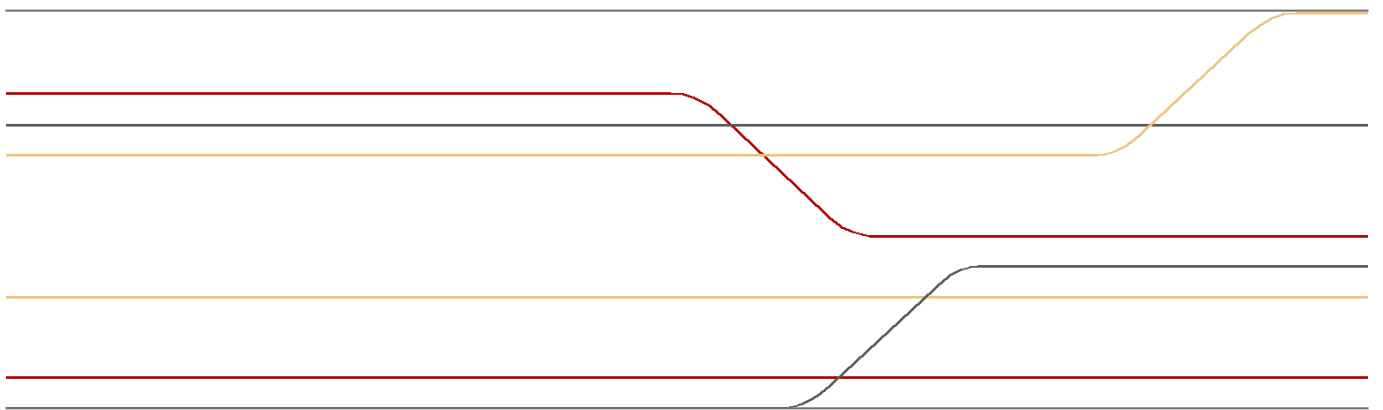


## w.1.03 TIM Server Setup Guide





# Table of Contents

<b>w.1.03 TIM Server Setup Guide</b> .....	i
.....	i
Glossary .....	1
<b>Overview</b> .....	1
What is the TIM Server .....	1
Feature List .....	1
How do I get the TIM Server .....	2
one-off and recurring costs .....	2
<b>How do I run the TIM Server</b> .....	2
Summary TIM Server .....	2
Installation on a Windows system .....	2
System requirements .....	2
<b>Definition of configuration parameters</b> .....	3
Configuration of the ECRI protocol .....	3
Using SIXml .....	3
Using MPDxml .....	3
Using O.P.I. ....	3
Using U.P.I. ....	3
Using cXI .....	3
Configuration of the EFTI protocol .....	4
Using SIXml over IP / LAN .....	4
Using SIXml over RS232 .....	4
Using IFSF eps over IP / LAN .....	4
Using IFSF eps over RS232 .....	4
Configuration of TIM Server working mode .....	4
Configuration of log and debug modes .....	5
Logging .....	5
Debugging .....	6



© 2017 SIX Payment Services Ltd.

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Created: May 24, 2017

## Glossary

Abbreviation / Term	Description
TIM	Till Integration Module
EFT	Electronic Funds Transfer
ECR	Electronic Cash Register
EP2	EFTPOS 2000
ECRI	Interface between ECR and TIM Server
EFTI	Interface between TIM Server and terminal
POS	Point of Sale
cXI	cCredit XML Interface
O.P.I	Open Payment Initiative
UPI	Unified Payment Interface

Table 1: Abbreviations and Terms

## Overview

### What is the TIM Server

The TIM Server is a multifunctional cash register interface driver for all terminals from SIX Payment Services AG. The TIM Server is able to provide all of the major communication protocols to the cash register over a simple homogenous interface. The driver controls all the relevant communication states and provides an economical and secure process of the payment.

The TIM Server offers a simple integration into all ECR-applications as well as system environments. Only one interface is needed to be able to communicate with all of the payment functions. The TIM Server is simple to configure and accomplishes payments reliably.

### Feature List

- secure, reliable transaction process
- full ep2 support
- independent of operating system
- modular technical architecture for any known payment procedure
- $n$  to  $n$  relation between ECR and terminal (both side sharing)
- log and debug function
- supported EFTI protocols: SIXml, cXI
- supported ECRI protocols: SIXml, MPD xml, O.P.I., U.P.I., cXI
- Windows 7 or higher
- Linux
- MAC OS X

## How do I get the TIM Server

please contact [pos-integration@six-group.com](mailto:pos-integration@six-group.com).

### one-off and recurring costs

The merchant (user) of TIM pays a usage fee. For further information please check SIX website.

## How do I run the TIM Server

### Summary TIM Server

For Windows systems it is recommended to register the TIM Server as a service. For other systems it is just called.

As the TIM Server driver supports different working modes and ways to connect a terminal it is required to set unambiguous configuration parameters. The configuration parameters are defined through the command line at startup. You can also create a configuration file *eftdvs.cfg* and put the command line options into it. Just keep them together on the first line of the file. The file *eftdvs.cfg* has either to be in the same directory as the *eftdvs* executable file or you define the path of the file in the command line.

Some configuration parameters can also be defined by the application. The definition in the configuration file overrules the definition by the application!

### Installation on a Windows system

#### **eftdvs.exe /Service**

Installs the program as a Windows Service.

Switches in combination with /Service:

#### **/AutoStart**

Install the Service to be started on system startup automatically (default: Manual Starting)

#### **/UserName uname /Password pw**

Account to run the service, default = System-account

#### **/UnRegServer**

Removes the Service

The SDK provided through the FTP also contains installation .bat files for different setups

### System requirements

The TIM Server is delivered as a 32 and 64-bit application and runs under the following systems:

Windows 7 or higher

Linux

MAC OS X

According to the individual TIM Server setup a firewall configuration may have to be adjusted:

static ports are:

SIXml over IP TIM Server broadcast UDP 33334 (out), 33335 (in)

TCP 7784 (out) is always required

## Definition of configuration parameters

### Configuration of the ECRI protocol

#### Using SIXml

No specific parameter required.

The TIM Api connects to TIM Server on port 7784.

#### Using MPDxml

##### **/ECRPort <port>**

Set the port where TIM Server is waiting for connection from ECR. default is 8137.

##### **/Pepper 1**

For compatibility with Pepper receipt handling.

#### Using O.P.I.

##### **/OPI**

start TIM Server in O.P.I. mode

##### **/OPI <mode>**

With mode the protocol compatibility can be set using a combination of the following values:

Name	Value	Description
V1.3	1	O.P.I. V1.3, default
V1.2	2	O.P.I. V1.2
cCredit	4	Authorization Response with ccredit:RctDetails
xmlns	16	Inserts the XML namespaces in the response
PrinterConnection	32	Keeps the connection to the OPI device persistent
SupShiftReceipt	64	Disable the printing of login and logoff receipts
SuppressReconReceipt	128	Disable the printing of reconciliation receipts

Table 2: Enumeration O.P.I. mode

##### **/OPISchemaLocation <path>**

set the path where the schemas are located, if needed

##### **/OPIServerPort <port>**

set the server port where TIM Server is listening, default is 20002

##### **/OPIDevicePort <port>**

set the device port (eg display printer), default is 20007

#### Using U.P.I.

##### **/UPI**

start TIM Server in U.P.I.mode

#### Using cXI

## **/cXI**

start TIM Server in cXI mode

## **Configuration of the EFTI protocol**

### **/Protocol <protocol>**

Driver uses only the given protocol

Value	Description
SIXML	SIXml
IFSF	IFSF eps

Table 3: overview EFTI protocols

### **Using SIXml over IP / LAN**

#### **/Protocol SIXMLTCP [IP Address]**

defines SIXml protocol over LAN, connect on terminal IP address  
 destination port 7784 cannot be modified  
 e.g. /Protocol SIXMLTCP 10.11.12.13

TIM Server broadcast destination port 33334, cannot be modified  
 terminal broadcast destination port 33335, cannot be modified  
 TIM Server connects on terminal IP, port 7784, cannot be modified

### **Using SIXml over RS232**

#### **/Protocol SIXMLSer [Port Speed]**

defines SIXml protocol over RS232 on given COM port and baudrate  
 e.g. /Protocol SIXMLSer \\.\COM1 115200

### **Using IFSF eps over IP / LAN**

#### **/Protocol IFSF [IP Address]**

defines IFSF eps protocol over LAN by connecting on terminal IP address  
 e.g. Protocol IFSF 10.11.12.13

### **Using IFSF eps over RS232**

#### **/Protocol IFSFSer [Port Speed]**

defines IFSF eps protocol over RS232 on given COM port and baudrate  
 e.g. /Protocol IFSFSer \\.\COM1 115200

## **Configuration of TIM Server working mode**

### **/AutoStart**

normally the device driver is started on the first connect. with /AutoStart set the device driver is started immediately after initialization.

### **/NoProbe**

normally the driver does a probing during initialization. with /NoProbe this is not done. /NoProbe needs /Port, /Speed, and /Protocol to be set

### **/DisableBroadcast**



do not send Broadcast messages to search for device in LAN

#### **/NoSerialPortScan**

do not search for device on serial ports

#### **/NoEvents**

TIM Server sever answers only on Requests. It does not send Status or Exception messages without request.

#### **/KeepAliveMsg**

this parameter is to be set for generating a regular message flow from the server to the client

#### **/TextNoRef**

text in messages has no references for special characters ('ä' → 'ae')

#### **/ReceiptOptions**

this parameter overdrives the Property ReceiptOptions and has be set with a decimal value

Name	Value	Description
SuppressHeader	1	suppresses receipt header delivered by the terminal
unused	2	not supported anymore
SuppressSignature	4	TIM Server does not generate a signature line in the receipt content
SmallFormat	8	small receipt format
SuperSmallFormat	16	super small receipt format
unused	32	not supported anymore
MerchantReceipt	64	different receipts for merchant and cardholder
SuppressECRInformation	128	receipt contains no ECR information
SuppressEFTInformation	256	receipt contains no EFT information
CompactFormat	512	compact receipt format
SuperCompactFormat	1024	super compact receipt format
UltraCompactFormat	2048	ultra compact receipt format

Table 4: Enumeration ReceiptOptions

#### **/ConfigDir <path>** on the commandline

Sets the directory to search for the configuration file instead of the loadir

## **Configuration of log and debug modes**

### **Logging**

#### **/TraceDir <path>**

Sets the directory for the trace output used for logging (result: efdvsYYYYMMDD.log)

#### **/NoZip**

do not compress the trace files

#### **/DeleteLogAge**

this parameter can be used to define the log file storage time in month. default is 12. to allow support / investigations log files should be stored at least for 3 months.

#### **/Trace <options>**

Sets trace and debugging options. Default is *logstd*.

Flag	Description
logstd	Standard log. Traces all relevant activity, suppresses trace in idle states, when no changes
Msg	Trace internal messages
Control	Trace controller messages
Port	Trace serial port traffic

Session	Trace EFT session
opi	Trace OPIDevice requests
Server	Trace socket server
Gateway	Trace gateway activity
Service	Service activity
Warning	Emit extended warnings
pci	creation of PCI audit log

Table 5: trace options

The trace is written to the following files within the selected directory:

- daily in efdvsYYYYMMDD.log (YYYYMMDD stands for e.g. 20170423)
- the old tracefiles will be zipped daily into efdvsYYYYMMDD.log.zip, per week, there is a new zip-file.
- the Zip-files are deleted according to the DeleteLogAge setting.

for Windows dzip32.dll is required (part of the SDK), for unix gzip is used ([www.gzip.org](http://www.gzip.org))

## Debugging

### /Console

Runs the program on console.

To start a troubleshooting session, perform the following steps:

1. Stop the EFT Multiprotocol Service if installed and running.
2. Open a command shell (cmd.exe) and switch to the installation directory.
3. On the command line, enter efdvs /Console /Trace -1

The driver will write all debug messages to the console output.