

# Enable ODBC trace for PHP

Last updated by | Akio Hose | Aug 30, 2021 at 12:01 AM PDT

---

## Contents

- [Enable ODBC Trace for PHP](#)
  - [Issue](#)
  - [Case background](#)
  - [Enable ODBC trace and download file \(Windows\)](#)
    - [Getting the ODBC trace file \(Windows\)](#)
  - [Enable ODBC trace and download file \(Linux\)](#)
    - [Enable ODBC trace](#)
    - [Download ODBC trace file \(Linux\)](#)
      - [Launch WinCSP tool](#)
      - [Download trace file](#)

## Enable ODBC Trace for PHP

### Issue

How to enable ODBC trace on PHP where the application is hosted on Web App Service.

### Case background

The customer is using a third party application which is based on PHP. The application connects to Azure SQL DB where the table has Always Encrypted columns configured. The application encounters errors on Insert operation.

```
[previous exception] [object] (PDOException(code: 22018): SQLSTATE[22018]: [Microsoft][ODBC Driver 17 for SQL Shg257']) at /var/www/html/vendor/laravel/framework/src/Illuminate/Database/Connection.php:459)
```



The "Operand type clash" in the error message can be signs of that the application is not performing encryption and decryption at the client side.

See Always Encrypted: [How it Works](#)

Note that PHP scripts rely on [ODBC drivers](#) to make connections to SQL Server/Azure SQL DB.

### Enable ODBC trace and download file (Windows)

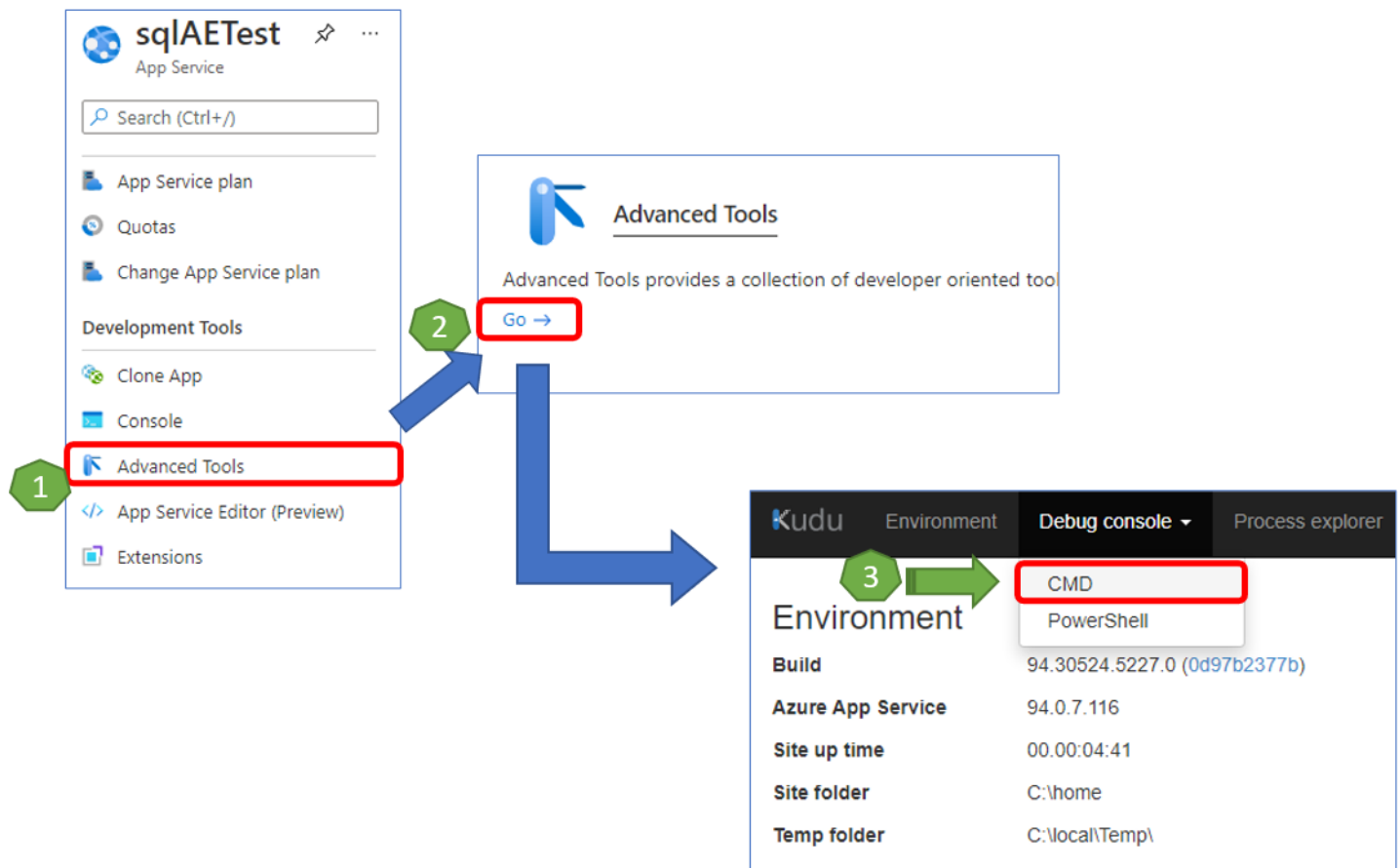
Specify "TraceOn" and "TraceFile" option in the PHP connection string as in the following example:

```
$connectionOptions = array(
    "Database" => "<databaseName>",
    "Uid" => "<userid>",
    "PWD" => "<password>" ,
    "ColumnEncryption" => "Enabled",
    "TraceOn" => "1", "TraceFile" => "odbcTrace.txt"
```

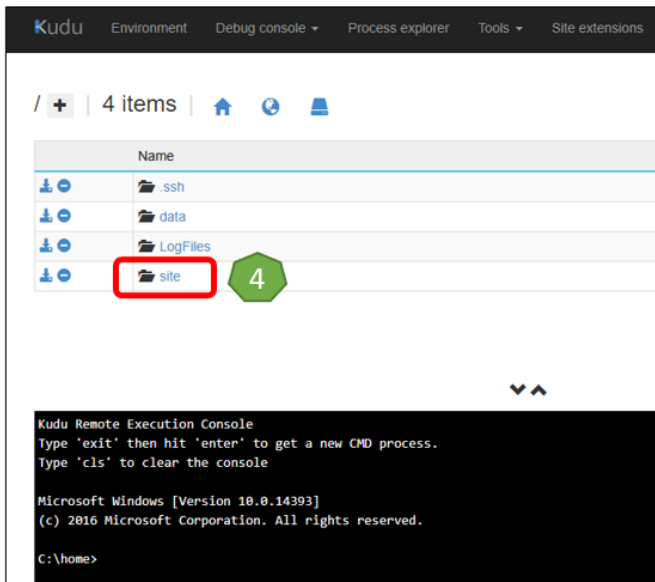
Ref: [PHP Connection Options](#)

### Getting the ODBC trace file (Windows)

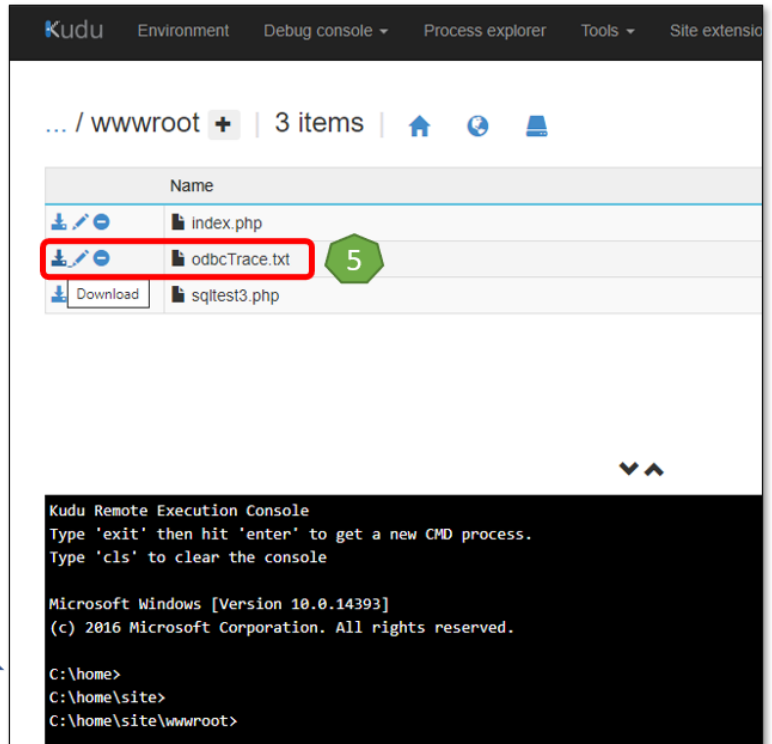
1. In Azure portal, navigate to your App Service, scroll down to Development Tools, select Advanced Tools.
2. Select Go on the Advanced Tools page.
3. Select CMD from the Debug console tab to bring up Kudu



4. Select the site link. Notice this will change the directory in the console. Continue this step until you get to the wwwroot directory. Once you reach the wwwroot directory, you should see the generated ODBC trace file.
5. Select the Download icon. This will display the contents of the trace file. Copy-paste the contents and save it to a local file.



Navigate to  
[c:\home\site\wwwroot]



## Enable ODBC trace and download file (Linux)

### Enable ODBC trace

Enable ODBC trace in PHP connection string. In this example we specify the path (/home/site/wwwroot/odbcTrace.txt) to ensure the output location.

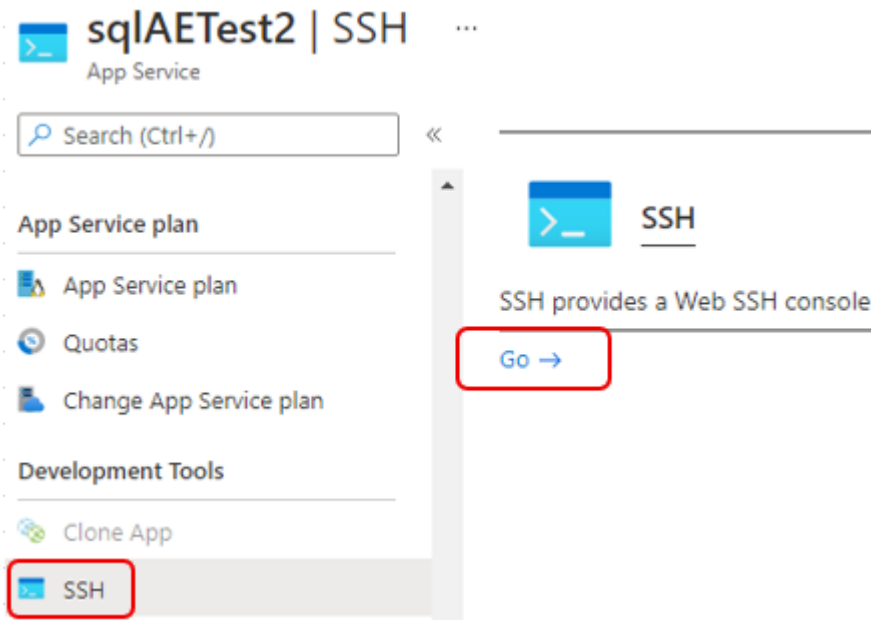
```
$connectionOptions = array(
    "Database" => "<databaseName>",
    "Uid" => "<userid>",
    "PWD" => "<password>" ,
    "ColumnEncryption" => "Enabled",
    "TraceOn" => "1", "TraceFile" => "/home/site/wwwroot/odbcTrace.txt"
```

### Download ODBC trace file (Linux)

Steps overview:

1. Confirm that the trace file is generated
2. Retrieve information to connect to the FTP site
3. Download the file through FTP

From Azure portal, go to App Service and select [Go] from the [Development Tools] blade.



An SSH console is launched on a new page

[illegible]

Go to the destination directory and confirm the trace file is generated.

[illegible]

- In this example we will use [WinCSP](#) to download the file from the FTP site.
- You will need the User Name and password to connect to the FTP site.

Here is how to get the information:

1. From App Service page, select [Deployment Center]
2. Select [Local Git/FTPS credentials] tab
3. Copy required information to connect to the FTP site:
  - o FTP host name
  - o User name and password

sqlAETest2 | Deployment Center

App Service

Search (Ctrl+/)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Security

Events (preview)

Deployment

Quickstart

Deployment slots

Deployment Center

Settings

Configuration

Save Discard Browse Manage publish profile Sync

Settings Logs **Local Git/FTP credentials**

App Service supports multiple technologies to access, publish and modify the content of the application or the user.

FTP endpoint

Git Clone Uri

Application scope

Application scope credentials are auto-generated and provide access only to this specific app or deployment slot. These credentials can be used with FTPS, Local Git and WebDeploy. They cannot be configured manually, but can be reset anytime. [Learn more](#)

Username

Password

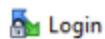
Retrieve the FTP site host name.  
[waws-prod-bn1-121.ftp.azurewebsites.windows.net]  
Notice we are not retrieving the entire string but ONLY the host name.

Copy user name  
(this will be the user name to connect to FTP site)

Copy password

## Launch WinCSP tool

A login dialog is prompt when the WinCSP tool launches. Fill in the required information and select Login to connect to the FTP site.



Session

File protocol: FTP Encryption: TLS/SSL Implicit encryption

Host name: waws-prod-bn1-121.ftp.azurewebsites.windows Port number: 990

User name: sqlAETest2\sqlAETest2 Password: .....

☐ Anonymous login

Save Advanced...

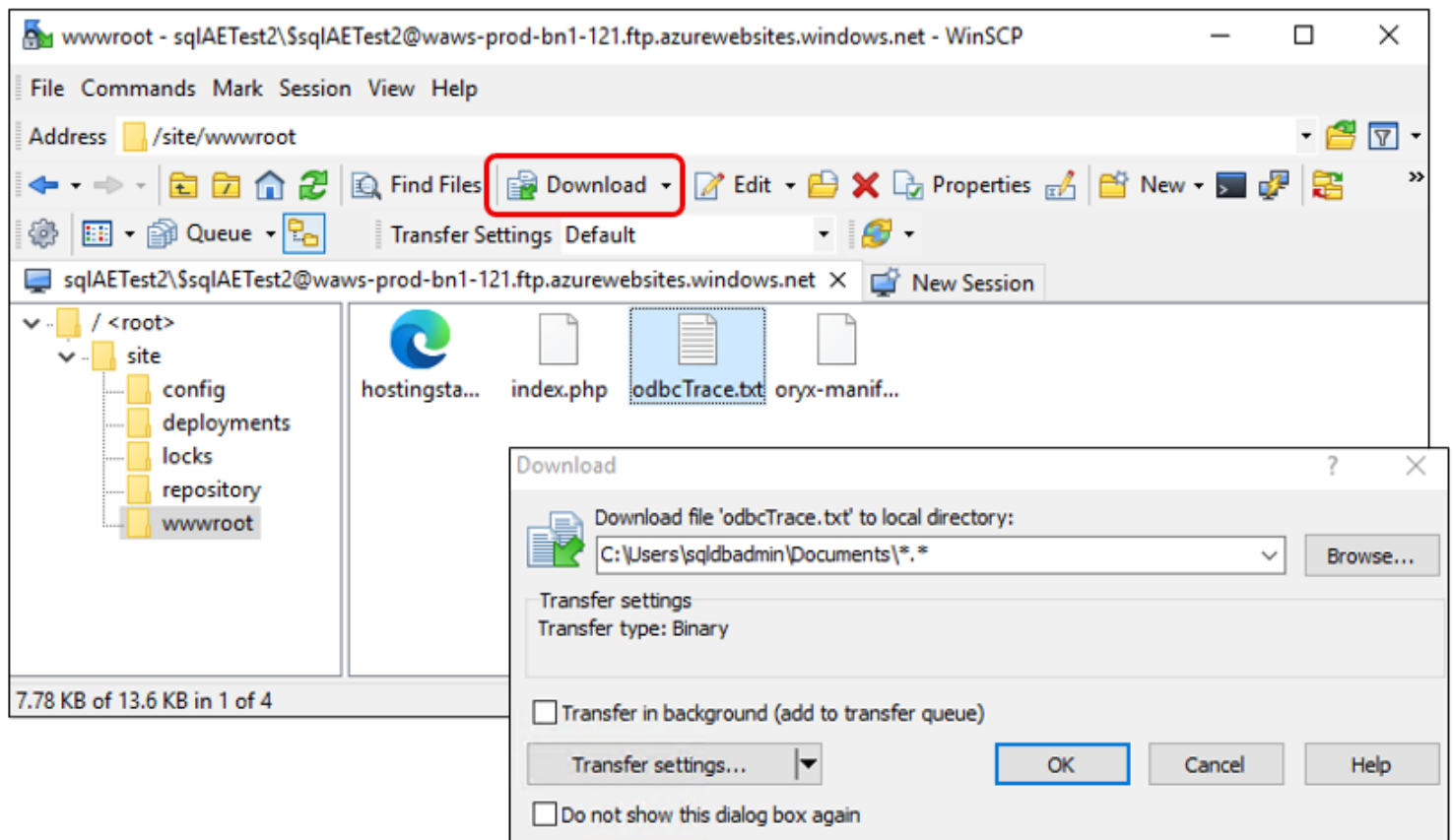
Tools Manage Login Close Help

☒ Show Login dialog on startup and when the last session is closed

### Download trace file

Once connected to the FTP site, a file explorer dialog is displayed.

- In the left pane, navigate to the path where the trace file is located
- Select the trace file
- Select [Download] button to display the Download dialog
- Download the file to your local location



\*\*How good have you found this content?\*\*

