

Access Control API 1

..... 1

BS2AccessGroup 1

BS2DoorSchedule 2

BS2AccessLevel 2

BS2TimePeriod 2

BS2DaySchedule 3

BS2WeeklySchedule 3

BS2DailySchedule 3

BS2HolidaySchedule 4

BS2Schedule 4

BS2Holiday 5

BS2HolidayGroup 5

Access Control API

API

- [BS2_GetAccessGroup](#): 가 .
- [BS2_GetAllAccessGroup](#): 가 .
- [BS2_SetAccessGroup](#): .
- [BS2_RemoveAccessGroup](#): .
- [BS2_RemoveAllAccessGroup](#): .
- [BS2_GetAccessLevel](#): 가 .
- [BS2_GetAllAccessLevel](#): 가 .
- [BS2_SetAccessLevel](#): .
- [BS2_RemoveAccessLevel](#): .
- [BS2_RemoveAllAccessLevel](#): .
- [BS2_GetAccessSchedule](#): 가 .
- [BS2_GetAllAccessSchedule](#): 가 .
- [BS2_SetAccessSchedule](#): .
- [BS2_RemoveAccessSchedule](#): .
- [BS2_RemoveAllAccessSchedule](#): .
- [BS2_GetHolidayGroup](#): 가 .
- [BS2_GetAllHolidayGroup](#): 가 .
- [BS2_SetHolidayGroup](#): .
- [BS2_RemoveHolidayGroup](#): .
- [BS2_RemoveAllHolidayGroup](#): .

BS2AccessGroup

```
typedef struct {
    uint32_t id;
    char name[BS2_MAX_ACCESS_GROUP_NAME_LEN];
    uint8_t numAccessLevels;
    uint8_t reserved[3];
    uint32_t accessLevels[BS2_MAX_ACCESS_LEVEL_PER_ACCESS_GROUP];
} BS2AccessGroup;
```

1. *id*

2. *name*

BioStar

3. *numAccessLevels*

4. *reserved*

5. *accessLevels*

BS2DoorSchedule

```
typedef struct {  
    uint32_t doorID;  
    uint32_t scheduleID;  
} BS2DoorSchedule ;
```

1. *doorID*

2. *scheduleID*

BS2AccessLevel

```
typedef struct {  
    uint32_t id;  
    char name[BS2_MAX_ACCESS_GROUP_NAME_LEN];  
    uint8_t numDoorSchedules;  
    uint8_t reserved[3];  
    BS2DoorSchedule doorSchedules[BS2_MAX_ACCESS_LEVEL_PER_ACCESS_GROUP];  
} BS2AccessLevel;
```

1. *id*

2. *name*

BioStar

3. *numDoorSchedules*

4. *reserved*

5. *doorSchedules*

BS2TimePeriod

```
typedef struct {  
    int16_t startTime;  
    int16_t endTime;
```

```
} BS2TimePeriod;
```

1. *startTime*

, 가 .

2. *endTime*

, 가 .

BS2DaySchedule

```
typedef struct {  
    uint8_t numPeriods;  
    uint8_t reserved[3];  
    BS2TimePeriod periods[BS2_MAX_TIME_PERIODS_PER_DAY];  
} BS2DaySchedule;
```

1. *numPeriods*

.

2. *reserved*

.

3. *periods*

5 .

BS2WeeklySchedule

```
typedef struct {  
    BS2DaySchedule schedule[BS2_NUM_WEEKDAYS];  
} BS2WeeklySchedule;
```

1. *schedule*

7 가 .

BS2DailySchedule

```
typedef struct {  
    uint32_t startDate;  
    uint8_t numDays;  
    uint8_t reserved[3];  
    BS2DaySchedule schedule[BS2_MAX_DAYS_PER_DAILY_SCHEDULE];  
} BS2DailySchedule;
```

1. *startDate*

.

2. *numDays*

.

3. reserved

4. schedule

startDate

BS2HolidaySchedule

```
typedef struct {
    uint32_t id;
    BS2DaySchedule schedule;
} BS2HolidaySchedule;
```

1. startDate

2. schedule

BS2Schedule

```
typedef struct
{
    uint32_t id;
    char name[BS2_MAX_SCHEDULE_NAME_LEN];
    uint8_t isDaily;
    uint8_t numHolidaySchedules;
    uint8_t reserved[2];
    union
    {
        BS2WeeklySchedule weekly;
        BS2DailySchedule daily;
    }schedule;
    BS2HolidaySchedule
    holidaySchedules[BS2_MAX_HOLIDAY_GROUPS_PER_SCHEDULE];
}BS2Schedule;
```

1. id

CAUTION

0 1 , 0 , 1 가

2. name

BioStar

3. *isDaily*

flag

4. *numHolidaySchedules*

5. *reserved*

6. *weekly*

isDaily가 0 가 가

7. *daily*

isDaily가 0 가 가

8. *holidaySchedules*

BS2Holiday

```
typedef struct {
    uint32_t date;
    uint8_t recurrence;
} BS2Holiday;
```

1. *date*

2. *recurrence*

0	
1	
2	
3	

BS2HolidayGroup

```
typedef struct
{
    uint32_t id;
    char name[BS2_MAX_SCHEDULE_NAME_LEN];
    uint8_t numHolidays;
    uint8_t reserved[3];
    BS2Holiday holidays[BS2_MAX_HOLIDAYS_PER_GROUP];
}BS2HolidayGroup;
```

1. *id*

2. *name*

BioStar

3. *numHolidays*

4. *reserved*

5. *holidays*

From:

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

Permanent link:

https://kb.supremainc.com/bs2sdk./doku.php?id=ko:access_control_api&rev=1676530460

Last update: **2023/02/16 15:54**