

Table of Contents

BS2_GetMasterAdmin	1
Declaration	1
Parameters	1
Return Value	2
Sample Code	2
See Also	3

BS2_GetMasterAdmin

Important

Suprema devices comply with CE RED (Radio Equipment Directive) and support the Master Admin feature.

Without admin configuration, anyone could access the admin menu and modify settings, creating a security vulnerability.

The Master Admin feature mandates full admin enrollment to prevent such risks.

Devices supporting this feature must complete master admin enrollment, and device usage is restricted until configuration is complete.

This differs from the existing operator [View BS2AuthOperatorLevel structure](#), so please be cautious.

All device features become available after master admin enrollment is successfully completed.

Below is information about devices and versions that support the Master Admin feature.

Device Type	Supported Version
BS3	V1.4.0 and later
XS2	V1.4.0 and later
BS2a	V1.2.0 and later
BEW3	Support planned

[+ 2.9.12] Gets master admin information from devices supporting the CE RED (Radio Equipment Directive) Master Admin feature.

Declaration

```
#include "BS_API.h"

int BS2_GetMasterAdmin(void* context, BS2_DEVICE_ID deviceId,
BS2UserFaceExBlob* masterAdmin);
```

[View BS2UserFaceExBlob structure](#)

Parameters

- [In] *context* : Context
- [In] *deviceId* : Device ID
- [Out] *masterAdmin* : Pointer to store the master admin information

Return Value

Returns BS_SDK_SUCCESS when performed successfully, and returns the corresponding error code when an error occurs.

Sample Code

C++

```
BS2UserFaceExBlob masterAdmin;
memset(&masterAdmin, 0x0, sizeof(masterAdmin));
int sdkResult = BS2_GetMasterAdmin(context_, id, &masterAdmin);
if (BS_SDK_SUCCESS != sdkResult)
{
    TRACE("BS2_GetMasterAdmin call failed: %d", sdkResult);
    return sdkResult;
}

print(masterAdmin);

if (masterAdmin.cardObjs && < masterAdmin.user.numCards)
    BS2_ReleaseObject(masterAdmin.cardObjs);

if (masterAdmin.fingerObjs && < masterAdmin.user.numFingers)
    BS2_ReleaseObject(masterAdmin.fingerObjs);

if (< masterAdmin.user.numFaces)
{
    if (masterAdmin.faceObjs)
        BS2_ReleaseObject(masterAdmin.faceObjs);
    else if (masterAdmin.faceExObjs)
        BS2_ReleaseObject(masterAdmin.faceExObjs);
}
```

C#

```
BS2UserFaceExBlob userBlob;
Console.WriteLine("Trying to get master admin");
BS2ErrorCode sdkResult = (BS2ErrorCode)API.BS2_GetMasterAdmin(sdkContext,
deviceID, out userBlob);
if (sdkResult != BS2ErrorCode.BS_SDK_SUCCESS)
{
    Console.WriteLine("Got error({0})." , sdkResult);
    return;
}

print(userBlob);

if (userBlob.cardObjs != IntPtr.Zero)
```

```
API.BS2_ReleaseObject(userBlob.cardObj);  
if (userBlob.fingerObj != IntPtr.Zero)  
    API.BS2_ReleaseObject(userBlob.fingerObj);  
if (userBlob.faceObj != IntPtr.Zero)  
    API.BS2_ReleaseObject(userBlob.faceObj);  
if (userBlob.faceExObj != IntPtr.Zero)  
    API.BS2_ReleaseObject(userBlob.faceExObj);
```

See Also

[BS2_SetMasterAdmin](#)

From:

<https://kb.supremainc.com/kbtest/> - **BioStar Device SDK**

Permanent link:

https://kb.supremainc.com/kbtest/doku.php?id=en:bs2_getmasteradmin

Last update: **2026/01/28 15:45**