise org, encrypting the contents locally, then uploading the Personal App data to Okta Personal. Only apps that are associated with usernames that belong to non-excluded domains can be migrated to Okta Personal.

[9] Enable self-service request feature
[10] Self-service for app integrations

The following steps are performed during the migration:

1. End users: obtain a list of Personal Apps in their Okta work account that are eligible for migration
2. End users: select which Personal Apps they wish to migrate
3. Retrieve credentials for selected Personal Apps
4. Verify that the Okta Personal Vault is unlocked
5. Encrypt credentials for each Personal App and upload to Okta Personal
6. Remove the selected Personal Apps from the end user's Okta work account

Security Considerations

Okta Browser Plugin

Permissions

The Okta Browser plugin requires permissions to interact with web pages that you visit. For a full list of permissions and their purpose, please see Okta Browser Plugin permissions for web extensions.

Okta Personal does not require additional permissions from those listed above.

Autofill

Okta Personal uses the standard autofill mechanisms from the Okta Browser Plugin. When you click an App on your dashboard, the Okta Browser Plugin navigates to the specified URL, autofills your username and password, and logs in on your behalf.

The Okta Browser Plugin implements strict URL matching to prevent leaking your credentials to phishing pages. See Security features of the Okta Browser Plugin.

Context Syncing

The Okta Browser Plugin uses browsers runtime API to communicate between the Okta Personal dashboard and the plugin. This provides enhanced functionality such as syncing your authentication context and keys, i.e. you may log into Okta Personal through either the plugin or webpage and your authentication status is reflected in the other.

Okta Personal for Workforce Migration

The Okta Browser Plugin has access to both your work and Okta Personal accounts after linking. The plugin serves as a bridge to retrieve and migrate Personal Apps from your work account to your Okta Personal account.

Okta Software Development Lifecycle

Okta Personal participates in Okta's Software Developer Lifecycle which involves architectural reviews, testing, system hardening, and security guidance.

External Reviews

Okta engages third party security auditors to review and perform penetration testing of the Okta Personal product. The results of these reviews are available upon request.

Privacy

Okta Personal processes your personal data in accordance with the Okta Privacy Policy. Okta Personal collects metadata such as requestor, requestor IP, and URLs for diagnostic and metrics purposes. Okta Personal does not log the underlying data such as cryptographic keys, plaintext or ciphertext associated with a request.

Okta Personal supports data subject rights.

- Okta Personal Users can request their account and data to be fully deleted from Okta systems. To delete your account and any saved information in your account, please email us at okta_personal@okta.com using the email associated with your Okta Personal account.
- If you would like to make a data subject request to exercise your rights pursuant to applicable privacy laws, please complete Okta's data deletion form.

Cookies

Okta Personal uses cookies as defined by our Cookies Policy.

Any products, features or functionality referenced in this material that are not currently generally available may not be delivered on time or at all. Product roadmaps do not represent a commitment, obligation or promise to deliver any product, feature or functionality, and you should not rely on them to make your purchase decisions.