

Identifying Web User Contacts that will have issues registering/logging in SW 2.0

09/22/2025 5:14 pm EDT

00136106

Identifying Web User Contacts that will have issues registering/logging in SW 2.0

Occasionally, customers who upgrade to SedonaWeb 2.0 from 1.0 or even newly adopt it may encounter issues if a customer's email is listed on multiple contacts /accounts. Sedona Web 2.0 requires the login to be unique and originally, the login was the customer's email. It is possible for a customer to have their email in more than one location such as an old terminated account, a master with multiple subs managed by a single contact, or from acquisitions/migrations.

Scott Pickens created a set of scripts to help potentially identify email contacts that would fall into this category and allow a mass removal.

-- Needs context for step --

```
select c.Customer_Number,c.Customer_Name,ws.* from WS_Account ws
```

```
inner join AR_Customer c on c.Customer_Id =ws.Customer_Id
```

```
where Account_Name ='the_email.com'
```

-- Creates Backup Table of the WS_Account table --

```
Select * into WS_Account_Bak from WS_Account
```

-- Select records where there is an Account_Name in the WS_Account table that does not exist in the AspNetUsers in the SedonaCloud Database --

```
SELECT asp.Id,asp.BaseCompanyId, ws.*
```

```
FROM WS_Account AS ws
```

```
LEFT OUTER JOIN SedonaCloud.dbo.AspNetUsers AS asp ON ws.Account_Name = asp.Email
```

```
WHERE (asp.Email IS NULL) --49
```

```
-- Update the Account_Name field adding 11 in front of the value ---
```

```
-- Where there is an Account_Name in the WS_Account table that does not exist in the AspNetUsers in the  
SedonaCloud Database --
```

```
Update WS_Account
```

```
set Account_Name = '11'+Account_Name
```

```
FROM WS_Account AS ws
```

```
LEFT OUTER JOIN SedonaCloud.dbo.AspNetUsers AS asp ON ws.Account_Name = asp.Email
```

```
WHERE (asp.Email IS NULL)
```

```
-- Restore Account Number from Backup Table if needed to undo changes made. --
```

```
UPDATE WS_Account
```

```
SET WS_Account.Account_Name = BAK.Account_Name
```

```
FROM WS_Account WS
```

```
INNER JOIN WS_Account_Bak bak ON ws.Account_Id = bak.Account_Id
```

```
-- Update ASPNetUsers to reuse email address where no WS_Account record exists.
```

```
select ws.Account_Name,asp.* from SedonaCloud.dbo.AspNetUsers asp
```

```
left outer join WS_Account ws ON ws.Account_Name = asp.Email
```

```
--left outer join AR_Customer_Bill_Email b on asp.Email =b.Email_Address

WHERE (ws.Account_Name IS NULL) and BaseCompanyId =1 and Ext_Employeeld is NULL --49


Update asp

set

Email ='11'+Email,

NormalizedEmail ='11'+NormalizedEmail,

NormalizedUserName ='11'+NormalizedUserName,

UserName='11'+UserName

--Select *

from SedonaCloud.dbo.AspNetUsers asp

left outer join WS_Account ws ON ws.Account_Name = asp.Email

--left outer join AR_Customer_Bill_Email b on asp.Email =b.Email_Address

WHERE (ws.Account_Name IS NULL) and BaseCompanyId =1 and Ext_Employeeld is NULL --49
```
