

User Management API 1

..... 1

BS2User 1

BS2UserSetting 2

BS2UserPhoto 3

BS2UserBlob 4

BS2Job 5

BS2UserBlobEx 5

BS2UserSmallBlob 6

BS2UserSmallBlobEx 7

BS2UserSettingEx 8

BS2UserFaceExBlob 10

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] FaceStation F2 .
- [BS2_GetUserInfosFaceEx](#): [+ 2.7.1] FaceStation F2 ID 가 .
- [BS2_GetUserDatasFaceEx](#): [+ 2.7.1] FaceStation F2 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 reserved2[1];
    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. *authGroupID*
 ID.

9. *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 .

2. endTime

가 0

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

(FS2 FW 1.0.0

가)

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 (FS2 FW 1.0.0 가)

11. accessGroupId

가 16

BS2UserSettingEx

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

FaceStation F2

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		
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	

	1	2	3
42	ID	PIN	
43	ID		
44	ID		PIN
45	ID		
46	ID		PIN
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* cardObj;
    BS2Fingerprint* fingerObj;
    BS2Face* faceObj; // 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* faceExObj; // 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

. (FS2 FW 1.0.0

가)

11. *accessGroupId*

가

16

12. *settingEx*

FaceStation F2

가

13. *faceExObjs*

FaceStation F2

user.numFaces[Face API](#)

From:

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

Permanent link:

https://kb.supremainc.com/kbtest/doku.php?id=ko:user_management_api&rev=1603341968Last update: **2020/10/22 13:46**