Nymi Connected Worker Platform

Storage and Transmission of Personal Data

Version 2.0

Table of Contents

Revision History

Date	Version	Explanation of Change
2020-Aug	1.0	Update for NEE 3.2.0 launch
2021-April-27	2.0	Update for CWP 1.1 launch

Personal Data

This document describes the storage and transmission of personal data in Nymi's Connected Worker Platform (CWP). CWP includes the following components:

1. Nymi Band
2. Nymi SDK and Runtime Components (Nymi Agent and Nymi Bluetooth Endpoint Service)
3. Nymi Enterprise Server (NES)
4. Nymi Band Application
5. Contact Tracing Services
6. Nymi Lock Control

An appendix of this document gives database queries to manage stored personal data.

This document applies to CWP 1.1.

Nymi Band 3.0

The Nymi Band 3.0 is the hardware component of CWP, and the only component that processes biometric data. Specifically, "biometric data" on the Nymi Band refers to the fingerprint images, fingerprint template and electrocardiogram (ECG) data that the wearer supplies.

During Nymi Band enrollment, a fingerprint template is constructed from the supplied user's fingerprint images. A fingerprint template is a mathematical representation of unique fingerprint features and is not a fingerprint image. The user's fingerprint template is securely stored on their Nymi Band, and it is not possible to extract/access the fingerprint template from memory on the Nymi Band. During authentication, fingerprint image and ECG sensor data is processed but is not stored after the authentication step. Performing the Delete User Data process on a Nymi Band erases all user information, including the biometric data. This design allows the Nymi Band hardware to process biometric data while maintaining the privacy of the wearer.

The Bluetooth Low Energy (BLE) MAC address and the Near Field Communications (NFC) Unique ID (UID) are transmitted by the Nymi Band under certain situations:

- The BLE MAC address is transmitted in BLE communications, e.g. in BLE advertising packets, and BLE data packets such as assert identity
- The NFC UID is transmitted when the Nymi Band is placed close to an NFC reader.

The BLE MAC address is advertised when the band is unauthenticated and authenticated.

When the Nymi Band is unenrolled, a randomly generated NFC UID is available each time it is tapped on an NFC reader when on charger or on-body. This randomly generated NFC UID differs in length from the static NFC UID available when the Nymi Band is enrolled.

The Nymi Band is comprised of three indirect identifiers:

- BLE Mac address
- NFC UID
- Unique serial number engraved on the back of the Nymi Band.

Alone these indirect identifiers cannot be used to infer the identity of the Nymi Band User. A record of each unique identifier is stored in the NES database and accessible only to verified administrators with access to the SQL database or Administrator Console.

For Nymi's contact tracing solution, the Nymi Band stores the following information:

- Proximity Events (Local BLE MAC Address, Remote BLE MAC Address, Time stamp)

A **Proximity Event** occurs when two contact tracing-enabled Nymi Bands are within range of each other for a short period of time. The Local BLE MAC is the address of the host Nymi Band, where the proximity event is registered. The Remote BLE MAC is the address of the nearby Nymi Band.

Nymi Band Application

Nymi Band Application (NBA) is used to enroll an employee with their Nymi Band. The application requires an employee to log in with their Active Directory username and password, which are validated against Active Directory via Nymi Enterprise Server (NES).

The Nymi Band Application creates a mapping between a Nymi Band and the employee.

The resulting mapping is stored in the Nymi Enterprise Server (NES). By default, all communication between the Nymi Band Application and Nymi Enterprise Server is encrypted. The Nymi Band Application itself does not store the username or password, or the above mapping, after the user logs out or closes the application.

Nymi Enterprise Server

NES is one of the server-side components of CWP and is a collection of several web services that reside on premise. NES relies on the corporate Active Directory to perform the initial authentication of the employees, for the purpose of creating the mapping between Nymi Bands and employees by the Nymi Band Application.

The Nymi Enterprise Server database stores the following information about user mapping:

- Username (as it exists in the Active Directory)
- Hardware identifiers that are associated with the Nymi Band that is enrolled to the Employee:
    - Near Field Communication Unique ID (NFC UID)

- Bluetooth Low Energy MAC address (Band ID)
- Nymi Band Serial Number

When Using Nymi Lock Control, the user's password is stored in the NES database encrypted with a key that is stored on the Nymi Band. NES stores this encrypted key and all other information in a Microsoft SQL Server database. Additional encryption can be applied to the database tables via SQL Server encryption mechanism.

The audit log functionality of the NES database generates a historical log of changes (creation, updates and deletion) in the above mapping, for audit and compliance purposes.

Contact Tracing Services (CTS)

The Contact Tracing Services compose the server-side component of Nymi's contact tracing solution. The Contact Tracing Services store contact tracing data in the contact tracing database for employees enrolled in the contact tracing program. The Contact Tracing Services communicate with NES for the purpose of retrieving user data for contact tracing reporting. The username is queried from NES, and the first name and last name of the user is queried against Active Directory.

The following information is stored in the CTS:

- Proximity Events (Local BLE MAC Address, Remote BLE MAC Address, Time stamp)
- Contact Events (Local BLE MAC Address, Remote BLE MAC Address, Time stamp)

A **Proximity Event** occurs when two contact tracing-enabled Nymi Bands are within range of each other for a short period of time.

A **Contact Event** occurs when there is a sequence of three consecutive Proximity Events .

Contact Tracing Dashboard

The Contact Tracing Dashboard is a web application that is used to visualize contact tracing data for the organization. The Contact Tracing Dashboard displays the data that is stored in the Contact Tracing Services.

The Contact Tracing Dashboard processes the following personal data:

- Contact Events (Local BLE MAC Address, Remote BLE MAC Address, Time stamp)
- Aggregation of Contact Events
- Full Name of user associated with a Contact Event

A **Contact Event** occurs when there is a sequence of three consecutive Proximity Events., approximately 15 minutes.

Nymi SDK and Runtime Components

Nymi SDK (nymi_api.dll) and Runtime components (Nymi Agent and Nymi Bluetooth Endpoint Service) process personal data in the following manner:

- BLE MAC address, NFC UID, and Active Directory username are passed between Nymi SDK, NES, and Nymi-Enabled Applications during certain operations (e.g. looking up Nymi Band based on the NFC UID that is received during an NFC tap, and user authentication during an e-signing operation). Retrieval of such information from NES is performed over an HTTPS (encrypted) connection by default.
- BLE MAC address is also processed by the Nymi Runtime components during (encrypted) communications with Nymi Bands over BLE.

Nymi SDK and Runtime components only process the personal data during the relevant transactions, and do not store it in non-volatile storage.

Nymi Contact Tracing Collection Agent (CTCA)

The Nymi Contact Tracing Collection Agent (CTCA) is specific to the Nymi's Contact Tracing solution and is used to download contact tracing data from the Nymi Band and to transfer data to the Contact Tracing Services (CTS).

CTCA processes the following personal data:

- Proximity Events (Local BLE MAC Address, Remote BLE MAC Address, Time stamp)

Log Files

Log files for NES, NBA and the Nymi SDK and Runtime components contain user information to support troubleshooting.

The following information can be found in log files:

1. Username
2. BLE MAC address
3. NFC UID
4. File locations

Appendix Sample Queries

Using the sample queries below, an authorized user is able to address a Data Subject Request (DSR) to access, delete, rectify, and restrict processing personal data if required. Note the contact tracing feature uses a data processing pipeline starting from the Nymi Band and ending at CTS. This pipeline may contain intermediate processing steps that store data for a limited amount of time and cannot be purged on a per-user basis. This data is automatically purged after the data lifetime expires. Access to the data on the pipeline is secured and restricted. The proximity data on the pipeline is processed exactly once to derive results.

Return personal data in NES

1. Connect to SQL Server as administrator in NES
2. Specify the domain name and username in the query
3. Run the query

```
declare @domain_username as nvarchar(30);
declare @user_core_id as int;
declare @user_name as nvarchar(164);
declare @user_domain as nvarchar(1280);
declare @user_core_id as int;
declare @user_name as nvarchar(164);
declare @user_domain as nvarchar(1280);

SET @domain_username = '<domain_name>\<username>'; /*<<< specify domain name and user name */

SELECT
@user_core_id = ID,
@user_name = Username,
@user_domain = Domain
@user_core_id = ID,
@user_name = Username,
@user_domain = Domain

FROM nub.UserCore
WHERE CONCAT(Domain,'\',Username)=@domain_username;
WHERE CONCAT(Domain,'\',Username)=@domain_username;

SELECT
@user_name 'User Name',
@user_domain 'Domain',
NymiBandID 'Mac Address',
```

```
HardwareID 'Serial Number',
NfcUID 'NFC Id',
case IsActive when 1 then 'Yes' else 'No' End Active ,
case IsPrimary when 1 then 'Yes' else 'No' End PrimaryBand ,
BandLabel,
FirmwareVersion
FROM nub.NymiBand
WHERE UserCoreID = @user_core_id;
@user_name 'User Name',
@user_domain 'Domain',
NymiBandID 'Mac Address',
HardwareID 'Serial Number',
NfcUID 'NFC Id',
case IsActive when 1 then 'Yes' else 'No' End Active ,
case IsPrimary when 1 then 'Yes' else 'No' End PrimaryBand ,
BandLabel,
FirmwareVersion
FROM nub.NymiBand
WHERE UserCoreID = @user_core_id;
```

Deleting personal data in NES

1. Connect to SQL Server as administrator in NES
2. Specify the domain name and username in the query below

```
declare @domain_username as nvarchar(30);
declare @user_core_id as int;
declare @user_name as nvarchar(164);
declare @user_domain as nvarchar(1280);
declare @user_core_id as int;
declare @user_name as nvarchar(164);
declare @user_domain as nvarchar(1280);

SET @domain_username = ' <domain_name>\<username>'';  /*<<< specify domain name and user name */

SELECT
@user_core_id = ID,
@user_name = Username,
```

```
@user_domain = Domain
@user_core_id = ID,
@user_name = Username,
@user_domain = Domain

FROM nub.UserCore
WHERE CONCAT(Domain,'\',Username)=@domain_username;
WHERE CONCAT(Domain,'\',Username)=@domain_username;

-- ============================================================================================================
--SQL query exists to delete personal information for a particular user.
--(all entries that include information about the Nymi Band serial number, username, NFC UID, BLE MAC address)
--query should include all Personal information records for the user of interest.
--SQL query exists to delete personal information for a particular user.
--(all entries that include information about the Nymi Band serial number, username, NFC UID, BLE MAC address)
--query should include all Personal information records for the user of interest.

-- Delete records from Private key store
DELETE FROM nub.PrivateKeyStore
WHERE nub.PrivateKeyStore.ID IN
( SELECT PrivateKeyStoreId from nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
)
);
-- Delete records from External Authenticator
DELETE FROM nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
);
---- Delete records from Nymi Band
DELETE FROM nub.NymiBand
WHERE UserCoreID = @user_core_id;
DELETE FROM nub.PrivateKeyStore
```

```
WHERE nub.PrivateKeyStore.ID IN
( SELECT PrivateKeyStoreId from nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
)
);
-- Delete records from External Authenticator
DELETE FROM nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
);
---- Delete records from Nymi Band
DELETE FROM nub.NymiBand
WHERE UserCoreID = @user_core_id;

---- Delete records from User OTP
DELETE FROM xrf.UserOtp
WHERE UserCoreID = @user_core_id;
DELETE FROM xrf.UserOtp
WHERE UserCoreID = @user_core_id;

---- Delete records from user
DELETE FROM nub.UserCore
WHERE ID = @user_core_id;
DELETE FROM nub.UserCore
WHERE ID = @user_core_id;
```

Rectify personal data in NES

1. Connect to NES database server as administrator
2. Specify the domain name and username and the data to be rectified in the query below

```
declare @domain_username as nvarchar(30);
declare @user_core_id as int;
```

```
declare @user_name as nvarchar(164);
declare @user_domain as nvarchar(1280);
declare @NymiBandID as nvarchar(64);

SET @domain_username = ' <domain_name>\<username> '; /*<<< specify domain name and user name */

SET @user_name='testqa11'; /*<<< information to be updated */
SET @user_domain='nymiqa1.com'; /*<<< information to be updated */
SET @NymiBandID='DF:5E:35:BA:56:E1'; /*<<< information to be udpted */

SELECT
@user_core_id = ID

FROM nub.UserCore
WHERE CONCAT(Domain,'\',Username)=@domain_username;

UPDATE nub.NymiBand
SET IsActive=1, IsPrimary=1
WHERE UserCoreID=@user_core_id AND
NymiBandID = @NymiBandID;

UPDATE nub.NymiBand
SET IsActive=0, IsPrimary=0
WHERE UserCoreID=@user_core_id AND
NymiBandID != @NymiBandID;

UPDATE nub.UserCore
SET Domain = @user_domain,
Username =@user_name
WHERE ID=@user_core_id;
```

3. Verify that data is updated by using below query

```
SELECT * from nub.NymiBand WHERE UserCoreID=@user_core_id
SELECT * from nub.UserCore WHERE ID=@user_core_id;
```

Return personal data in CTS

1. Connect to Contact Tracing database server with a database account having DML privileges
2. Specific the Nymi Band ID assigned to the specific user (e.g. AABBCCDDEEFF)
   ```
   SELECT * from CovidContacts where CovidContacts.source_mac like '<NymiBandID>' OR CovidContacts.remote_mac like '
   <NymiBandID> '
   ```
3. Run the query

Delete personal data of in CTS

1. Connect to Contact Tracing database server with a database account having DML privileges
2. Specific the Nymi Band ID assigned to the specific user (e.g. AABBCCDDEEFF) to delete the personal data
   ```
   DELETE CovidContacts where CovidContacts.source_mac like ' <NymiBandID> ' OR CovidContacts.remote_mac like '
   <NymiBandID> ';
   ```
3. Verify that data is deleted from database by using below query.
   ```
   SELECT * from CovidContacts where CovidContacts.source_mac like ' <NymiBandID> ' OR CovidContacts.remote_mac like
   ' <NymiBandID> '
   ```

Restrict processing personal data in CTS

1. Create backup of MSSQL DB (both Contact Tracing DB and NES DB)
2. Connect to NES database server with a database account having DML privileges, delete the user data from DB.
3. Specify the domain name and username in the query below

```
declare @domain_username as nvarchar(30);
declare @user_core_id as int;
declare @user_name as nvarchar(164);
declare @user_domain as nvarchar(1280);
declare @user_core_id as int;
declare @user_name as nvarchar(164);
declare @user_domain as nvarchar(1280);

SET @domain_username = ' <domain_name>\<username>'';  /*<<< specify domain name and user name */

SELECT
@user_core_id = ID,
```

```
@user_name = Username,
@user_domain = Domain
@user_core_id = ID,
@user_name = Username,
@user_domain = Domain

FROM nub.UserCore
WHERE CONCAT(Domain,'\',Username)=@domain_username;
WHERE CONCAT(Domain,'\',Username)=@domain_username;

-- ============================================================================================================
--SQL query exists to delete personal information for a particular user.
--(all entries that include information about the Nymi Band serial number, username, NFC UID, BLE MAC address)
--query should include all Personal information records for the user of interest.
--SQL query exists to delete personal information for a particular user.
--(all entries that include information about the Nymi Band serial number, username, NFC UID, BLE MAC address)
--query should include all Personal information records for the user of interest.

-- Delete records from Private key store
DELETE FROM nub.PrivateKeyStore
WHERE nub.PrivateKeyStore.ID IN
( SELECT PrivateKeyStoreId from nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
)
);
-- Delete records from External Authenticator
DELETE FROM nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
);
---- Delete records from Nymi Band
DELETE FROM nub.NymiBand
WHERE UserCoreID = @user_core_id;
```

```
DELETE FROM nub.PrivateKeyStore
WHERE nub.PrivateKeyStore.ID IN
( SELECT PrivateKeyStoreId from nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
)
);
-- Delete records from External Authenticator
DELETE FROM nub.ExternalAuthenticator
Where IIF(ISNULL(NymiBandID,0)=0,0,NymiBandID) IN
(
SELECT ISNULL(ID,0) FROM
nub.NymiBand WHERE UserCoreID=@user_core_id
);
---- Delete records from Nymi Band
DELETE FROM nub.NymiBand
WHERE UserCoreID = @user_core_id;

---- Delete records from User OTP
DELETE FROM xrf.UserOtp
WHERE UserCoreID = @user_core_id;
DELETE FROM xrf.UserOtp
WHERE UserCoreID = @user_core_id;

---- Delete records from user
DELETE FROM nub.UserCore
WHERE ID = @user_core_id;
DELETE FROM nub.UserCore
WHERE ID = @user_core_id;
```

4. Connect to Contact Tracing database server with a database account having DML privileges, delete the user data form DB.
5. Specific the Nymi Band ID assigned to the specific user (e.g. AABBCCDDEEFF) to delete the personal data

```
DELETE CovidContacts where CovidContacts.source_mac like ' <NymiBandID> ' OR CovidContacts.remote_mac like '
<NymiBandID> ';
```

6. Verify that data is deleted from database by using below query.

```
SELECT * from CovidContacts where CovidContacts.source_mac like ' <NymiBandID> ' OR CovidContacts.remote_mac like
' <NymiBandID> '
```