

MIGS Payment Client Installation Guide

EGate User Manual

CopyrightThe information contained in this manual is proprietary
and confidential to MasterCard International
Incorporated (MasterCard) and its members.

This material may not be duplicated, published, or disclosed, in whole or in part, without the prior written permission of MasterCard.

Trademarks Trademark notices and symbols used in this manual reflect the registration status of MasterCard trademarks in the United States. Please consult with the Customer Operations Services team or the MasterCard Law Department for the registration status of particular product, program, or service names outside the United States.

All third-party product and service names are trademarks or registered trademarks of their respective owners.

Table of Contents

1	Pref	ace	5
	1.1	Who Should Read This Guide	5
	1.2	Related Documents	5
	1.3	Software Versions Covered	5
	1.4	Web Server Versions Covered	6
	1.5	Other Software Dependencies	6
	1.6	Revision History	6
	1.7	Where to Get Help	6
2	Pay	ment Client Installation	7
	2.1	Supported Platforms	7
	2.2	Installation Package	7
	2.3	Windows Payment Client 3.0 Installation	8
	2.4	Windows Payment Client 3.1 Installation	10
	2.5	Unix Payment Client 3.0 Installation	15
	2.6	Unix Payment Client 3.1 Installation	18
3	Test	ing the Payment Client Installation	21
	3.1	Testing the Windows Payment Client	21
	3.1.	1 Windows Payment Client – MOTO Test	22
	3.1.2	2 Windows Payment Client – SSL Test	24
	3.2	Testing the Unix Payment Client	27
4	Pay	ment Client Sample Code Installation	28
	4.1	Supported Programming Languages	28
	4.2	Installation Package Conventions	28
	4.3	ASP Samples ~ Win32 Internet Information Services (IIS)	29
	4.4	Java Samples ~ Win32 Apache Jakarta Tomcat	30
	4.4.	1 Important Information Regarding Java Sources	31
	4.4.2	2 Editing and Recompiling Java Sources	31
	4.5	Java Socket Samples ~ Win32 Apache Jakarta Tomcat	32
	4.5.	1 Starting the Java Socket Listener	32
	4.5.2	2 Important Information Regarding Java Socket Sources	32
	4.6	Perl Samples ~ Win32 Apache HTTP Server	32
	4.6.	1 Important Information Regarding Perl Sources	33
_	4.6.2	2 Editing and Deploying Perl Sources	34
5	Pay	ment Client Uninstallation	35
	5.1	Windows Payment Client Uninstallation	35
	5.2	Unix Payment Client Uninstallation	36

Table of Figures

Figure 1 - Windows Payment Client 3.0 Installation Program	8
Figure 2 - Windows Payment Client 3.0 Installation Options	8
Figure 3 - Windows Payment Client 3.1 Installation Program	10
Figure 4 - Windows Payment Client 3.1 Installation Options	11
Figure 5 - Windows Payment Client 3.1 COM Configuration	12
Figure 6 - Windows Payment Client 3.1 Choose a Java Virtual Machine	12
Figure 7 - Windows Payment Client 3.1 Enter Sockets Configuration	13
Figure 8 - Unix Payment Client 3.0 Installation Folder	15
Figure 9 - Unix Payment Client 3.0 Link Location	16
Figure 10 - Unix Payment Client 3.0 Pre-Installation Summary	16
Figure 11 - Unix Payment Client 3.1 Installation Options	18
Figure 12 - Unix Payment Client 3.1 Java Virtual Machine Selection	19
Figure 13 - Unix Payment Client 3.1 Sockets Configuration	20
Figure 14 - Testing the Windows Payment Client - Payment Client Diagnostics	21
Figure 15 - Testing the Windows Payment Client - MOTO Test Completed	23
Figure 16 - Testing the Windows Payment Client - Client Return URL for SSL Test	24
Figure 17 - Testing the Windows Payment Client - SSL 3-Party Card Select	25
Figure 18 - Testing the Windows Payment Client - SSL 3-Party Card Details	25
Figure 19 - Testing the Windows Payment Client - SSL Return URL	26
Figure 20 - Testing the Windows Payment Client - SSL Test Completed	26
Figure 21 - ASP version of Payment Client Sample Code - IIS Directory	29
Figure 22 - ASP version of Payment Client Sample Code - Contents	30
Figure 23 - Java Sockets version of Payment Client Sample Code - Listener	
Figure 24 - Windows Payment Client Uninstallation - Confirmation	35
Figure 25 - Windows Payment Client Uninstallation - Complete	36

1 Preface

This Payment Client Installation Guide outlines the installation process for the MIGS Payment Client. The purpose of this document is to provide an overview of the following Payment Client installation tasks:

- 1. Installing the Payment Client
- 2. Setting up the *Payment Client Sample Code*

1.1 Who Should Read This Guide

This guide is specifically aimed at business analysts and systems integrators who intend to integrate the *Payment Client* into merchant applications, and personnel who will be involved with the integration and support processes. The purpose of this document is to provide an overview of the installation process. The reader is expected to be familiar with configuring their respective operating systems and web servers.

1.2 Related Documents

This document must be read in conjunction with various other documents which describes the *Payment Client* technical details as well as business logic surrounding transaction processing.

Document Name		Provider
MIGS Payment Client Reference Manual		MasterCard International
MIGS Payment Client Integration Guide	2.3	MasterCard International

1.3 Software Versions Covered

Software	Version
Payment Server	2.2
Payment Client	3.0 and 3.1
Payment Client Sample Code	ASP/COM
	Java
	Java Sockets
	Perl

1.4 Web Server Versions Covered

Due to the multitude of web server implementations and platforms, this document will not attempt to cover all versions, but will instead illustrate various examples.

Web Server	Version	Reference URL
Apache HTTP Server	1.3.27	http://httpd.apache.org/
Apache Jakarta Tomcat	4.1.18	http://jakarta.apache.org/
Microsoft IIS (Internet Information Services)	5.0	http://www.microsoft.com/iis

1.5 Other Software Dependencies

Software	Version	Reference URL
Java Runtime Environment (to run samples)	SUN J2SE 1.3.1	http://java.sun.com/j2se/1.3/
OR	OR	OR
Java Development Kit (for compilation)	SUN J2SE 1.4.2	http://java.sun.com/j2se/1.4.2
Perl (needed for Perl samples only)	5+	http://www.perl.com/

NOTE: Payment Client 3.0 should be used with SUN Java 1.3.1 while Payment Client 3.1 should be used with SUN Java 1.4.2.

1.6 Revision History

Version	Date	Author	History
Draft 0.01	18-Mar-2003	Thomson	Document creation
1.01	06-Aug-2003	Thomson	Document updated
1.2	06-Apr-2004	Thomson	Document updated

1.7 Where to Get Help

For assistance with *Payment Client* integration, please contact your support organisation's help desk. Contact details will be supplied upon service sign-up.

2 Payment Client Installation

The *Payment Client* will only be distributed to authorised systems integrators via an authorised consultant. Please contact a *Payment Client* representative for the *Payment Client* software.

2.1 Supported Platforms

The following platforms are supported by *Payment Client*:

- Windows NT4 (SP6) ~ (All sample code versions)
- Windows 2000 (SP2 or later) ~ (All sample code versions)
- Windows XP ~ (ASP/COM sample code supported in Payment Client 3.1 only)
- Solaris 8 ~ (ASP/COM sample code not supported)
- Red Hat Linux $7.x/8.x \sim$ (ASP/COM sample code not supported)

2.2 Installation Package

NOTE: Either the *Windows Payment Client* or the *Unix Payment Client* will be provided, depending on the operating system of the target machine. Either the *Payment Client 3.0* OR the *Payment Client 3.1* will be provided, depending on your intended deployment Java version.

Installation packages are compressed using the *zip* format, and requires a *zip* decompression program to extract the files from the package. Please contact internal IT support if the program has not been installed.

IMPORTANT: it is the responsibility of the bank and the merchant to keep this installation file in a secure place. These files contain bank and merchant specific encryption keys used in transactions. Therefore, it is imperative that this file is kept in a secure location.

2.3 Windows Payment Client 3.0 Installation

- 1. Unzip all the files from the installation package into a unique temporary directory (e.g. *C:\temp\MIGS_PC*\)
- 2. Open the temporary directory (e.g. *C:\temp\MIGS_PC*) and double-click on the file named *paymentclient_3-0.exe* to start the installation program as illustrated in the following screenshot:



Figure 1 - Windows Payment Client 3.0 Installation Program

- 3. Read through, and then press *Next* on the "*Introduction*" screen.
- 4. Two installation options will be presented on the "Features" screen:





If ASP/COM is to be supported, please choose the option labelled *COM Support*. Otherwise, choose the *Standard Installation*. Press *Next* to continue. **NOTE:** for COM support, the Payment Client 3.0 installation program registers the Payment Client object as a COM object with a utility *javareg.exe*. This allows the Payment Client to run inside the Microsoft VM.

- Choose the installation folder. For consistency it is recommended that the *Payment Client* be installed in the *C:\QSIPayments* directory. Press *Next* to continue.
- 6. A *"Choose Shortcut Folder"* selection screen will be displayed. For consistency the default setting is recommended. Press *Next* to continue.
- 7. A *"Pre-Installation Summary"* confirmation screen will be displayed. Press *Install* to continue.
- 8. An *"Install Complete"* screen will be presented to confirm successful installation of the program. Press *Done* to finish.
- The installation program will close. NOTE: The temporary directory created in *Step 1* (e.g. *C:\temp\MIGS_PC*) should be removed. The installation package should be stored in a secure location.
- 10. Test the payment client by using the *Test Payment Client* program. Please refer to section 3 for help on *Testing the Payment Client Installation*.

2.4 Windows Payment Client 3.1 Installation

Please ensure that SUN J2SE 1.4.2 is installed <u>prior</u> to the installation of Payment Client 3.1.

- Unzip all the files from the installation package into a unique temporary directory (e.g. C:\temp\MIGS_PC\)
- 2. Open the temporary directory (e.g. *C:\temp\MIGS_PC*) and verify that the following files are available:

CardNumbers.profile – preset test card for diagnostics tests and test merchant IDs.

paymentclient_3.1.1.0.exe – Payment Client 3.1 installation program.

qsi.3 / qsi.4 – encryption keys linked to the specific merchant ID(s).

serverurl.properties - URL setting for accessing the Payment Server.

truststore.properties – encryption truststore settings.

NOTE: the encryption keys **qsi.3** and **qsi.4** may have been distributed separately from the Payment Client zip file. Please ensure that these are copied to the temporary directory alongside the other installation files, otherwise installation will fail.

C:\temp\MIG5_PC	
File Edit View Favorites Tools Help	
📙 🖙 Back 🔹 🔿 👻 🛅 🎽 🏹 🗐 🔯 Search 🛛 🖓 Folders	③History 階 🥵 🗙 ᡢ 🔳・
Address C:\temp\MIGS_PC	▼ ∂Go
CardNumbe. paymentclient PC 3.1.1.0 qsi.3 	qsi.4 serverurl.pr truststore
7 object(s)	2.78 MB 🖳 My Computer 🥢

Figure 3 - Windows Payment Client 3.1 Installation Program

Double-click on the file named *paymentclient_3.1.1.0.exe* to start the installation program.

3. Read through, and then press *Next* on the "*Introduction*" screen.

4. Three installation options will be presented on the "*Choose installation options*" screen.

🖳 Dialect Solutions Payment Clien	t 3.1.1.0
	Choose Installation Options
	Please select the configuration of the Payment Client you wish to install. More than one option may be selected. Installation Options Java Sockets IOM
InstallAnywhere by Zero G	
Cancel	Previous

Figure 4 - Windows Payment Client 3.1 Installation Options

Java – select this option if you intend to use Java servlets with direct API calls to the Payment Client.

Sockets – select this option if you intend to use Java servlets or Perl CGI with socket calls to the Payment Client.

COM – select this option if you intend to use ASP/COM with the Payment Client.

Note: although multiple options may be selected, please select the **COM** option only (i.e. do not select Java or Sockets) if you intend to use ASP/COM. Unlike Payment Client 3.0 where the Payment Client object is installed as a COM object via *javareg.exe* and run inside the Microsoft VM, Payment Client 3.1 installs a *PCCOM.dll* file that connects to the Payment Client that off the SUN Java VM (J2SE).

5. If **COM** was selected, a "Select COM Configuration" screen will be displayed:

Dialect Solutions Payment Client 3.1.1.0				
	Select COM Configuration			
- Marine -	Select the COM configuration you wish to install NOTE: For performance reasons the recommended interface to use for the COM Object is the COM Sockets interface (Options 1 or 2).			
	 1. Use COM with an existing Payment Client PCService. 2. Use COM and install a new Payment Client PCService! 3. Use COM and install a Payment Client using Java JNI. Option 1 Use this option to allow COM to connect to an existing Payment Client PCService that has already been installed on the local network. The PCService may have to be configured to accept connections from this host. 			
InstallAnywhere by Zero G				
Cancel	Previous			

Figure 5 - Windows Payment Client 3.1 COM Configuration

It is recommended that option 2 should be used (Use COM and install a new **Payment Client PCService**). Payment Client 3.1 installs a *PCCOM.dll* file that connects to the PCService which runs off the SUN Java VM (J2SE). Press *Next* to continue.

6. Select the correct SUN Java VM at the following screen. Payment Client 3.1 should be used with J2SE 1.4.2.



Figure 6 - Windows Payment Client 3.1 Choose a Java Virtual Machine

- Choose the installation folder at the "Choose Install Folder" screen. For consistency it is recommended that the Payment Client should be installed in the default C:\Dialect directory. Press Next to continue.
- 8. Please select the appropriate proxy configuration at the "*Choose Proxy Configuration*" screen. If you are unsure, please contact your internal network support for information.
- 9. The following sockets configuration screen will be displayed if sockets was selected in step 4, or if COM was selected and option 1 or 2 was selected in step 5:

Dialect Solutions Payn	ent Client 3.1.1.0
	The following settings are required for a sockets installation: - The port number that the PCService will monitor - A list of valid IP addresses that requests will be accepted from, separated by a space - If the PCService is to be installed as a NT Service, the Service Name to use PCService Port 9050 Enter multiple IP Addresses, separated by a space IP Addresses 127.0.0.1 ✓ Install the Payment Client sockets listener as an NT Service Service Name Dialect PCService (Port 9050)
InstallAnywhere by Zero	Previous

Figure 7 - Windows Payment Client 3.1 Enter Sockets Configuration

PCService Port – this is the TCP/IP port on which the PCService socket listener will accept incoming connects on. For general installations on a single server, the localhost should be able to connect to itself via the default port 9050.

Enter multiple IP Addresses – this is the list of IP addresses that will be allowed to connect to the PCService socket listener. For general installations on a single server, the machine connects to itself, therefore the localhost IP of **127.0.0.1** will be sufficient.

Install the Payment Client sockets listener as an NT Service – if this option is checked, the PCService socket listener will be started automatically each time the machine is started. This option is recommended, as it will allow the machine to run the PCService socket listener <u>without</u> the need for a user to be logged onto the system, however this option will require administrative privileges on the system.

Service Name – the NT Service name, only applicable where the option above is checked.

- 10. A screen will be displayed requesting you to "*Choose [the] Shortcut Folder*". For consistency the default setting is recommended. Press *Next* to continue.
- 11. A *"Pre-Installation Summary"* confirmation screen will be displayed. Ensure all the settings are correct, then press *Install* to continue.
- 12. An *"Install Complete"* screen will be presented to confirm successful installation of the Payment Client. Press *Done* to finish.
- The installation program will close. NOTE: The temporary directory created in Step 1 (e.g. C:\temp\MIGS_PC\) should be removed. The installation package should be stored in a secure location.
- 14. Test the payment client by using the *Test Payment Client* program. Please refer to section 3 for help on *Testing the Payment Client Installation*.

2.5 Unix Payment Client 3.0 Installation

- 1. Unzip all the files from the installation package into an empty, unique temporary directory (e.g. ~/*MIGS_PC/*)
- 2. Open a shell and run the following commands:

```
cd ~/MIGS_PC
chmod u+x paymentclient_30.bin
./paymentclient_30.bin
```

NOTE: the installation program requires a graphical display of at least 256 colours. If a graphical display is not available, please start the text mode installation via the command:

```
./paymentclient_30.bin -i console
```

- 3. The installation program will begin. Read through, and then press *Next* on the introduction screen.
- Choose the installation folder. For consistency it is recommended that the *Payment Client* should be installed in the */usr/local/QSIPayments* directory. Press *Next* to continue.

Y	QSI Payments Payment Cli	ent 3.0		///////// - ×
			Choose Ins	stall Folder
Γ		Where Would You Like to	Install?	
]/usr/local/QSIPayments		
			Restore Default Folder	Choose
	nstallAnywhere by Zero G —		Duraulaura	
	Cancel		Previous	

Figure 8 - Unix Payment Client 3.0 Installation Folder

5. A screen will be displayed allowing the user to choose a link directory. This is the directory where symbolic links to the *PaymentClientDiagnostics*, *PCService*, and *Uninstallation* utilities will be created.

💙 QSI Payments Payment Cli	ent 3.0		//////// - ×
		Choose Lini	k Location
	Where would you like to create links? In your home folder Other: Don't create links	Choose Lin	Cht-se
InstallAnywhere by Zero G – Cancel		Previous	Next

Figure 9 - Unix Payment Client 3.0 Link Location

For consistency, it is recommended that the default setting should be used. This will create links in the directory:

~/PaymentClient/

Press Next to continue.

6. A "Pre-Installation Summary" confirmation screen will be displayed. Press *Install* to continue.

QSI Payments Payment Cli	ent 3.0 🗕 🗙
	Pre-Installation Summary
	Please Review the Following Before Continuing:
-	Product Name: QSI Payments Payment Client 3.0
AM ANA	/usr/local/QSIPayments Link Folder: /root
	Feature Set: Standard Installation
	QSIPayments Payment Client 3.0
	Disk Space Information (for Installation Target): Required: 2,114,893 bytes Available: 3,482,804,224 bytes
InstallAnywhere by Z <u>ero G</u> —	
Cancel	Previous Install

Figure 10 - Unix Payment Client 3.0 Pre-Installation Summary

7. An "Install Complete" screen will be presented to confirm successful installation of the program. Press *Done* to finish.

- 8. The installation program will close. **NOTE**: The temporary directory created in Step 1 (e.g. ~/MIGS_PC/) should be removed. The installation package should be stored in a secure location.
- 9. Test the payment client by using the *Test Payment Client* program. Please refer to section 3 for help on *Testing the Payment Client Installation*.

2.6 Unix Payment Client 3.1 Installation

- 1. Unzip all the files from the installation package into an empty, unique temporary directory (e.g. ~/*MIGS_PC/*)
- 2. The encryption keys **qsi.3** and **qsi.4** may have been distributed separately from the Payment Client *zip* file. Please ensure that these are copied to the temporary directory alongside the other installation files, otherwise installation will fail.
- 3. Open a shell and run the following commands: cd ~/MIGS_PC (or your temporary directory from step 1) chmod u+x paymentclient_3_1_1_0.bin ./paymentclient_3_1_1_0.bin

NOTE: the installation program requires a graphical display of at least 256 colours. If a graphical display is not available, please start the text mode installation via the command:

./paymentclient_3_1_1_0.bin -i console

- 4. The installation program will begin. Read through, and then press *Next* on the *"Introduction"* screen.
- 5. Two installation options will be presented on the *"Choose Installation Options"* screen.

Y	Dialect Solutions Payment C	Client 3.1.0.0
		Choose Installation Options
		Please select the configuration of the Payment Client you wish to install. More than one option may be selected. Installation Options ☐ Java IV Sockets
	nstallAnywhere by Zero G	Previous Next
	Cunter	

Figure 11 - Unix Payment Client 3.1 Installation Options

Select the **Java** option if you intend to use Java servlets with direct API calls to the Payment Client.

Otherwise select the **Sockets** option if you intend to use sockets to connect to the Payment Client (e.g. Java sockets or Perl CGI socket calls).

NOTE: either option will allow direct Java API calls, however the sockets option will install the PCService socket listener which opens a socket (defaulting to port 9050) to listen for incoming requests.

6. Select the correct SUN Java VM at the following screen. Payment Client 3.1 should be used with J2SE 1.4.2.

Dialect Solutions Payment C	lient 3.1.0.0 📃 🗶			
Choose a Java Virtual Machine				
The following Java Runtime Environments have been detected on this machine. Please select a Java Runtime Environment 1.3 or 1.4 (or IBM 1.3) to be used.				
	/usr/bin/j2sdk1.4.2_03/bin/java			
InstallAnywhere by Zero G				
Cancel	Previous			

Figure 12 - Unix Payment Client 3.1 Java Virtual Machine Selection

 Choose the installation folder at the "Choose Install Folder" screen. For consistency it is recommended that the Payment Client should be installed in the default /usr/local/Dialect directory. Press Next to continue.

NOTE: installation to the */usr/local/Dialect* directory may require administrative privileges. If you do not have administrative access to your system, you may install the Payment Client in your home directory, however the user that owns the web server process or the PCService process must be able to access the Payment Client and log directories.

8. Please select the appropriate proxy configuration at the "*Choose Proxy Configuration*" screen. If you are unsure, please contact your internal network support for information.

9. The following sockets configuration screen will be displayed if sockets was selected in step 5:

Dialect Solutions Payment Cl	ient 3.1.0.0 – 🗙 Enter Sockets Configuration
	The following settings are required for a sockets installation: - The port number that the PCService will monitor - A list of valid IP addresses that requests will be accepted from, separated by a space PCService Port 9050 Enter multiple IP Addresses, separated by a space IP Addresses 127.0.0.1
InstallAnywhere by Zero G	Previous

Figure 13 - Unix Payment Client 3.1 Sockets Configuration

PCService Port – this is the TCP/IP port on which the PCService socket listener will accept incoming connects on. For general installations on a single server, the localhost should be able to connect to itself via the default port 9050.

Enter multiple IP Addresses – this is the list of IP addresses that will be allowed to connect to the PCService socket listener. For general installations on a single server, the machine connects to itself, therefore the localhost IP of **127.0.0.1** will be sufficient.

- 10. A screen will be displayed requesting you to "*Choose Link Location*". For consistency, the default setting is recommended. Press *Next* to continue.
- 11. A *"Pre-Installation Summary"* confirmation screen will be displayed. Ensure all the settings are correct, then press *Install* to continue.
- 12. An "*Install Complete*" screen will be presented to confirm successful installation of the Payment Client. Press *Done* to finish.
- 13. The installation program will close. **NOTE:** the temporary directory created in Step 1 should be removed. The installation package and encryption key files should be stored in a secure location.
- 14. Test the Payment Client by using the Test Payment Client program. Please refer to section 3 for help on Testing the Payment Client Installation.

3 Testing the Payment Client Installation

The *Payment Client* installation package contains a utility that will allow the systems integrator to test the functionality of the *Payment Client* and connectivity with the *Payment Server*.

3.1 Testing the Windows Payment Client

The Payment Client Diagnostics can be accessed by following the menu: Start Menu > Programs > QSIPayments > PaymentClient > Test Payment Client OR Start Menu > Programs > Dialect > PaymentClient > Test Payment Client

<u>Alternatively</u>, *Payment Client Diagnostics* can be started by running the following commands in a *cmd shell*:

```
C:

cd C:\QSIPayments\PaymentClient\

.\PaymentClientDiagnostics.exe

or

C:

cd C:\Dialect\PaymentClient\bin\

.\PaymentClientDiagnostics.exe
```





Depending on the merchant's setup, they may do MOTO and/or SSL transaction mode tests. Merchants should consult with their bank to determine their transaction mode.

3.1.1 Windows Payment Client – MOTO Test

Prerequisites for the MOTO test:

• MOTO must be *Enabled* in *Merchant Manager* > *Global Permissions*

To perform a MOTO test, start the *Payment Client Diagnostics* program as described in section 3.1.

- 1. Enter the number 1 and press *Enter* to start the MOTO test.
- 2. The program will display the following message: *Please enter a merchant id:*

Generally, the user should use the TEST merchant ID. This is the merchant ID prefixed with *TEST*.

3. The program will display the following message: *Please select a card: ... or to enter another card type Z*

Type Z and press *Enter* to specify a test card. A full list of test cards can be found in *Appendix 3* of the *MIGS Payment Client Integration Guide*.

As a quick reference, the following card can be used for any merchant IDs prefixed with *TEST*:

Card Number	5123456789012346
Expiry Date	0504

a. The program will display the following message: *Please enter a card number:*

The card number should be entered with no spaces or hyphens.

b. The program will display the following message: *Please enter the card expiry date (MMYY):*

NOTE: the actual card expiry date could be in the format *YYMM*, depending on the setup of the merchant.

4. The *Payment Client Diagnostics* program will attempt to contact the *Payment Server* to carry out the test transaction. **NOTE**: depending on how the program was started (see section 3.1) the program will finish up and either close automatically or return to the *cmd shell*.

```
CWUNNT/System32/cmd.exe
Please select a transaction type:
type 1 for MOTO
type 2 for SSL

Please enter a merchant id:
TESTIESAUDALUCYF
Please select a card:
type 2 shortCutRey for x cardNamex x cardNumberx x cardExpiryx
or to enter another card type Z
Z
Please enter a card number:
5123456789012346
Please enter the card expiry date (MMYY):
0504
The following user data has been collected:
test type: MOTO
merchant id: TESTTESNUDALUCYF
card number: 5123456789012346
card expiry: 0504
The MOTO test will be executed with the following data:
merchant id: TESTTESNUDALUCYF
session id: session-id
card expiry: 0504
purchase amount: 100
locale id: en_AU
dr url:
Testing redirected messages from stdout:
Logging initilies estarting
DEBUG: XX0000 - Loading property data: C:\QSIPayments\PaymentClient\config\con.q
sipayments.utility.logging.properties
...Log messages will ne longer appear on standard out
...Log messages will no longer appear on standard out
...Log messages will no longer appear on standard out
...Log the sessage
Lesting stand or the PaymentServer ... OK
Checking the results from the PaymentServer ... OK
Chec
```

Figure 15 - Testing the Windows Payment Client - MOTO Test Completed

5. A log of the test transaction will be stored in the following file: C:\QSIPayments\PaymentClient\PaymentClientDiagnosticsDDMMYYYYHHMMSS.log or

C:\Dialect\PaymentClient\bin\PaymentClientDiagnosticsDDMMYYYYHHMMSS.log

where DDMMYYYHHMMSS is the date and time of the test transaction.

6. The MOTO test has been completely successfully if the following message can be found in the log file:

Checking the results from the PaymentServer ... OK The MOTO test has been completed successfully.

3.1.2 Windows Payment Client – SSL Test

Prerequisites for the SSL test:

- SSL+ must be *Enabled* in *Merchant Manager* > *Global Permissions*
- A *Client Return URL* must be specified in *Merchant Administration* > *Configuration*

Configuration Details Merchant ID: TESTTESAUDALUCYF		Today's date: 4/8/04 Help CLICK HERE
Merchant		
Merchant Name	Lucy's Flower Shop	
Merchant ID	TESTTESAUDALUCYF	
Internationalization		
Locale	English (United States)	
Time Zone	Australia/Sydney	
SSL		
Client Return URL : eg. http://example.com	http://127.0.0.1/SSL/ASP_3party_DR.asp?DR=\${DR}	

Figure 16 - Testing the Windows Payment Client - Client Return URL for SSL Test

The *Client Return URL* needs to be configured in Merchant Administration since no value can be provided in the *Payment Client Diagnostics*. Once the test is complete, the sample codes can be used and the *Return URL* can be sent with the digital order (thus overriding the Merchant Administration setting).

To perform a SSL test, start the *Payment Client Diagnostics* program as described in section 3.1.

- 1. Enter the number 2 and press *Enter* to start the SSL test.
- 2. The program will display the following message: *Please enter a merchant id:*

Generally, the user should use the TEST merchant ID. This is the merchant ID prefixed with *TEST*.

3. The user's default web browser will open, and the server hosted card selection page will be displayed. Click on a *card logo* to select that card.

QSI Payment Technologies - Secure Payme	nt Server - Microsoft Internet Explorer
File Edit View Favorites Tools Help	
Address 🛃 iMvPVitsxc8pWajuX.2nIs4c4nlcJYhyLNT	- ISF3*eLhZ7yLGwVQLHQPvjqrFDYrIOwySHDSYdZMMdkMyl2UiOBa9k7PxJnaK6g_&o=pt ▼ 🔗 Go
MasterCard Internet Gateway Service	Your details will be sent to and processed by The MasterCard Internet Gateway Service and will not be disclosed to the merchant
Mershant name	I ustria Elevert Shan
Pay securely using SS	vould you like to pay? 3L+ by clicking on the card logo below:
	FROM DURING
×	MasterCard VISA
Copyright @200	03 Dialect Solutions Holdings Pty Ltd. All Rights Reserved.
	SECURE PAYMENTS 📩 POWERED BY DIALECT
Al Done	

Figure 17 - Testing the Windows Payment Client - SSL 3-Party Card Select

4. The "Card Details" page will be displayed. Enter a valid test card. A full list of test cards can be found in *Appendix 3* of the *MIGS Payment Client Integration Guide*.

🚰 QSI Payment Technologies - Secure Payment Serve	er - Microsoft Internet Explorer provided by MasterCard International		
File Edit View Favorites Tools Help	100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100		
😓 Back 🕶 🔿 🖉 🖉 🚮 🔞 Search 📾 Favorites 👹 History 🖏 🖬 🖆 🖓 💽			
Address 🙋 https://migs-mtf.mastercard.com.au/ssl?session	id=PAY1060645105617263 ▼ 🗟 🔽		
Your	tetails will be sent to and processed by The		
Mastercaro	erCard Internet Gateway Service and will not be		
Internet Gateway Service	ised to the merchant		
	TEST MODE		
Merchant name: Lucy	s Flowers		
(martine)			
Enter your card details:			
🔒 MasterCard:	rou nave chosen masterCard as your method of navment. Please enter your card details into the		
	form below and click "pay" to complete your		
	purchase.		
Caud Mumbeu 22			
card humber			
Expiry Date !!!	/ month/year		
Security Code 🔠			
	The 3 digits after the card number on the signature		
Durahana ana ana an	panel of your card.		
Purchase Amount :::	A0D \$1.00		
	-		
Ø Done	A LA		

Figure 18 - Testing the Windows Payment Client - SSL 3-Party Card Details

5. A "Processing" page will be displayed, followed by a "Results" page. **NOTE**: Depending on the Client Return URL specified in section 3.1.2 and whether the sample code has been installed, a further "Receipt Page" or an "Error Page" will be displayed. Either one is acceptable for the SSL test, as long as the URL contains the encoded receipt. *Copy* the URL from the browser.



Figure 19 - Testing the Windows Payment Client - SSL Return URL

6. *Paste* the URL to the *Payment Client Diagnostics*.



Figure 20 - Testing the Windows Payment Client - SSL Test Completed

7. A log of the test transaction will be stored in the following file:

C:\QSIPayments\PaymentClient\PaymentClientDiagnosticsDDMMYYYYHHMMSS.log or

C:\Dialect\PaymentClient\bin\PaymentClientDiagnosticsDDMMYYYYHHMMSS.log

where DDMMYYYYHHMMSS is the date and time of the test transaction.

 The SSL test has been completely successfully if the following message can be found in the log file:
 Checking the results from the PaymentServer ... OK The SSL test has been completed successfully.

3.2 Testing the Unix Payment Client

The *Payment Client Diagnostics* can be accessed via the *Test_Payment_Client* link that was created during the installation process. The procedure for testing the Unix Payment Client is similar to the procedure for testing the Windows Payment Client.

4 Payment Client Sample Code Installation

The *Payment Client Sample Code* will only be distributed to authorised systems integrators via an authorised *Payment Client* consultant. These packages contain bank specific customisations, and **MUST NOT BE REDISTRIBUTED** to any other parties. Please contact a *Payment Client* representative for the *Payment Client Sample Code* package.

4.1 Supported Programming Languages

The *Payment Client Sample Code* will contain sample code in various different programming languages. The *Payment Client Sample Code* may be provided in any combination of the following languages:

- ASP/COM
- Java
- Java Sockets
- Perl

4.2 Installation Package Conventions

The *Payment Client Sample Code* installation package generally contains the following directories containing sample code for different transactions:

•	3DS_2.5P	3D Secure "authentication and payment" sample code. Merchant hosted model where the merchant collects the
		card details and sends this through in the digital order.
•	3DS_2.5P_MTS	Similar to the above, with merchant transaction sources.
•	3DS_3P	3D Secure "authentication and payment" sample code.
		Server hosted model where cardholder enters their card
		details on the Payment Server, reducing the risk of
		cardholder information being stolen from the merchant's systems.
•	AMA_CAPTURE	Advanced Merchant Administration command for
		performing captures after a previous authorisation request.
		This is the second step of a two-step authorisation-then-
		capture payment model.
•	AMA_FINTRANS	Command for performing transaction lookup via the
		original transaction number returned with all
		authorisation/purchase transactions.
•	AMA QUERYDR	Command for transaction lookup via the MerchTxnRef
		which is supplied by the merchant with each digital order.
•	AMA REFUND	Command for fully or partially refunding a transaction via
	—	the original transaction number returned with all
		authorisation/purchase transactions.
•	EPS	External Payment Selection sample code. Similar to the
		SSL Server Hosted sample code, however the card type is
		sent in the digital order and thus bypasses the card type

		selection screen on the Payment Server.
•	МОТО	Merchant Hosted payments sample code. Card details are
		captured by the merchant's systems, which communicates
		with the Payment Server for authorisation.
•	MOTO_MTS	Similar to the above, with merchant transaction sources.
•	SSL	Server Hosted payments sample code. Cardholder is
		redirected to the Payment Server to enter their card details,
		reducing the risk of cardholder information being stolen
		from the merchant's systems. Communication is done via
		redirection of the cardholder's browser to the server hosted
		pages.

NOTE: depending on the transaction types supported by your merchant profile, a subset of the sample codes may be provided.

4.3 ASP Samples ~ Win32 Internet Information Services (IIS)

1. The contents of the sample code package should be extracted and placed in the *IIS wwwroot* directory. This directory may be different for different *IIS* installations, however the default value is:

C:\Inetpub\wwwroot

The following screenshot shows the default value in the IIS administration:

Default IIS Web Site Properties				
Directory Security HTTP Headers Custom Errors	Server Extensions			
When connecting to this resource, the content should come from:	ory Documents			
A directory located on this computer				
C A redirection to a URL				
Local Path: C:\Inetpub\wwwroot	Browse			
 Script source access ✓ Log visits ✓ Read ✓ Index this resource ✓ Write ✓ Directory browsing 				
Application Settings Application name: Default Application	Remove			
Starting point: <default iis="" site="" web=""></default>	Configuration			
Application Protection: Medium (Pooled)	Unload			
OK Cancel App	ly Help			

Figure 21 - ASP version of Payment Client Sample Code - IIS Directory

NOTE: for security purposes, ensure that *Script source access* is \Box *unchecked* and *Execute Permissions* is set to *Scripts Only*. \Box *Check* the *Directory Browsing* box if you would like to see the files displayed like Figure 22.

 Test to ensure that the sample code has been placed in the correct directory by pointing the internet browser to the following URL: <u>http://localhost/</u>

A listing similar to the following will be displayed. **NOTE:** *Directory browsing* (see above screenshot) needs to be enabled for the following screen to be displayed. Other directories may also be visible depending on other applications existing on the *IIS* installation. Directory browsing should be turned off in an production environment.

🕘 localho	ost - / - Micr	osoft Inte	ernet l	Explore	r			
File E	dit View	Favorites	Tools	Help	4	• =	• - 🕝 🖸 🖓 🗟 🏈	1
Address	🔄 http://loc	alhost/						▼ ∂‰
loca	alhos	t - /						<u>^</u>
ר ר ר ר ר ר ר ר ר ר ר ר ר ר ר ר ר ר ר	Thursday, Thursday, Thursday, Thursday, Thursday, Thursday, Thursday, Thursday, Thursday,	March March March March March March March March March	11, 11, 11, 11, 11, 11, 11, 11, 11, 11,	2004 2004 2004 2004 2004 2004 2004 2004	11:31 11:31 11:31 11:31 11:31 11:31 11:31 11:43 11:51 12:03 12:11	AM AM AM AM AM AM AM PM PM	<pre><dir> 3DS 2.5P <dir> 3DS 2.5P MTS <dir> 3DS 3P <dir> AMA CAPTURE <dir> AMA FINTRANS <dir> AMA QUERYDR <dir> AMA REFUND <dir> EPS <dir> MOTO <dir> MOTO <dir> SSL</dir></dir></dir></dir></dir></dir></dir></dir></dir></dir></dir></pre>	
e							Eccal in	tranet

Figure 22 - ASP version of Payment Client Sample Code - Contents

3. Any of the available transaction types may now be used to test the *ASP version* of the *Payment Client Sample Code*.

4.4 Java Samples ~ Win32 Apache Jakarta Tomcat

This example assumes that the *Payment Client Sample Code* will be deployed in the *Tomcat Root Context* on a new *Tomcat* installation on port 80. Please refer to *Tomcat* documentation if *Virtual Hosts* is required, or deployment in a *new context* is required. Please consult the *Tomcat* installation owner if the code is being deployed to an existing installation. **NOTE:** if *Virtual Hosts* or a *new context* is used, change the value of *FORM ACTION*, and the *ReturnURL input field* in the sample code.

- Determine the directory for the *Tomcat* installation, e.g. *C:\jakarta-tomcat-4.1.18*\. This will be referred to as %TOMCAT_HOME%.
- 2. The contents of the sample code package should be extracted and placed in a new

%TOMCAT_HOME%\webapps\MIGS_SC directory.

3. Make a backup of the *%TOMCAT_HOME%\conf\server.xml* file. Open the file in a text editor and search for the following lines:

```
<!-- Tomcat Root Context -->
<!--
<Context path="" docBase="ROOT" debug="0"/>
-->
```

Notice that by default these lines are commented out. Change these lines to the following example:

- 4. Start the *Jakarta Tomcat* web server.
- 5. Test to ensure the setup is correct by pointing the internet browser to the URL: *http://localhost/*

The sample code directories should be listed. **NOTE:** the *WEB-INF* directory is a special directory containing the compiled *Java class* files. It is hidden from the internet browser.

6. The *QSI.properties* file may need to be copied into *Jakarta Tomcat*. This can be done by copying the following file:

From C:\WINNT\java\trustlib\QSI.properties

To %TOMCAT_HOME%\bin\QSI.properties

7. Any of the available transaction types may now be used to test the *Java version* of the *Payment Client Sample Code*.

4.4.1 Important Information Regarding Java Sources

The *Java source* files have been included in the directories corresponding to the transaction type (e.g. *MOTO*). Do not to leave these sources in their default directory otherwise they will be publicly downloadable. The *Java source* files can be stored within the *WEB-INF/classes* directory itself.

4.4.2 Editing and Recompiling Java Sources

Java source files can be edited however once they are changed they must be recompiled using the Java Development Kit as specified in Section 1.5 – Other Software Dependencies. Compiled Java class files must be placed back into the WEB-INF/classes directory, and Apache Jakarta Tomcat must be restarted to recognise the new class file.

In order to compile the *Java source* files, add an entry into the system *CLASSPATH*. On *WindowsNT* platforms, this can be done via the command prompt:

set CLASSPATH=%CLASSPATH%;C:\QSIPayments|PaymentClient\classes\Paym entClient.jar

The QSI PaymentClient class files would have been installed during the Payment

Client installation (refer to Section 2 – Payment Client Installation).

4.5 Java Socket Samples ~ Win32 Apache Jakarta Tomcat

Installation of the *Java Socket Samples* is very similar to the *Java Samples*. Please refer to *Section 3.4 – Win32 Apache Jakarta Tomcat*.

4.5.1 Starting the Java Socket Listener

Before testing the *Java Socket Samples*, socket listener must be started. On *WindowsNT* platforms, this can be done via the command prompt:

set CLASSPATH=%CLASSPATH%;C:\WINNT\java\trustlib
java PaymentClient.PCService

The following screenshot is an example of a successfully started listener:



Figure 23 - Java Sockets version of Payment Client Sample Code - Listener

4.5.2 Important Information Regarding Java Socket Sources

The *Java Socket source* files have been included in the directories corresponding to the transaction type (e.g. *MOTO*). It is a good idea not to leave these sources in their default directory otherwise they will be publicly downloadable. The *Java Socket source* files can be stored within the *WEB-INF/classes* directory itself.

4.6 Perl Samples ~ Win32 Apache HTTP Server

This example assumes that the *Payment Client Sample Code* will be deployed in the *DocumentRoot* on a new *Apache HTTP Server* installation on port 80. Please refer to *Apache* documentation if *Virtual Hosts* are requied. Please consult the *Apache HTTP Server* owner if code is being deployed on an existing installation. **NOTE:** if *Virtual Hosts* are used, change the value of *FORM ACTION*, and the *ReturnURL input field* in the sample code.

1. Determine the directory of the *Apache HTTP Server*, e.g. *C:\apache_1.3.27*. This will be referred to as *%APACHE_HOME%*.

- 2. The contents of the sample code package should be extracted and placed a new %APACHE HOME%/MIGS SC directory.
- 3. Make a backup of the *%APACHE_HOME%\conf\httpd.conf* file. Open the file in a text editor and search for the following lines:

```
#
#
DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "C:/apache_1.3.27/htdocs"
```

This should be changed to:

DocumentRoot: The directory out of which you will serve your
documents. By default, all requests are taken from this directory, but
symbolic links and aliases may be used to point to other locations.
DocumentRoot "C:/apache_1.3.27/MIGS_SC"
Search for the following lines:
This should be changed to whatever you set DocumentRoot to.
CDirectory "C:/apache_1.3.27/htdocs">
This should be changed to:
This should be changed to whatever you set DocumentRoot to.
CDirectory "C:/apache_1.3.27/htdocs">
Search for the following lines:
This should be changed to whatever you set DocumentRoot to.
CDirectory "C:/apache_1.3.27/htdocs">
Search for the following lines:
This should be changed to whatever you set DocumentRoot to.
CDirectory "C:/apache_1.3.27/MIGS_SC">
Search for the following lines:
This should be changed to whatever you set DocumentRoot to.
CDirectory "C:/apache_1.3.27/MIGS_SC">
Search for the following lines:
This should be changed to whatever you set DocumentRoot to.
CDirectory "C:/apache_1.3.27/MIGS_SC">
Search for the following lines:
This should be changed to whatever you set DocumentRoot to.
CDirectory "C:/apache_1.3.27/MIGS_SC">
Search for the following lines:
CDirectory "C:/apache_1.3.27/cgi-bin/"
This should be changed to:
CDirectory "C:/apache_1.3.27/cgi-bin/"

This should be changed to: scriptAlias /cgi-bin/ "C:/apache_1.3.27/MIGS_SC/cgi-bin/"

Search for the following line: <Directory "C:/ apache_1.3.27/cgi-bin">

This should be changed to: <Directory "C:/ apache_1.3.27/MIGS_SC/cgi-bin">

- 4. Start the *Apache HTTP Server*.
- 5. Test to ensure the setup is correct by pointing the internet browser to the URL: <u>http://localhost/</u>

The sample code will be listed.

- 6. Start the *Java Socket Listener*. Please refer to *Section 3.5.1 Starting the Java Socket Listener*.
- 7. Any of the available transaction types may now be used to test the *Perl version* of the *Payment Client Sample Code*.

4.6.1 Important Information Regarding Perl Sources

The *Perl source* files (*.cgi) have been included in the directories corresponding to the transaction type (e.g. *MOTO*). It is a good idea not to leave these sources in their default directory otherwise they will be publicly downloadable. Remove these and leave all *cgi* files in the *cgi-bin* directory.

Also note that depending on the *Apache HTTP Server* and *Perl* setup, the first line of all the *cgi* files may need to be changed. The first line of the samples may have been

defaulted to: #!C:/Per1/bin/per1

This will need to be changed to the location of an existing Perl interpreter.

4.6.2 Editing and Deploying Perl Sources

As mentioned above, *cgi* sources have been left in their corresponding directories for reference only. Changed *cgi* files must be placed back in the *cgi-bin* directory to function correctly.

5 Payment Client Uninstallation

5.1 Windows Payment Client Uninstallation

To uninstall the *Windows Payment Client*, the following steps need to be followed:

- (Step 1 is for COM Installations only) IMPORTANT: if the *Payment Client* had been installed using the *COM Support* option for ASP/COM: temporarily stop the IIS web server prior to uninstalling the *Payment Client*. NOTE: the web server will be down, and must be restarted after completing the uninstallation process. Please contact the web server owner for correct procedures.
- 2. Use the following shortcut to start the uninstallation process: Start Menu > Programs > QSIPaymentS > PaymentClient > Uninstall Payment Client
- 3. A confirmation screen will be displayed. Click on the *Uninstall* button to begin the uninstallation process.



Figure 24 - Windows Payment Client Uninstallation - Confirmation

4. Once the uninstallation process is complete, a summary screen will be displayed. Press *Quit* to close the uninstallation program.

🖳 InstallAnywhere Uninstaller		
	Uninstall QSI Payments Payment C	lient 3.0
	Uninstall Complete	
	The following items could not be removed:	
	Unable to remove directory: C:\QSIPayments\PaymentClient\log Unable to remove directory: C:\QSIPayments\PaymentClient\config	<u> </u>
InstallAnywhere by Zero G	<u>×</u>	T
Cancel		Quit

Figure 25 - Windows Payment Client Uninstallation - Complete

5. As illustrated in the screenshot above, certain files/directories cannot be removed automatically:

```
C:\QSIPayments\PaymentClient\log
C:\QSIPayments\PaymentClient\config
```

These are configuration files and log files. Backup the files by using the following command:

ren C:\QSIPayments\PaymentClient PaymentClientBackup

 (Step 6 is for COM Installations only) IMPORTANT: if the Payment Client had been installed using the COM Support option for ASP/COM: check that the directory C:\WINNT\java\TrustLib\PaymentClient has been removed.

5.2 Unix Payment Client Uninstallation

The *Uninstall_Payment_Client* link that was created during the installation process can be used to start the uninstallation process.