

GuardPointPro - SALTO integration

Table of contents

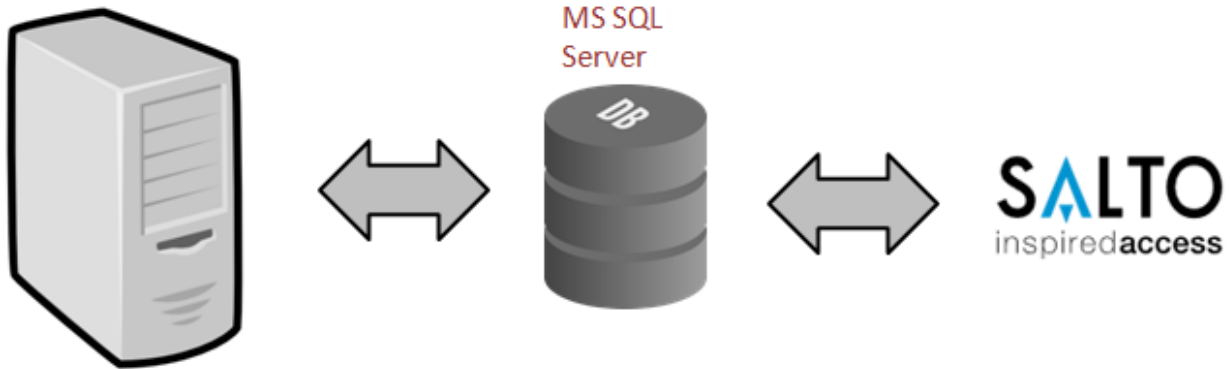
Introduction2
Requirements2
Installation2
Explanation of how it works3

GuardPointPro - SALTO integration

Introduction

To Synchronize the Amadeus5 server card holder data with the SALTO System.

Access Control Server



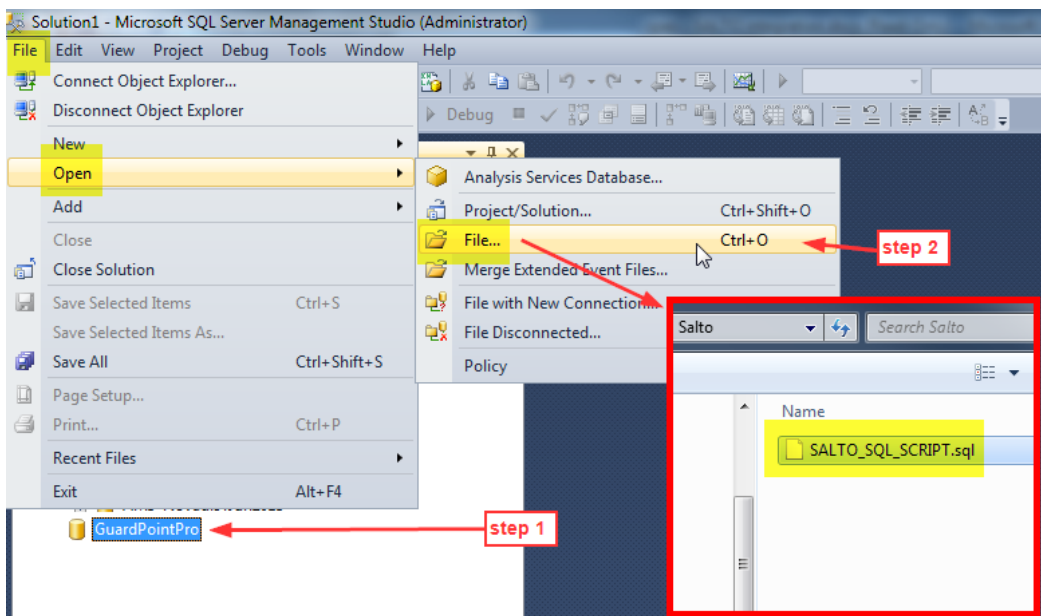
Requirements

- GuardPointPro version 3.2.029 or greater.
- GuardPointPro must be running with Microsoft SQL Server. (does not work with MS Access)

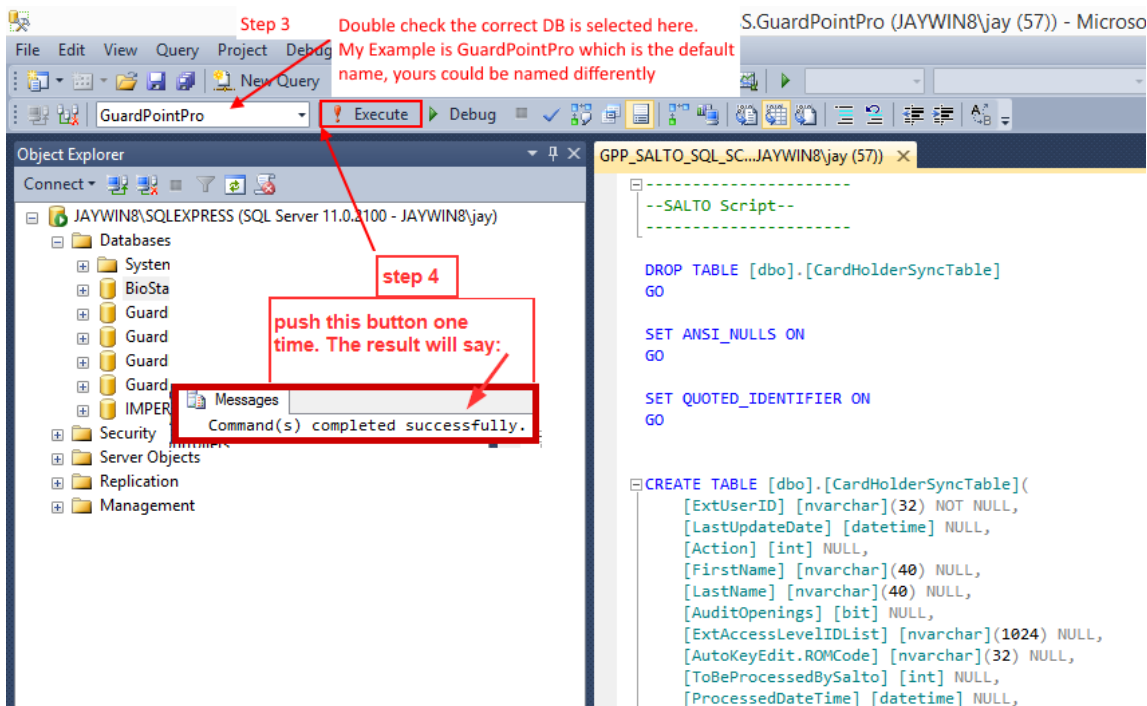
Installation

Run the [SQL script](#) and on the GuardPointPro SQL Database.

(http://public.sensoraccess.co.uk/Software%20Downloads/GuardPoint%20Pro/Integration/GPP_SALTO_SQL_SCRIPT.SQL)



GuardPointPro - SALTO integration



Explanation of how it works

Every change on below specific fields of the cardholders will be saved in to the newly created table, "CardHolderSyncTable".

This table is now local to the GuardPointPro DB, and will be used by the SALTO system to synchronize with GuardPointPro

Actions to update on:

- Delete cardholder
- New cardholder
- Update cardholder fields:
 - o First name
 - o Last name
 - o Access group list (from GuardPointPro cardholder screen)
 - o Card code (the code of the cardholder's card in GuardPointPro badge screen)

Each cardholder will have one record in "CardHolderSyncTable".

The SALTO system will read the records from "CardHolderSyncTable".

The changes on a cardholder will be saved in "CardHolderSyncTable", regardless if the previous changes of the cardholder were already read by SALTO or not.

This means that on update of any one card holder field, the old value of the field will not be preserved and will be replaced fully by the new value.

GuardPointPro - SALTO integration

The fields of “CardHolderSyncTable” and their source in GuardPointPro:

[ExtUserID] Unique ID from GuardPointPro.
[LastUpdateDate] The date the update was saved.
[Action] 3=Update/Create, 4=Delete
[FirstName] Cardholder's first Name in GuardPointPro
[LastName] Cardholder's last Name in GuardPointPro
[AuditOpenings] Set to “1” always
[ExtAccessLevelIDList] In GuardPointPro cardholder's Access groups names separated with delimiter.
[AutoKeyEdit.ROMCode] In GuardPointPro the cardholder's card code
[ToBeProcessedBySalto] This should be set to “1” when a post is updated or inserted
[ProcessedDateTime] Left empty from DDS side, Salto will set the date and time here when it has been synchronised
[ErrorCode] Left empty from DDS side, Salto will set this if an error has occurred
[ErrorMessage] Left empty from DDS side, Salto will set this if an error has occurred

Annex

Note: This new script will automatically look for and attempt to remove the previous DB changes before inserting the new changes. Because of this, if this the first time the script is run on the GuardPointPro database you will see the below error messages. You may safely ignore these errors.

Msg 3701, Level 11, State 5, Line 5

Cannot drop the table 'dbo.CardHolderSyncTable', because it does not exist or you do not have permission.

Msg 3701, Level 11, State 5, Line 36

Cannot drop the trigger 'dbo.CRDHLDSyncTrigDel', because it does not exist or you do not have permission.

Msg 3701, Level 11, State 5, Line 65

Cannot drop the trigger 'dbo.CRDHLDSyncTrigUpdate', because it does not exist or you do not have permission.

Msg 3701, Level 11, State 5, Line 102

Cannot drop the trigger 'dbo.CardSyncTrigDel', because it does not exist or you do not have permission.

Msg 3701, Level 11, State 5, Line 132

Cannot drop the trigger 'dbo.CARDSyncTrigUpdate', because it does not exist or you do not have permission.

GuardPointPro - SALTO integration

```
-----  
--SALTO Script--  
-----
```

```
DROP TABLE [dbo].[CardHolderSyncTable]  
GO
```

```
SET ANSI_NULLS ON  
GO
```

```
SET QUOTED_IDENTIFIER ON  
GO
```

```
CREATE TABLE [dbo].[CardHolderSyncTable](  
    [ExtUserID] [nvarchar](32) NOT NULL,  
    [LastUpdateDate] [datetime] NULL,  
    [Action] [int] NULL,  
    [FirstName] [nvarchar](40) NULL,  
    [LastName] [nvarchar](40) NULL,  
    [AuditOpenings] [bit] NULL,  
    [ExtAccessLevelIDList] [nvarchar](1024) NULL,  
    [AutoKeyEdit.ROMCode] [nvarchar](32) NULL,  
    [ToBeProcessedBySalto] [int] NULL,  
    [ProcessedDateTime] [datetime] NULL,  
    [ErrorCode] [int] NOT NULL Default 0,  
    [ErrorMessage] [nvarchar](512) NULL,  
    CONSTRAINT [PK_CardHolderSyncTable] PRIMARY KEY CLUSTERED  
(  
        [ExtUserID] ASC  
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, AL-  
LOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]
```

```
GO
```

```
DROP TRIGGER [dbo].[CRDHLDSTrigDel]  
GO
```

```
CREATE TRIGGER [dbo].[CRDHLDSTrigDel] ON [dbo].[CRDHLD]  
AFTER DELETE  
AS  
BEGIN
```

```
    SET NOCOUNT ON;
```


GuardPointPro - SALTO integration

```

insert into CardHolderSyncTable
([ExtUserID], [LastUpdateDate], [Action],[FirstName], [LastName], [AuditOpenings],
[ExtAccessLevelIDList], [AutoKeyEdit.ROMCode], [ToBeProcessedBySalto])
(select id,      GetDate(),      4,      First_Name, Last_Name,      1,      case len(isnull
(ag_name, '')) when 0 then ''
else left(rtrim(ag_name), len(rtrim(ag_name))-1)
END,      NULL ,      1
from deleted
where
deleted.id not in (select ExtUserID from CardHolderSyncTable))

update CardHolderSyncTable set
[LastUpdateDate] = GetDate(), [Action] = 4,[FirstName] = First_Name, [LastName] = Last_Name,
[AuditOpenings] = 1, [ExtAccessLevelIDList] = case len(isnull(ag_name, '')) when 0 then ''
else left(rtrim(ag_name), len(rtrim(ag_name))-1)
END, [ToBeProcessedBySalto] = 1
from DELETED
where
DELETED.id in (select ExtUserID from CardHolderSyncTable)
AND CardHolderSyncTable.ExtUserID = DELETED.id

END

GO

DROP TRIGGER [dbo].[CRDHLDSTrigUpdate]
GO

CREATE TRIGGER [dbo].[CRDHLDSTrigUpdate] ON [dbo].[CRDHLD]
AFTER UPDATE,INSERT
AS
BEGIN

    SET NOCOUNT ON;

    insert into CardHolderSyncTable
    ([ExtUserID], [LastUpdateDate], [Action],[FirstName], [LastName], [AuditOpenings],
    [ExtAccessLevelIDList], [AutoKeyEdit.ROMCode], [ToBeProcessedBySalto])
    (select id,      GetDate(),
    case when INSERTED.type = 3 then 4
    when INSERTED.type <> 3 then 3
    end,
    First_Name, Last_Name,      1,      case len(isnull(ag_name, '')) when 0 then ''
    else left(rtrim(ag_name), len(rtrim(ag_name))-1)
    END ,      NULL , 1
    from INSERTED
    where
    INSERTED.id not in (select ExtUserID from CardHolderSyncTable))

```

GuardPointPro - SALTO integration

```

update CardHolderSyncTable set
    [LastUpdateDate] = GetDate(),
    [Action] = case when INSERTED.type = 3 then 4
                  when INSERTED.type <> 3 then 3
                end
    , [FirstName] = First_Name, [LastName] = Last_Name, [AuditOpenings] = 1, [ExtAccessLevelIDList] =
case len(isnull(ag_name, '')) when 0 then ''
else left(rtrim(ag_name), len(rtrim(ag_name))-1)
END , [ToBeProcessedBySalto] = 1
from INSERTED
where INSERTED.id in (select ExtUserID from CardHolderSyncTable)
AND CardHolderSyncTable.ExtUserID = INSERTED.id

END

GO

DROP TRIGGER [dbo].[CardSyncTrigDel]
GO

CREATE TRIGGER [dbo].[CardSyncTrigDel] ON [dbo].[Card]
AFTER DELETE
AS
BEGIN

    SET NOCOUNT ON;

    insert into CardHolderSyncTable
        ([ExtUserID], [LastUpdateDate], [Action], [FirstName], [LastName], [AuditOpenings],
        [ExtAccessLevelIDList], [AutoKeyEdit.ROMCode], [ToBeProcessedBySalto])
        (select CRDHLD.id, GetDate(), 4, CRDHLD.First_Name, CRDHLD.Last_Name, 1, case len(isnull
        (CRDHLD.AG_Name, '')) when 0 then ''
        else left(rtrim(CRDHLD.AG_Name), len(rtrim(CRDHLD.ag_name))-1)
        END, DELETED.Code + '000000' , 1
        from DELETED inner join CRDHLD on CRDHLD.id = DELETED.Owner
        where
        CRDHLD.id not in (select ExtUserID from CardHolderSyncTable))

    update CardHolderSyncTable set
        [LastUpdateDate] = GetDate(), [Action] = 4, [FirstName] = CRDHLD.First_Name, [LastName] =
        CRDHLD.Last_Name, [AuditOpenings] = 1, [ExtAccessLevelIDList] = case len(isnull(CRDHLD.AG_Name, ''))
        when 0 then ''
        else left(rtrim(CRDHLD.AG_Name), len(rtrim(CRDHLD.ag_name))-1)
        END , [AutoKeyEdit.ROMCode] = DELETED.Code + '000000' , [ToBeProcessedBySalto] = 1
        from DELETED inner join CRDHLD on CRDHLD.id = DELETED.Owner
        where
        CRDHLD.id in (select ExtUserID from CardHolderSyncTable)
        AND CardHolderSyncTable.ExtUserID = CRDHLD.id
    
```

GuardPointPro - SALTO integration

END

GO

```
DROP TRIGGER [dbo].[CARDsyncTrigUpdate]
GO
```

```
CREATE TRIGGER [dbo].[CARDsyncTrigUpdate] ON [dbo].[Card]
AFTER UPDATE, INSERT
AS
BEGIN
```

```
    SET NOCOUNT ON;
```