MindLink Mobile

Technical Overview

Version 19.3

Table of Contents

1 Overview

MindLink Mobile enables MindLink functionality – including Microsoft Skype for Business Instant Messaging ("IM"), Presence and Persistent Chat ("PChat") – on a range of cross-platform mobile devices, and inside a number of Mobile Application Management ("MAM") containers.

1.1 Mobile Platform Support

MindLink Mobile clients are available for the following devices:

- Android – phone and tablet
- Apple iOS – iPad and iPhone

The application is offered in a version that runs natively and unmanaged on each of the above devices – referred to in this documentation as the "vanilla" version of the application. The Android application will run as a managed app on a device enrolled with Android for Work.

In addition, versions of the application are offered that run inside the following MAM containers:

- BlackBerry Dynamics – iOS, Android
- MobileIron – iOS only
- AirWatch – iOS only

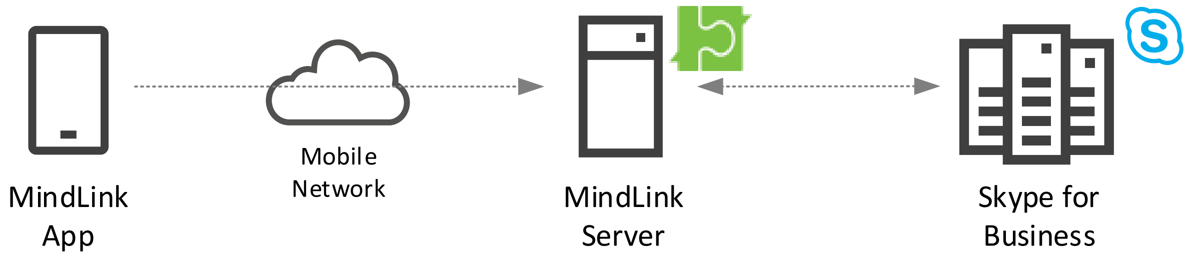
All versions of the application follow the same core architecture and offer the same set of features, unless otherwise indicated.

This document describes this core architecture and highlights differences for each mobile OS and MAM container where appropriate.

1.2 High-level Architecture

To enable connectivity to the Skype for Business components, an organization must deploy the MindLink Mobile server within their internal IT infrastructure.

A single MindLink Server can serve clients connecting from all supported device platforms.



The MindLink Mobile server performs the following responsibilities:

- Hosts the MindLink Foundation, which coordinates the core MindLink functionality and communicates with Microsoft Skype for Business as the underlying backend.
- Handles connections from mobile clients.
- Maintains session state across mobile network disconnections and acts as an intelligent buffer for updates to be sent to clients.

- Sends push notifications to mobile devices.

The MindLink Mobile server is a .NET application that runs as a Windows Service. The host Windows Server machine can be virtualized. The server component is installed by running a standalone .MSI executable and then using a graphical management utility application to configure the system.

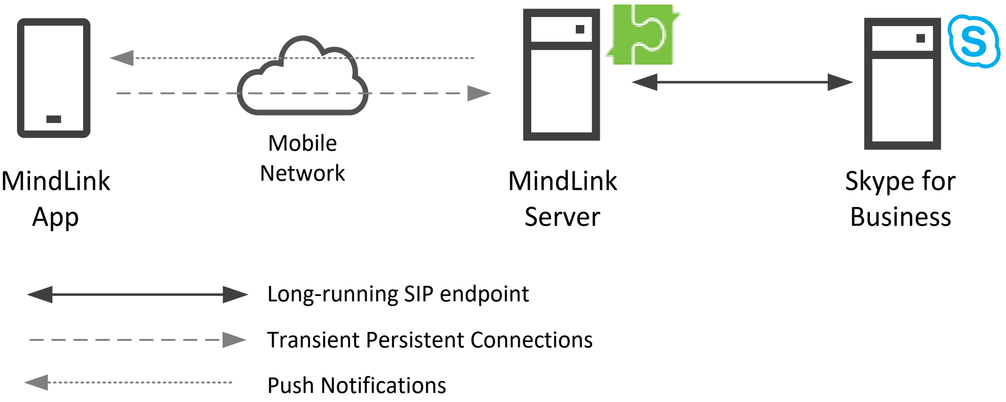
The MindLink Mobile server exposes a number of performance counters with which to monitor its load and network traffic.

2 Application Lifecycle

MindLink provides the user with an always-on mobile Skype for Business endpoint. The MindLink Server maintains the Skype for Business endpoint on behalf of the user, even when the user is not actively using the client application. This allows the user to receive push notifications of new messages when not using the application.

We refer to the lifetime of the server-side Skype for Business endpoint as the "session".

When the user is actively using the application, the application establishes a two-way connection with the server and receives updates of new messages immediately.



2.1 Configuration Bootstrapping

When the application starts up without a pre-existing session, it makes a one-time HTTP call to the MindLink server web address.

The user is prompted to enter this server address upon opening the app for the first time, and is able to change the address between sessions. Alternatively, many flavours of the application can be pre-configured with the MindLink server address. In this configuration, the user need not take any action to initially configure the app, and cannot change the server address between sessions.

The MindLink server responds with basic configuration about how further connectivity will be managed, and the capabilities enabled by the administrator. Having received this configuration, the app displays the log on screen.

2.2 Logging On

The logging on process happens once at the start of the MindLink session. Since MindLink sessions are long-running and persist across application or device restarts, this process happens relatively infrequently.

Once logged on, the client application is issued by the server with a one-time token which it subsequently uses to re-establish the session when required.

To log on, a user must manually enter their Active Directory credentials – the account name, and the password. The credentials are sent over a secure connection to the MindLink Server.

Alternatively, many flavours of the application can be pre-configured with a log on name, obtained from the account associated with the MAM container, or otherwise. In this configuration, the user only needs to supply their password, and cannot change the account associated with the application.

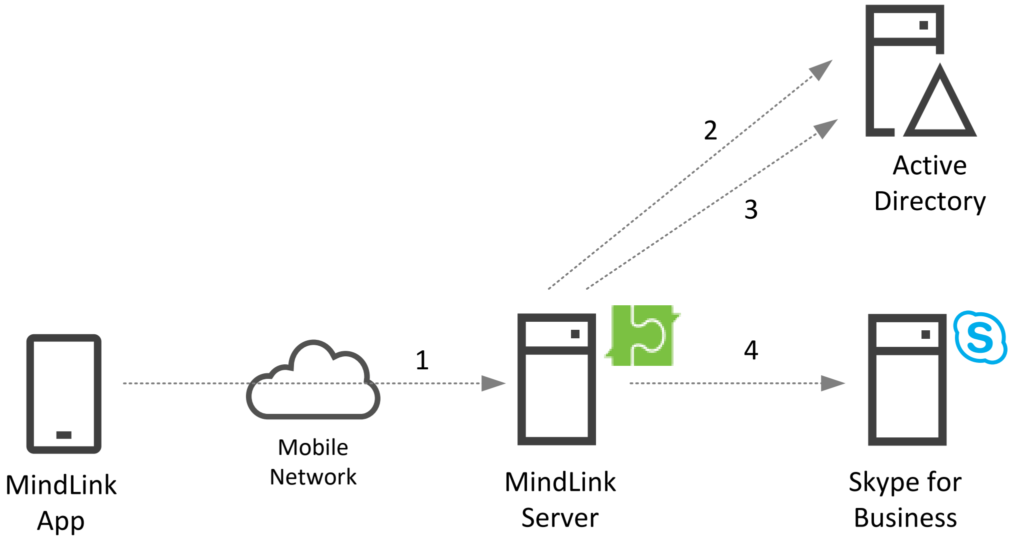
2.2.1 Skype for Business Server

The entered credentials should correspond to the user's enabled user account. In a "resource" or "central" forest Skype for Business deployment, the credentials of the linked user account should be entered.

The account name can be supplied in a down-level or UPN format. Explicit UPNs with UPN suffixes are supported for accounts in the same forest as the MindLink Server.

The MindLink Server authenticates these credentials with Active Directory by performing a fast bind with an LDAP server in the user account's domain. Active Directory in the Skype for Business forest is then queried via the Global Catalog to obtain the user's SIP address.

The SIP address is then used to establish the connection with Skype for Business. Since the MindLink Server is trusted by the Skype for Business infrastructure, neither the user name nor password is forwarded to Skype for Business directly.

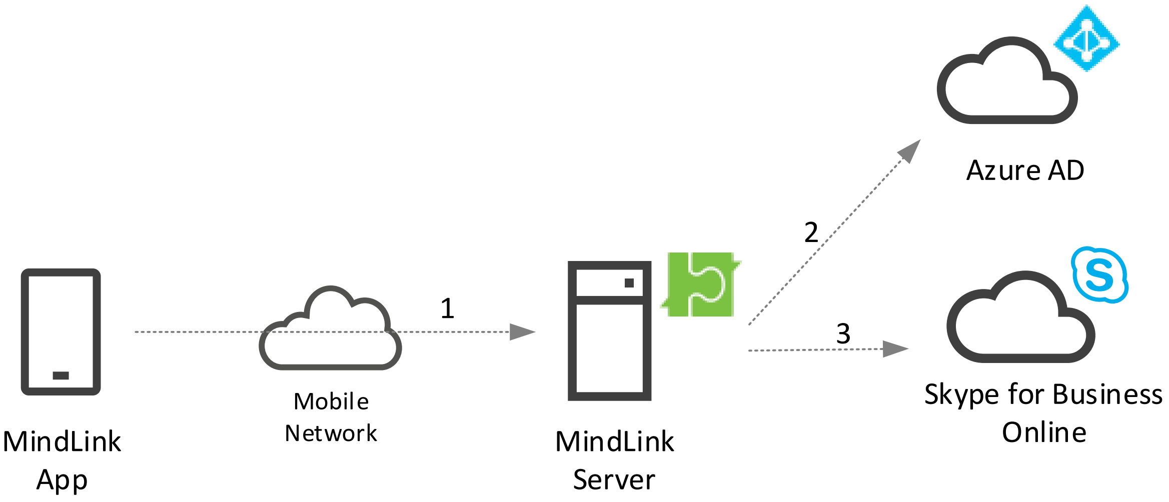


1) Client sends user name/password to server.
2) Server authenticates with fast concurrent bind to Active Directory LDAP server.
3) Server queries for SIP address from Active Directory Global Catalog.
4) Server establishes Skype for Business endpoint with SIP address using trusted connection.

2.2.2 Skype for Business Online

The entered credentials should correspond to the user's O365 account. The MindLink server authenticates the credentials against Azure Active Directory in Exchange for access to Skype for Business Online resources on behalf of the user.

1) Client sends credentials to server.
2) Server authenticates credentials in exchange for on-behalf-of token against Azure AD for Skype for Business resources.
3) Server establishes Skype for Business Online endpoint for O365 account.



2.3 Connection Lifecycle

When a session has been established it continues to persist (including maintaining the connection to Skype for Business) until:

- The user manually logs off.
- The user manually logs on as a different user on the same device, or uninstalls and reinstalls the application.
- The Skype for Business endpoint is disconnected due to an unrecoverable error with the Skype for Business infrastructure.
- The user is disabled on the Skype for Business system.
- The administrator has configured a session expiration interval and the user has not been active on the app for that interval.

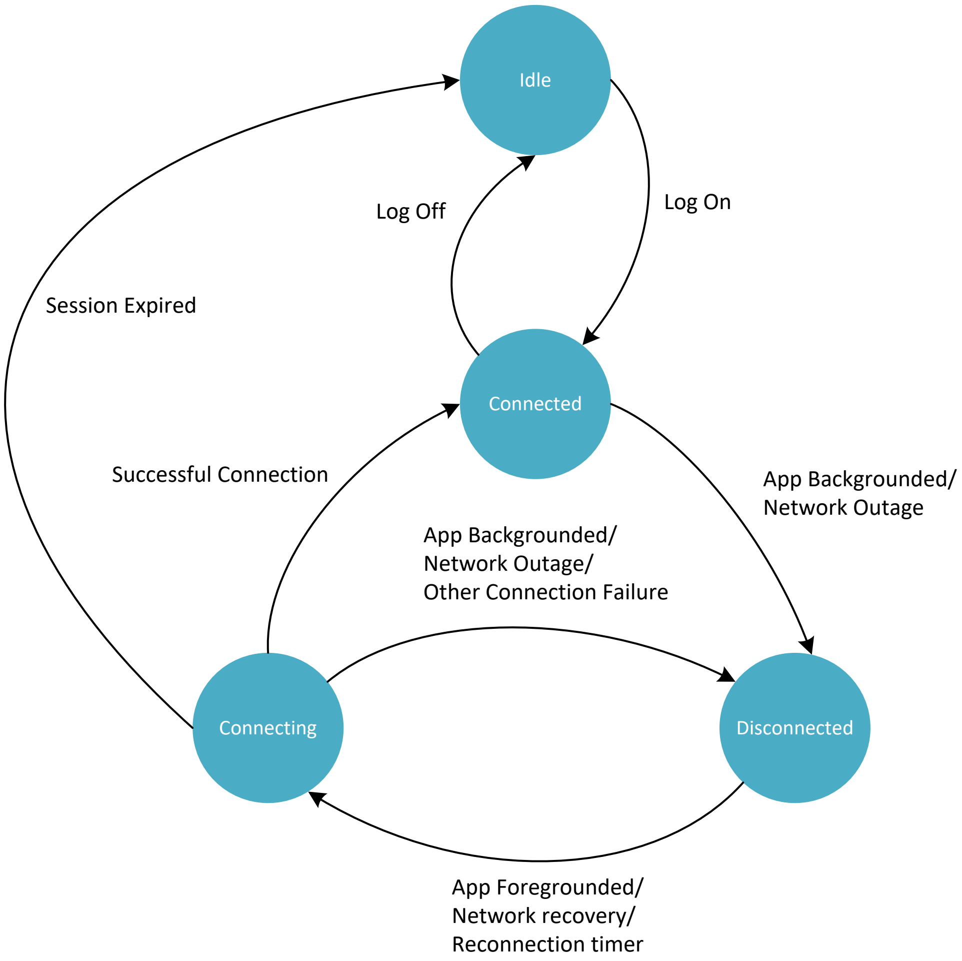
The session is always in one of two modes:

- **Connected to the client via a persistent connection**
  - The application will attempt to establish a two-way "persistent" connection to the server whenever it is in the foreground on the device screen.
  - The application will report itself as "connected" when in this state.
  - The user will be able to interact with the application including changing their profile, loading new messages, searching for content, and changing their presence.
  - The user will receive new messages and updates (e.g. presence state) immediately.
  - The client and server minimize data usage by using a subscription-based "client virtualization" protocol to negotiate only sending the data that is actually required on the device.
- **Disconnected from the client**
  - The application will report itself as "disconnected" in this state.

- The session will transition to this state when in the background on the device, the device is locked, or the application is forcefully "closed" by the user.
- The client application uses no network data and no CPU time in this state. The operating system manages lightweight marshalling of inbound push notifications.
- The user will receive a push notification when a new message in a chat room or IM conversation is received.
- The MindLink Server acts as buffer to any new messages or updates that are received in this time, ready to send them down to the device when it next reconnects.

Management of this lifecycle is automatic. The client disconnects when put into the background on the device and will immediately reconnect when brought into the foreground. The user will see a "Reconnecting…" message while the client reconnects to the server when the app is opened or brought into the foreground. Disconnection of the client when not in use saves battery and network usage.

The server records the last time that a user was connected. The administrator can define a policy such that sessions that have not been connected for a given amount of time are automatically disconnected.



2.4 Persistent Connectivity

When the client reports itself as connected, it is maintaining a continuous two-way "persistent" connection to the server. This connection allows the device to send and receive real time updates. If the connection is dropped due to bad network connectivity, then automatic reconnection will take place. A connection attempt is made every 1 second with a timeout of 4 seconds.

2.5 Push Connectivity

When the client is not connected – but the server is maintaining the user's session – new messages will be sent to the device as a push notification.

The notification signals to the user that new messages are available and that they should open the MindLink application to fetch and read them. The information sent with the notification is typically enough to indicate in which conversation new messages are available, but does not contain the full message text. The exact type of information sent about each message can be configured by the administrator.

Notifications are sent via the native push notification infrastructure for the corresponding device OS. On the device, notifications are added to the native OS notification center/hub and may trigger a sound of vibration (depending on the operating system).

The user can configure which types of messages will trigger a notification by setting the notification settings inside the app. In addition, they have the option to completely disable push notifications.

2.6 Stored Data

MindLink Mobile has been designed as a thin, or "stateless" client. This means that the application only holds session state – including message content – in memory, and only while the application is running.

When the application is opened, it connects to the server and downloads the necessary session state fresh from the cached data that the server is holding. This data persists in memory on the device while the application is open (foreground or background), and is purged when the application is closed or the device reboots.

When not running, the application will only store the following data at-rest on the device's permanent storage:

- One-time session token – Used to reconnect to the long-running server-side session between application or device restarts.
- Log-on user settings – Used to store user preferences as selected on the log-on screen.
- User journey state – Used to store the progress of the user through the tutorial workflow that appears when the application is first used.

Message data is not stored on the device at any time.

2.7 Data Loss Prevention

The app respects DLP MDM controls enforced by the device operating system and managed by any EMM vendor.

Additionally:

1) The administrator may configure whether copy/paste of message content out of the app is enabled or disabled. This is applied globally to all logged on devices.
2) The administrator may apply managed key/value configuration keys to fine-tune interaction with the OS and other apps.

DLP controls are discussed in greater depth later in the document.

3 Skype for Business Integration

MindLink uses Skype for Business as the engine for the core functionality. Messages sent on MindLink can be received by Skype for Business users using any compatible client, and vice versa.

3.1 Supported Versions

MindLink requires an on-premise Skype for Business Server deployment or a Skype for Business Online tenancy.

For Hybrid topologies, MindLink must be deployed in the on-premise topology to be utilized only by users homed on-premise, and must be additionally deployed against the O365 infrastructure to serve users homed in the cloud.

3.1.1 Skype for Business Server

For chat room ("persistent chat") functionality, Persistent Chat servers must be deployed within the topology. For Lync 2013 and later, users who need to log on to MindLink to use chat rooms must be enabled for Persistent Chat via the Persistent Chat Policy.

The following versions of Skype for Business Server are supported:

- Microsoft Lync 2013 – with Persistent Chat optionally deployed and users enabled for Persistent Chat.
- Microsoft Skype for Business Server 2015 – with Persistent Chat optionally deployed and users enabled for Persistent Chat.

A mixed version topology is also supported – for instance using a Lync 2013 Persistent Chat server in a Skype for Business 2015 topology.

3.1.2 Skype for Business Online

An active O365 tenancy is required to use MindLink against Skype for Business Online. All MindLink users must be enabled for Skype for Business Online.

3.2 Connectivity

3.2.1 Skype for Business Server

The MindLink Server connects to the Skype for Business infrastructure via SIP as a trusted application.

The trusted connection allows the Skype for Business infrastructure to treat the MindLink Server as an equal peer and enables efficient routing of SIP traffic to and from the MindLink Server. The establishment of this trust requires that the Skype for Business servers be able to resolve the DNS name of the MindLink Server.

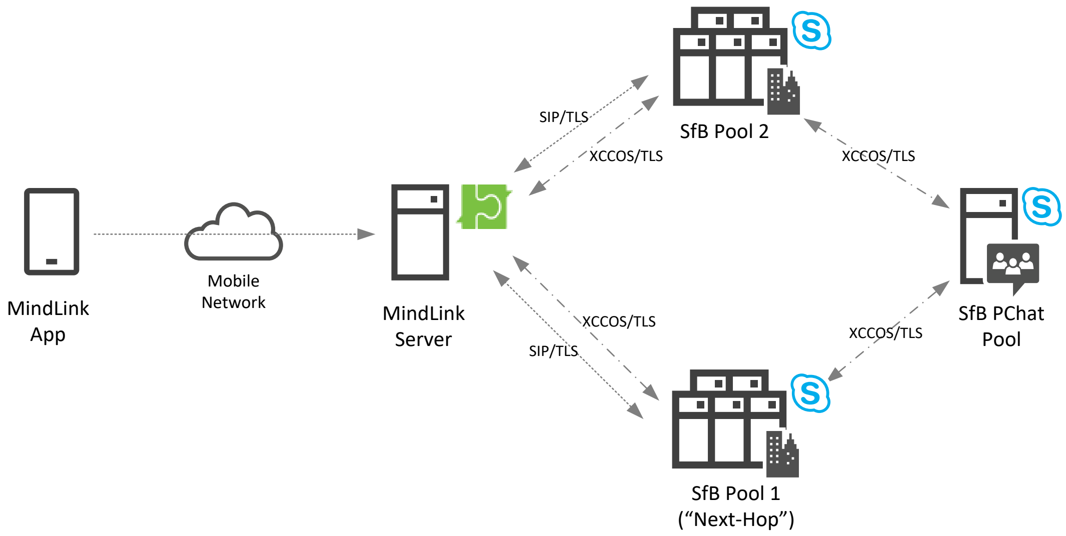
Configuration of this involves:

- Creating a trusted application pool containing the MindLink Windows host machine in the Skype for Business topology.
- Adding MindLink as a trusted application on the pool.

- Creating and assigning a certificate to the MindLink Server to establish trust with the Skype for Business servers.

A Skype for Business trusted application – in this case, the MindLink Server – must be configured with a "next-hop" Frontend pool. This is the Skype for Business frontend pool to which any initial connection will be made.

A MindLink user may be homed on any frontend pool in the Skype for Business infrastructure – the MindLink Server will subsequently connect directly to the necessary home pool to register each user. As such, a single MindLink Server may serve any Skype for Business user in the topology, subject to scale and geolocation decisions. It is generally recommended that the MindLink Server be located as physically close to the end users as possible.

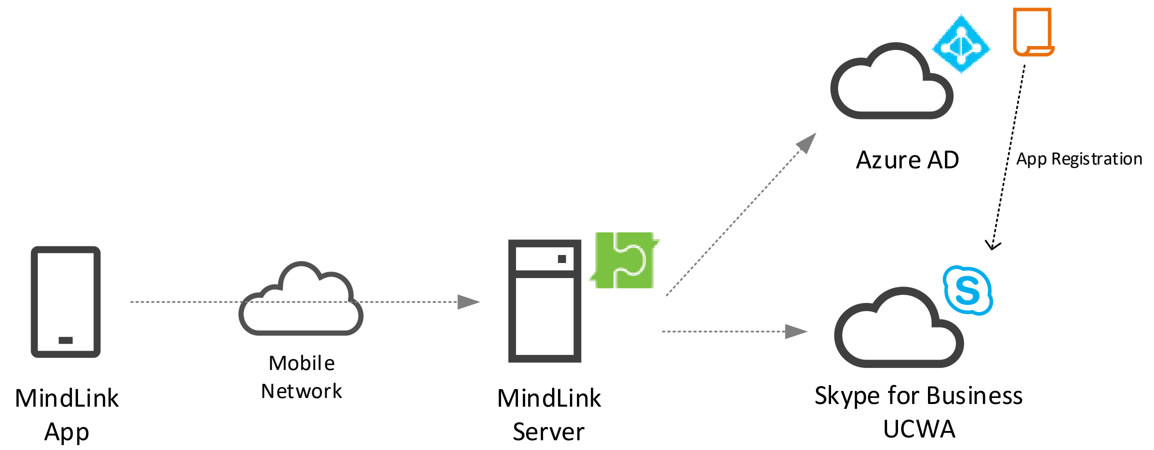


However, a single MindLink installation can only support one Persistent Chat pool. If there are multiple Persistent Chat pools, multiple MindLink Servers must be deployed.

3.2.2 Skype for Business Online

The MindLink Server must be deployed on a Windows server that can be running in an on-premise or cloud data center. The MindLink Server connects to the Skype for Business infrastructure over HTTP via the internet to the UCWA API.

MindLink must be registered as an application in the tenant's Azure Active Directory, and be granted delegated permissions to Skype for Business Online.



3.3 Compliance

The MindLink Server acts as a stateless proxy between the MindLink client and the Skype for Business infrastructure - no additional message data is stored in the MindLink infrastructure.

Any message sent via MindLink is routed through the Skype for Business system – even IM messages sent between two users both on the MindLink client. As such, all messages sent to or from MindLink will be captured by the Skype for Business IM and PChat compliance engines, or third-party products that filter frontend traffic.

3.4 User Profile

A user configures their MindLink client with a set of chat rooms to be permanently joined to, and a "contact list" of users that they wish to see the presence of and may want to message frequently.

These lists are separate to the joined chat rooms and contact list of the desktop Skype for Business client. Typically, a user will only require updates from only the most important chat rooms when mobile, and will only need to communicate with a subset of users. Maintaining a separate "mobile" user profile allows the user to filter the information they actually require when away from the desktop.

The MindLink client allows a user to see their desktop chat rooms and contacts and pull items from these lists to configure their mobile profile.

3.5 Lifecycle

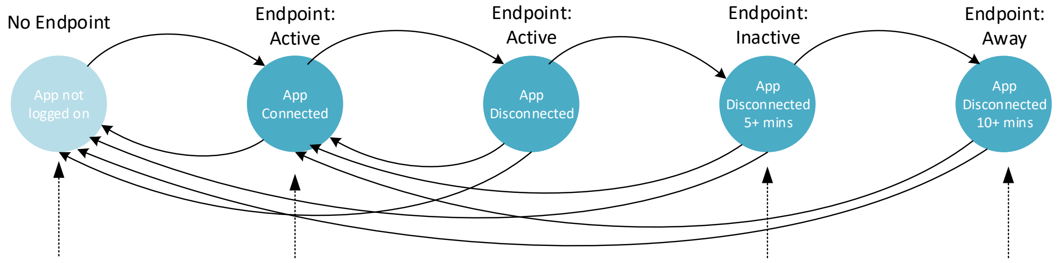
A user may choose to use MindLink for Persistent Chat and IM communication, or only Persistent Chat communication or IM communication individually. The available or optional modalities (IM vs Persistent Chat) may be configured globally for all MindLink users by the administrator.

When using MindLink with IM enabled, MindLink automatically participates in the Skype for Business multiple-points-of-presence (MPOP) system to ensure that IM messages are delivered to the most appropriate endpoint.

When the user opens the application, the MindLink Server will report the user as active on the MindLink endpoint to the Skype for Business presence engine. When the user backgrounds the application or the application gets disconnected for any other reason, the MindLink Server will report the user's non-activity according to the standard Skype for Business activity presence engine rules:

- After 5 minutes inactivity, the user will be reported as "inactive".
- After a further 10 minutes inactivity, the user will be reported as "away".

The aggregation system inside the Skype for Business presence engine uses this information to intelligently update the user's presence state and to rank the user's available endpoints in preference order for consumption of incoming IM messages.



Depending on user's other endpoints:

- User may show as offline

- MindLink preferred endpoint for IMs
- User shows as chosen presence state

Depending on user's other endpoints:

- MindLink not preferred endpoint for IMs
- User may show as inactive

Depending on user's other endpoints:

- MindLink not preferred endpoint for IMs
- User may show as away

When IM is not enabled, the MindLink endpoint does not publish presence information to the Skype for Business presence system and hence has no effect on the user's presence state.

3.6 Active Directory

3.6.1 Skype for Business Server

MindLink supports Skype for Business deployments in single or multi (resource or central) forest topologies.

The MindLink Server must have read access to Active Directory via the Global Catalog or LDAP server such that it can look-up users' Skype for Business SIP addresses in the Skype for Business forest.

In addition, the MindLink Server must be able to connect to an LDAP server in each of the authentication forests to pre-authenticate users using a fast-concurrent LDAP bind. This requires that auto-discovery of Active Directory infrastructure via DNS is working correctly.

3.7 User Access

A user must be enabled on Skype for Business to log on to MindLink. Conversely, disabling a user on Skype for Business will disable them on MindLink, including terminating any active MindLink sessions.

3.7.1 Skype for Business Server

There are no additional provisioning steps required to allow a user access to MindLink. However, user access to MindLink can be restricted to a subset of Skype for Business users by assigning MindLink users to an Active Directory group.

3.7.2 Skype for Business Online

MindLink must be registered as a web application in the tenant Azure AD.

User access can be granted/denied using the permissions and consent mechanisms built into the AAD application security model.

3.8 Monitoring

MindLink endpoints are registered against the Skype for Business registrar in the standard way. Post-mortem MindLink usage can be identified by querying the Skype for Business registration monitoring report logs, filtering by a MindLink User-Agent string.

Similarly, MindLink IM activity is recorded in the Skype for Business monitoring reports.

3.9 Conversation History

3.9.1 Skype for Business Server

MindLink endpoints may optionally integrate with the Skype for Business Server "Conversation History" system, whereby IM messages are saved to the user's "Conversation History" folder in their Exchange mailbox. MindLink will additionally retrieve previous messages from this folder when a conversation is re-opened at a later time.

The MindLink Server coordinates saving and retrieval of history messages against all Skype for Business backend versions, independent of the "Server-Side Conversation History" (SSCH) platform managed by Skype for Business itself. Furthermore, conversation history for MindLink endpoints can be enabled/disabled independently of SSCH and conversation history in the Microsoft SfB desktop client.

The following versions of Exchange Server are supported, independent of the backend Skype for Business version:

- Exchange 2010 SP2
- Exchange 2013
- Exchange 2016

3.9.2 Skype for Business Online

MindLink endpoints participate in server-side conversation history when it is enabled in the tenancy and the user has an Exchange mailbox.

3.10 Exchange Integration

3.10.1 Skype for Business Server

The MindLink Server will connect to Exchange to access the user's mailbox in the following circumstances:

- When conversation history is enabled on the MindLink Server.
- When the user is enabled for the Unified Contact Store (Lync 2013 and later)

In any of these cases the MindLink Server will locate the appropriate Exchange server, and then connect to Exchange on behalf of the user via Exchange Web Services.

This integration requires:

- If Autodiscovery is being leveraged, correct DNS configuration to support the Exchange Autodiscovery process, given the user’s primary email address. This is recommended but also necessary if there are multiple Exchange servers in the environment.
- Exchange ApplicationImpersonation rights assigned to the MindLink Server’s service account.

4 iOS

MindLink Mobile for iOS is a universal application that supports iPhone, iPod Touch and iPad devices.

It is also available in flavours that integrate with the following MAM containers:

- MindLink for BlackBerry (BlackBerry Dynamics, formerly Good Dynamics)
- MindLink for MobileIron
- MindLink for AirWatch

4.1 Persistent Connectivity

The application connects by making a TLS connection and an HTTP connection on two different ports to the MindLink Server.

These connections must be routed from the external mobile network to the MindLink Server, which must be installed on the internal network. The application has been designed to be agnostic as to how this routing occurs so that organisations can leverage their existing network infrastructure.

Various strategies are available:

- Direct connection through external firewall – the organisation opens the external firewall so that devices on the external firewall can connect directly to the MindLink Server.
- Proxied connection via a network security gateway – the organisation routes the connection via a security gateway deployed in the DMZ, e.g. using a F5 Big-IP or Citrix NetScaler.
- VPN connection – the connection is tunnelled via a VPN appliance to the internal network. The MindLink client supports iOS manual VPN, VPN on-demand, and per-app VPN.

In the recommended configuration, the app makes “persistent” connections over a TLS socket. This provides the most efficient connectivity in terms of network and battery usage, and the most stable user experience.

However, it is possible to configure the server to accept persistent connections over HTTP. In this configuration, the application will issue continuous “pending-GET” HTTP requests to the server to simulate a two-way connection. The use of HTTP may be required due to limitations in network gateway or tunnel platforms.

4.1.1 MindLink for BlackBerry

The MindLink for BlackBerry application connects to the server using the BlackBerry Dynamics infrastructure. The connection is tunnelled securely via the BlackBerry Network Operations Center and BlackBerry Proxy server directly to the MindLink Server on the internal network.

4.1.2 MindLink for MobileIron

The MindLink for MobileIron application can be configured to use the integrated MobileIron AppTunnel proxy. The server must be configured to allow persistent connections over HTTP, due to limitations in the AppTunnel platform.

Alternatively, the MobileIron Tunnel per-app VPN client may be used to proxy all traffic, which leverages the same MobileIron Sentry and Core components as the AppTunnel system.

4.1.3 MindLink for AirWatch

The MindLink for AirWatch application can be configured to use integrated app tunnelling via the AirWatch Tunnel Proxy. The server must be configured to allow persistent connections over HTTP, due to limitations in the AirWatch App Tunnel platform.

Alternatively, the AirWatch Tunnel per-app VPN client may be used to proxy all traffic, which leverages the same AirWatch Tunnel proxy components as the integrated tunnel system.

4.2 Push Connectivity

Push notifications are sent to the device using the Apple Push Notification Service (APNs).

The MindLink Server connects to the APNs infrastructure by making an outbound TLS connection. The ability to make this connection is a pre-requisite for starting the server. The connection is a TLS connection and hence cannot be made using a standard HTTP proxy.

The connection to the APNs service is secured using a certificate that the server must present to APNs to identify itself. This certificate will be provided by MindLink as part of the installation delivery, and must be updated every year.

On launching the app for the first time, the user is prompted to accept whether push notifications can be sent by the application. The user can configure how notifications should be shown on the device using the standard iOS notification settings system in the iOS Settings application.

Push notifications are sent according to the standard APNs Quality of Service implementation. If a push notification is sent when a device is out of coverage, then the push notification will be stored and delivered when the device is back in coverage. Only the latest push notification will be stored for delivery, however.

4.3 Storage

The MindLink app stores minimal information in the at-rest device storage.

All data that is stored is stored encrypted in the device keychain using keys only known to the MindLink application. This is enforced by the iOS security model.

4.3.1 MindLink for BlackBerry

All data is stored securely in the BlackBerry Dynamics storage container.

4.3.2 MindLink for MobileIron

All data is stored in a file that is encrypted using the MobileIron secure storage system.

4.3.3 MindLink for AirWatch

All data is encrypted with a key managed by the anchor application – AirWatch Agent or Container.

4.4 Access Rights

On start-up, the app will ask the user whether it should be allowed to send push notifications. This permission is enforced by the iOS app security model. The user will still be able to log on and use MindLink regardless of whether push notifications are enabled or disabled.

On accessing the camera and camera roll, the app will ask the user whether it should be allowed to access those resources. This permission is enforced by the iOS app security model. If the request is denied the camera will not be opened. If the camera has been disabled via MDM controls then the camera will not be available.

When the app is installed, it asks the operating system for permission to use the device keychain to store secure values. iOS sandboxes this access to an area that can only be used by that particular app.

The app does not access any personal information stored on the device – the iOS security model enforces that this is the case.

4.5 File Download

Files sent in chat rooms can be downloaded by the user to the device. The app farms the actual downloading process off to Safari, from where the file can be opened.

Interaction with Safari can be disabled by a managed configuration key or by MDM controls.

4.5.1 MindLink for BlackBerry

The download is farmed off to the BlackBerry Access browser, if installed. Otherwise the download will be farmed off to Safari, unless blocked completely.

4.5.2 MindLink for MobileIron

The download is processed according to the AppConnect Open-In policy rules for URLs. The download may be farmed off to the MobileIron Web@Work browser, Safari, or blocked completely.

4.5.3 MindLink for AirWatch

The download is farmed off to the AirWatch browser, if installed. Otherwise the download will be farmed off to Safari unless blocked.

4.6 Data-Loss Prevention

The app respects all iOS-level DLP constructs enforced by any EMM vendor – e.g. Managed Open-in.

In addition to the data-loss prevention mechanisms described elsewhere, the application supports the following additional controls:

- Copy/paste of message data outside of the application can be prevented by the administrator.
- Opening of URLs in a third-party browser can be controlled or blocked.
- Composing of emails via a third-party email app can be controlled or blocked.
- Calling phone numbers via a third-party dialler app can be controlled or blocked.

4.6.1 MindLink for BlackBerry

The application respects the copy/paste policies applied via the BlackBerry Dynamics container. Browsing, composing or calling in non-secure apps can be blocked via managed configuration keys.

4.6.2 MindLink for MobileIron

The application respects the copy/paste policies applied via the AppConnect container. Browsing, composing or calling in non-secure apps can be blocked via managed configuration keys and AppConnect policies.

4.6.3 MindLink for AirWatch

The application respects the copy/paste settings applied via the AirWatch app policies. Browsing, composing or calling in non-secure apps can be blocked via managed configuration keys and app policies.

4.7 Deployment

The application is deployed to the user’s device as a publicly distributed app from the iTunes store.

An enterprise-app store system may be used to advertise the availability of the application to the user, but the actual delivery of the app binary is from iTunes.

4.7.1 MindLink for BlackBerry

The application must be assigned to the user via the BlackBerry Control/UEM console, and the user must be provisioned with an activation key. The MindLink application definition is made available to the BlackBerry Control/UEM instance as a Partner app via the BlackBerry Marketplace.

4.7.2 MindLink for MobileIron

The user’s device must have a MobileIron AppConnect container – i.e. they must have deployed the MobileIron Mobile@Work application and be registered and enabled in the MobileIron Core console. The application must be authorized via the policy settings.

4.7.3 MindLink for AirWatch

The user’s device must be enrolled using either Agent or Container activation, and the application must be enabled for the user in the AirWatch Console.

4.8 On-Boarding

By default, the user must configure the application to talk to the MindLink Server. On first installation, the application will ask for the address of the server to connect to. This can be either:

1. The DNS name of the MindLink Server, plus the port on which the web service is configured to listen.
2. The user’s email address. The app will use the address https://mindlinkdiscover.<domain from email address>. This mechanism requires that a “mindlinkdiscover” DNS alias has been created for the MindLink Server address.

This value is then stored and re-used for subsequent sessions. The iOS Settings app can be used to change the value.Alternatively, the app can be pre-configured with the server address via iOS managed app configuration – configured via any EMM technology.

4.8.1 MindLink for BlackBerry

The server URL can be provided via the BlackBerry Dynamics configuration system.

4.8.2 MindLink for MobileIron

The server URL can be provided by the AppConnect key/value pair configuration system.

4.9 Authentication

By default, the user must enter their Active Directory credentials. These are authenticated by the MindLink server and exchanged for a session token.

Alternatively, the app can be pre-configured with the user's log on name via iOS managed app configuration – configured via any EMM technology.

4.9.1 MindLink for BlackBerry

The app can be configured to use the UPN associated with the account used to activate the BlackBerry Dynamics container.

4.9.2 MindLink for MobileIron

A preconfigured user name can be provided via the AppConnect configuration.

4.10 Logging

Client-side app logging may be enabled to diagnose functional or connectivity issues. The user is responsible for enabling logging via the app settings. When enabled, the app will log at all logging levels to:

1. The device console – this is can be inspected from a developer tool on a desktop machine e.g. XCode when connected via USB.
2. A rolling log file in the app sandbox – this can be exported by the user via email or other suitable third-party app. This mechanism respects any iOS "open-in" policies applied via any EMM technology.

The ability to enable logging can be disabled by the administrator via iOS managed configuration. Similarly, verbose logging of secure information can be disabled.

4.10.1 MindLink for BlackBerry

The app logs to a rolling log file inside the secure container. The log file can be exported via a BlackBerry secure email app if available, or else via any third party app – this interaction respects the DLP policies applied.

The ability to enable logging can be disabled by the administrator via a managed configuration key.

Logging levels are enabled according to the logging level applied to the BlackBerry Dynamics container.

4.10.2 MindLink for MobileIron

The app logs via the AppConnect platform.

The logs are written to the device console and to a non-secure file in the app sandbox, when enabled. This file can be exported via a third-party email app – this interaction respects the DLP policies applied.

The logging level is controlled by the administrator via an AppConnect policy. The user must enter a debug key provided by the administrator to enable sensitive values in the log files.

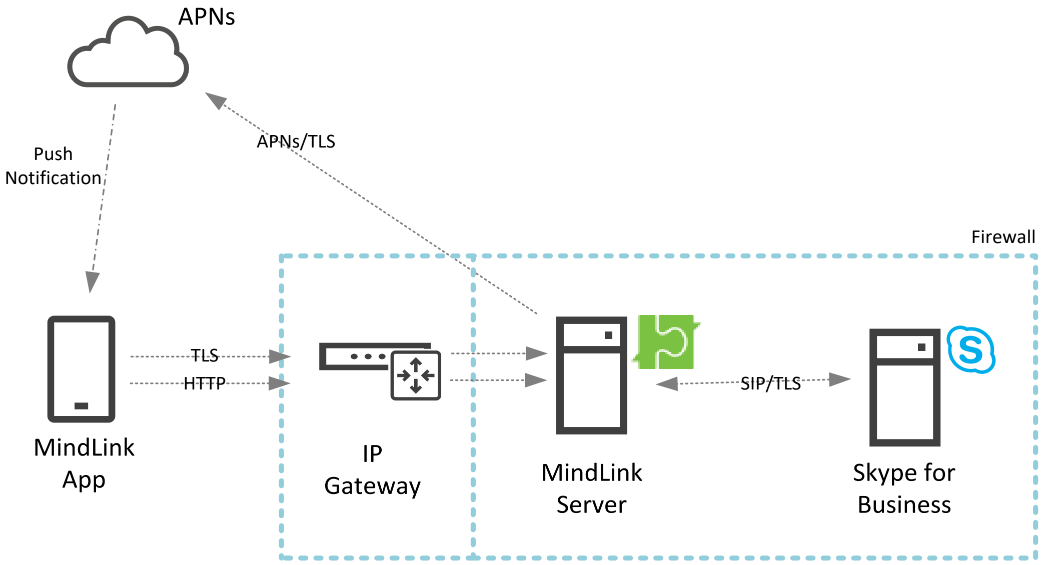
4.10.3 MindLink for AirWatch

The app logs via the AirWatch SDK platform.

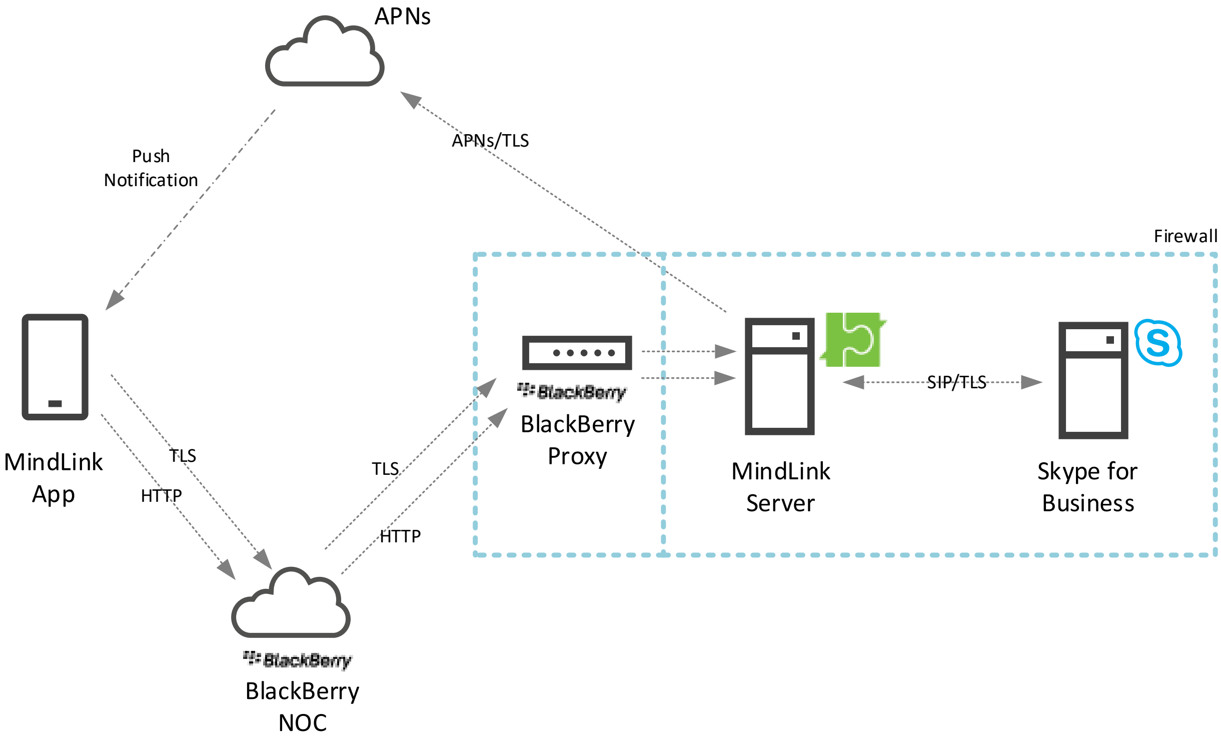
Logs are written internally to the SDK and can be uploaded directly to the administration console on demand.

The administrator has control over the enabled logging level, configured via the app policy.

4.11 Reference Deployment

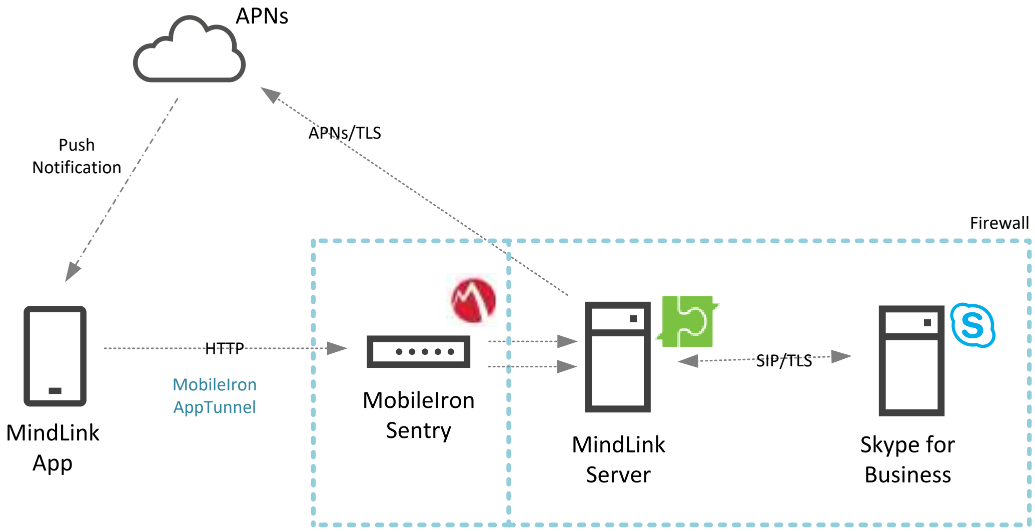


4.11.1 MindLink for BlackBerry

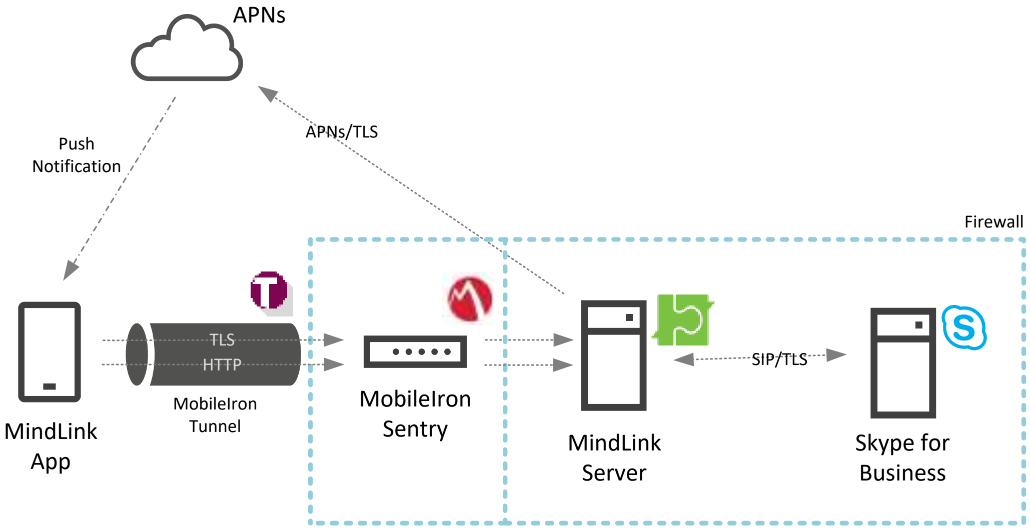


4.11.2 MindLink for MobileIron

Leveraging MobileIron AppTunnel:

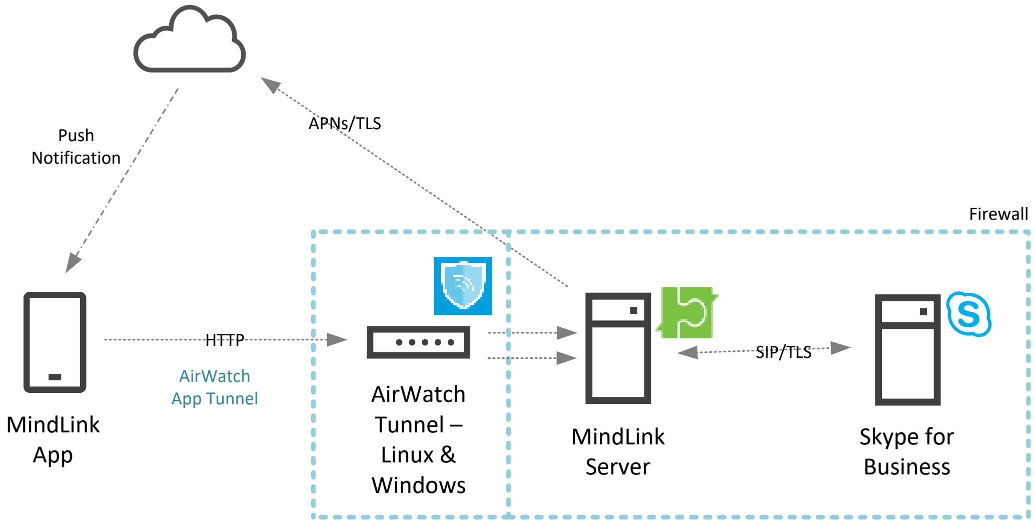


Leveraging MobileIron Tunnel:

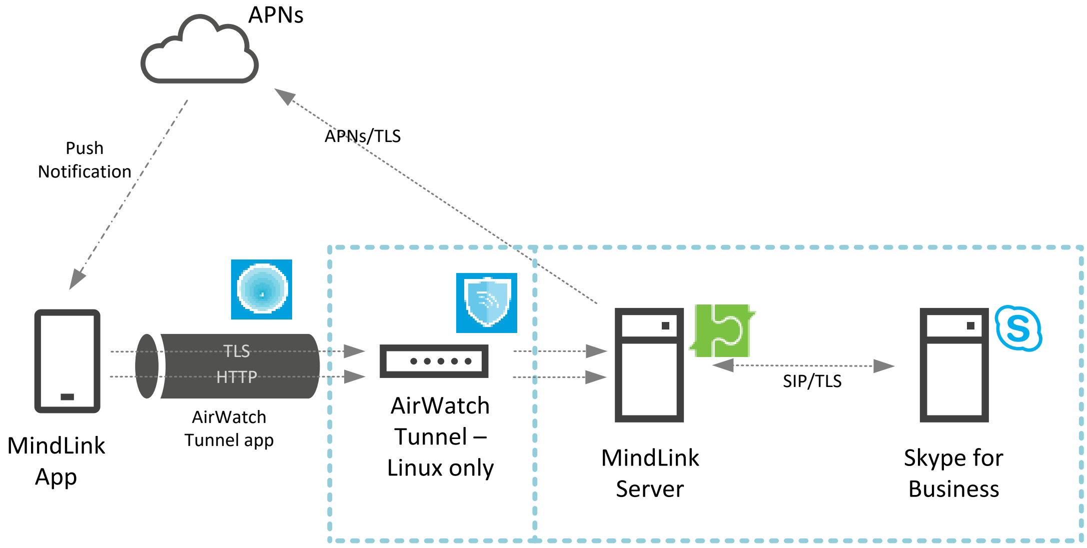


4.11.3 MindLink for AirWatch

Leveraging AirWatch integrated App Tunnel:



Leveraging AirWatch Tunnel per-app VPN:



5 Android

MindLink Mobile for Android supports Android phone and tablet devices with any screen size. The app will run on unmanaged devices and also as a managed app within the Android for Work container, on AfW enrolled devices.

It is also available in flavours that integrate with the following MAM containers:

- MindLink for BlackBerry (BlackBerry Dynamics, formerly Good Dynamics)

5.1 Persistent Connectivity

The application connects by making a TLS connection and an HTTP connection on two different ports to the MindLink Server.

These connections must be routed from the external mobile network to the MindLink Server, which must be installed on the internal network. The application has been designed to be agnostic as to how this routing occurs so that organisations can leverage their existing network infrastructure.

Various strategies are available:

- Direct connection through external firewall – the organisation opens the external firewall so that devices on the external firewall can connect directly to the MindLink Server.
- Proxied connection via a network security gateway – the organisation routes the connection via a security gateway deployed in the DMZ. E.g. using a F5 Big-IP or Citrix NetScaler.
- VPN connection, including per-app VPN. This VPN configuration can be managed by AfW or other MDM technology.

5.1.1 MindLink for BlackBerry

The MindLink for BlackBerry application connects to the server using the BlackBerry Dynamics infrastructure. The connection is tunnelled securely via the BlackBerry Network Operations Center and BlackBerry Proxy components server directly to the MindLink Server on the internal network.

5.2 Push Connectivity

Push notifications are sent to the device using Firebase Cloud Messaging (FCM).

The MindLink Server connects to the FCM infrastructure by making outbound HTTP requests. The server must be able to communicate with FCM for push notifications to be received. The server can be configured to use an HTTP proxy to make this connection.

A user can choose whether push notifications are sent to the device by configuring a setting inside the application. When push notifications are received, they are added to the Android notification drawer. If a new message is received while the app is in the background (before it has automatically disconnected), then a notification of this new message will also be added to the notification drawer. The Android Settings application may also be used to configure how MindLink notifications are added to the notification drawer.

If a push notification is sent when a device is out of coverage, then the push notification will be stored and delivered when the device is back in coverage. Only the latest push notification will be stored for delivery, however.

5.3 Storage

The MindLink app stores very little information in the at-rest device storage, and no message content.

The data that is stored is saved on the device's internal storage in "private" mode, such that the operating system restricts access to the application itself.

When installed as a managed Android for Work app, the stored data is managed by the AfW encryption container.

5.3.1 MindLink for BlackBerry

All data is stored securely in the BlackBerry Dynamics storage container.

5.4 Access Rights

The application requests the following permissions from the operating system on installation:

- Network access
  - To communicate with the MindLink Server
- Inspect network connections
  - To coordinate the connection to the MindLink Server
- Control vibration
  - To insert notifications into the notification drawer
- Prevent the phone from sleeping
  - To momentarily clean-up network resources when disconnecting.

The app does not access any personal information stored on the device – the Android security model enforces that this is the case.

Android 6.0+ devices

Android will prompt the user for permission to access external storage when the user enables logging for the first time. Similarly, a prompt for access to the camera will appear when a photo is attempted to be sent for the first time.

Pre-Android 6.0 devices

On installation, Android will prompt the user to accept that the application can perform the following actions:

- Access External Storage
  - This is required to write log files to the SD Card when logging is enabled.

5.4.1 MindLink for BlackBerry

The BlackBerry Dynamics container will request access to read the "phone state". This includes the phone number and the cellular network information. This is a requirement of the BlackBerry Dynamics platform, used to identify the correct BlackBerry servers for the application.

5.5 File Download

Files sent in chat rooms can be downloaded by the user to the device. The app farms the actual downloading process off to the device browser, from where the file can be opened.

Interaction with the device browser can be disabled by a managed configuration key.

5.5.1 MindLink for BlackBerry

The download is farmed off to the BlackBerry Access browser, if installed. Otherwise the download will be farmed off to the default device browser, unless blocked completely.

5.6 Data-Loss Prevention

In addition to the data-loss prevention mechanisms described elsewhere, the copy/paste of message data outside of the application can be prevented by the administrator.

When installed as a managed Android for Work app, the app will respect the DLP policies applied to the AfW container. Additionally, the app supported the following controls via managed configuration keys:

- Opening of URLs in a third-party browser can be controlled or blocked.
- Composing of emails via a third-part email app can be controlled or blocked.

5.6.1 MindLink for BlackBerry

The application respects the copy/paste policies applied via the BlackBerry Dynamics container. Browsing or composing in non-secure apps can be blocked via managed configuration keys.

5.7 Deployment

The application is deployed to the user's device as a publically distributed app from the Google Play store.

An enterprise-app store system may be used to advertise the availability of the application to the user, but the actual delivery of the app binary is from Google Play.

5.7.1 MindLink for BlackBerry

The application must be assigned to the user via the BlackBerry Control/UEM console, and the user must be provisioned with an activation key. The MindLink application definition is made available to the BlackBerry Control instance as a Partner app via the BlackBerry Marketplace.

5.8 On-Boarding

By default, the user must configure the application to talk to the MindLink Server. On first installation, the application will ask for the address of the server to connect to. This can be either:

1) The DNS name of the MindLink Server, plus the port on which the web service is configured to listen.
2) The user's email address. The app will use the address **https://mindlinkdiscover.<domain from email address>**. This mechanism requires that a "mindlinkdiscover" DNS alias has been created for the MindLink Server address.

This value is then stored and re-used for subsequent sessions.

Alternatively, when installed as a managed Android for Work app, the app can be pre-configured with the server address via Android App Restrictions configuration – configured via any EMM technology.

5.8.1 MindLink for BlackBerry

The server URL can be provided via the BlackBerry Dynamics configuration system.

5.9 Authentication

By default, the user must enter their Active Directory credentials. These are authenticated by the MindLink server and exchanged for a session token.

Alternatively, the app can be pre-configured with the user's log on name via managed app restrictions – configured via any EMM technology.

5.9.1 MindLink for BlackBerry

The app can be configured to use the UPN associated with the account used to activate the BlackBerry Dynamics container.

5.10 Logging

Client-side app logging may be enabled to diagnose functional or connectivity issues. The user is responsible for enabling logging via the app settings. When enabled, the app will log at all logging levels to:

1. The device LogCat console – this is can be inspected from a developer tool on a desktop machine e.g. Android Studio when connected via USB.
2. A rolling log file on the device storage card – this can be opened by the user as a text file or exported via email or other suitable third-party app.

When installed as a managed Android for Work app:

1) the app will log to the secure side of the device storage card. The AfW container will enforce which apps can be used to open or export the log file.
2) The ability to enable logging can be disabled by the administrator via managed configuration. Similarly, verbose logging of secure information can be disabled.

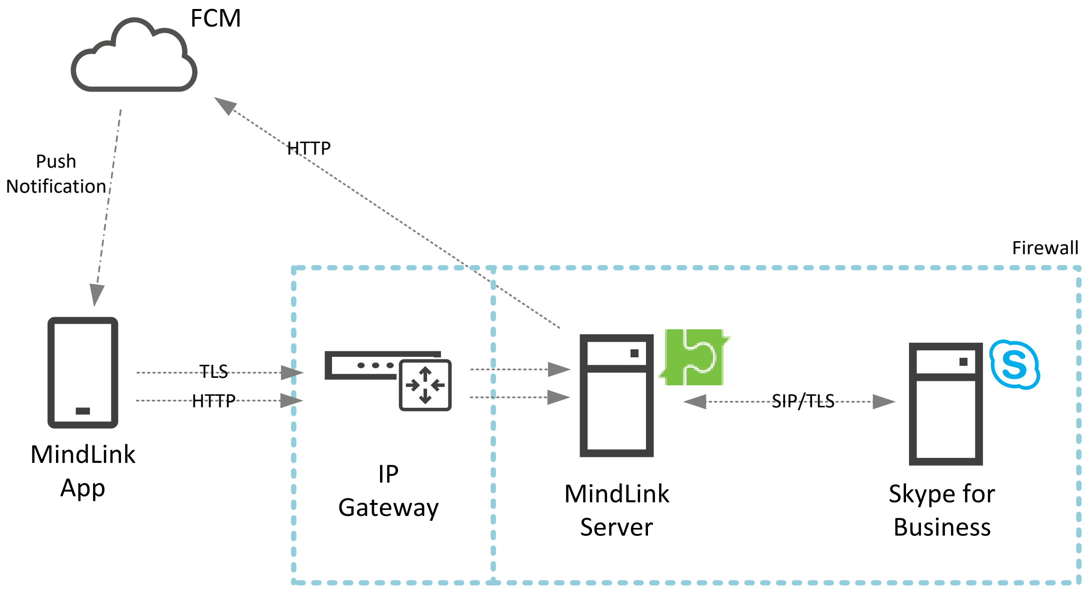
5.10.1 MindLink for BlackBerry

The app logs to a rolling log file inside the secure container. The log file can be exported via a BlackBerry secure email app if available, or else via any third party app.

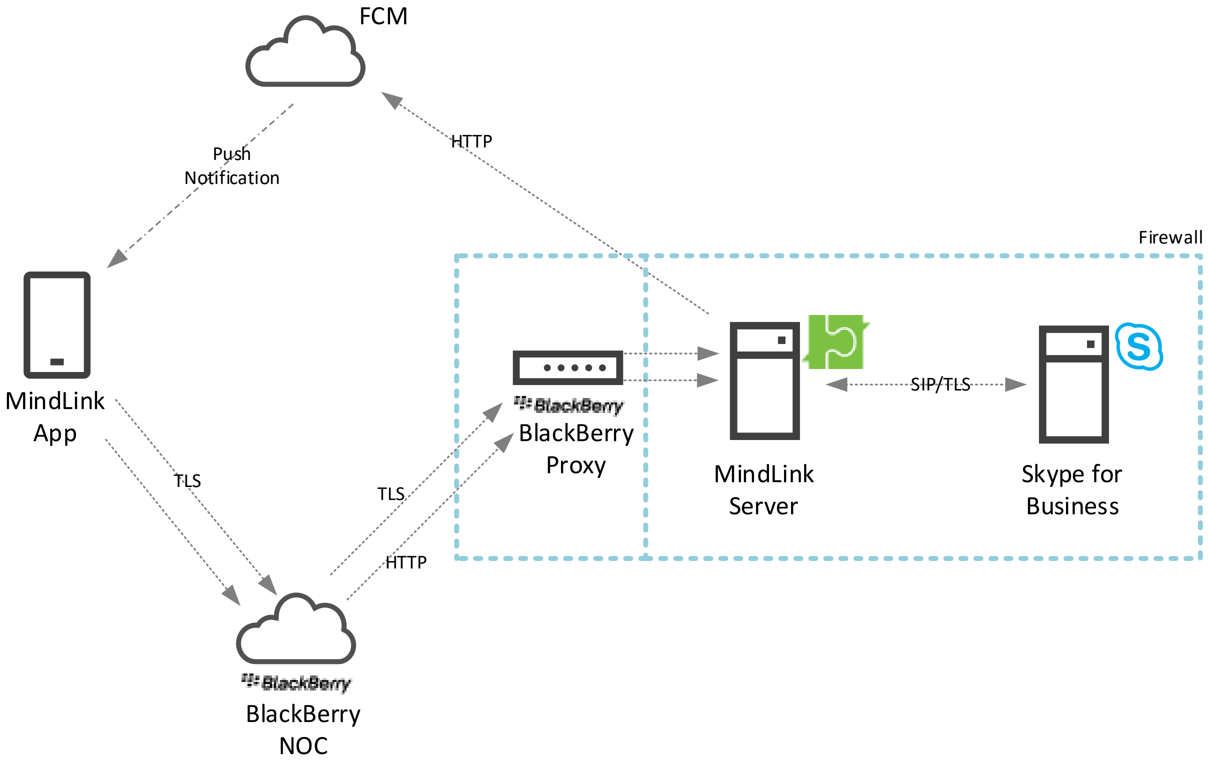
The ability to enable logging can be disabled by the administrator via a managed configuration key.

Logging levels are enabled according to the logging level applied to the BlackBerry Dynamics container.

5.11 Reference Deployment



5.11.1 MindLink for BlackBerry



6 Persistent Chat Add-ins

MindLink for iPad and MindLink for Android on Android tablet devices supports Chat Room Add-ins. An add-in is a panel that is displayed alongside the chat room message content for the purposes of displaying related or relevant information. The Add-in can be used to enhance the productivity and usefulness of the conversation within the chat room.

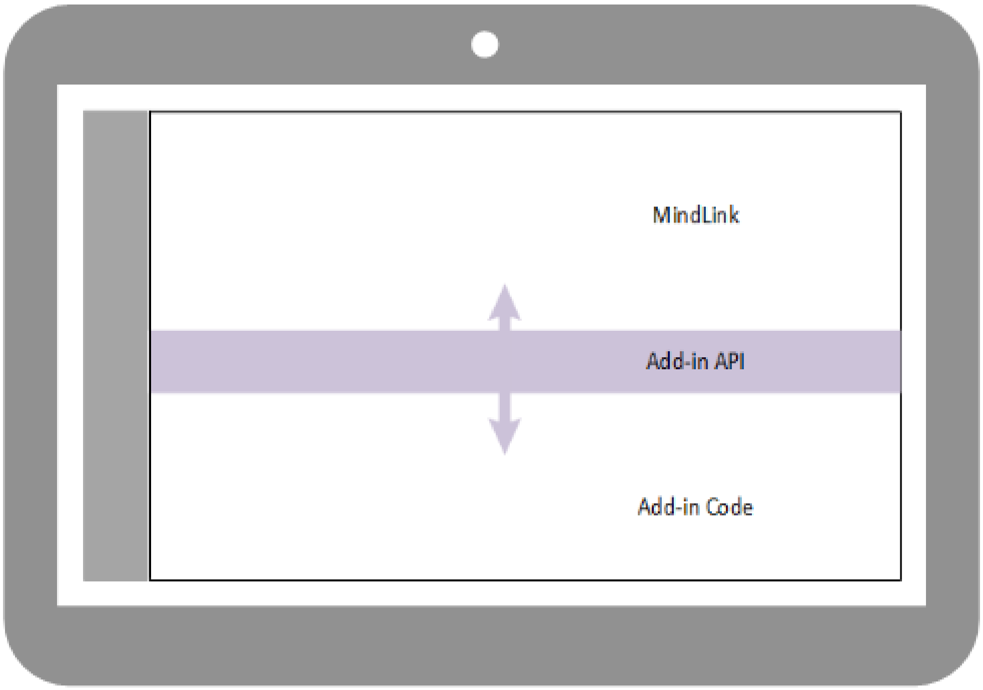
An Add-in can be any web page. The system administrator configures which panel appears in which chat room using the Persistent Chat administration tools.

The MindLink application hosts the Add-in content and also exposes an API with which the Add-in can interact with the conversation in the chat room. Whilst any static web page content can be shown as an Add-in, specially designed Add-ins can be implemented to interact with the rest of the application using the API. For example, the Add-in may be written to interact with the chat room messages when a condition is met – such as an Add-in hosting a live data stream from a third-party line of business system which then posts relevant information to the chat room in the parent pane.

The Add-in architecture consist of following components:

- A standard web page which contains code and content.
- The page is contained within the Add-in wrapper which essentially sits inside a browser frame with an API – "MindLink JavaScript Add-in API".
- The JavaScript API, which provides the capability to support interaction between the Add-in and the parent panes.

More information is available within the Add-in developer guide.



7 Scale Out and High Availability

With the minimum hardware requirements, a single MindLink Server instance will support 2000 devices.

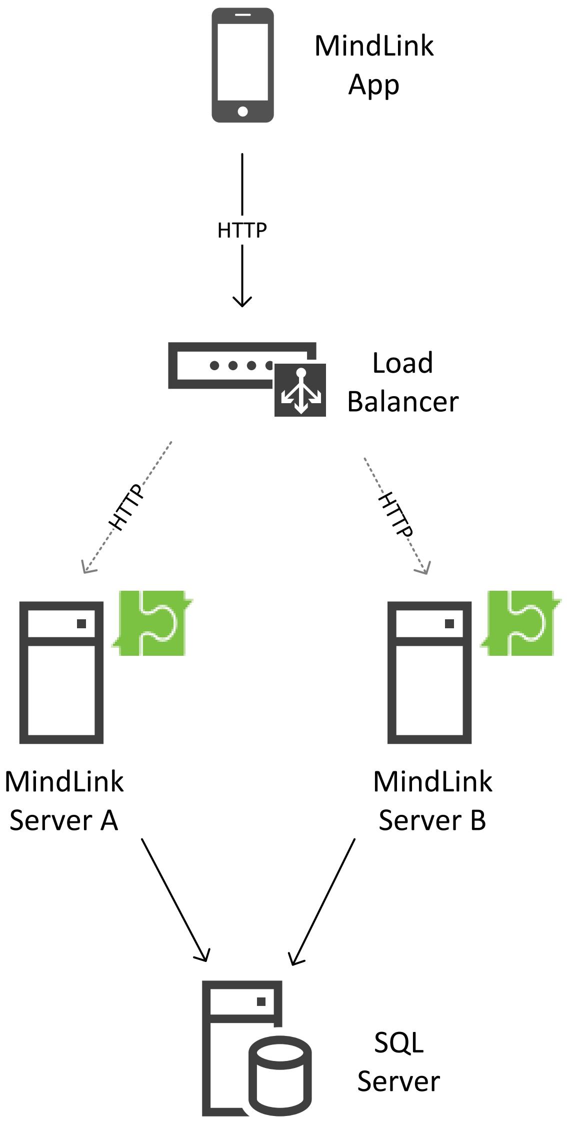
MindLink Servers can be deployed in a pooled cluster for the purpose of:

- Scaling out the number of devices:
  - Deploy more servers to linearly scale the supported number of devices.
  - Session load is distributed throughout the pool.
- Fault tolerance during server failure:
  - Deploy ***f*** additional servers to support ***f*** server failures.
- Cross-site high-availability:
  - Deploy MindLink Servers in a cross-site stretched pool for site-level active/active HA.

7.1 Infrastructure Requirements

Deployment of a standalone MindLink Server requires no additional infrastructure outside of the host Windows Server.

However, deployment of multiple MindLink Servers in a pooled cluster requires additional infrastructure components – a Microsoft SQL server and an HTTP Load Balancer. The clustering mechanism is implemented at the application level and does not require Windows Server clustering or other OS-level configuration.



7.1.1 Load Balancer

The load balancer's role is to distribute new session log-ons across the available servers. Any standard HTTP load balancer implementing a round-robin balancing algorithm is supported, and no client affinity mechanism is required.

Only the initial bootstrapping HTTP request is made via the load balancer. This establishes an association between a specific node in the pool and the device. The chosen node then coordinates the Skype for Business session on behalf of the device as usual. Subsequent TLS connections throughout the lifetime of the session are made directly from the device to the specific node as usual.

The load balancer is aware of which servers are active in the pool by periodically pinging an HTTP health-check service exposed by each MindLink Server.

7.1.2 SQL Server

The role of the SQL server is to allow the servers in the pool to coordinate amongst themselves as to which node is responsible for which device.

SQL Mirroring, AlwaysOn Failover Cluster and AlwaysOn Availability Groups are all supported high-availability solutions for the SQL layer.

7.2 Scale Out

Adding more servers to a MindLink Server pool will add capacity to serve more devices. Each server is responsible for maintaining the long-running Skype for Business server for a subset of the connected devices.

Each device session is managed by exactly one member of the MindLink Server pool. This affinity is assigned by the load balancer when each session is initiated.

7.3 High Availability

Deploying MindLink Servers in a clustered pool with extra server capacity allows service to be maintained even when a server fails. An extra node should be deployed for every server failure that should be tolerated.

In normal operation, device sessions are load balanced evenly across all nodes. When a server fails, the long-running sessions managed by the node are ended.

When the device next tries to re-connect to the failed server to resume its session, the app will prompt the user that their session cannot be resumed. At this point, the app will then create a new session which will be assigned a new remaining live node by the load balancer.

Sessions for devices homed on other nodes are not affected by the node failure and will continue as normal.

7.4 Cross-Site High Availability

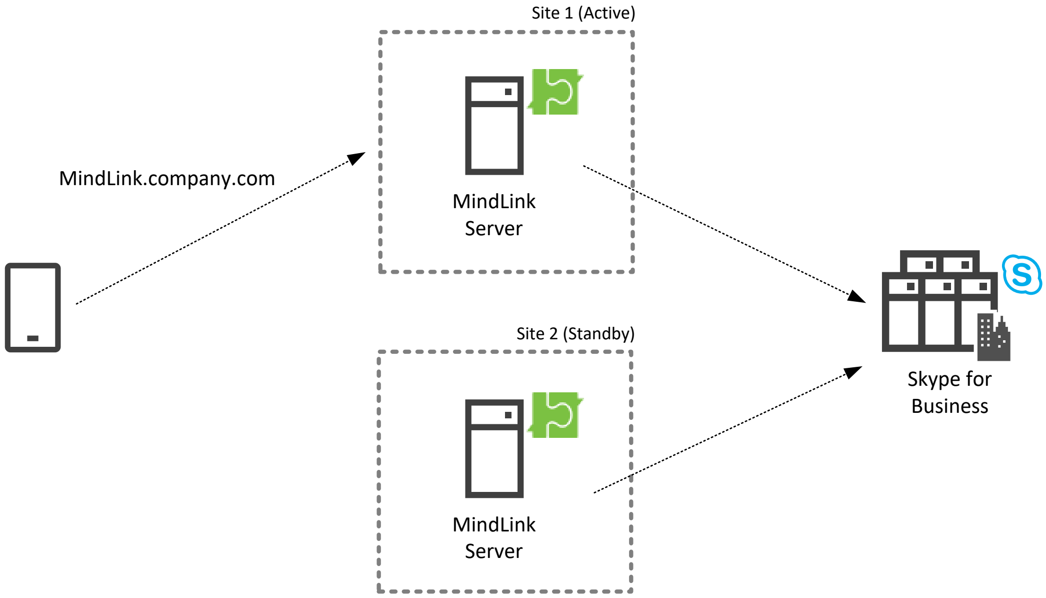
MindLink Servers may be deployed in a cross-site pool for active/active site-level resilience. In normal operation, device sessions are load balanced evenly across the two sites. When an entire site fails, the end-user experience is as described above.

The network connection between data centres must have a latency of less than 5ms.

8 Failover

For failover at the site-level in an active/passive configuration, a mirror-image MindLink deployment should be configured at another site.

On failover, the address of the MindLink Server (or the address of the load balancer when deployed as a pool) as configured on the client should be switched to point to the secondary installation via DNS or otherwise.



In the above diagram the address of the MindLink Mobile server pool has been configured as MindLink.company.com, and all devices have been configured to connect to that address. MindLink.company.com is currently resolving via DNS to the IP of the MindLink Server in Site 1.

On failover to Site 2, the DNS configuration will be changed such that MindLink.company.com resolves to the IP of the MindLink Server in Site 2.

8.1.1 Skype for Business Server

Failover of the MindLink components can happen independently of the Skype for Business frontend and Persistent Chat pools.

Similarly, as the MindLink components in the standby site should already be defined as trusted application servers in the Skype for Business topology, no additional Skype for Business configuration changes are required to failover the MindLink tier.

8.1.2 Pooled Failover

If the MindLink Servers are deployed in a pooled cluster, an identical pool should be defined in each site. The data stored in the SQL server used by the pool relates only to the member server nodes in the pool, and as such does not need to be synchronized between sites – the active and standby SQL databases can be completely decoupled.

9 Deployment

The MindLink Mobile Server is distributed as a self-contained MSI. The MSI contains the server components and the Management Center (a graphical configuration utility).

The installation process consists of:

1) Extracting the MSI.
2) Configuring the system via the Management Center – including Skype for Business connectivity and frontend app settings.
3) Starting the MindLink Server Windows Service.

For Skype for Business Online integration, some additional Azure AD configuration must be supplied via a cmdlet available in a PowerShell module installed alongside the server

Updates are made available on roughly a 6-week cadence. Upgrades are installed in-place and will migrate existing configuration. Users will experience an outage while the MindLink Server service is restarted after the upgrade.

On deploying the server, the client should be downloaded by the user from the appropriate public App Store.

Internal Enterprise App Stores may be leveraged to prompt the user that the app is available, but the binary download will be from the public App Store.

10 Versioning

This section applies to the MindLink apps that are publicly distributed, as it deals with the versioning concerns of having regularly updated public-store client apps connecting against customers' on-premise MindLink Server deployments.

On release of a new MindLink Mobile version (currently scheduled on a 6-week cadence), MindLink Software will update the download available in the iTunes store and Google Play store to the latest version, so that the iTunes and Google Play client apps always match the latest server version available from MindLink Software.

10.1 Backwards Compatibility

As such, it will be likely that eventually the client version available in the iTunes or Play store will be a newer version than that of the MindLink Server which an organization has deployed. For example, new MindLink users will be downloading version 4 of MindLink from iTunes, when their organization may have version 2 deployed.

This backwards-compatibility of new client versions will be supported by MindLink Software for server versions up to a year old.

10.2 Forwards Compatibility

Conversely, on upgrade of an organization's MindLink Server, existing users will potentially still have an older version of the MindLink Mobile client installed on their devices.

In this case, the MindLink client will show a message informing the user to upgrade their installed application to the latest version from the iTunes or Play store, and prevent them from logging in to the app.

11 Licensing

The MindLink Server requires a license to run. This will be provided to you by MindLink and must be applied to the installation via the Management interface. The license will either allow an unlimited number of devices to connect, or will specify a maximum capacity.

11.1.1 Skype for Business Server

If a maximum capacity is specified, devices will be allowed to start sessions up until the number of sessions on the server (or across the pool) reaches the capacity limit. At this point, further device logons will be denied. This will occur until existing sessions are ended – the user logs out, the session expires, etc.

11.1.2 Skype for Business Online

A maximum capacity is not enforced by the software against Skype for Business Online, though customers may still be licensed on a device-capacity model.