# Table of Contents

Possible integration options

There are three possible integration approaches in integrating Suprema readers to a 3rd party system: using Protocol, BioStar Device SDK and BioStar API(with BioStar2 Local API Server).

- BioStar Device SDK: Developing own application using the APIs provided by Suprema. It's not possible to communicate with a server (e.g. BioStar Server). Developers need to implement the functionality to manage and control devices. BioStar Device SDK is in the form of Windows DLL, so the application you're developing should be a Windows based application.
- BioStar Local API (BioStar2 Local API Server): Developing own application using the RESTful APIs that the BioStar 2 Local API Server provides. If you use BioStar Local API, you're communicating with your local BioStar 2 server through the BioStar 2 Local API Server.
- BioStar2 Local API Server Download
- Database Linking: Developing own application that directly accesses databases of the two systems that you want to integrate. In this approach, you don't have to use BioStar-specific APIs or libraries. Instead, the application synchronizes the two systems at the database level.
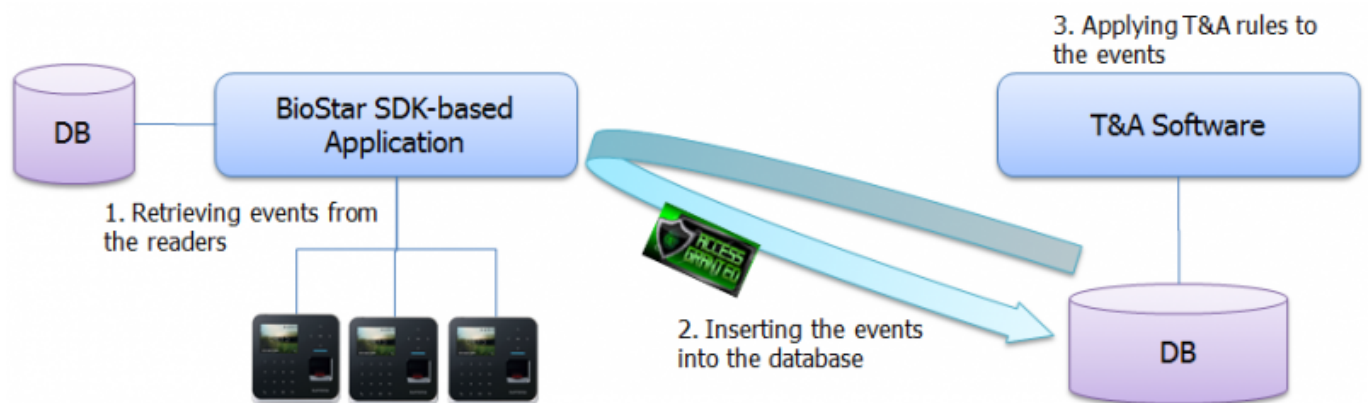
BioStar Device SDK

The following system diagram is basic usage of BioStar Device SDK.

Let's say you have your own T&A application. You want to automatically transfer users that are created on the T&A application to Suprema readers in real-time while transferring event data from the readers to your T&A application.
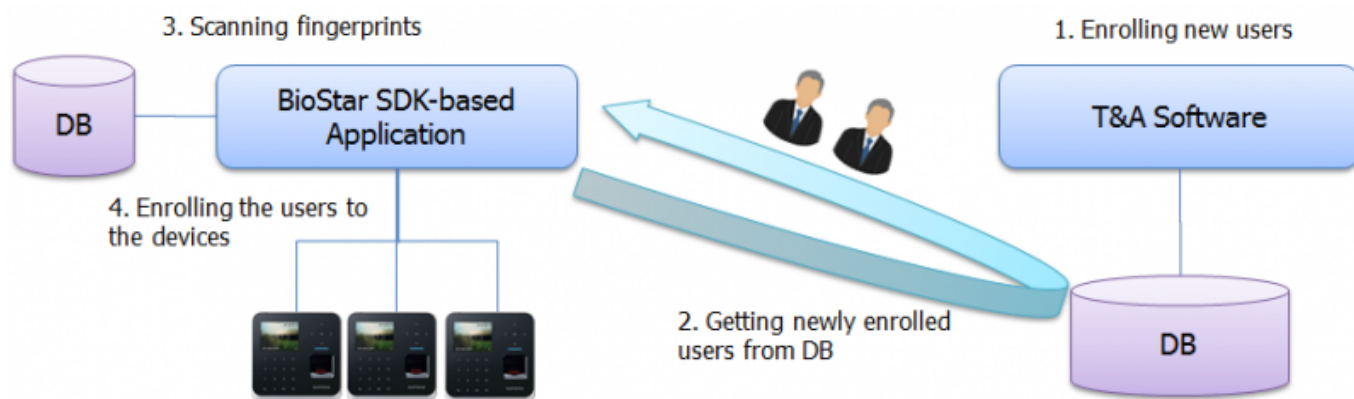
[Figure 1. Automatic User Transfer]

[Figure 2. Automatic Event Transfer]

As you can see from the figure 1, the SDK-based application gets the list of newly created users periodically and if there are newly created users, it inserts those users into its own database. Then, we need to receive fingerprint templates from the users, because we get user information only from the T&A software. Then, we can transfer the users to the devices.

Figure 2 explains how events are transferred from the readers to the T&A software. When the users identify themselves on the readers, the authentication events are stored on the devices. The BioStar SDK-based application retrieves the events from the devices at a certain interval. When there are newly stored authentication events, the application inserts the events into the database of the T&A software. If your T&A software offers server-level API, the SDK-based application will call the APIs instead of accessing the database directly.

## BioStar2 Local API(BioStar2 Local API Server)

So far we have looked at BioStar Device SDK. With the BioStar 1 system, we only had BioStar Device SDK, which means that it was not possible to either request data from a BioStar 1 server or control devices connected to the server by sending commands to the server. You always had to deal with the devices directly. Dealing with the devices directly is not an easy task because you have to implement user and device management on your own. If you need to use advanced access control rules like access groups, things get more complicated.

To ease these difficulties, we are now offering BioStar2 Local API and BioStar2 Local API Server, server-level RESTful API, with which you can communicate with BioStar 2 server. Therefore, you can get data from the server, create users and access control properties like access groups and time schedules. You can also control devices through BioStar2 Local API & BioStar2 Local API Server. We are not going to take a look at what RESTful API is about, because it's out of scope of this article. You can find detailed information on RESTful API from Wikipedia (https://en.wikipedia.org/wiki/Representational_state_transfer).

With the server-level API, you can focus your own logic leaving all the complicated tasks like user/device/access control management to BioStar 2 server. This will greatly shorten the development time.
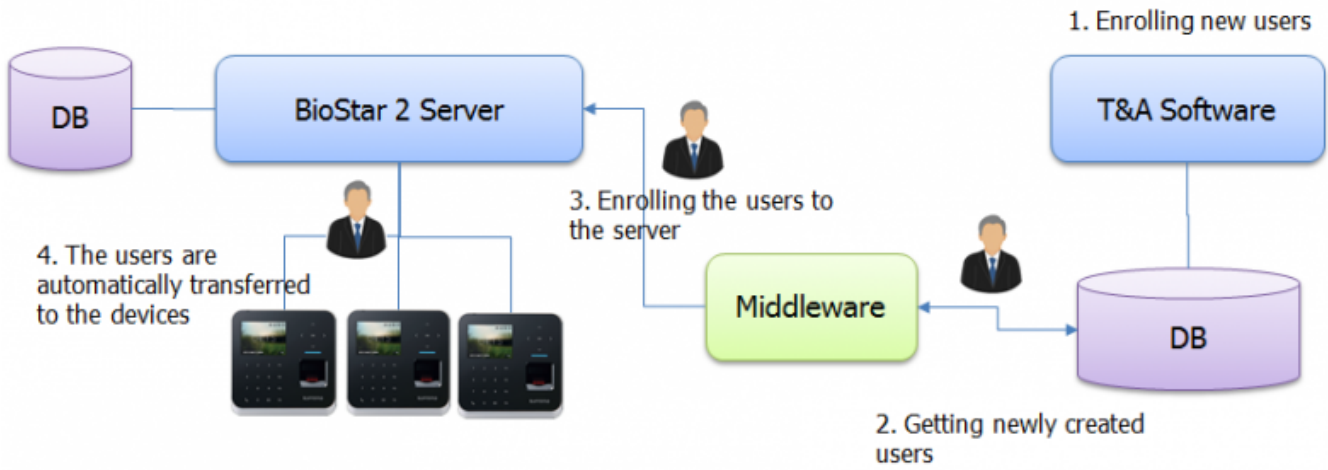
[Figure 3. Automatic User Transfer]
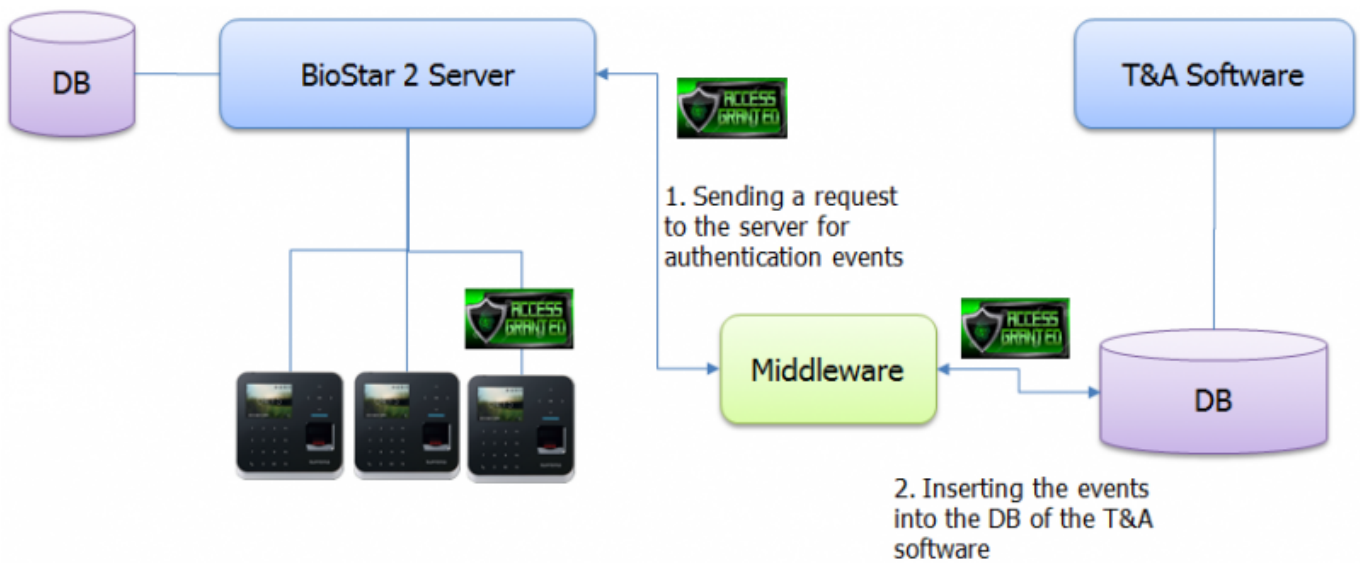
[Figure 4. Automatic Event Transfer]

If you look at Figure 3, you will notice that the middleware is communicating with BioStar 2 Server instead of dealing with the devices. This means that you don't have to implement user and device management functionality, which is quite a daunting job. Therefore, all the functions you need to implement is getting newly created users from the T&A software and insert them into BioStar 2 by calling BioStar user registration API.

Figure 4 explains how authentication events can be transferred from BioStar 2 to the T&A software. When users authenticate themselves, the authentication events are automatically transferred to BioStar 2 by default. The middleware can request BioStar 2 Server to send the events after a certain time. Then, the middleware inserts the events into the database of the T&A software.

BioStar Device SDK vs BioStar2 Local API

Now that we have taken a closer look at what BioStar Device SDK and BioStar2 Local API are about, it's time to discuss advantages and disadvantages of each option.

	BioStar Device SDK	BioStar API
Simple Installation	3	1
Ease of Development	1	5
Multi Operating System Support	3	5
Multi Programming Language Support	3	5
Seamless User Interface	5	2

- Simple Installation: With BioStar Device SDK, you can implement the features that you need for your project, so you can create a simple and lightweight Suprema reader and user management tool. If you use BioStar2 Local API, you must install BioStar 2 Server and set up BioStar2 Local API Server. BioStar2 Local API Server Installation Guide
- Ease of Development: As I mentioned above, developing applications with BioStar API is a lot easier because you can use BioStar 2 as an access control system and all you need to do is developing a bridge between BioStar 2 and a 3rd party system.
- Multi Operating System Support: Although BioStar 2 Server supports Windows only, you can create applications on any operating system as long as the operating system supports HTTP communications. For example, you can create an Android mobile application that communicates with BioStar server installed on a Windows PC. On the other hand, BioStar Device SDK is provided in the form of Windows DLL and Linux library, which means that you cannot create a mobile application with BioStar Device SDK.
- Multi Programming Language Support: Since BioStar Device SDK is provided in the form of Windows DLL and Linux library, you can implement your own application in C, C++, C#, Visual Basic and so on. BioStar API is a web-service API and you get what you want by using HTTP methods with an URL. This means that you can develop your own application in any language that supports HTTP methods.
- Seamless User Interface: If you develop your own application using BioStar Device SDK, you can provide seamless user interface to users. In particular, when you are able to modify a 3rd party application like T&A application, you can directly embed functionality to communicate with Suprema devices into the 3rd party application. This will give users more seamless and consistent user experience than having to manage two different systems.

From:
http://kb.supremainc.com/knowledge/ -

Permanent link:
http://kb.supremainc.com/knowledge/doku.php?id=en:possible_integration_options_in_biostar_2&rev=1557462974

Last update: **2019/05/10 13:36**