# BlackBerry Dynamics Security White Paper

**Version 2.2**

# Overview

This document provides detail about the security provided by BlackBerry Dynamics. The intended audience is CIOs, IT managers, software architects, and people with a similar level of technical knowledge.

This document assumes some knowledge about the features and purpose of the BlackBerry Dynamics Platform. For background, you might want to review documentation at BlackBerry UEM and BlackBerry Dynamics.

BlackBerry Dynamics platform enables mobile enterprise applications to employ the industry-leading security features such as:

- Proxy infrastructure that enables connections between mobile clients and application servers that are behind the enterprise firewall. There is no need for a VPN, or for ports to be opened in the enterprise firewall.
- End-to-end encryption of data in transit between mobile clients and application servers.
- Storing enterprise data on the device in a separate secure container that can be remotely wiped by an administrator.
- Encrypting with AES 256-bit cipher. This technology is used to protect data at rest, in the secure container, and to protect data in transit between client and server.
- Enforcing password and device compliance policies, when accessing enterprise data.

BlackBerry has developed few applications using Dynamics platform such as BlackBerry Work, BlackBerry Access etc. Many ISVs (Independent Software Vendors) have also built applications using BlackBerry Dynamics. Enterprises can build their own applications using Dynamics libraries to meet their custom business needs.

## Components

A BlackBerry Dynamics deployment has the following components

- **BlackBerry Dynamics Application:** An application with embedded calls to the BlackBerry Dynamics runtime which provides services/features to the user. Sometimes also referred as Dynamics application.
- **BlackBerry Dynamics Runtime:** Every BlackBerry Dynamics application includes an instance of the BlackBerry Dynamics runtime. The runtime has APIs that give the application access to user authentication, secure communications, secure storage, and communication behind the firewall. The runtime also handles enforcement of security policies on behalf of the application. An instance of the BlackBerry Dynamics runtime may sometimes be referred to as a Container or Dynamics Runtime. Source code and APIs that is shared with the developers is referred to as Dynamics library.
- **BlackBerry Cloud:** The BlackBerry Cloud infrastructure provides the secure communications infrastructure between the Dynamics runtime on the device, and the BlackBerry Dynamics enterprise servers behind the firewall.

- **Enterprise Servers**: There are two BlackBerry Dynamics components installed behind the enterprise firewall.

    1. **Management Server**: This server provides management of the enterprise's users, applications and security policies and is called BlackBerry UEM server/core. BlackBerry UEM server supports MDM and enterprise app store functionality in addition to Dynamics applications.

    2. **The BlackBerry Proxy** service provides the secure communications infrastructure between the Dynamics application and application servers that are behind the enterprise firewall. In a BlackBerry UEM deployment this service is part of **BlackBerry Connectivity Node** (BCN).

## Security Features

The following table lists the major features of the BlackBerry Dynamics platform and the specific security capability associated with that feature.

| Security Element | Features |
|---|---|
| Container Access | • Different policies for different Users<br>• Remote lock<br>• Compliance verification<br>• Auto-Lock & local container access authentication |
| Container Data Storage | • Secured & Managed container to protect enterprise data<br>• Data encrypted with AES-256<br>• Remote delete & lock<br>• FIPS140-2 certified crypto module. |
| Data Transmission | • TLS connections<br>• AES-256 encryption<br>• FIPS140-2 certified crypto algorithms<br>• Connection monitor |
| Enterprise Resource Protection | • No opening of the firewall<br>• Role based administration<br>• No need for enterprise credentials outside the firewall<br>• Centralized and cross platform security control<br>• Connections to permitted application servers or domains. |

## How Data Is Protected

Data can be grouped into two categories:

- On-device: Data already stored on a device
- In-transit: Data in process of being communicated

### On-Device Data

These tables provide a summary of how on-device data is protected by the BlackBerry Dynamics platform.

| How data is protected… | Answer |
|---|---|
| Enterprise data is saved inside the BlackBerry Dynamics app | Encrypted with AES-CBC using with 256 bit key. |
| BlackBerry Dynamics application encryption key | Protected with user password and on device security processor.<br>Password strength requirement is set from the management console. |
| After uninstallation | Files are deleted. |
| Device stolen | User is asked for password. Files on device is encrypted. |
| User is no longer entitled to the application | Files are deleted. Initiated by IT administrator using management console. |

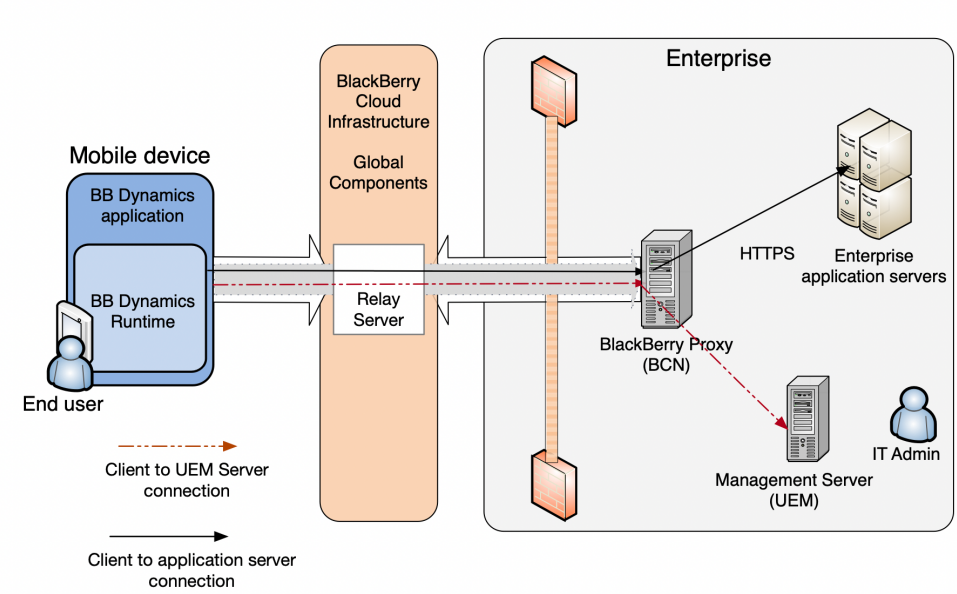
## In-Transit Data

These tables provide a summary of how in-transit data is protected by the BlackBerry Dynamics platform.

| How data is protected… | Answer |
|---|---|
| Data in transit between the Dynamics app and BlackBerry proxy server. | Encrypted with AES-CTR. |
| Data in transit between application client and application server | Application developer controls this. The Dynamics runtime library provides TLS and HTTPS APIs to establish secure connections. |
| BlackBerry UEM server to BlackBerry Cloud | TLS |
| BlackBerry Proxy server to BlackBerry Cloud | TLS |
| BlackBerry UEM server to BlackBerry Proxy server and vice versa | TLS |

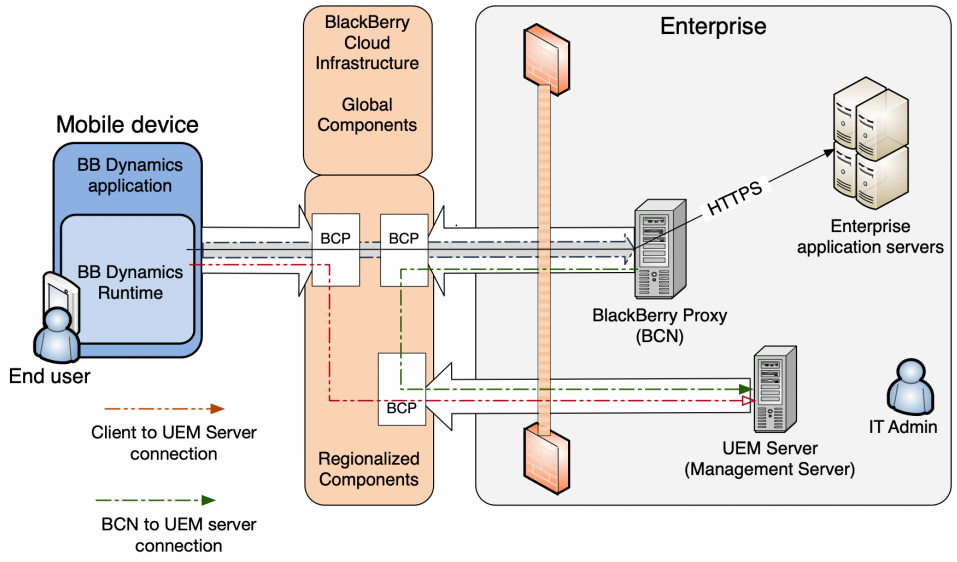
# Global Deployment

All BlackBerry global cloud components are shared by every UEM deployments.




# Regional Deployment

Deployment uses some global components and some components that are present in the local region.



# BlackBerry Dynamics App

The BlackBerry Dynamics SDK includes a client library, which is used to build a BlackBerry Dynamics application for mobile devices. The Dynamics SDK provides the APIs to perform app activation, enterprise connectivity, and user authentication. For management, the Dynamics runtime supports remote application and container management, including the ability to delete app data. The Dynamics SDK is available to developers for iOS and Android platforms to build applications. In addition, BlackBerry supports cross-platform frameworks such as Cordova, React Native and Xamarin. Features supported for each platform can be found at BlackBerry developer site and in the API Reference Documentation. In addition, on Windows and MacOS BlackBerry provides BlackBerry Access application which is built using Dynamics SDK.

Application activation is the process by which a Dynamics runtime receives initial provisioning and configuration data from the management server. Security aspects of application activation are described in App Activation section.

Once activation is completed a Dynamics app may:

- Establish a HTTPS, TLS or TCP connection to an application server behind the enterprise firewall by using APIs exposed in the Dynamics SDK. Security aspects of enterprise connectivity are described in Enterprise Connectivity section. The Dynamics runtime supports TLS 1.2, TLS1.1 etc. The Dynamics runtime also supports NTLM v2 and Kerberos authentication for HTTP/S connections.
- Create encrypted files and databases using apis provided by the Dynamics runtime.

## Data Storage on the Device

A BlackBerry Dynamics app is recommended to store all user and management data in encrypted files and databases using APIs provided by Dynamics runtime. These files and databases are encrypted by the Dynamics runtime with AES (CBC mode), using a 256-bit random key (Data Encryption Key), and random IV's.

The Data Encryption Key itself is encrypted with a key based on the user provided secret and stored in the Startup file (see next section). The mechanism used to protect Data Encryption Key varies based on the user authentication mode enabled by the administrator and device hardware security capabilities. The startup file is disabled from being backed up. The startup file is used during the app startup.

An encrypted copy of Data Encryption Key is also saved in the Recovery/Restore file (as described in the section Application Unlock and Restore). Recovery file could be backed up if app data backup is enabled on the device. This file is used to restore the app on a different device.

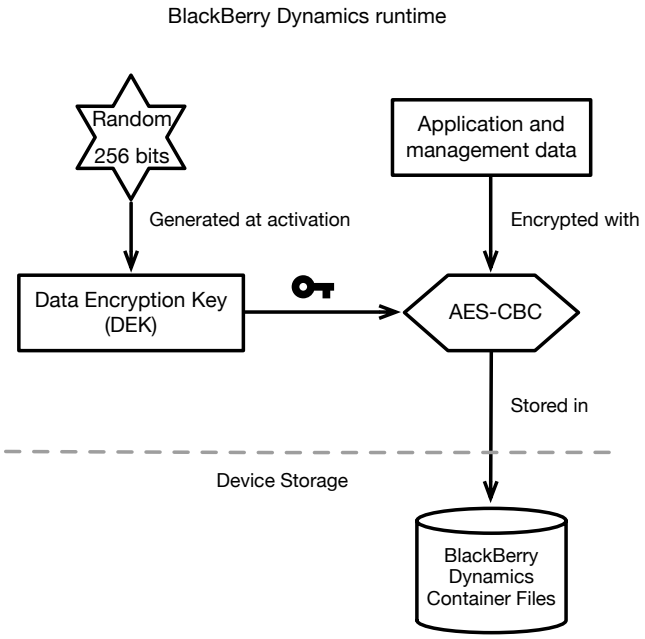



***Figure: BlackBerry Dynamics runtime data persistence***

## User Authentication and Key Storage

A BlackBerry Dynamics app can only be used by the user who has activated the app. After app activation is completed, the user is asked to set a password. This password (User Secret) is used to secure the Data Encryption Key and authenticate the user on subsequent app startup. When the app delegates authentication to another Dynamics app, the User Secret is provided by the other Dynamics app (see the Authentication Delegation section).

A cryptographic hash of the *User Secret*, called the User Key, is used to encrypt Data Encryption Key (DEK). Encrypted Data Encryption Key is kept inside the Startup file. SHA512 hash of User Key is also saved in the Startup file and on subsequent app startup is used to authenticate the user. Depending on the device OS and hardware security capabilities, some cryptographic operations are completed inside the secure enclave or device key store as shown in the following diagrams.

## Android

The figure below shows how Data Encryption Key is secured on an Android device.

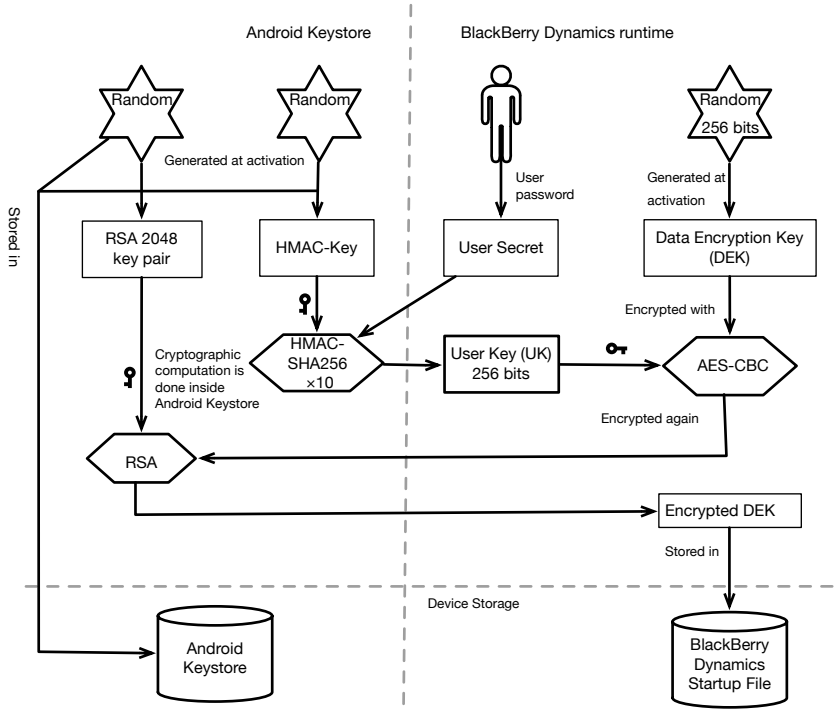



***Figure: Securing Data Encryption Key on Android***

## iOS

The figure below shows how Data Encryption Key is secured on an iOS device. Secure Enclave support was added in SDK version 4.2.x.

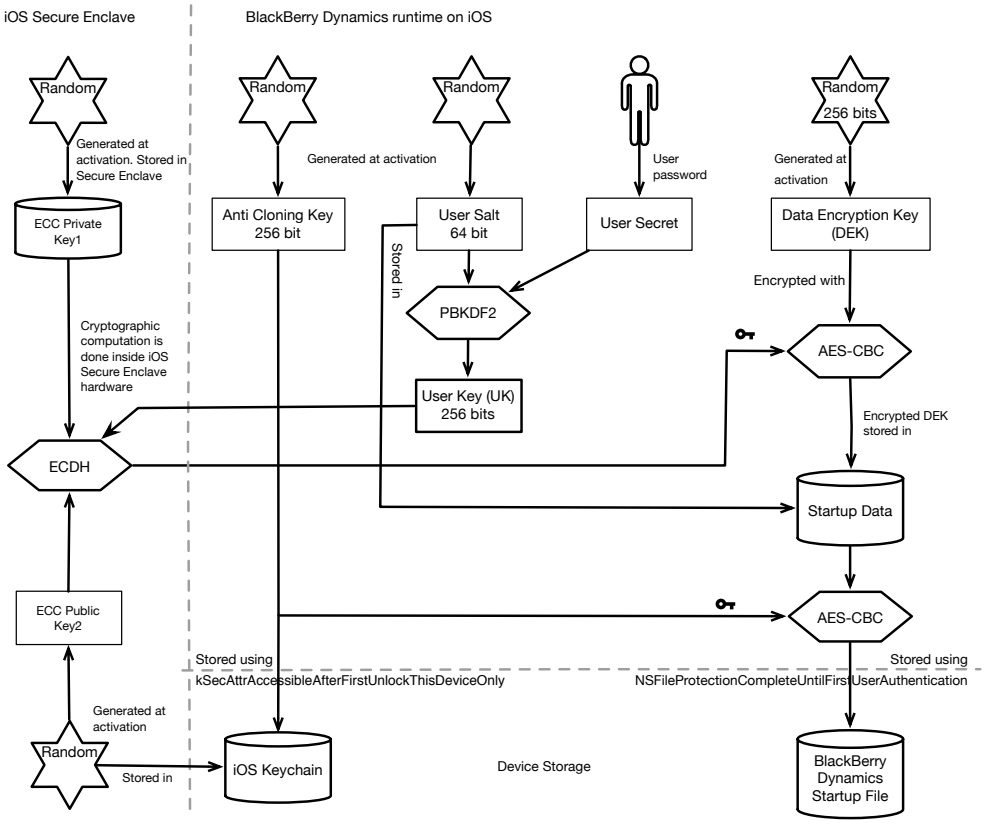



***Figure: Securing Data Encryption Key on iOS***

## macOS

The figure below shows how Data Encryption Key is secured on a macOS device.

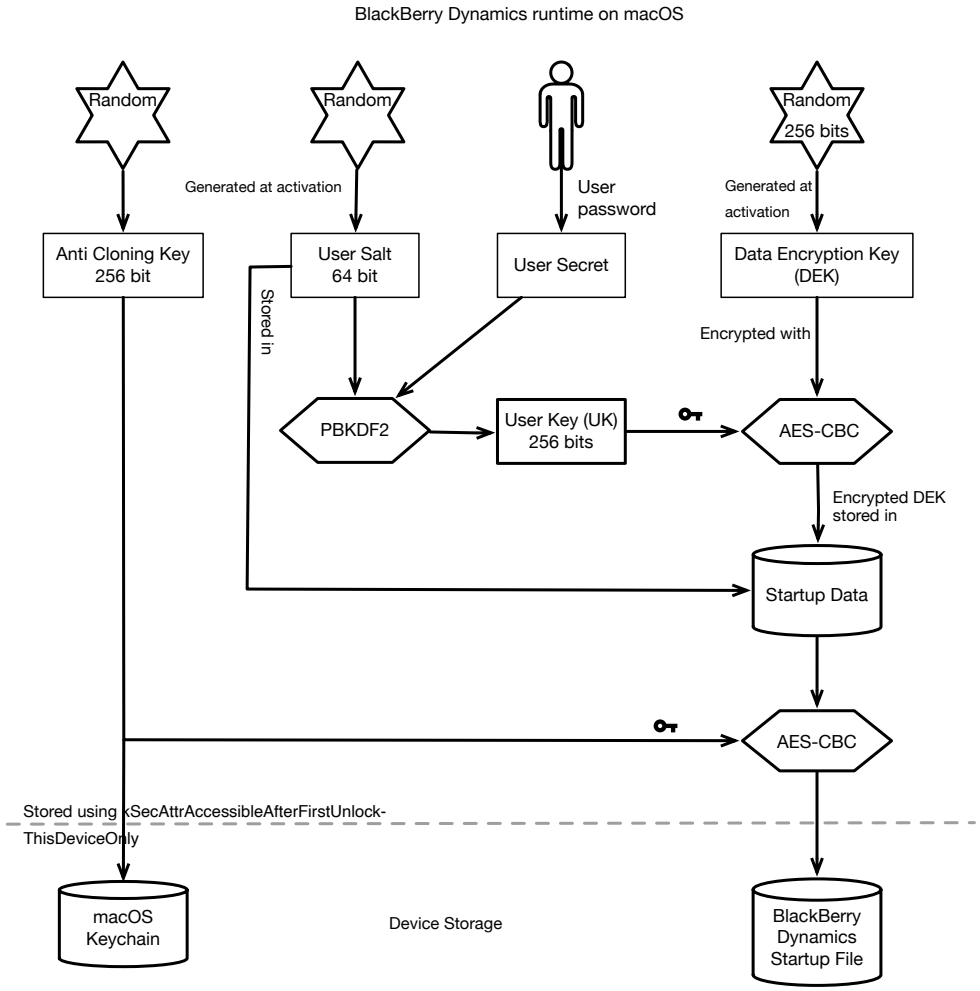



***Figure: Securing Data Encryption Key on macOS***

## Windows

The figure below shows how Data Encryption Key is secured on a Windows device.

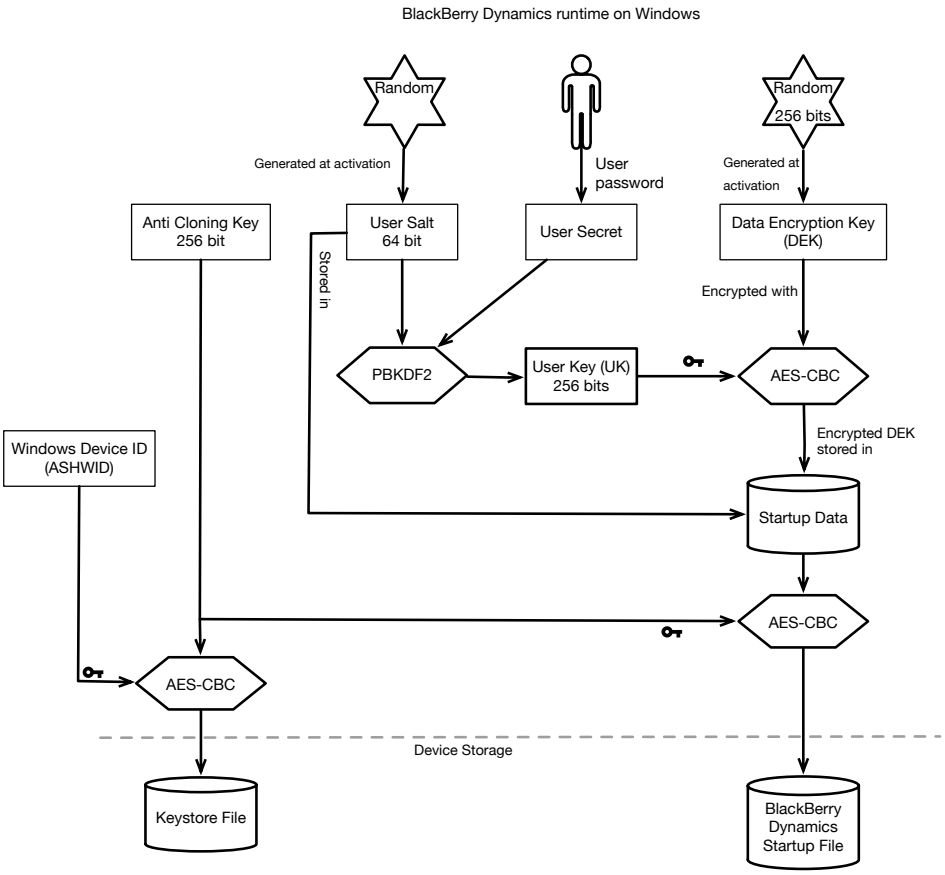



***Figure: Securing Data Encryption Key on Windows***

## Biometric User Authentication

BlackBerry Dynamics runtime supports user authentication based on fingerprint recognition system from Apple Touch ID and Face ID on iOS devices, Android Fingerprint and Samsung Pass on Android devices. Collectively these features are called biometric authentication below. Biometric authentication is not supported on macOS and Windows OS. When this feature is enabled, user can perform biometric authentication instead of providing password. Dynamics runtime support of biometric authentication supplements user password.

No work is required from the Dynamics app developers. Operating system's standard behavior of biometric authentication is not changed by Dynamics runtime. This feature also does not impact Authentication Delegation feature. Administrator can control this feature from the management console in the security policy settings. No biometric meta-data is received by Dynamics runtime. Operating system only informs Dynamics runtime if the biometric authentication has succeeded or failed.

Administrators can

- Enable biometric authentication. This setting allows Dynamics runtime to use biometric authentication when app requires re-authentication (i.e., app is already running) such as idle lock, restore from background, easy activation etc.
- Enable biometric authentication after cold start. This is the case when device is shut down and started or app is started for the first time. If this is not enabled, user will be prompted for password in these situations.
- Provide a time interval after which user will be prompted for password (even when user has already performed biometric authentication)

If biometric authentication is enabled by the administrator, user is still required to set password after the app activation. In addition, if the policy allows, users can choose to enable biometric authentication. If the device passcode is changed, or when the biometrics enrolled on the device OS is changed, Dynamics runtime will ask for the user password.

## Android

On Android devices when biometric authentication is enabled for cold start, a new 256 bit Cold Start Key (CSK) is generated in the Android keystore with userAuthenticationRequired property and saved in the keystore. This CSK is used as depicted in the diagram below. This is in addition to how user secret is handled and used as shown in the User Authentication and Key Storage section (Figure: Securing Data Encryption Key on Android).

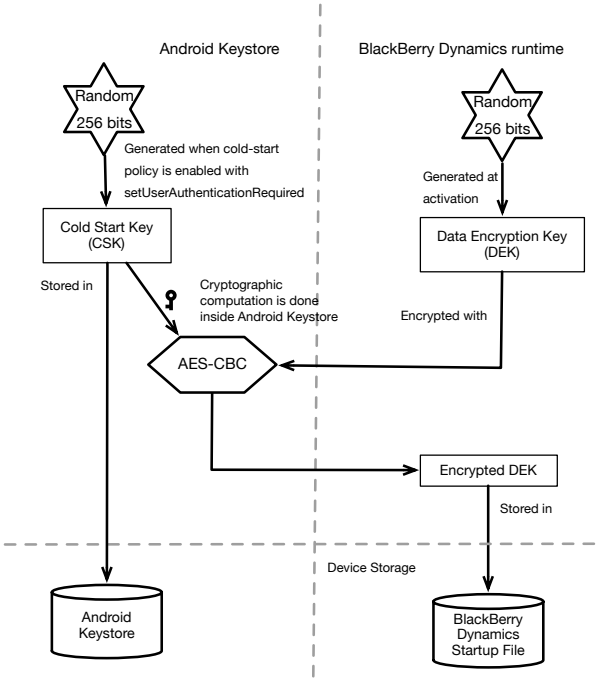



***Figure: Biometric authentication with cold start on Android***

### iOS

When Face Id/Touch ID authentication is enabled for cold start on iOS devices, the User key is saved inside the device keychain with attribute kSecAttrAccessibleWhenPasscodeSetThisDeviceOnly. If the passcode is removed/disabled, User key is removed from the device keychain.

### No Password

Administrators also have the option to not require the password to be set by BlackBerry Dynamics app users. When this "No Password" feature is enabled, the Dynamics runtime creates 256 random key called User Secret Replacement (USR) and is used in the same way as user password. USR is saved inside the Startup file.

## Idle Lock

The BlackBerry Dynamics runtime supports idle lock feature. When the user has not interacted with the app for a specified amount of time as set by the administrator, the Dynamics runtime will display a Dynamics unlock screen to the user. The Dynamics unlock screen is super imposed on the app user interface screen. The app is said to be idle locked and user must authenticate to see the app's user interface again.

While Dynamics unlock screen is displayed, the app is still running; app data is present in the runtime memory; the app has access to the data in the file system, and it can connect to the app server.

## Bypass Unlock

Bypass unlock feature allows the app to display certain user interface screens to the user while idle lock is in effect. This feature can be useful to the apps that require swift user response such as inbound call in a telephony app or immediate storage of external data such as picture or note to the secure storage.

Bypass unlock is only supported on iOS and Android.

The user interface screens that are enabled for Bypass Unlock are specified by the developer when the app is built. Access to Bypass Unlock is restricted and must be requested from BlackBerry. Apps that are granted access to Bypass Unlock are issued a unique signed registration token by BlackBerry. The token must be embedded in the app declaration at build-time.

Developers are required to display the list of user interface screens that are enabled for Bypass Unlock feature to the administrator using app specific policies in the UEM console. In addition, developer is recommended to provide control to the administrator to turn on/off this Bypass Unlock feature via the same app specific policy.

For more information on how to use Bypass Unlock feature and how to get registration token see Bypass Unlock: Application Developer Guide

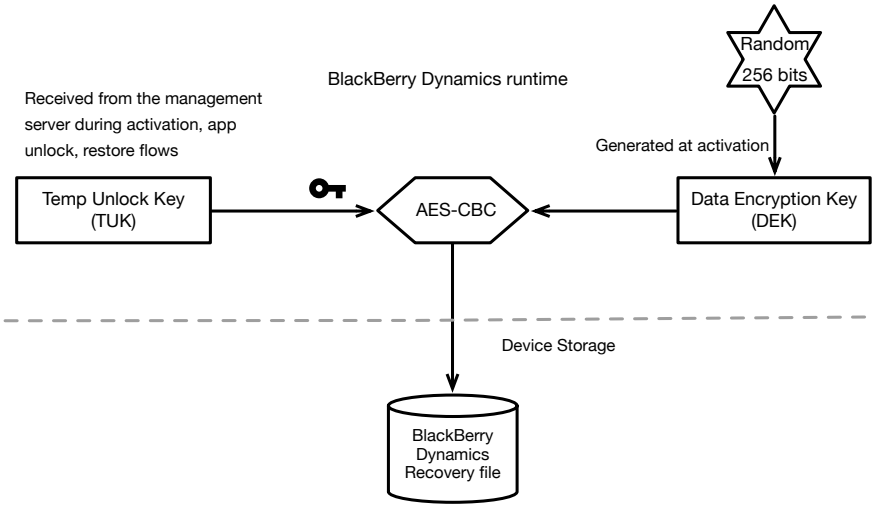
# App Unlock and Restore

A Dynamics app can be unlocked if a user has forgotten the password, or the app has been remotely locked by the enterprise IT administrator. To unlock the app user needs an app unlock key issued by the management server.

This app unlock key can only be used to unlock the container for which it was created and will expire in 24 hours. The key is used to authenticate the user and identify the instance of a Dynamics app to the management server in the unlock process.

The Dynamics app then uses the same process as described in App Activation. It authenticates itself to the management server using app unlock key and gets the Temp Unlock Key (TUK, 256bit) from the management server and unlocks the Dynamics app (i.e., decrypt the Data Encryption Key, DEK).

After the initial enterprise activation, a copy of DEK encrypted with TUK is saved in the *Recovery* file. TUK is created for each container by the management server during the activation process. The same TUK is sent to the client during the app unlock/restore process.




## App Restore on a New Device

A Dynamics app supports data being restored on a new device. A user who is using a Dynamics app on one device can backup data for the Dynamics app using OS provided services such as iCloud or equivalent. This data can then be restored on a different device running the same OS. However, Dynamics app data is encrypted inside the container. To unlock and restore the data inside the container user needs to get an app unlock key from the management server and input this on the Dynamics app. This is the same key and process as described above.

# Background Authorize

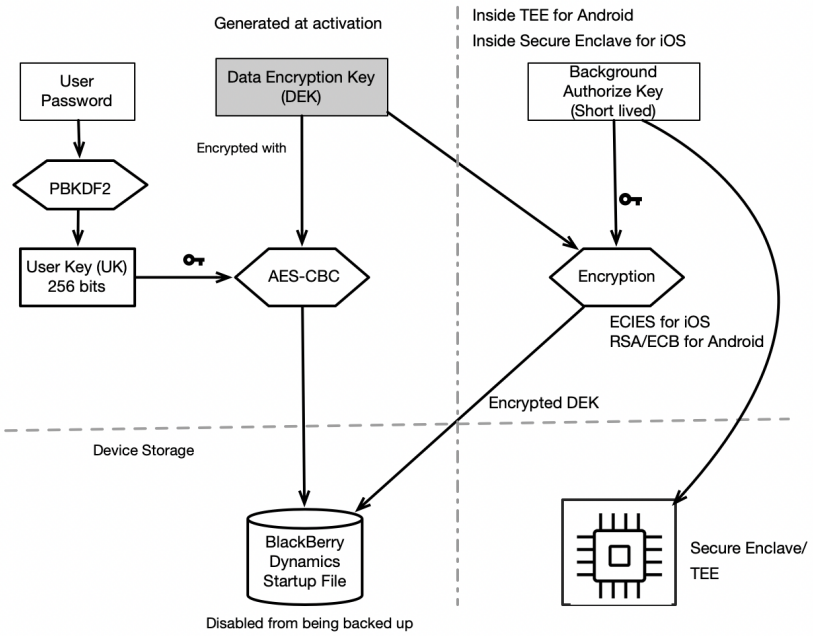
Background Authorize feature enables a recently locked application to utilize the principal Dynamics APIs like secure storage and secure communications when the application is running in the background. This feature is helpful when an application may have stopped (because the operating system unloaded it from runtime memory, or the application has crashed). An application may be started in the background in response to having received an

APNS message (for example, a new email has been received). In this scenario, if the Background Authorize feature is enabled, the application can download the new messages and store them in the secure container even though the user did not unlock the Dynamics container. When the user brings the application to the foreground, they are required to authenticate to see their new messages. User benefits by being able to see the content faster. In the email app scenario, new emails will be present by the time user opens the email app. This feature is supported on iOS platform from Dynamics SDK version 6.0 and on Android platform from SDK version 9.0.

For a Dynamics application to support Background Authorize developer needs to:

- Request to use the feature from BlackBerry and embed a signed authorization token inside their application.
- Define an application specific policy which enables an administrator to turn on this feature and specify duration for which this feature is enabled. Duration is the time since the user last authenticated to the Dynamics container. For example: 12 hours, or 1 day.
- Dynamics SDK checks if the policy has been enabled by the administrator and app has the signed token from BlackBerry authorizing this feature to the application.

To unlock a Dynamics container the user key is required as described in "User Authentication and Key Storage" section. User key is derived from user password or provided by Auth Delegate application. When Background Authorize feature is allowed and the container is unlocked, a short-lived Background Auth key is generated in the iOS Secure Enclave (inside TEE on an Android device). Now Background Authorize feature is active. Background Auth key is used to encrypt Data Encryption Key (DEK) and this encrypted DEK is saved in the BlackBerry Dynamics Startup file. When Background Authorize time has elapsed (from the time user authenticated the container), Background Auth key is removed from the iOS Secure Enclave (Android TEE) and Background Authorize feature becomes inactive. The security risk of this feature is same as No Password feature during the time this feature is enabled.



# Sharing Data between Dynamics Apps on the same Device

Using Shared Services, a Dynamics app may securely send and receive data with another Dynamics app installed on same device. The security model is described in Shared Services Framework.

### Data Leak Prevention

Using policy rules in their management server, administrators can prevent users from copying data from a Dynamics app to a non-Dynamics apps on the device, as well as prevent dictation, custom keyboards, screen capture and AI Writing Tools. Capabilities available on each platform vary and can be seen the management console. In the UEM console, the **Data Leakage Prevention** section of the security policy controls these capabilities. In the UEM server console, these settings are managed in the BlackBerry Dynamics profile.

Secure cut-copy-paste is accomplished by encrypting the data saved in the copy/paste buffer (using AES 256, CBC) along with keyed hash (HMAC-SHA512). All containers belonging to a user are provided same copy-paste key by the management server.

# Device Integrity

BlackBerry Dynamics runtime performs various checks to verify the device and application integrity. On Android and iOS, admin can enable rooted/jailbreak detection using the compliance policies. Similar jailbreak detection policies are not available on Windows and Mac platforms as users on these platforms most likely to have administrative privileges. Therefore, desktop applications lack some security protections offered by closed mobile platforms. Please check the desktop application admin guide for available mitigations. Dynamics runtime also supports Android SafetyNet attestation checks and iOS application integrity verification. Rules for various compliance checks are pushed to clients by the management server and updated dynamically.

# FIPS Compliance

Federal Information Processing Standards (FIPS) are U.S. government regulations regarding computing and computing security (see FIPS Pub 140-2 by NIST for more information). FIPS compliance is supported on Android and iOS platforms and can be enabled in a security policy. BlackBerry Dynamics SDK uses FIPS validated crypto module called OpenSSL FIPS Object Module.

When an admin enables FIPS compliance in a policy, the major effect is on associated apps. Enabling FIPS compliance enforces the following constraints in conformance with FIPS:

- App must be built with FIPS enabled as documented in the references below. Apps that do not have FIPS enabled will not conform to the security policy and are blocked on user devices. Users must contact an administrator to be unblocked. Administrators can unblock the user by disabling FIPS compliance in the policy at either the user level or the app level.
- MD4 and MD5 are prohibited by FIPS, which means that access to NTLM- or NTLM2-protected web pages and files is blocked. Wrapped apps are blocked. In secure socket key exchanges with ephemeral

keys, with servers that are not configured to use Diffie-Hellman keys of sufficient length, Dynamics retries with static RSA cipher suites.

## Dynamics App Connections

Dynamics application can connect to services behind the enterprise firewall without requiring VPN. Administrator can control which services Dynamics applications can connect to. Connections to services on internet can also be routed via the enterprise network. Administrator can also specify which root certificate authorities are trusted for the TLS connections. See "Enterprise Connectivity" section for additional information.

All connections from Dynamics app to BlackBerry hosted services use only TLS v1.2 with strong cipher suites. Only selected set of well-known root certificate authorities are trusted by the Dynamics runtime for connections to BlackBerry hosted services.

## WatchOS

BlackBerry Dynamics supports secure communication and storage on WatchOS when used alongside a Dynamics companion application running on iOS. WatchOS implements processes to securely pair the watch and handheld, restrict when the watch may be unlocked and secure communications. See 'System security for watchOS' on Apple website.

Dynamics provides the following additional protections:

- The Dynamics Watch app is paired with the Dynamics iOS companion app by the user confirming a 6 digit code displayed on both screens. This process facilities a Diffie-Hellman key exchange where the exchanged public keys are stored in the iOS Secure Enclave.
- Messages exchanged are encrypted with a ECC 256bit key-pair which is generated in the Secure Enclave. Encryption uses the private key generated on the originating Device (watch or handheld), and decryption uses the public key which was sent to the other device during initial pairing.
- Files that are sent between the WatchOS app and iOS companion app are encrypted in transit and decrypted on receiving device using the same public/private keys and are stored in Dynamics Secure Container.

# Enterprise Servers

There are two types of BlackBerry Dynamics enterprise servers managed by customers: a. BlackBerry UEM server (management server) b. BlackBerry Connectivity Node are installed by administrator behind their network firewall. BlackBerry Proxy server is a component of BCN. BlackBerry hosted UEM server on the public internet (BlackBerry UEM Cloud) can also be used by the customers.

These on-premises Dynamics enterprise servers only make outbound connections to the BlackBerry Cloud, and do not require inbound ports to be opened in the firewall. Both these servers work with outbound proxy servers if required.

## BlackBerry UEM

Both on-premises UEM server and Cloud UEM server, offer trusted end-to-end security and provide the control that organizations need to manage all endpoints and device ownership models. UEM server manages users, devices, policies, Dynamics apps, and BlackBerry proxy servers. Collection of all UEM servers and proxy servers constitute a Dynamics deployment. A Dynamics deployment typically have multiple instances of UEM server.

The IT administrator uses the UEM server to create and manage users, provision activation password/keys, manage access, policies, and app entitlements. The administrator can create different policies based on security needs for different users. Within a security policy, the administrator controls various requirements and conditions such as user authentication, user password strength, auto lock, copying of data outside of the secure Dynamics container, compliance with OS version, hardware manufacturer or models, jailbreak and rooted detection. To find more information about security and compliance policies please see Managing BlackBerry Dynamics apps in UEM admin guide.

The IT administrator can also use the UEM server to define and deploy apps to groups of users or devices and set app-specific policies. Using a UEM server an IT administrator has precise control over which app servers and domains a Dynamics app can establish a connection with.

An IT administrator may install multiple UEM servers for load balancing and fault tolerance. In a global deployment client randomly picks which UEM server to connect to. In a regional deployment BCP server forwards the connection to one of the UEM servers.

### Administrator Roles and Rights

With Role-Based Access Control (RBAC), your organization can easily restrict access to BlackBerry UEM functions and offload a group of tasks to certain administrators or help desk support specialists. Role privileges are enforced globally across all UEM servers in your deployment, so administrators have the same rights and access for any UEM server console they log into. Custom roles can be created by giving them a smaller set of rights.

An administrator can have multiple roles. In this case, the administrator inherits the cumulative rights granted to all roles to which the administrator's account belongs.

UEM server administrators are authenticated against enterprise Microsoft Active Directory (AD). Administrators have option of providing their AD password or use Kerberos single sign-on in an on-premises UEM deployment.

## Installation

Installation of the first BlackBerry UEM server requires a UEM SRP identifier and a SRP authentication key. During installation, an overall administrator (super-admin) is setup. The super-admin must already be present in the corporate directory. Additional administrators, with different roles, can be added by the super-admin, but they must also be present in the corporate directory. Only designated administrators can log in to the UEM console and manage users, containers and apps. All actions taken by administrators are logged for audit purposes.

All policy, configuration, and container information are saved by the UEM server in the database provided by the IT administrator during the UEM server installation. Access to the database server is authenticated.

To add additional UEM server instances to an existing deployment, admin needs to provide the URL and credentials of the UEM server database created when first UEM server instance was installed.

## Root Certificate Authorities

At the time first on-premises BlackBerry UEM server is installed, UEM server generates two 2048bit RSA public-private key pairs. 1. UEM RSA root CA, 2. Dynamics root CA. Public key of each key pair is self-signed. Each root CA signs corresponding intermediate CA certificate. UEM intermediate CA certificate is then used to issue TLS certificate for console, BCN etc. Dynamics intermediate CA is used to issue certificates for the Dynamics containers. See section on Certificates for additional information.

# BlackBerry Proxy Server

A BlackBerry proxy server, which is a component of BCN, enables a Dynamics app to connect to application servers that are inside the enterprise firewall. Proxy server connects to the app server over TCP when requested by Dynamics applications.

BlackBerry proxy servers are typically grouped in a cluster and used to provide connection to collection of application servers or servers in a sub-domain. An administrator can specify which proxy cluster will be used to complete connection at sub-domain level or at individual application server level. In a global deployment client randomly picks a proxy server in a cluster to connect to. In a regional deployment BCP server forwards the connection from a Dynamics apps to one of the proxy servers in a cluster.

## Installation

Installation of BCN requires an activation file from the UEM server. This file contains BCP address to reach to UEM server, and password to use as secret in the activation flow. During the activation, BCN receives a x509 public certificate signed by the UEM RSA root CA which it uses to authenticate its connection to the UEM server. See section on BCP

to learn how BCP is used to connect to the UEM server.

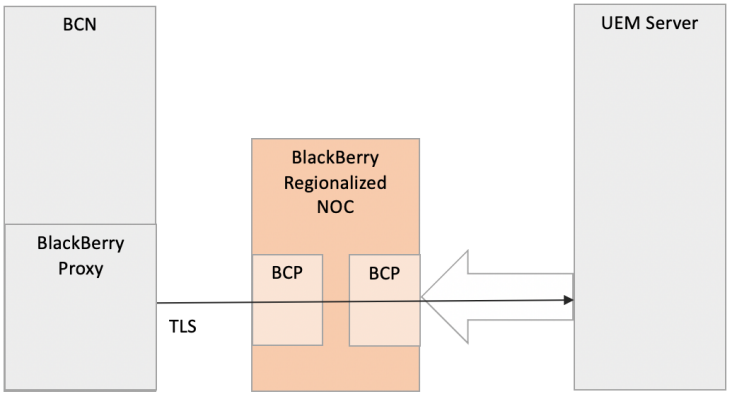



***Figure: BlackBerry Proxy to UEM connection***

# BlackBerry Cloud

The BlackBerry Cloud has multiple types of services running on the internet, hosted by BlackBerry. Some of these services are global and others are regional i.e., similar group of servers running in multiple physical locations across the world.

## Global Services

### Catalog Service

This component provides application management and entitlement to the UEM server and to Dynamics applications. When ISVs create applications, they are registered with the Catalog service using the BlackBerry portal. When enterprise create custom application, application is registered with this service using the UEM console.

The Dynamics runtime checks for entitlement of the app for its user and deletes or locks the app when entitlement for the installed app is removed. In a Dynamics deployment enabled for regional services, app entitlement check is performed by Dynamics runtime against its UEM server.

### BlackBerry Enterprise Identity

The BlackBerry Enterprise Identity (EID) acts as an identity provider to the BlackBerry services running on the internet. The identity provider issues authentication token to the Dynamics apps to connect to optional additional services such as BlackBerry Mobile Threat Detection, BlackBerry Persona Mobile etc. The identity provider issues authentication token based on a public certificate issued by the BlackBerry Certificate authority to each Dynamics application. The identity provider also provides optional 2FA service in conjunction with UEM server and UEM client application to the Web applications.

### BlackBerry Push Notification service

This service is used by Dynamics apps internally to send and receive notification from the BlackBerry services while the application is running. All Dynamics apps maintain a persistent connection to this component of the BlackBerry Cloud.

### Activation/Discovery Service

This service assists in activation of Dynamics application. This service identifies which UEM deployment a user is enabled for activation.

### Relay Service

A Dynamics application establishes connection to their enterprise BlackBerry proxy server using the Relay service. Both Dynamics apps and proxy servers connect to this component by using the Good Relay Protocol (GRP) over TCP. See **Error! Reference source not found.**. Relay service is a global service.

# Regional Services

## BlackBerry Communication Proxy (BCP)

The BlackBerry Communication Proxy (BCP) is a service that enables an entity behind a firewall to receive incoming TCP connection from applications outside firewall. This is used by BlackBerry proxy server and Dynamics applications to create secure connection servers inside enterprise firewall when regionalization is enabled in the UEM. The BCP service is regionalized. Each region has its own BCP service. Each UEM deployment is assigned to a region and connects BCP service in its region.

# App Activation

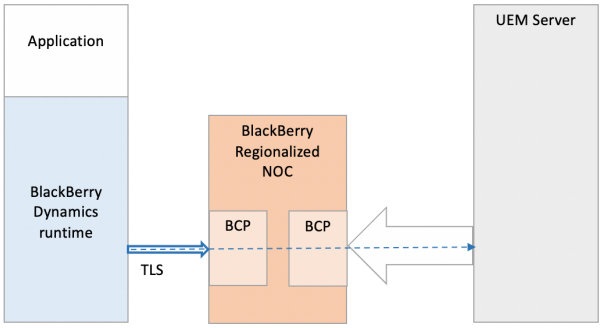
App activation is the process by which a Dynamics app is activated with both the enterprise's UEM server and the BlackBerry Cloud. As a result of activation, the Dynamics app can be managed by the UEM server and has permissions to connect to the appropriate application servers inside the enterprise firewall. A Dynamics runtime receives the keys and configuration data to securely communicate with the BlackBerry Cloud and BlackBerry servers in the enterprise. A Dynamics app can be activated using user's corporate password (for example, AD password) or activation password or a 15-digit access key.

## Activation Password based Activation using BCP

User is issued an activation password (or QR code) by the UEM server. User inputs their email address and activation password in Dynamics activation screen. Dynamics runtime completes activation flow using BCP service by connecting to the UEM server.

Dynamics runtime discovers the UEM server identity by performing a lookup against the activation service using user identity. A Dynamics runtime instance then establishes a TLS connection to the BCP server and requests connection to the UEM service activation endpoint. BCP server proxies the request to the UEM server. TLS connection is terminated at the BCP server. Dynamics runtime performs a key exchange using SPEKE protocol with Elliptical-curve cryptography. This negotiated key is then used to encrypt activation messages end to end between Dynamics runtime and the UEM server. During this process Dynamics runtime creates a public-private key pair on the device and sends the CSR to the UEM server to issue a signed public certificate (BCP client certificate). In addition, in this exchange Dynamics runtime receives master session key, BlackBerry proxy information, connectivity profile, Application server address and other configuration data.

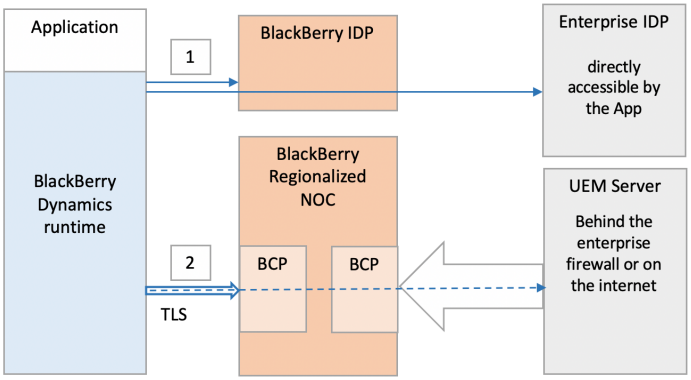
When an **activation password** is issued, user identity and the UEM server identity is registered in the activation service in the BlackBerry Cloud. Password or its hash is not sent to the BlackBerry activation service. This registration is purged after a defined time period. There is only one activation password in the UEM server for a user. It can be used to activate one application or multiple applications based on the setting. Activation password expires after a defined time. Activation password can be set by the user or admin or auto generated. Administrator controls the complexity of the password. The user identity and password are also displayed to the user in form of a QR code on the console or sent in the email. It is also possible to perform activation without registering information with the activation service. In this scenario activation process requires address of the UEM server which must be read from QR code or input by the user manually. Dynamics SDK version from 8.x.x onwards support this feature.




## Activation using Enterprise IDP

In this activation mode a user provides their enterprise (Active directory) credentials to complete the activation of the app. A customer should provide their own identity provider (IDP) that supports OpenID Connect. During activation the end user is redirected by BlackBerry's IDP to the customer's Enterprise IDP where the user authenticates. The Enterprise IDP issues a JWT which is verified and accepted by the BlackBerry IDP. The BlackBerry IDP then issues its own JWT security access token to the client, which is sent in the activation request to the UEM server. The JWT is bound to the public certificate presented to the BlackBerry IDP by the client. The BlackBerry Dynamics runtime must present proof of possession of the private key of the public certificate bound in the JWT presented to the UEM server. In response the UEM server returns the activation payload. The subsequent steps mirror the same activation flow as using BCP which is described in the previous section.

A UEM administrator needs to enable this feature and set their IDP related configuration in the UEM. This feature is supported on iOS and Android from Dynamics SDK version 9.1 and requires UEM server version 12.14 and above.




## Access Key based Activation

UEM also supports activation using a 15-digit access key for the older clients that don't support above mentioned activation modes. Access-key based activation will be phased out in future. User provides email address and

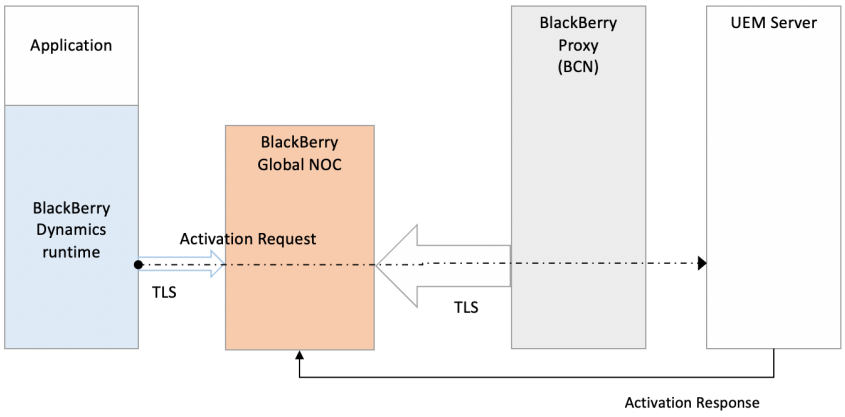
access-key in the activation screen. Dynamics runtime exchanges the activation message with UEM server via BlackBerry Cloud and BlackBerry Proxy server, which are end to end encrypted.

When an **access key** is issued, user identity and access key hash is registered in the BlackBerry activation service and is associated with the UEM server. This registration is purged after a defined time period configured in the UEM server. PBKDF2 hash of the access key (256 bit long HMAC-SHA512, 16384 iterations) is sent to the activation service. The access key is only available to the user and the UEM server. Access key expires after its use and can-not be used to activate other Dynamics applications. The access key is 15-digit random alpha-numeric data [a-z, 0-9] (same strength as random 72bits) and is delivered in an email.

After user inputs user id (email address) and access key in a Dynamics activation screen, Dynamics runtime authenticates to the activation service using user id and hash of the access key and receives information from BlackBerry Cloud such as UEM server address, list of BlackBerry proxy servers, Relay server, master link key etc.

1. Dynamics runtime establishes a shared encryption key with UEM server for an end-to-end secure channel by authenticated ECDH parameter exchange. This eliminates the possibility of someone in the BlackBerry Cloud being able to execute a MITM attack.

2. Over the end-to-end encrypted message channel, a Dynamics runtime receives enterprise provisioning data from the UEM server.




## Easy Activation

Easy Activation simplifies the provisioning process by allowing a Dynamics app to "hand off" activation to an app already activated on the device that can act as the activation delegate. User does not to input their AD credentials or activation password or access key. An user only performs local authentication on the device by providing dynamics container password or by performing biometric authentication. Easy Activation is supported for Android and iOS platforms.

When a user installs a new Dynamics app and starts the activation process, the new app will check for the availability of a suitable installed activation delegate app. If an activation delegate is not discovered, the user will

be prompted to use the standard activation process.  If a suitable activation delegate app is detected, the "Easy Activation" setup option is also presented to the user.

When "Easy Activation" option is selected by the user, activation delegate app is launched. User is asked to provide the container password by previously activated app to authenticate the request. After authenticating the user, activation delegate app requests an activation key. This activation key is returned back to the app that requested easy activation. App being activated will now complete the activation process in the same way as if user entered the access key depending upon Dynamics SDK version and UEM server version. When all components are new SPEKE flow will be used.

The new Dynamics app and the activation delegate app exchange the request and response over secure ICC channel (as described in Secure ICC Handshake). In its request for an activation key, the new Dynamics app also sends a randomly generated nonce (term in cryptography meaning "a number or bit string used only once "). The activation key received in this process can only be used along with the nonce used to create it. Thus, only the Dynamics app that requested the access key can complete the activation using it.

# Enterprise Connectivity

Dynamics Platform provides capability for Dynamics applications to connect to the services behind the enterprise firewall without VPN or opening the firewall. Administrator can completely control which services a client application can connect to. This is managed by creating Dynamics Connectivity profile and routing rules. A connectivity profile allows administrator to specify if client can connect to a given server or a server in a sub-domain on internet directly or needs to use a BlackBerry proxy server to connect to it. Administrator can also specify if a server or all servers in a sub-domain should not be allowed.

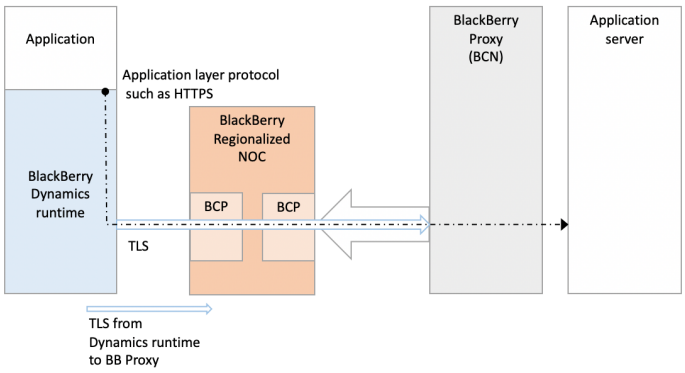
Dynamics runtime will use BCP or a proxy in DMZ (Direct Connect) path to establish connection to a BlackBerry proxy for connecting to an application server (typically on-premise). Administrator can also specify if proxy server should use another outbound web-proxy for completing connections to an application server. A web-proxy can also be specified via means of a PAC file.

## BCP

The BCP is a service that enables an entity behind a firewall to receive incoming TCP connection from applications outside firewall. BCP is used by Dynamics applications and BlackBerry proxy servers to create secure connection to servers inside enterprise firewall. This service is also used by UEM server and proxy server to connect to each other. UEM servers and proxy servers registers with BCP to receive connections for a given type of service such as "enterprise connectivity", "container management", "proxy server management" etc. This is a long-lived persistent connection. The BCP service performs health check of each BCP service node. If a BCP service node does not respond as ready to accept new connections, it is removed from the pool of nodes available for new connections.

BCP service registration from a UEM server requires SRP ID and key for authentication and server is issued an auth-token good for a certain duration. Auth-token is refreshed on expiry by BCP.

BCP service registration from a BCN requires a certificate signed by its UEM server. UEM server must be running at the time BCN does service registration and renews the registration, since UEM server verifies the authentication of BCN servers.




***Connections using BCP Service***

When a Dynamics runtime requests for a connection of a service type to a BCP, connection request is forwarded to the corresponding service endpoint (on UEM server or proxy server) over an existing connection tunnel to BCP (from UEM server or proxy server). If multiple service provider instances are registered for a service type, BCP picks a random connected service endpoint and forwards the connection to it. End to end TLS connection is established between the Dynamics runtime and the BCP service endpoint. A Dynamics runtime is provided a root certificate authority to trust and the claims to look for in the public certificate of the server. A Dynamics runtime does not authenticate to the BCP server. Authentication of the client is performed by the UEM server or BlackBerry proxy via client certificate-based authentication in TLS handshake. Dynamics runtime uses BCP client certificate for this.

UEM deployments use BCP based connectivity by default.

### Connection to a BlackBerry proxy server

A Dynamics client establishes connection to BlackBerry proxy server over the BCP connection. The connection is end-to-end encrypted between the Dynamics runtime and the proxy server using a symmetric session encryption key that is not known to the relay server. Session encryption key and session authentication token expire periodically and are renegotiated. Key exchange process uses Master Session Key (the key exchanged during app activation). HMAC-SHA512 message authentication code is used to ensure integrity of the key exchange.

## Direct Connect

The BlackBerry Dynamics Direct Connect feature (referred to as Direct Connect) provides Dynamics clients ability to connect to BlackBerry proxy servers without connecting through the BlackBerry Cloud.

This feature benefits following entities:

- Users whose Dynamics apps experience long connection establishment latency and low throughput due to large TCP round trip time (RTT) between Dynamics apps and the BlackBerry Cloud, or between BlackBerry proxy servers and the BlackBerry Cloud.
- Organizations which have additional data privacy requirements which restrict user data leaving from their networks and do not want their user and enterprise data to be routed via the BlackBerry Cloud.

An enterprise administrator can enable this feature by configuring Direct Connect for each BlackBerry proxy server in the server settings section in the UEM console. When this feature is enabled, Dynamics clients will make connection to the proxy server directly instead of connecting via BCP or the Relay service to connect to an application server inside the enterprise network.

#### Deployment Models

There are two deployment configurations for Direct Connect.

1. A BlackBerry proxy server is deployed in the DMZ where it is reachable from the Internet. DMZ is also known as perimeter network.

2. A HTTP Proxy server installed in the DMZ which forwards the connection from a Dynamics app to a BlackBerry proxy server. This server is referred to as a DMZ proxy.

In both configurations a Dynamics client establishes an TLS connection to the BlackBerry proxy server and authenticates to the proxy server as described in Relay service section above. However, in the case of Direct Connect, encryption is performed by TLS layer which uses TLS 1.2 and negotiates ECDHE-RSA-AES256-SHA384 cipher suite. The BlackBerry proxy server uses the TLS certificate signed by the UEM Dynamics Intermediate CA to authenticate to the Dynamics client. Administrators can provide their own enterprise TLS certificate to be used by BlackBerry proxy server.

Connection to the enterprise app servers will never be attempted via BlackBerry Cloud when configured for Direct Connect.

When connecting to the UEM server, BCP is preferred path when regionalization is enabled. When regionalization is not enabled UEM server will failover to use the Relay service in BlackBerry Cloud if Direct Connect path is not working.

The figure below shows the protocols in use for Direct Connect.

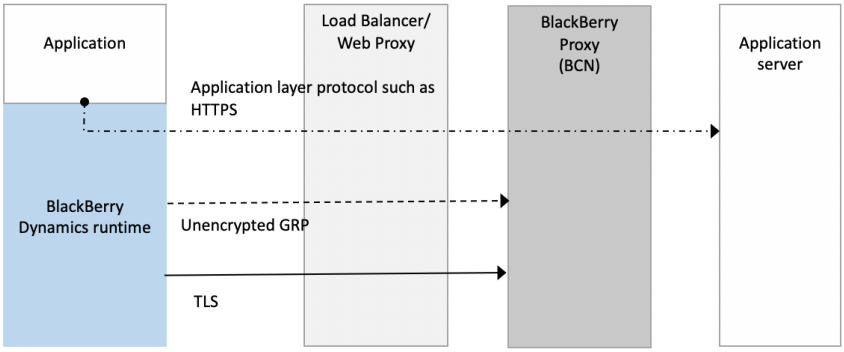



***Figure: Layers and protocols in Direct Connect***

## BlackBerry Proxy in the DMZ

In this configuration, an enterprise admin installs a BlackBerry proxy server in the DMZ. The proxy server must be reachable from the Internet. All enterprise app servers used by the Dynamics application must also be reachable from the proxy servers in the DMZ. Externally reachable proxy server hostname can be set in the UEM console. Administrator can assign different externally visible hostname to each proxy server node or assign one hostname and use a load-balancer in front of proxy servers.

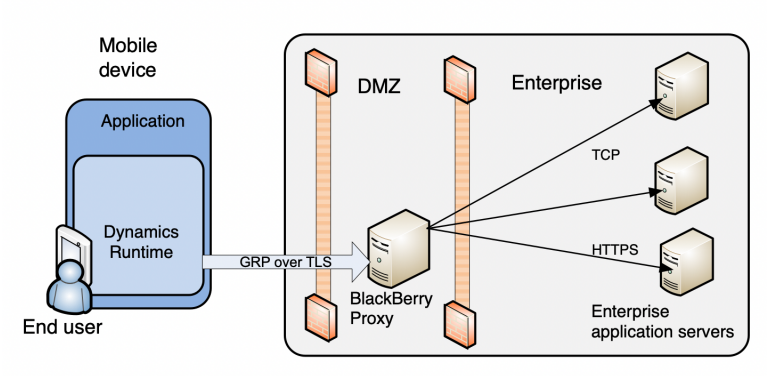



***Figure: BlackBerry proxy in the DMZ***

## DMZ Proxy

In this configuration, an enterprise admin installs an HTTP forward proxy server in the DMZ that supports HTTP CONNECT. The BlackBerry proxy server would remain inside the internal network as in a typical Dynamics deployment. In this model, BlackBerry proxy servers would be reachable from the DMZ proxy as opposed to all application servers being exposed to the DMZ in the previous model.

Dynamics clients will first make an HTTP CONNECT request to the DMZ proxy and request a connection to the BlackBerry proxy server. No authentication is done against the DMZ proxy by default. The DMZ proxy completes this connection to the proxy server. Then Dynamics client establishes a TLS connection to the proxy server.

The enterprise admin must provide the FQDN of the DMZ proxy in the UEM console and associate the DMZ proxy with the BlackBerry proxy server on the server settings screen. DMZ proxy could be a load balancer. Load balancer could be optionally configured to challenge the Dynamics client to perform client certificate-based authentication in the TLS connection. This would require load balancer to terminate TLS and start a new TLS session towards the BlackBerry proxy server.

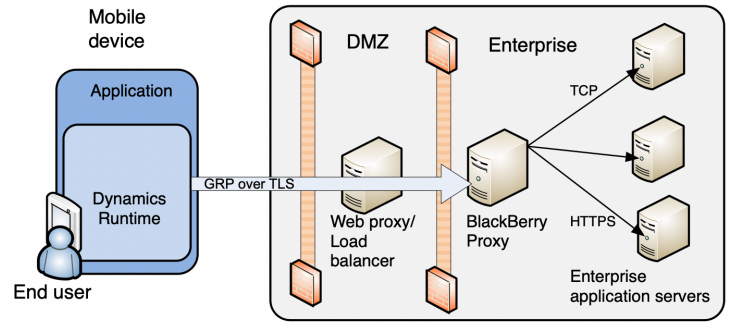



***Figure: DMZ Proxy***

# Summary of Connection Methods

Table below summarizes the different connection methods supported by the BlackBerry Dynamics.

| Connection method | Client authentication performed by | Encryption | DMZ connection requirements | Intranet connection requirements from DMZ |
|---|---|---|---|---|
| BCP in regional BlackBerry Cloud | BlackBerry proxy server | TLS | Outbound to a regional BCP server | None |
| Direct Connect with BlackBerry proxy in DMZ | BlackBerry proxy Server | TLS | One inbound IP address per BlackBerry Proxy server | Multiple inbound IP address, one per app server |
| Direct Connect with DMZ Proxy | BlackBerry proxy server.<br>Optionally by DMZ proxy also. | TLS | One inbound IP address per DMZ proxy | One inbound IP address per BlackBerry proxy server |

# Certificates

X.509 certificates are used in the Dynamics deployments for various purposes such as in client and server authentication for TLS connections, SMIME emails. Certificates are used by the application layer logic, by Dynamics runtime, UEM and BlackBerry Proxy server.

## Platform Certificates

BlackBerry Dynamics component uses TLS certificates and client certificates by default. These certificates are issued by root certificate authorities created at UEM server installation time. Dynamics runtime trusts this root certificate authority and is provided with root CA public certificate at the time of activation and is updated over management channel post activation. This section describes various certificates in use and their purpose. Some of the certificates can be changed by the enterprise administrator.

### Certificate Authorities

UEM RSA Root CA: Self signed root certificate, created by each on-premises UEM deployment. Cloud UEM has one global RSA root CA.

UEM RSA Intermediate CA: Intermediate CA certificate signed by UEM RSA Root CA.

Dynamics Root CA: Self signed root certificate, created by each UEM deployment.

Dynamics Intermediate CA: Intermediate CA certificate signed by Dynamics root CA.

### Server-side Certificates

**UEM server TLS certificates** issued by UEM RSA Intermediate CA is used in following connections as server-side certificate

1. TLS connections initiated by browsers to the UEM console. TLS certificate for this connection can be changed by the enterprise administrator.
2. TLS connections initiated by BlackBerry proxy server to the UEM server.
3. TLS connections initiated by Dynamics apps to the UEM server for management channel and application catalog server.

**UEM Server Client certificate** issued by UEM RSA Intermediate CA is used to authenticate to BlackBerry Proxy server when connecting to it.

**BlackBerry Proxy TLS certificates** are issued by UEM RSA Intermediate CA and Dynamics RSA Intermediate CA **and** is used in following connections as server-side certificate

1. TLS connections initiated by Dynamics apps to BlackBerry proxy servers via Direct Connect. Certificate for this connection is issued by Dynamics RSA Intermediate CA by default and can be changed by the enterprise administrator.

2. TLS connections initiated by Dynamics apps to BlackBerry proxy servers via BCP. Certificate for this connection is issued by UEM RSA Intermediate CA.
3. TLS connections initiated by the UEM server to the BlackBerry proxy server. Certificate for this connection is issued by UEM RSA Intermediate CA.
4. TLS connections from the app servers to the BlackBerry Proxy server (to verify BlackBerry Dynamics Auth Token etc.). Certificate for this connection is issued by Dynamics RSA Intermediate CA and can be changed by the administrator.

**BlackBerry Proxy server Client certificate** issued by UEM RSA Intermediate CA is used to authenticate to UEM server when connecting to it.

**BlackBerry Cloud.** Servers in the BlackBerry Cloud use TLS certificate issued by Thawte (well-known external CA) to host the TLS connections.

### Dynamics Client certificate

**ICC Certificate** is issued by Dynamics Intermediate CA to each activated BlackBerry Dynamics app.

1. ICC certificate is used for both client cert and TLS server cert when connecting from one BlackBerry Dynamics app to another Dynamics app
2. ICC certificate may additionally be used as client cert in Direct Connect mode when configured by the IT admin to authenticate to an optional web proxy in the DMZ. If enterprise has issued their own client certificates to the user apps, those will be used as client certificate as prompted by the web proxy server in the DMZ.

**Dynamics Client certificate** is issued by UEM RSA Intermediate CA is used to connect to UEM app catalog server and to the UEM server management channel.

## Enterprise Certificates

An enterprise administrator can control certificates that are trusted for the app client to server connections and can issue client certificates for each user device.

### Trusted Certificate Authorities

An enterprise administrator can specify public certificates that should be trusted for app client to server connections by the Dynamics runtime. In addition, administrator can control if the certificate authorities present in the device OS are to be trusted for the TLS connections. This feature is independent of the MDM control which allows administrator to add trusted certificate authority to the device key store.

### User Certificate Usage

Sometimes a user may need a public-private key-pair to send and receive signed or encrypted emails (SMIME) or to authenticate to an app server instead of a user password.

BlackBerry Dynamics runtime supports user authentication on TLS connections using client (user) certificates. In addition, Dynamics runtime also supports Kerberos authentication in combination with user certificates for pre-authentication (PKINIT). Client App developer does not need to do any implementation work to use this feature.

Additional details about PKINIT and controlling which users can get certificates, can be found in the UEM admin guide.

## User Certificate Enrollment

An administrator can enroll the user certificates (public private key-pair) to a Dynamics runtime from many different sources by creating and assigning certificate profile to the user/device group. A certificate profile can be marked as mandatory or optional by the administrator. For a mandatory certificate profile, Dynamics runtime after activation will require and complete the user certificate enrollment and only after user certificate is enrolled, user can start using the app. For an optional certificate profile, user can cancel the certificate enrollment when prompted (requires Dynamics SDK version 9.1 and above and UEM server 12.14 and above). Once the certificate is enrolled in one Dynamics app, all other Dynamics apps on the same device will share the same key-pair. Sharing of the user certificate from one Dynamics app to another Dynamics app uses secure ICC framework. User's private key is never provided to the app layer.

### Manual Enrollment

The administrator or the user can upload their own public private key-pair in a pkcs12 formatted file from the management console, which is then sent to user's all Dynamics apps (runtime). In addition, user's pkcs12 file is removed from the management server after pre-configured time.

### Dynamics PKI Connection

This method supports creation of custom PKI Connector server by the customer to distribute credentials (key pairs) from their own certificate issuer infrastructure. Your PKI Connector server must implement the HTTP interfaces as documented in the **Error! Reference source not found.** link in References. When this provider is configured, Dynamics apps will automatically make a request to the UEM server after activation. The UEM server makes the request for credentials to the enterprise's Dynamics PKI connector. Optionally, an administrator can require a user to input an additional one-time password (issued by the enterprise certificate issuer infrastructure) on the Dynamics app to get the certificate. This one-time password is sent to the enterprise Dynamics PKI connector for authentication. The Enterprise Dynamics PKI connector returns back credentials in pkcs12 format.

### Microsoft NDES SCEP Connection

Dynamics runtime uses SCEP to perform certificate enrollment directly against Microsoft NDES server. User is not prompted for password. SCEP password is never sent to the Dynamics runtime. Instead, UEM server assists Dynamics runtime with creation of SCEP payloads (see reference: Assisted certificate enrollment). Certificate private-public key pair is generated on the device and CSR is composed on the device.

### Entrust SCEP Connection

Dynamics runtime uses SCEP to perform certificate enrollment directly against Entrust CA server (for example Entrust IdentityGuard server). User is not prompted for password. SCEP password is never sent to the Dynamics runtime. Instead, UEM server assists Dynamics runtime with creation of SCEP payloads. Certificate private-public key pair is generated on the device and CSR is composed on the device.

### Entrust IdentityGuard based Smart Credentials

BlackBerry UEM Client is required. UEM Client interacts with Entrust IdentityGuard to enroll user credentials. Public key is sent to the Entrust server for signing. The user is required to enter or scan a password QR code. Entrust IdentityGuard supports derived credentials. UEM Client shares the key-pairs with other Dynamics applications on the device.

### App based Credentials

Dynamics runtime supports import of user certificates from the Dynamics applications. Enterprises can create their own Dynamics application to enroll user certificate and then export them to Dynamics runtime.

Purebred application and UEM client application make use of this feature to import certificates in the Dynamics runtime. The Purebred solution provides a native client app and a server. The Purebred client app receives user certificates of different types (signing, encryption, authentication) on the device from the Purebred server. On iOS platform, UEM Client on the device requests certificates from the Purebred client app when triggered by the user and saves them inside Dynamics runtime.

However, on Android platform, the Purebred client app adds the user key-pair to the device key store.

### Device Key Store Credentials

In all the above certificate profiles the private key for the user certificate is saved in the Dynamics container and the runtime performs the cryptographic operations using the private key. With the "device key store" profile support, the runtime can support private keys which are outside of the Dynamics container. The runtime will call the iOS (keychain services) or Android system (device key store) APIs to perform cryptographic operations

On Android the certificate private key is placed in the device key store by a device management agent or by a third-party application such as Purebred. On iOS a third-party app (such as Purebred) acts as crypto-token provider and keeps the keys in its own app keychain.

The UEM administrator controls which certificates from the native key store are used via the User Credentials Profile. They can also choose to enable this feature per platform supported.

On iOS this feature requires Dynamics SDK version 10.2 and above and UEM server version 12.16 and above.

# Additional Features

BlackBerry Dynamics provides the following additional features

- Delegate authentication and password security from one Dynamics app to another.
- Securely exchange data between two Dynamics apps.
- Enable IT administrators to manage app specific policies from the UEM server.
- Securely provide a user's identity to an app server.
- Add security and manageability to an iOS app by wrapping.

## Authentication Delegation

The authentication delegation feature allows one Dynamics app to hand off user authentication to another Dynamics app, which is running on the same device. This feature is supported on Android and iOS platforms.

To set up authentication delegation, the IT administrator must identify the app that will act as the authenticator from the UEM console. Any Dynamics app if designated by the UEM administrator can act as the authenticator.

**Authentication Delegate**: A Dynamics app selected by the admin to perform user authentication.

**Active Authentication Delegate**: The Dynamics app that is performing the task of user authentication on the device. A device may have one or more active authentication delegates. Active authentication delegate app also acts as Easy Activation Delegate (see section:

Easy Activation).

### Process for Delegating

To perform authentication delegation, the designated authenticator app must be first installed and activated by the user. When the authenticator app has activated, the user sets a password for it. This password is then used to authenticate the user on all the Dynamics apps on the same device.

#### Setting Delegation

When the next Dynamics app is activated on the same device and requires a password to be set, it invokes the authenticator app to set user credential. The newly activated Dynamics app is informed about the apps that can act as authentication delegates during the activation process. In addition, UEM SERVER provides the native app identifier (bundle ID for iOS, package name for Android OS) to be used, to send the request to the authenticator using secure ICC handshake (as described in Secure ICC Handshake).

The newly activated Dynamics app requests the authenticator app to provide the User Key (defined in User Authentication and Key Storage) instead of asking the user to set a security password. This User Key is then used to secure the contents of the Dynamics app as described in User Authentication and Key Storage.

The authenticator app may prompt the user for a password depending on its own state. The User Key is different for each Dynamics app that requests authentication delegation. The User Key is derived by the authenticator app

using an authentication delegation key and a salt. The salt is the mobile OS specific app address (bundle ID/package name) of the app requesting authentication delegation. The authentication delegation key is sent by the UEM server during activation.

### Using Delegation

When a Dynamics app is started (post activation) or when it is in locked state, it typically asks the user to provide password to authenticate the user. A Dynamics app that has delegated authentication to another Dynamics app will invoke the authenticator app. The authenticator app authenticates the user by asking the authenticator app password. If the authentication is successful, the authenticator app returns the User Key, which is used to unlock the app's Dynamics container.

Authenticator app also ensures the correct Dynamics app gets the User Key by using native OS services as described below.

## Multiple Authentication Delegates

UEM server also allows administrator to specify more than one authentication delegates. This list of delegates is a prioritized list. Multiple authentication delegate feature allows administrator to

a. Migrate from the active authentication delegate app to a new authentication delegate app.
b. Support the case where a given authentication delegate is only available on one platform. Admin can provide authentication delegates available for each platform.

Dynamics runtime uses this list to pick the authentication delegate immediately after activation. It will select the highest priority authentication delegate present at the time to get the User Key and will continue to use that app as active authentication delegate when container needs to be unlocked.

When a new app is activated, which is a higher priority authentication delegate, it will ask the user to set a password. Subsequently, existing apps, when unlocked next time, will update their authentication delegate to the newly activated authentication delegate app. The active (current) authentication delegate must be present at this time and user may be required to authenticate (i.e., enter password) in the active authentication delegate before the switch to the new authentication delegate can be completed.

If the current authentication delegate app is removed by the user, then user will not be able to unlock other apps present on the device (which delegated authentication to the deleted app). In this case it is recommended that user reinstall and activate the deleted app. Alternately, it is possible for the user to request for temporary passwords to unlock each app present on the device and delegation authentication to an alternate app. Apps unlocked in this manner will then immediately delegate authentication to the highest priority authentication delegate present on the device.

Admin can also enable self-authentication. When this is enabled, it acts as lowest priority authentication delegate. If none of the higher priority authentication delegates are present at the time of activation, then app being activated will ask the user to set the password itself.

# Secure ICC Handshake

Secure ICC handshake is used for Authentication Delegation, Easy Activation, and during App Kinetics. Secure ICC handshake establishes a secure channel between two apps on the same device when User Key is being requested for Authentication Delegation or when access key is being requested for Easy Activation.

This process uses following ciphers.

- An ECC curve P-521 is used for ECDH and an ANSI X9.63 key derivation function with SHA-512 as the underlying hash function for symmetric key derivation.
- Data exchanged by the ICC is encrypted by the AES-CBC cipher using 256bit key negotiated using ECDH key exchange.
- The IV is returned in the clear by the Dynamics Authenticator app.

## iOS

Source app uses the openURL API to send the request to the destination app. The destination app uses the source app (iOS app identifier: bundle ID) information provided by the openURL iOS API to identify which app is requesting the service and to send the response back to it.

## Android

On Android, Intent is used to request service from the other app. The source app uses Android's app identifier Package name (sent by the UEM server) to request service by using explicit Intent. Inside the request it sends its own Package Name. The destination app sends the response back to app identifier sent inside the request. Dynamics client depends on the OS to send the response to the identified app.

# Shared Services Framework

The Shared Services framework allows Dynamics apps or servers to expose their services/capabilities to other Dynamics apps. A service that is provided by one Dynamics app may be utilized by another Dynamics app. The Dynamics runtime provides APIs for the Dynamics apps to discover services present on the device or on a server.

- For more information, see BlackBerry Dynamics Shared Services Framework

## App-Based Services

The Dynamics runtime provides the means to securely exchange service requests and responses over a socket connection between service provider and service consumer Dynamics app on iOS and Android platforms.

The process of establishing secure socket connection between the two Dynamics apps is a two-step process as shown below:

1. Exchange certificates and establish a port number using secure ICC handshake.

This process is described in Secure ICC Handshake. After this step both apps know the identity of the other app and possess ICC certificate of the other Dynamics app.

2. Establish TLS connection.
   The service provider app's certificate is used for the TLS connection. The service consumer app presents its certificate for client authentication. Both sides trust only certificates exchanged during secure ICC handshake. The cipher suite for the TLS connection is TLS_RSA_WITH_AES_256_CBC_SHA256. The RSA key length for the certificates is 3072 bits.

### Server-Based Services

The Shared Services framework also enables any Dynamics apps to use services provided by the servers. For an example presence service exposed by presence server could be consumed by an email app or a document reader app built by some other developer.

Entitlement of server-based services is done at user level by binding the service to a BlackBerry Dynamics App ID. This Dynamics App ID need not have a corresponding client app. Any Dynamics app activated for a user can discover the server-side services it has entitlement too. In addition, the Shared Services framework provides the list of app servers that can be used to request the service. The service provider can require user-level authentication before providing the service.

## App-specific policies

This feature allows the IT admin to set policies/configurations which are specific to an app within the UEM console. Changes to an apps policy are tracked by the UEM server and sent to the impacted Dynamics runtime over an TLS connection. App policy definition is published by the app developer in an XML format from the UEM console or BlackBerry Dynamics portal. The app policy definition file is saved in the BlackBerry Dynamics Cloud so that it is available to all UEM servers in an organization.

Additional information about this feature is available in the *Technical Brief: Application Policies*

## BlackBerry Dynamics Authentication Token

The BlackBerry Dynamics platform includes rigorous authentication of the end user. This is used when, for example, identifying whether the user is entitled to run the current app, and when applying security policies. The BlackBerry Dynamics Authentication Token (Dynamics Auth token) mechanism enables apps to take advantage of the authentication processes of the Dynamics platform.

Dynamics Auth tokens can be requested by the Dynamics app on the device after the app has completed activation with UEM. During the app activation users identify is established. Once a token has been issued, the app on the device can send the token to the app server at the back end. The Dynamics Auth token can then be checked by the app server, using a verification service provided by the BlackBerry enterprise servers. If the token is verified, the user's identity (email address), app identifier, app server name (for which token was requested), an optional challenge string is returned to the app server.

Internally, the integrity of the Dynamics Auth token is checked with a security token. This is a keyed hash (HMAC-SHA512) of the contents of the Dynamics Auth token with a key (GRP Auth token) that is only known to the Dynamics runtime and the BlackBerry enterprise servers.

## Kerberos Constrained Delegation

The Kerberos constrained delegation (KCD) feature allows Dynamics clients to support Kerberos based authentication without requiring users to enter their domain passwords. See the references section to learn about KCD.

An advantage of using KCD is that since the user is never asked for their domain password, the user's domain password cannot be stolen while user is typing it on the mobile device. Additionally, in some deployments, the user does not even have a domain password, since hardware-based authentication tokens are used.

The KCD feature must be enabled by an administrator on the UEM console. Additionally, the administrator must configure the Kerberos service account for the UEM server in Active Directory as trusted, for Kerberos constrained delegation, for all the app servers which are to be authenticated by the use of this mechanism. When this permission is set, UEM is able to fetch Kerberos service tickets on behalf of all users for the app servers which are set for constrained delegation in AD. The only restriction is that the user account, the UEM Kerberos service account, and the app service account all must belong to the same domain in order for the UEM to be able to fetch the service ticket.

When a Dynamics client is challenged to authenticate to an app server using Kerberos over HTTP/S, the Dynamics client requests a service ticket for the app from the UEM server. This request is authenticated using Dynamics Auth token. The UEM server, if it is enabled for KCD, attempts to fetch the service ticket from the Ticket Granting Service running on the Kerberos Key Distribution Center (KDC) on behalf of the user, and it sends the service ticket and unencrypted Kerberos session key for the app server to the Dynamics client. The Dynamics client uses these tokens to authenticate to the app server. The service ticket is cached by the Dynamics client as specified in the service ticket itself and used in future requests to the same app server. If the Dynamics client is unable to get the service ticket from the UEM server, it prompts the user to provide their domain password and completes Kerberos authentication to the app server. This feature can be applied selectively for some app servers, if desired.

# References

- BlackBerry UEM
- Developing BlackBerry Dynamics Apps
- Managing BlackBerry Dynamics Apps
- Technical Brief: Inter-Container Communication
- Technical Brief: Application Policies
- Bypass Unlock: Application Developer Guide
- KCD overview
- FIPS Pub 140-2 by NIST
- Assisted certificate enrollment
- BlackBerry Dynamics Shared Services Framework

# Acronyms/Glossary

| Term | Definition |
|---|---|
| AD | Active Directory |
| AES, AES-CBC | Advanced Encryption Standard, - Cipher Block Chaining mode |
| ARM | Popular RISC-oriented instruction set architecture (operating system) |
| BlackBerry Cloud | Collection of servers and services hosted by BlackBerry on the internet |
| CA | Certification Authority |
| DMZ | "De Militarized Zone" or perimeter network |
| ECC | Elliptic Curve Cryptography |
| EDEK | "Encrypt Decrypt Encrypt", also called Triple DES (Data Encryption Standard) |
| FIPS | Federal Information Processing Standard |
| FQDN | Fully Qualified Domain Name of a host |
| GRP | Good Relay Protocol |
| HMAC-SHA | Hash-based Message Authorization Code with Secure Hash Algorithm |
| HTTP/HTTPS | Hyper Text Transport Protocol/Secured |
| ICC | Inter-Container Communication protocol |
| ISV | Independent Software Vendor |
| KCD | Kerberos Constrained Delegation |
| MD4, MD5 | Message Digest 4 and 5 |
| MDC | Mobile Data Conduit protocol |
| PBKDF2 | PBKDF2 is a key derivation function that is part of RSA Laboratories' Public-Key Cryptography Standards series, specifically PKCS #5 v2.0, also published as Internet Engineering Task Force's RFC 2898. |
| RSA | One of the widely used public-key cryptosystems and is widely used for secure data transmission; named after Ron Rivest, Adi Shamir, and Leonard Adleman. |

| Term | Definition |
|---|---|
| SDK | Software Development Kit |
| TLS | Transport Layer Security, successor to SSL |
| TUK | Temp Unlock Key |
| UEM | Unified Endpoint Management server |