Purpose of this Stored Procedure

This SP **identifies orphan database users across all databases** on a SQL Server instance and **emails an HTML report**.
It detects:

- **Missing Login** → Database user exists but server login does not
- **SID Mismatch** → Login exists but SID does not match the database user

Step-by-Step Explanation:

1. Disable row count messages
SET NOCOUNT ON;
Prevents extra "x rows affected" messages (important for SQL Agent jobs).

2. Get SQL Server Instance Name
@InstanceName = MachineName[\InstanceName]

- Builds the full instance name
- Used only for reporting (email subject + HTML header)

3. Create Temporary Table
#OrphanUsers
Stores:

- Database name
- User name
- Login name
- Issue type (Missing Login / SID Mismatch)

This table collects results from **all databases**.

4. Scan All User Databases

- Loops through **all ONLINE databases (excluding system DBs)**
- Uses **dynamic SQL** to switch database context
- Reads:
  - sys.database_principals (database users)
  - sys.server_principals (server logins)

Logic:
LEFT JOIN server principals ON user name
IF login is NULL → Missing Login
ELSE → SID Mismatch
Contained users and system users are excluded.

5. Build HTML Report

- Creates an HTML table manually
- Adds:
  - Instance name
  - One row per orphan user
- Uses a **cursor** to safely concatenate rows into HTML

6. Send Email

Uses:

msdb.dbo.sp_send_dbmail

Behavior:

- If orphan users exist → send **HTML report**
- If none found → send **plain text message**

Email subject includes the SQL Server instance name.

Key Benefits

✓ SQL Agent–safe

✓ Works across all databases

✓ Handles SID mismatch correctly

✓ Clean HTML output

✓ No auto-fix (read-only & safe)

Typical Use Case

- Schedule as a **weekly SQL Agent Job**
- Used by DBAs for **security audits**
- Helps after **server migrations or restores**

What This Script Does NOT Do

- Does not auto-fix users
- Does not modify logins or permissions
- Does not run on Azure SQL Database (Managed Instance only)


```
USE DBAScripts
GO
SET NOCOUNT ON;
-------------------------------------------------------------
-- 0. Instance Name
-------------------------------------------------------------
DECLARE @InstanceName SYSNAME;

SELECT @InstanceName =
   CAST(SERVERPROPERTY('MachineName') AS SYSNAME)
   + CASE
      WHEN SERVERPROPERTY('InstanceName') IS NULL THEN ''
      ELSE '\' + CAST(SERVERPROPERTY('InstanceName') AS SYSNAME)
     END;

-------------------------------------------------------------
-- 1. Temp table (NO InstanceName column)
-------------------------------------------------------------
IF OBJECT_ID('tempdb..#OrphanUsers') IS NOT NULL
   DROP TABLE #OrphanUsers;

CREATE TABLE #OrphanUsers
```

```
(
    DatabaseName SYSNAME,
    UserName     SYSNAME,
    LoginName    SYSNAME NULL,
    IssueType    VARCHAR(50)
);

-------------------------------------------------------------
-- 2. Collect orphan users
-------------------------------------------------------------
DECLARE @SQL NVARCHAR(MAX) = N'';

SELECT @SQL = @SQL + N'
USE ' + QUOTENAME(name) + N';

INSERT INTO #OrphanUsers
SELECT
    DB_NAME(),
    dp.name,
    sp.name,
    CASE
        WHEN sp.name IS NULL THEN ''Missing Login''
        ELSE ''SID Mismatch''
    END
FROM sys.database_principals dp
LEFT JOIN sys.server_principals sp
    ON dp.name = sp.name
WHERE dp.type IN (''S'',''U'',''G'')
  AND dp.authentication_type <> 2
  AND dp.name NOT IN (''dbo'',''guest'',''sys'',''INFORMATION_SCHEMA'')
  AND (sp.name IS NULL OR dp.sid <> sp.sid);
'
FROM sys.databases
WHERE state_desc = 'ONLINE'
  AND database_id > 4;

EXEC sys.sp_executesql @SQL;

-------------------------------------------------------------
-- 3. Build HTML Report
-------------------------------------------------------------
DECLARE @HTML NVARCHAR(MAX);

SET @HTML = N'
<html>
<head>
<style>
table { border-collapse: collapse; font-family: Arial; font-size: 12px; }
th, td { border: 1px solid #999; padding: 6px; }
th { background-color: #2F4F4F; color: white; }
```

```
</style>
</head>
<body>
<h2>SQL Server Orphan Users Report</h2>
<p><b>Instance:</b> ' + @InstanceName + N'</p>
<table>
<tr>
<th>Database</th>
<th>User</th>
<th>Login</th>
<th>Issue</th>
</tr>';

DECLARE
   @DB SYSNAME,
   @User SYSNAME,
   @Login SYSNAME,
   @Issue VARCHAR(50);

DECLARE cur CURSOR LOCAL FAST_FORWARD FOR
SELECT DatabaseName, UserName, LoginName, IssueType
FROM #OrphanUsers
ORDER BY DatabaseName, UserName;

OPEN cur;
FETCH NEXT FROM cur INTO @DB, @User, @Login, @Issue;

WHILE @@FETCH_STATUS = 0
BEGIN
   SET @HTML = @HTML +
      N'<tr>
      <td>' + @DB + N'</td>
      <td>' + @User + N'</td>
      <td>' + ISNULL(@Login, N'N/A') + N'</td>
      <td>' + @Issue + N'</td>
      </tr>';

   FETCH NEXT FROM cur INTO @DB, @User, @Login, @Issue;
END

CLOSE cur;
DEALLOCATE cur;

SET @HTML = @HTML +
N'</table>
<br/>
<p>Generated on: ' + CONVERT(NVARCHAR(30), GETDATE(), 120) + N'</p>
</body>
</html>';
```

```
-------------------------------------------------------------
-- 4. Send Email
-------------------------------------------------------------
DECLARE @MailSubject NVARCHAR(255);
DECLARE @NoDataBody  NVARCHAR(MAX);

SET @MailSubject = N'SQL Server Orphan Users Report - ' + @InstanceName;
SET @NoDataBody  = N'No orphan users found on instance: ' + @InstanceName;

IF EXISTS (SELECT 1 FROM #OrphanUsers)
BEGIN
   EXEC msdb.dbo.sp_send_dbmail
      @profile_name = 'DBAAlerts',
      @recipients   = 'sqldbachamps2025@gmail.com',
      @subject      = @MailSubject,
      @body         = @HTML,
      @body_format  = 'HTML';
END
ELSE
BEGIN
   EXEC msdb.dbo.sp_send_dbmail
      @profile_name = 'DBAAlerts',
      @recipients   = 'sqldbachamps2025@gmail.com',
      @subject      = @MailSubject,
      @body         = @NoDataBody,
      @body_format  = 'TEXT';
END
```


Sample Mail Notification Received as follows:

SQL Server Orphan Users Report

Instance: [redacted]

Database	User	Login	Issue
Advworks	Karthik	Karthik	SID Mismatch
Advworks	Menang	N/A	Missing Login
DBADashDB	Karthik	Karthik	SID Mismatch
PMSQLDBA	Karthik	Karthik	SID Mismatch
PMSQLDBA	Menang	N/A	Missing Login
SamsungDB	Karthik	Karthik	SID Mismatch
SamsungDB	Menang	N/A	Missing Login

Generated on: 2025-12-24 10:12:32

Note: DB Mail need to be configured to receive mail notification from within SQL Server.