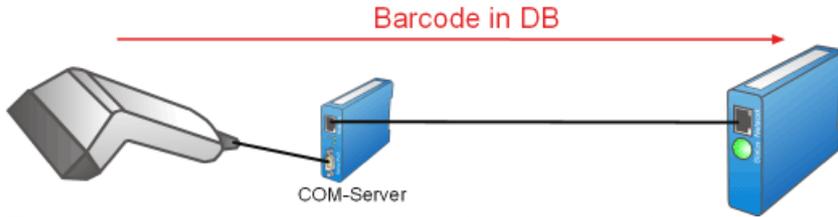


Application:

Log serial data in SQLite database



The example shown here highlights the following features of the pure.box2:

- Execution of a PHP script when data are received on a TCP port
- Reading of the received data through the standard input
- Saving the read information in a box-internal SQLite database

Objective

A company has a number of loading ramps onto which packages are loaded for delivery in trucks. Before the individual packages are loaded, their bar codes need to be read by an employee with a hand scanner. The scanned codes need to then be saved with time stamp at a central location in a database.

The hand scanner used has an RS232 port and is connected to a [W&T Com-Server](#). When a barcode is scanned it is passed through the Com-Server via network to the pure.box2, which then takes in the received data and stores it internally in an SQLite database. The database can then be read by other programs using the traditional SQL syntax.

The PHP file for implementing this example, **save_data.php**, can be downloaded [here](#). In the compressed file you will also find the file **show_data.php**, which opens, reads and displays the database when opened in the browser.

Configuring the COM-Server

In order for the COM-Server used to be able to pass the data read by the hand scanner via network to the pure.box, the following configuration steps are necessary:

- The serial configuration (baud rate, handshake, parity etc.) must be matched to the hand scanner when setting up the COM-Server.
- The COM-Server must be set to TCP Client mode, which enables passing of serially received data to a preset IP address and port number (here: Port 8000) on the pure.box.

Configuration of the pure.box 2

Save the file **save_data.php** using FTP or SMB sharing in the folder named **programs** in the pure.box 2, and the file **show_data.php** in the folder named **websites**. Then the following configuration steps are necessary:

<p>Open the Homepage of the pure.box 2 by entering the IP address in the address line of your browser.</p>	
<p>Select Login in the menu tree and log on using your password in the opened dialog box.</p>	<p style="text-align: center;">Anmelden</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px auto; width: 80%;"> <p>Passwort <input style="width: 90%;" type="password"/></p> <p style="text-align: center;"><input type="button" value="Senden"/></p> </div>

Navigate to the page **Home >> Configuration >> User files**. In the file overview click on the folder named **programs**. The file **save_data.php** is displayed.

Startseite	Ausführung	Pfad	Größe	Änderungsdatum
		userfiles		
		programs		
		save_data.php	1 KB	18.05.2010 16:20:43
1 Objekte(e)				

Clicking on the icon **Edit file properties** (left next to **save_data.php**) opens the Edit dialog.



Check the option **Event controlled**. Select **TCP** and set **Port** to 8000. Save your changes by clicking on **Save...** you're done!

Ereignisgesteuert	<input checked="" type="checkbox"/>	Ausführung bei Datenempfang auf angegebenem Port
Protokoll	<input checked="" type="radio"/> TCP <input type="radio"/> UDP	
Port	<input type="text" value="8000"/>	1024 - 65535

Now when data are sent from the handheld scanner through the Com-Server to TCP port 8000 on the pure.box 2, the script **save_data.php** is run. The received data are read in by the script, a time stamp added and the data are stored in an SQLite database on the pure.box In general you should note that running of the PHP interpreter is automatically stopped after two minutes. The programmed action should therefore be capable of being processed in a shorter time.

Source text

In the first line the standard input for reading data is opened.

```
$in = fopen('php://stdin', 'r');
```

The incoming data are read character by character in a loop and the string variable **\$input** is appended. The loop is exited when a line break (Line Feed, ASCII table: 10) is detected.

```

$input = "";
do
{
    $input .= fread($in, 1);
}
while(substr($input, -1, 1) != chr(10));

```

Then the standard input is closed again.

```
fclose($in);
```

To be able to save the data in the database, a database object for write access must be created, here: **\$db**. If the database does not yet exist, e.g. the first time the script is invoked, the required table **received** with the two columns, **time** for the time stamp and **data** for the received data, are created.

```

$db = new PDO('sqlite:../data.sqlite');
$db -> exec('CREATE TABLE IF NOT EXISTS received (time INTEGER, data TEXT)');

```

By using an SQL-INSERT command a new entry is added in the table **received**. The current time is written to the **time** column in UTC format, and **data** contains the received text. Finally the database is closed again.

```

$db -> exec('INSERT INTO received (time, data) VALUES (' . time() . ', ' . $input . ')');
$db = null;

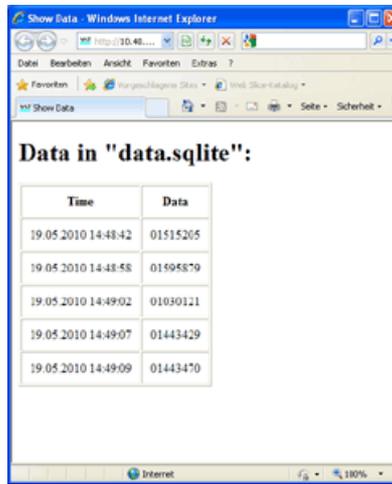
```

The PHP script has now fully run and is exited. When data are received again the script is started again and runs as described above.

If you have previously copied the file **show_data.php** to the sub-folder **websites**, you can open it in your browser using the following URL.

```
http://<IP-Adresse>/websites/show_data.php
```

The page reads the data received so far from the database and displays it in table format.



Time	Data
19.05.2010 14:48:42	01515205
19.05.2010 14:48:58	01595879
19.05.2010 14:49:02	01030121
19.05.2010 14:49:07	01443429
19.05.2010 14:49:09	01443470

Summary

Simple server applications can be implemented with the pure.box 2 just by knowing PHP. The server functionality is already built into the box. All that is left is to program the action you wish to occur when the data are received.



We are available to you in person:

Wiesemann & Theis GmbH
Porschestra. 12
42279 Wuppertal
Phone: +49 202/2680-110 (Mon.-Fri. 8 a.m. to 5 p.m.)
Fax: +49 202/2680-265
info@wut.de

© Wiesemann & Theis GmbH, subject to mistakes and changes: Since we can make mistakes, none of our statements should be applied without verification. Please let us know of any errors or misunderstandings you find so that we can become aware of and eliminate them.

[Data Privacy](#)