

User Management API 1

..... 1

BS2User 1

BS2UserSetting 4

BS2UserPhoto 5

BS2UserBlob 5

BS2Job 6

BS2UserBlobEx 7

BS2UserSmallBlob 8

BS2UserSmallBlobEx 9

BS2UserSettingEx 10

BS2UserFaceExBlob 12

BS2UserStatistic 13

User Management API

API

- [BS2_GetUserList](#): ID 가
- [BS2_RemoveUser](#):
- [BS2_RemoveAllUser](#):
- [BS2_GetUserInfos](#): ID 가
- [BS2_GetUserInfosEx](#): [+ 2.4.0] ID 가 (Job code, User phrase)
- [BS2_EnrollUser](#):
- [BS2_EnrollUserEx](#): [+ 2.4.0] (Job code, User phrase)
- [BS2_EnrollUser](#): [+ 2.6.3]
- [BS2_EnrollUserEx](#): [+ 2.6.3] (Job code, User phrase)
- [BS2_GetUserDatas](#): 가
- [BS2_GetUserDatasEx](#): [+ 2.5.0] 가 (Job code, User phrase)
- [BS2_GetSupportedUserMask](#): 가
- [BS2_EnrollUserSmall](#): [+ 2.6.3]
- [BS2_EnrollUserSmallEx](#): [+ 2.6.3]
- [BS2_GetUserSmallInfos](#): [+ 2.6.3] , ID
- [BS2_GetUserSmallInfosEx](#): [+ 2.6.3] , ID
- [BS2_GetUserSmallDatas](#): [+ 2.6.3] ,
- [BS2_GetUserSmallDatasEx](#): [+ 2.6.3] ,
- [BS2_EnrollUserFaceEx](#): [+ 2.7.1] Visual Face
- [BS2_GetUserInfosFaceEx](#): [+ 2.7.1] Visual Face ID 가
- [BS2_GetUserDatasFaceEx](#): [+ 2.7.1] Visual Face 가
- [BS2_PartialUpdateUser](#): [+ 2.8.3]
- [BS2_PartialUpdateUserEx](#): [+ 2.8.3]
- [BS2_PartialUpdateUserSmall](#): [+ 2.8.3]
- [BS2_PartialUpdateUserSmallEx](#): [+ 2.8.3]
- [BS2_PartialUpdateUserFaceEx](#): [+ 2.8.3] Visual Face
- [BS2_GetUserStatistic](#): [+ 2.8.3] 가 가

BS2User

```
typedef struct {
    char userID[BS2_USER_ID_SIZE];
    uint8_t formatVersion;
    uint8_t flag;
```

```

uint16_t version;
uint8_t numCards;
uint8_t numFingers;
uint8_t numFaces;
uint8_t infoMask;
uint32_t authGroupID;
uint32_t faceChecksum;
} BS2User;
    
```

1. *userID* , 1 ~ 4294967295 가 .

2. *formatVersion* .

3. *flag* flag OR 가 .

0x00	None
0x01	가
0x02	가
0x04	가
0x80	가

4. *version* .

5. *numCards* .

6. *numFingers* .

7. *numFaces* .

8. *infoMask* [+ 2.8.3] 가 가 , infoMask ,
 , Partial update ([BS2_PartialUpdateUser](#), [BS2_PartialUpdateUserEx](#),
[BS2_PartialUpdateUserSmall](#), [BS2_PartialUpdateUserSmallEx](#), [BS2_PartialUpdateUserFaceEx](#)) user
 mask . ()

0x01	BS2_USER_INFO_MASK_PHRASE
0x02	BS2_USER_INFO_MASK_JOB_CODE
0x04	BS2_USER_INFO_MASK_NAME
0x08	BS2_USER_INFO_MASK_PHOTO
0x10	BS2_USER_INFO_MASK_PIN
0x20	BS2_USER_INFO_MASK_CARD
0x40	BS2_USER_INFO_MASK_FINGER

0x80	BS2_USER_INFO_MASK_FACE

, infoMask

	BS2_USER_MASK	infoMask
	unmasking	unmasking
	masking	masking
	unmasking	masking

infoMask , partial update user mask , unmasking

infoMask , partial update user mask , masking

infoMask masking , partial update user mask unmasking

Credential(/ /)

BS2User numCards, numFingers, numFaces가 0 , fingerObjs, cardObjs, faceObjs, faceExObjs credential 가 , 가 infoMask

/ / 0 , infoMask mask credential

BS2_USER_INFO_MASK_FINGER masking 2 가 , infoMask , numFingers = 1, fingerObjs

가가 1 , 2 1 가 , 3

Credential(/ /)

/ / credential 0 , infoMask masking , 가 credential

Credential(/ /)

/ / credential 0 , infoMask unmasking , credential

9. authGroupID

ID.

10. faceChecksum

BS2UserSetting



```
typedef struct {
    uint32_t startTime;
    uint32_t endTime;
    uint8_t fingerAuthMode;
    uint8_t cardAuthMode;
    uint8_t idAuthMode;
    uint8_t securityLevel;
} BS2UserSetting;
```

1. startTime

가 , 0
 978307200 (2001-01-01 00:00:00)

2. endTime

가 , 0
 1924991999 (2030-12-31 23:59:59)

3. fingerAuthMode

0	
1	PIN
254	
255	()

4. cardAuthMode

2	
3	
4	PIN
5	PIN
6	, , PIN
254	
255	()

5. idAuthMode

ID

7	ID
8	ID PIN
9	ID PIN
10	ID PIN
254	
255	()

6. securityLevel

0	
1	
2	
3	
4	
5	

BS2UserPhoto

```
typedef struct {
    uint32_t size;
    uint8_t data[BS2_USER_PHOTO_SIZE];
} BS2UserPhoto;
```

1. size

2. data

16kb

BS2UserBlob

```
typedef struct {
    BS2User user;
    BS2UserSetting setting;
    uint8_t name[BS2_USER_NAME_SIZE];
    BS2UserPhoto photo;
    uint8_t pin[BS2_PIN_HASH_SIZE];
    BS2CSNCard* cardObjs;
    BS2Fingerprint* fingerObjs;
    BS2Face* faceObjs;
    uint32_t accessGroupId[BS2_MAX_NUM_OF_ACCESS_GROUP_PER_USER];
} BS2UserBlob;
```

1. user

2. *setting*

3. *name*

UTF-8

4. *photo*

Jpeg

5. *pin*

PIN *BS_MakePinCode*

6. *cardObjs*

user.numCards

[Smartcard API](#)

7. *fingerObjs*

user.numFingers

[Fingerprint API](#)

8. *faceObjs*

user.numFaces

[Face API](#)

9. *accessGroupId*

가

16

BS2Job

```

typedef struct {
    uint8_t numJobs;
    uint8_t reserved[3];

    struct {
        BS2_JOB_CODE code;
        BS2_JOB_LABEL label;
    } jobs[BS2_MAX_JOB_SIZE];
} BS2Job;

```

1. *numJobs*

Job

2. *reserved*

3. *jobs*

T&A Job

BS2UserBlobEx

```
typedef struct {
    BS2User user;
    BS2UserSetting setting;
    uint8_t name[BS2_USER_NAME_SIZE];
    BS2UserPhoto photo;
    uint8_t pin[BS2_PIN_HASH_SIZE];
    BS2CSNCard* cardObjs;
    BS2Fingerprint* fingerObjs;
    BS2Face* faceObjs;
    BS2Job job;
    BS2_USER_PHRASE phrase;
    uint32_t accessGroupId[BS2_MAX_NUM_OF_ACCESS_GROUP_PER_USER];
} BS2UserBlobEx;
```

1. *user*

2. *setting*

3. *name*

UTF-8

4. *photo*

Jpeg

5. *pin*

PIN

BS_MakePinCode

6. *cardObjs*

user.numCards

[Smartcard API](#)

7. *fingerObjs*

user.numFingers

[Fingerprint API](#)

8. *faceObjs*

user.numFaces

[Face API](#)

9. *job*

10. *phrase*

UI

FaceStation 2	V1.0.0
FaceStation F2	V1.0.0

X-Station 2	V1.0.0

11. *accessGroupId*
가

16

BS2UserSmallBlob

```
typedef struct {
    BS2User user;
    BS2UserSetting setting;
    uint8_t name[BS2_USER_NAME_SIZE];
    BS2UserPhoto* photo;
    uint8_t pin[BS2_PIN_HASH_SIZE];
    BS2CSNCard* cardObjs;
    BS2Fingerprint* fingerObjs;
    BS2Face* faceObjs;
    uint32_t accessGroupId[BS2_MAX_NUM_OF_ACCESS_GROUP_PER_USER];
} BS2UserSmallBlob;
```

1. *user*

2. *setting*

3. *name*

UTF-8

4. *photo*

Jpeg

5. *pin*

PIN

BS_MakePinCode

6. *cardObjs*

user.numCards

[Smartcard API](#)

7. *fingerObjs*

user.numFingers

[Fingerprint API](#)

8. *faceObjs*

user.numFaces

[Face API](#)

9. *accessGroupId*
가

16

BS2UserSmallBlobEx

```
typedef struct {
    BS2User user;
    BS2UserSetting setting;
    uint8_t name[BS2_USER_NAME_SIZE];
    BS2UserPhoto* photo;
    uint8_t pin[BS2_PIN_HASH_SIZE];
    BS2CSNCard* cardObjs;
    BS2Fingerprint* fingerObjs;
    BS2Face* faceObjs;
    BS2Job job;
    BS2_USER_PHRASE phrase;
    uint32_t accessGroupId[BS2_MAX_NUM_OF_ACCESS_GROUP_PER_USER];
} BS2UserSmallBlobEx;
```

1. *user*

2. *setting*

3. *name*

UTF-8

4. *photo*

Jpeg

5. *pin*

PIN

BS_MakePinCode

6. *cardObjs*

user.numCards

[Smartcard API](#)

7. *fingerObjs*

user.numFingers

[Fingerprint API](#)

8. *faceObjs*

user.numFaces

[Face API](#)

9. *job*

10. *phrase*

UI

FaceStation 2	V1.0.0
FaceStation F2	V1.0.0

X-Station 2	V1.0.0
BioStation 3	V1.0.0

11. *accessGroupId*
가

16

BS2UserSettingEx

FaceStation F2, BioStation 3
BS2UserSetting

```
typedef struct {
    uint8_t faceAuthMode;
    uint8_t fingerprintAuthMode;
    uint8_t cardAuthMode;
    uint8_t idAuthMode;
    uint8_t reserved[28];
} BS2UserSettingEx;
```

1. *faceAuthMode*

	1	2	3	4
11				
12				
13		PIN		
14		PIN		
15			PIN	
254				
255) (

2. *fingerprintAuthMode*

	1	2	3	4
16				
17				
18		PIN		
19		PIN		

	1	2	3	4
20			PIN	
254				
255) (

3. *cardAuthMode*

	1	2	3
21			
22			
23			
24		PIN	
25			
26		PIN	
27		PIN	
28		PIN	
29			
30			PIN
31			
32			PIN
33			PIN
34			PIN
35			PIN
254			
255) (

4. *idAuthMode*

ID

	1	2	3
36	ID		
37	ID		
38	ID	PIN	
39	ID		
40	ID	PIN	
41	ID	PIN	
42	ID	PIN	
43	ID		
44	ID		PIN
45	ID		
46	ID		PIN

	1	2	3
47	ID		PIN
48	ID		PIN
49	ID		PIN
254			
255	(

5. *reserved*

BS2UserFaceExBlob

```
typedef struct
{
    BS2User user;
    BS2UserSetting setting;
    BS2_USER_NAME user_name;
    BS2UserPhoto* user_photo_obj;
    BS2_USER_PIN pin;
    BS2CSNCard* cardObjs;
    BS2Fingerprint* fingerObjs;
    BS2Face* faceObjs; // FS2, FL
    BS2Job job;
    BS2_USER_PHRASE phrase;
    BS2_ACCESS_GROUP_ID accessGroupId[BS2_MAX_NUM_OF_ACCESS_GROUP_PER_USER];

    BS2UserSettingEx settingEx; // F2
    BS2FaceEx* faceExObjs; // F2
} BS2UserFaceExBlob;
```

1. *user*

2. *setting*

3. *name*

UTF-8

4. *photo*

Jpeg

5. *pin*

PIN

BS_MakePinCode

6. *cardObjs*

user.numCards

Smartcard API

7. *fingerObjs*

user.numFingers

Fingerprint API

8. *faceObjs*

FaceStation2, FaceLite

user.numFaces

Face API

9. *job*

10. *phrase*

UI

FaceStation 2	V1.0.0
FaceStation F2	V1.0.0
X-Station 2	V1.0.0

11. *accessGroupId*

가

16

12. *settingEx*

FaceStation F2

가

13. *faceExObjs*

FaceStation F2

user.numFaces

Face API

BS2UserStatistic

```
typedef struct {
    uint32_t numUsers;
    uint32_t numCards;
    uint32_t numFingerprints;
    uint32_t numFaces;
    uint32_t numNames;
    uint32_t numImages;
    uint32_t numPhrases;
} BS2UserStatistic;
```

1. *numUsers*

2. *numCards*

3. *numFingerprints*

4. *numFaces*

5. *numNames*

6. *numImages*

7. *numPhrases*

From:

<https://kb.supremainc.com/bs2sdk/> - **BioStar 2 Device SDK**

Permanent link:

https://kb.supremainc.com/bs2sdk/doku.php?id=ko:user_management_api&rev=1716887864

Last update: **2024/05/28 18:17**