cardlink

Modirum MPI

Version 4.0.2.56

Merchant Interface v4

18. Mar 2019. a

Table of Contents

Introduction

Modirum has developed a range of software products to support the use of 3-D Secure protocol for payer authentication. Modirum MPI is a product that is developed for merchants, acquirers or payment service providers that want to implement the 3-D Secure Merchant Plug-In (MPI) protocol. Modirum MPI is compliant with Visa, MasterCard, American Express, Diners and JCB schemas (protocol version 1.0.2). Since version 4.0.2.51 also ThreeDSecure 2.0 and 2.1 support has been established. As of now 2.0 is discontinued and Modirum MPI is certified to 2.1 protocol.
This document is the reference manual of Modirum MPI merchant interface. It has been targeted to merchants who connect to and process 3D Secure through Modirum MPI
The structure of this document is as follows:

- Section HTTP POST Interface describes the merchant application interface of the product. The interface enables implementation of 3-D Secure protocol without any programming work. This section includes a separate summary of different result values of the 3-D Secure process (mdStatus, eci).

The new interface specification introduces recently emerged new MPI extension such as Brazil Market and Trusted Third Party and ThreeDSecure 2.x specifics and extensions. Please check for new fields in this version to ensure compability (look for “Since 4.0.2.54”)

1.1 Prerequisites

It is assumed that the reader has basic knowledge of payment cards in the traditional commerce and in electronic commerce. It is also assumed that the reader has basic knowledge about the acquiring process of the payment cards.

1.2 Glossary

This section includes selected terms that are commonly used in the document. An extensive 3-D Secure glossary is available in 3-D Secure: System Overview, available through the “Vendors & Merchants” link on http://corporate.visa.com.

ACS =Access Control Server	A component that operates in the Issuer Domain, verifies whether authentication is available for a card number, and authenticates specific transactions.
API	Application Programming Interface
HTML	HyperText Markup Language
PSP/Processor	Payment services provider
MPI Manager/Admin Tool	User interface implementation of the MPI database. The tool can be used to maintain merchant and directory server data as well as to browse transactions.
Merchant Plugin	A component that operates in the Acquirer Domain; incorporated into the merchant’s Web storefront to perform functions related to 3-D Secure on behalf of the merchant, such as determining whether authentication is available for a card number and validating the digital signature in a 3-D Secure message.
MPI	See Merchant Plug-in.
PAN	Primary Account Number – often referred to as card number
PATransReq	Payer Authentication Transaction Request; a record of authentication activity sent by the ACS to the Authentication History Server
PATransRes	Payer Authentication Transaction Response; Authentication History Server response to PATransReq
PAReq = Payer Authentication Request	A message sent from the Merchant Server Plugin to the Access Control Server via the cardholder device. Requests the issuer to authenticate its cardholder and contains the cardholder, merchant, and transaction-specific information necessary to do so.

PARes = Payer Authentication Response	A message formatted, digitally signed, and sent from the Access Control Server to the Merchant Server Plug-in (via the cardholder device) providing the results of the issuer's 3D Secure cardholder authentication.
VEReq = Verify Enrollment Request	A message from the Merchant Server Plug-in to the Visa Directory or from the Visa Directory to the ACS, asking whether authentication is available for a particular card number
VERes = Verify Enrollment Response	A message from the ACS or the Visa Directory, telling the Merchant Server Plug-in whether authentication is available
Directory Server	A server hardware/software entity which is operated in the Interoperability Domain; it determines whether authentication is available for a specific card number, and if so, it returns the URL of the appropriate Access Control Server to the Merchant Server Plugin.
WAP	Wireless Application Protocol
WML	Wireless Markup Language
3DS Server	A server component used in ThreeDSecure 2.x that resempbles somewhat MPI functionality of ThreeDSecure 1.0 and is responsible with communications and interfacing with Directories and 3DS Requestors and ACS redirections in browser and redirection mode.
3DS Requestor	An end entity or Merchant participating in 3DS Secure 2.x protocol.
AReq	Authentication request in 3DS 2.x protocol
Ares	Authentication response in 3DS 2.x protocol
RReq	Results request in 3DS 2.x protocol
RRes	Results request response in 3DS 2.x protocol
CReq	Challenge request in 3DS 2.x protocol
Cres	Challenge request response in 3DS 2.x protocol

Table 1: Glossary

1.3 Getting Started

The purpose of this section is to describe the basic functionality of Modirum MPI. Functions of the product are walked- through with help of examples. More detailed description of the product is included in the rest of the document.

Note that the examples can be tested with a demo merchant and a live Modirum MPI. In order to get access to Modirum demo site, please contact Modirum support.

In order to try the following examples with Modirum software, please note the following recommendations:

- With the example merchant application, it is recommended to use FireFox, Chrome, or Internet Explorer 9 (or newer). The software has also been tested with other browsers, but there have been some technical problems with earlier browser versions.
- It is highly recommended to use a display with a minimum 1024x786 graphics resolution.

1.4 Which MPI Interface Should Be Used

Modirum MPI supports several alternative protocols for integration to the payment page:

1. HTTP POST as Foreground
2. HTTP XML or XML+SOAP API

Which one to use depends on how much integration effort you are willing to invest. The biggest architectural decision you need to make is whether you will be using the MPI in “foreground” or “background” mode.

In “foreground” mode, the MPI is exposed directly to the Internet as a separate web application, typically on a separate dedicated server or a separate web application on some server also used for other purposes.

In “XML” mode, the MPI may not be directly exposed to Internet for end users but only for inbound directory connection for 3DS2 RReq, but some other application, typically the one displaying the payment page, handles the user browser session and passes data between the cardholder browser and the MPI.

HTTP POST as foreground is the easiest and the technical integration effort should be up to few days, whereas XML integration effort may be a few weeks.

1.4.1 HTTP POST as Foreground

Payment page and MPI are separate applications residing on same or different servers. In this model the transaction flow is:

1 The payment page places MPI input values into HTTP fields on HTML page and redirects the user browser to the MPI (using html form with Javascript).
2 MPI contacts directory and either returns immediately (not enrolled of frictionless authentication) or redirects the user browser session to the ACS for challenge authentication.
3 ACS authenticates
4 MPI receives the user session back from the ACS.
5 MPI verifies the ACS response and Directory Response in 3DS2.
6 The MPI redirects the user browser back to the payment page and payment page application reads the return values from HTML fields.

1.4.2 XML API or with SOAP

Payment page and MPI are separate applications residing on same or different servers. In this model, MPI never receives the user browser session. The payment page application has the user session, performs necessary redirections and receives the response.

In this model the transaction flow is:

1. The payment page places MPI input values into XML message and sends a background system-to-system HTTP(S) call to the MPI.
2. MPI contacts directory and returns either final response, continue with enrollment and perform 3ds method or finalized HML redirect form with included formatted PAReq/CReq message and ACS URL to the payment page application.
3. If cardholder was enrolled and or challenge authentication requested, payment page application redirects the user browser to the ACS with posting the PAReq/CReq from acquired from MPI.
4. ACS authenticates
5. Payment page application receives the user session back from the ACS with PARes/CRes.
6. Payment page application submits the ACS response PARes/CRes to the MPI in XML message for verification.
7. MPI verifies the ACS response and DS direct response in case of 3DS2.
8. The MPI places results to merchant with final XML message with verified results and payment page application reads the return values from XML message fields.

Note: The communications in background use XML or XML SOAP messages defined in interface specification sections in this document.

So, the payment application should be aware of this and should be capable to send such requests to mpi and extract the needed values from response XML/XML SOAP.

2 HTTP POST Interface

The simplest way to add 3-D Secure support to existing Merchant site is to use HTTP Post interface of the Modirum MPI.

The control flow using HTTP Post interface is the following:

1. Merchant payment page asks the user all the relevant payment data, such as card number, expiry, etc.
2. Merchant payment page calculates digest (version 3.0 or earlier) or the signature (v 4.0, detailed below) of all the fields to be posted to Modirum MPI including the shared secret (version 3.0 or earlier) and POSTs these fields except the shared secret to the Modirum MPI.
3. Modirum MPI checks the card participation from the 3-D Secure directory and return the redirection page to the Issuer ACS. The right directory is found with either card scheme id or matching the beginning of the card number.
4. After the ACS is finished with the user authentication, ACS returns the control to the Modirum MPI. The MPI Server verifies the signature
5. of ACS and POSTs the response to either success URL or fail URL, which the MPI has earlier passed to the ACS.
6. Merchant payment page reads the response and continues with the authorization of the payment (or does the error processing).

2.1 Parameters of the Post to MPI Server

2.1.1 Start or process an authentication session

The following table describes the parameters of the POST from the payment page to Modirum MPI.
HTTP POST interface version has been upgraded to 4.0.

Field **General parameters**	**Required / Optional/ Conditional**	**Description**
version	R	4.0
cardType	O/R	Card scheme can be defined explicitly. Values with one digit are valid. The value is matched to directory server entry that is mapped to the merchant with that particular cardType The field can also be left empty, which means that the directory server is determined from the card number
pan	R	Card number. 13-19 digit account number Note: must nor present if cardEncData field is used
expiry	R	Expiration Date supplied to the merchant by cardholder (YYMM). Note that this field is not checked in most of the production ACS installations. Note: must nor present if cardEncData field is used
cardEncData	C	In case Client-side card data encryption used with VPOS support. And then fields pan and expiry must be missing

deviceCategory	R	Integer length 1, Indicates the type of cardholder device. Supported values are: 0 = www 1 = legacy mobile (deprecated, means wml interfaces) 4 = dtv (deprecated) 5 = mobile SDK (for ThreeDSecure 2.x only) 6 = for ThreeDS Requestor initiated (3RI) flow (for ThreeDSecure 2.x only) Since version 4.0.2.37 this field is required. Most cases this value shall be always 0 as wap phones are no longer mainstream and may not be supported by ACSes.
purchAmount	O	Max. 12-digit numeric amount in minor units of currency with all punctuation removed. (Optional for NPA only) Examples: Display Amount USD 123.45 Purchase Amount 12345
exponent	O	Exponent number length 1 (Optional for NPA only) The minor units of currency specified in ISO 4217. For example, US Dollars has a value of 2; Japanese Yen has a value of 0.
description	O	Purchase description. Brief description of items purchased, determined by the merchant. Maximum size is 125 characters, but merchant should consider the characteristics of the cardholder's device when creating the field. Note: Most Issuers do not display the description to the user.
currency	O	Currency. Determined by merchant. ISO 4217, 3-digit numeric. (Optional for NPA only)
merchantID	R	ID to identify the merchant to Modirum MPI.
merchantName	O	**Length**: 1-25 characters **Format**: any characters Send this parameter if you want to override PAReq.Merchant.name. Effective only in Multinamed or PSP mode only merchant only. Else of by default it is taken from mpi database. ***
xid	R	**Unique** transaction identifier determined by merchant. Contains a 20 byte statistically unique value that has been Base64 encoded, giving a 28-byte result. Note: used only in or for 3DS1.0.2 protocol, but still needed for any case to be present for later reference lookups etc.
okUrl	R/-	Fully qualified URL to Merchant. Modirum MPI will do POST when finished to this URL in all other cases, except when authentication or signature validation fails (mdStatus = 0). Max length 2048 chars **Not applicable if deviceCategory = 5 or 6**
failUrl	R/-	Fully qualified URL to Merchant. Modirum MPI will do POST to this URL when authentication or signature validation fails. mdStatus = 0. Max length 2048 chars **Not applicable if deviceCategory = 5 or 6**
MD	O	The MD ("Merchant Data") field: merchant state data that must be returned to the merchant. The content of this field is passed unchanged and without assumptions about its content to the return POST. This field is used to accommodate the different ways merchant systems handle session state. If the merchant system does not maintain state for a given shopping session, the MD can carry whatever data the merchant needs to continue the session. This field must contain only ASCII characters in the range 0x20 to 0x7E; **Since version 4.0.2.15 characters '<' and '>' are not allowed in this parameter.** If other data is needed, the field must be Base64 encoded. The size of the field (after Base64 encoding, if applicable) is limited to 254 bytes. If MD includes confidential data (such as the PAN), it must be encrypted.
recurFreq	O	Recurring frequency for PAReq/AReq Purchase.Recur.frequency (integer days, 28 means monthly)
recurEnd	O/R	Recurring end date for PAReq/AReq format YYYYMMDD, if recurFreq is present then recurEnd is required
installments	O	Number of Installments for PAReq/AReq.Purchase.install integer value >1 and <=999. I**nstall and recurring parameters can not be present at the same time.**

panMode	O	Possible values: **VPOSToken** – to indicate that instead of real pan pan parameter will contain VPOS token.
3DS 2.x general extra fields (reflecting the fields of EMVCo spec)		
TDS2.acctID	O	Variable string..64 extra account info
TDS2.acctType	O	Fixed str 2: 01 = Not Applicable, 02 = Credit, 03 = Debit
TDS2.addrMatch	O	Fixed string Y/N: Y = Shipping Address matches Billing Address. N = Shipping Address does not match Billing Address.
TDS2.cardholderName	O	Var string 2..45. Cardholder name
TDS2.email	O	Var string ..254
TDS2.homePhone	O	Var string ..19 (..3-..15): Home phone format cc-subscriber 1-686123456
TDS2.mobilePhone	O	Var string ..19 (..3-..15): Mobile phone format cc-subscriber 1-686123456
TDS2.workPhone	O	Var string ..19 (..3-..15): Work phone format cc-subscriber 1-686123456
TDS2.messageCategory	O	Fix string 2: 01 = PA(payment), 02 = NPA(non-payment), defaults to 01
TDS2.messageVersion	O	In case it is required to force MPI to use particular message version for this transaction (in case MPI and Issuer supports it). If not supported, then MPI will use one of supported messageVersion-s Format: 2.0.1 or 2.1.0 Length: 5. Mostly for specific testing only. Not recommended to be used as MPI resolves on its own appropriate version.
TDS2.purchaseDate	O/C	Purchase date in GMT: Format YYYYMMDDHHMMSS. Optional for normal purchases and will de defaulted to current time. Shall be present for installments and recurrings and set to original purchase agreement date .
TDS2.transType	O	Fix string 2: 01 = Goods/ Service Purchase 03 = Check Acceptance, 10 = Account Funding, 11 = Quasi-Cash Transaction, 28 = Prepaid Activation and Load
TDS2.threeDSRequestor3RIInd	O	Fix string 2: Unsure its use case: 01 = Recurring transaction 02 = Installment transaction, 03 = Add card,04 = Maintain card information, 05 = Account verification. (this is field threeRIInd in case of 2.1), to be used only if deviceCategory=6.
TDS2.threeDSRequestorAuthenticationInd	O	Type: string Length: 2 Accepted values: 01 - Payment, 02 -recurring, 03 - instalment, 04 – add card??, 05 – maintain card??, 06 – verification as part token EMV token ID
TDS2.threeDSRequestorChallengeInd	O	Fix string 2. Challenge request prefrences indicator. 01 = No preference 02 = No challenge requested, 03 = Challenge requested: 3DS Requestor Preference, 04 = Challenge requested: Mandate
TDS2.threeDSRequestorID and TDS2.threeDSRequestorID.{schemaId}	O	Type: string Length: 1-35. Note: Normally filled by MPI from its configuration *** If You have multiple schema values and unsure with what schema card goes to send schema specific values for all possible schemas
TDS2.threeDSRequestorName and TDS2.threeDSRequestorName.{schemaId}	O	Type: string Length: 1-40 Note: Normally filled by MPI from its configuration *** If You have multiple schema values and unsure with what schema card goes to send schema specific values for all possible schemas
TDS2.threeDSRequestorNPAInd	O/R	Fix string 2, unsure of use case: 01 = Add card, 02 = Maintain card information, 03 = Cardholder verification as part of EMV token ID&V According to EMVCo 3DS Spec Required if TDS2.messageCategory is 02
TDS2.threeDSRequestorURL	O	Fully qualified URL of 3DS Requestor website or customer care site. If not provided, then url will be taken from merchant config in database. Length: Var string..2048 (applies also for 3DS1) Example: http://server.domainname.com
TDS2.challengeWindowSize	O	Desired challenge window size for 2.x. Number 2. Values accepted 01 = 250 x 400, 02 = 390 x 400, 03 = 500 x 600, 04 = 600 x 400, 05 = Full screen
TDS2.payTokenInd	O	Type: string Length: 4
TDS1.acquirerBIN and TDS1.acquirerBIN.{schemaId}	O	Type: numeric, Used as first priority if 3DS1 and present. Length: 1-11 (if present overrides acqBIN ***) If You have multiple schema values and unsure with what schema card goes to send schema specific values for all possible schemas

TDS2.acquirerBIN and TDS2.acquirerBIN.{schemaId}	O	Type: numeric Length: 1-11 (if present overrides acqBIN ***) If You have multiple schema values and unsure with what schema card goes to send schema specific values for all possible schemas
TDS1.acquirerMerchantID and TDS1.acquirerMerchantID.{schemaId}	O	Type: string. Used as first priority if 3DS1 and present. Length: 1-35 (if present overrides merchantId, ***) If You have multiple schema values and unsure with what schema card goes to send schema specific values for all possible schemas
TDS2.acquirerMerchantID and TDS2.acquirerMerchantID.{schemaId}	O	Type: string Length: 1-35 (if present overrides merchantId, ***) If You have multiple schema values and unsure with what schema card goes to send schema specific values for all possible schemas
TDS2.merchantName	O	Type: string (applies also for 3DS1 if merchantName not set) Length: 1-40 (if present overrides merchantName, **)
TDS2.mcc and TDS2.mcc.{schemaId}	O	Type: numeric Length: 4 (if present overrides mcc in profile ***) If You have multiple schema values and unsure with what schema card goes to send schema specific values for all possible schemas
TDS2.merchantCountryCode	O	Type: numeric Length: 3 (if present overrides country code in profile ***)
		*** Effective only if merchant PSP mode (3DS1 effect also)
		** Effective only if merchant PSP or Multi named mode (3DS1 also)
3DS 2.0 Merchant Risk fields		
TDS2.mriShipIndicator	O	Format 2 numeric characters, See table A.7.2 EMVCO spec
TDS2.mriDeliveryTimeframe	O	Type: numeric Length: 1-2
TDS2.mriDeliveryEmailAddress	O	Format up to 254 numeric characters eamil, See table A.7.2 EMVCO spec
TDS2.mriReorderItemsInd	O	Type: numeric Length: 1-2
TDS2.mriPreOrderPurchaseInd	O	Type: numeric Length: 1-2
TDS2.mriPreOrderDate	O	Format 8 character, date YYYYMMDD See table A.7.2 EMVCO spec
TDS2.mriGiftCardAmount	O	Format max len 15 numeric value in minor units.
TDS2.mriGiftCardCurr	O	Format 3 numeric characters ISO4217
TDS2.mriGiftCardCount	O	Format 2 numeric characters
3DS 2.0 extra account info fields		
TDS2.chAccAgeInd	O	Length of time that the cardholder has had the account with the 3DS Requestor. Format and values according to EMVCo specification
TDS2.chAccDate	O	Fix YYYYMMDD, Date that the cardholder opened the account at merchant
TDS2.chAccChangeInd	O	Length of time since the cardholder's account information with the 3DS Requestor was last changed. Including Billing or Shipping address, new payment account, or new user(s) added. Format and values according to EMVCo specification
TDS2.chAccChange	O	Fix YYYYMMDD, Date that the cardholder's account with the 3DS Requestor was last changed. Including Billing or Shipping address,
TDS2.chAccPwChangeInd	O	Length of time since the cardholder's account with the 3DS Requestor had a password change or account reset. Format and values according to EMVCo specification
TDS2.chAccPwChange	O	Fix YYYYMMDD, Date that cardholder's account with the 3DS Requestor had a password change or account reset.
TDS2.nbPurchaseAccount	O	Var num string..4: Number of purchases with this cardholder account during the previous six months.
TDS2.provisionAttemptsDay	O	Var num string..3: Number of Add Card attempts in the last 24 hours.
TDS2.txnActivityDay	O	Var num string..3: Number of transactions (successful and abandoned) for this cardholder account with the 3DS Requestor across all payment accounts in the previous 24 hours.
TDS2.txnActivityYear	O	Var num string..3: Number of transactions (successful and abandoned) for this cardholder account with the 3DS Requestor across all payment accounts in the previous year.
TDS2.shipAddressUsageInd	O	Indicates when the shipping address used for this transaction was first used with the 3DS Requestor. Format and values according to EMVCo specification
TDS2.shipAddressUsage	O	Fix YYYYMMDD, Date when the shipping address used for this transaction was first used with the 3DS Requestor
TDS2.shipNameIndicator	O	Type: numeric Length: 1-2

TDS2.paymentAccInd	O	Indicates the length of time that the payment account was enrolled in the cardholder's account with the 3DS Requestor. Format and values according to EMVCo specification
TDS2.paymentAccAge	O	Fix YYYYMMDD, Date that the payment account was enrolled in the cardholder's account with the 3DS Requestor.
TDS2.suspiciousAccActivity	O	Indicates whether the 3DS Requestor has experienced suspicious activity (including previous fraud) on the cardholder account. Type: numeric Length: 1-2
3DS 2.0 extra billing addres fields (may be required by some schemas)		
TDS2.billAddrCity	R	Var string..50, City
TDS2.billAddrCountry	R	Fix num 3, country code 3166-1 numeric
TDS2.billAddrLine1	R	Var string..50, Address line 1
TDS2.billAddrLine2	O	Var string..50, Address line 2
TDS2.billAddrLine3	O	Var string..50, Address line 3
TDS2.billAddrPostCode	R	Var string 16, Zip code
TDS2.billAddrState	R	Var str 2 3166-2 country subdivision code
3DS 2.0 extra shipping addres fields		
TDS2.shipAddrCity	O	Var string..50, City
TDS2.shipAddrCountry	O	Fix num 3, country code 3166-1 numeric
TDS2.shipAddrLine1	O	Var string..50, Address line 1
TDS2.shipAddrLine2	O	Var string..50, Address line 2
TDS2.shipAddrLine3	O	Var string..50, Address line 3
TDS2.shipAddrPostCode	O	Var string 16, Zip code
TDS2.shipAddrState	O	Var str 2 3166-2 country subdivision code
3DS 2.0 extra authentication info **Information about how the 3DS Requestor authenticated the cardholder before or during the transaction.**		
TDS2.AIAuthMethod	O	Fix num 2: 01 = No 3DS Requestor authentication occurred (i.e. cardholder "logged in" as guest) 02 = Login to the cardholder account at the 3DS Requestor system using 3DS Requestor's own credentials 03 = Login to the cardholder account at the 3DS Requestor system using federated ID 04 = Login to the cardholder account at the 3DS Requestor system using FIDO Authenticator (maps to threeDSReqAuthMethod)
TDS2.AIAuthTimestamp	O	Date YYYYMMDDHHMM, Date and time in UTC of the cardholder authentication. (maps to threeDSReqAuthTimestamp)
TDS2.AIAuthData	O	Var str 2048. Data that documents and supports a specific authentication process. (maps to threeDSReqAuthData)
3DS 2.0 extra authentication info **Information about how the 3DS Requestor authenticated the cardholder as part of a previous 3DS transaction.**		
TDS2.PAIRef	O	Var sting 36. (threeDSReqPriorRef) This data element contains a ACS Transaction ID for a prior authenticated transaction
TDS2.PAIAuthMethod	O	Fir num 2. (threeDSReqPriorAuthMethod) Mechanism used by the Cardholder to previously authenticate to the 3DS Requestor. 01 = Frictionless authentication occurred by ACS 02 = Cardholder challenge occurred by ACS
TDS2.PAIAuthTimestamp	O	Date YYYYMMDDHHMM UTC. Date and time in UTC of the prior cardholder authentication.
TDS2.PAIAuthData	O	Var strr 2048. (threeDSReqPriorAuthData) Data that documents and supports a specific authentication process.
signature	R	Base64 encoded value of signature (RSA with SHA2-256) of all the field values above concatenated together having ";" semicolon in between (including trailing semicolon after the last value). signature=base64(RSA with SHA2-256(utf8bytes(value1;value2;...;value n;))) If missing or mismatching error is displayed and request in not further processed **Requests are signed by merchant private key and validated with merchant Certificate (merchant certificate generation is referred to section 2.2)**

Table 2: HTTP POST parameters for authentication session

2.1.2 Query authentication results

This interface makes it possible to ask for the status of already processed authentication in MPI.

Field **General parameters**	**Required / Optional**	**Description**
version	R	Version **4.0** of Modirum MPI HTTP POST protocol.
merchantID	R	ID to identify the merchant to Modirum MPI.
service	R	Value: statusQuery
format	R	Valid values: for application/x-www-form-urlencoded format is "NVP"
xid	R	Unique transaction identifier determined by merchant to query. Contains a 20 byte statistically unique value that has been Base64 encoded, giving a 28 byte result.
lastcres	O	If APP posses last CRes it shall post it with status query. (base64 url encoded) If this field is present CRes is validated and processed before returning results.
signature	R	Base64 encoded value of signature (RSA with SHA2-256) of all the field values above concatenated together having ";" semicolon in between (including trailing semicolon after the last value). Signature = base64(RSA with SHA2-256(utf8bytes(value1;value2;...;value n;))). If missing or mismatching, error is displayed, and the request is not further processed. **Requests are signed by merchant private key and validated with merchant Certificate (merchant certificate generation is referred to section 2.2)**

Table 3: HTTP POST parameters for query service

2.2 Calculation of the Signature

The digest in the incoming POST and in the return POST is calculated by the following rule.

1. Concatenate all the values of all the possible fields with semicolon ";" in between (including trailing semicolon after the last value) as listed and in same order as in the table. If a parameter is missing (not sent) or sent value is empty string "", then value and separator is not concatenated.
2. Calculate RSA with SHA2-256 signature of step 1 (using of UTF-8 char encoding when converting string to bytes) using your private key.
3. Return the signature in base64 encoded form.

Notes: Never implement the signature calculation in the browser using JavaScript or similar as this way you expose your private key to the world. Only implement it on server side executed code as (jsp/servlet/asp/php etc.).

For example:
Signature=base64(RSA with SHA2-256(utf8bytes(value1;value2;...;value n;)))

Example code for Java users:

```
java.security.PrivateKey privateKey = getPrivateKey(); //fetch your private key
java.security.Signature signature = Signature.getInstance("SHA256withRSA");
signature.initSign(privateKey);
signature.update(concatenatedValues.toString().getBytes(StandardCharsets.UTF_8));
byte[] sigBytes=signature.sign();
String sigStr=Base64.encode(sigBytes);
```

Example for .Net users:

https://docs.microsoft.com/en-us/dotnet/api/system.security.cryptography.rsacryptoserviceprovider.signdata?view=netframework-4.7.2
See also sigexample.cs

```
public static string sign(string valueToSign, String privateKey)

{
        byte[] keybytes= parseKey(privateKey);
        byte[] toSign = System.Text.Encoding.UTF8.GetBytes(valueToSign);
        RSACryptoServiceProvider RSAalg = DecodePrivateKeyInfo(keybytes);
        byte[] signature=RSAalg.SignData(toSign, "SHA256"); //RSAwithSHA256
        string sigStr= Convert.ToBase64String(signature);
        return sigStr;
}
```

Note: easiest ways to generate Your merchant private key and self-signed certificate is:

a) Java keytool:
keytool -genkeypair **-keyalg RSA -sigalg SHA256withRSA** -alias merchant -keystore merchant.p12 -storepass mypassword -validity 1440 -keysize 2048

keytool -export -alias merchant -keystore merchant.p12 -rfc -file merchant_x509.cer

b) Or using openssl:
openssl req -sha256 -newkey rsa:2048 -nodes -keyout merchant.pem -x509 -days 1440 -out merchant_x509o.cer

2.3 Redirection to ACS

The MPI Server generates a redirection page, which includes a form that is posted to the ACS. The format of this page depends on the channel. PAReq redirection-default templates of different channels are defined in WEB-INF/templates xsl files.

2.4 Returning from MPI Server to Merchant Site

The standard method for returning control back from the MPI Server to the merchant is a redirection through the user browser.

The standard control flow after receiving the PARes (or CPRS with condensed messages) is:

1. ACS Posts the PARes back to MPI Server via the user browser using the redirection.
2. MPI Server determines the status of the transaction and posts the status values to merchant payment page via the user browser using the redirection.
3. Merchant server continues with authorization and after the authorization displays the confirmation page to the user.

2.5 Parameters of return POST from MPI Server

The following table describes the parameters of the POST to okURL or failURL by Modirum MPI.

Field	Required / Optional	Description
version	R	Version of Modirum MPI HTTP POST protocol. Copied from the HTTP POST request. Currently **4.0**
merchantID	R	From merchantID field of the incoming POST
xid	R	From xid field of the incoming POST
mdStatus	R	End status of the transaction mdStatus field provides all the information that is needed to determine, how to manage the transaction in the merchant system See section 2.6 Modirum MPI Transaction mdStatus
mdErrorMsg	R	Up to 128 chars alphanumeric description of the error.
veresEnrolledStatus	R	The actual value of veres enrollement status like "Y", "N", "U" or "-" if value is not available due errors etc. (since 4.0.2.31). **Note:** In case of UPOP protocol this value is always Y except status response code 77. In case of 3DS2 this is set to Y if directory is contacted.
paresTxStatus	R	The actual value of pares tx status "Y", "N", "U", "A", “R”, “C” or "-" if value is not available due errors or not enrolled. (since 4.0.2.31) **Note:** In case of UPOP protocol this value is equal to **activateStatus** (Y, A, F, L, N) if respCode is 00. - in case respcode 77 and ant other respcode cases N (not authenticated). In case of 3DS2 this is ARes or RRes transStatus whatever was latest event
iReqCode	O	Two-digit numeric code provided by ACS indicating data that is formatted correctly, but which invalidates the request. This element is included when business processing cannot be performed for some reason. Never provided if mdStatus=0. 3-D Secure iReqCode field. **Note:** In case of UPOP protocol this value is equal to **respCode** if respCode is not 00.
iReqDetail	O	May identify the specific data elements that caused the Invalid Request Code (so never supplied if Invalid Request Code is omitted). See Table 20 on page 60 for details. Note: In case of UPOP protocol this value is equal to respMsg if respCode is not 00.
vendorCode	O	Error message describing iReqDetail error.
eci	O	Electronic Commerce Indicator With Visa cards, the value to be passed in Authorization Message (exactly 2 decimal digits). ECI fields determine the final status of the transaction See section 2.6 Modirum MPI Transaction mdStatus
cavv	O	Cardholder Authentication Verification Value Determined by ACS. Contains a 20-byte value that has been Base64 encoded, giving a 28 byte result. Some Visa regions may require that this value be included in the VIP authorization message (Visa EU is not using CAVV as of 10.10.02). **Note:** In case of UPOP protocol this value is equal to UPOP **vcode**.
cavvAlgorithm	O	A positive integer indicating the algorithm used to generate the Cardholder Authentication Verification Value. Current defined values are: 0 = HMAC (as per SET™ TransStain) 1 = CVV 2 = CVV with ATN 3 = MasterCard AAV Note that the value is copied directly from the PARes/RReq message.
MD	O	From MD field of the incoming POST
PAResVerified	O	If signature validation of the return message is successful, the value is true. If PARes message is not received or signature validation fails, the value is false. (Compliance testing facility requirement). If signature validation is omitted the value will be empty In case of 2.x there is no signatures but signature valid flag is set to true for backward compability, as data is trusted from mutual TLS authentication.
PAResSyntaxOK	O	If PARes validation is syntactically correct, the value is true. Otherwise value is false. (Compliance testing facility requirement)

protocol	O	Protocol used in processing eg: 3DS1.0.2, 3DS2.1.0, SP5 Will be present if processing started. **Note: new Since 4.0.1.54**
cardType	O	Card type resolved by MPI if not in request or same value as in request. **Note: new Since 4.0.1.54**
fssScore	O	Fraud score calculated by the Modirum FSS. Present if the MPI is configured for use with the FSS, if setting fss.includeScoreInResponse has been set to true and if the score is available.
3DS 2.x general fields		
TDS2.transStatus	O/C	Provided if available (means if 3DS2)
TDS2.transStatusReason	O/C	Provided if available Since 4.0.2.54
TDS2.threeDSServerTransID	O/C	Provided if available
TDS2.dsTransID	O/C	Provided if available
TDS2.acsTransID	O/C	Provided if available
TDS2.acsRenderingType	O/C	Provided if available
TDS2.acsReferenceNumber	O/C	Provided if available
TDS2.acsSignedContent	O/C	Provided if available and request deviceCategory = 5
TDS2.authTimestamp	O/C	Provided if available YYYYMMDDHHMM, 3DSServer set timestamp of authentication received in GMT (may change in future)
TDS2.messageVersion	O/C	3DS Message version used. (example 2.1.0 or 2.0.1) Since 4.0.2.54
TDS2.acsChallengeMandated	O/C	Provided if available Since 4.0.2.54
TDS2.authenticationType	O/C	Provided if available Since 4.0.2.54
TDS2.acsOperatorID	O/C	Provided if available Since 4.0.2.54
TDS2.cardholderInfo	O/C	Provided if available. If provided. Merchant shall show contents of this field to cardholder. Since 4.0.2.54
TDS2.acsUrl	O/C	Provided if available Since 4.0.2.54
TDS2.challengeCancel	O/C	Provided if available **Since 4.0.2.56**
TDS2.AResExtensions	O	Contains ARes.messageExtension content if it was present in message. Example: [{"name":"name","id":"id","criticalityIndicator": false,"data": {"text":"test"}}, {…}]
TDS2.RReqExtensions	O	Contains RReq.messageExtension content if it was present in message. Example: [{"name":"name","id":"id","criticalityIndicator": false,"data": {"text":"test"}}, {…}]
signature	R	Base64 encoded value of signature (RSA with SHA2-256) of all the field values above concatenated together having ";" semicolon in between (including trailing semicolon after the last value). Signature = base64(RSA with SHA2- 256(utf8bytes(value1;value2;...;value n;))). If missing or mismatching, do not continue to payment as may be counterfeit or error message **Responses are signed by processor private key and validated with Processor certificate (processor certificate is referred to Section 5. page 51)**

Table 6: Parameters of return POST from MPI Server

Note: In case of deviceCategory = 5 then no html is returned but name value pairs content only in application/x-www- form-urlencoded format. Response to service query also returns this set of parameters in application/x-www-form-urlencoded format.

2.6 Modirum MPI Transaction mdStatus

mdStatus and eci fields

Modirum MPI returns many fields in the HTTP POST return interface, but there are two fields that have the most significance from business transaction management point of view.

- mdStatus field provides all the information that is needed to determine, how to manage the transaction in the merchant system:

- 0 = Not Authenticated, do not continue transaction
- 1 = Fully Authenticated, continue transaction
- 2 = Not enrolled (3DS1 only, may continue to transaction)
- 3 = Not enrolled cache (not used, was used in early 3DS1)
- 4 = Attempt (Proof of authentication attempt, may continue to transaction)
- 5 = U-received (grey area, continue to transaction depends schema rules)
- 6 = Error received (from Directory or ACS)
- 8 = Block by Fraud Score (used only if set up with FSS and scores configured)
- 9 = Pending (this code is not sent in red, internal, status only, except XML API or SDK in challenge)
- 50 = Execute 3DS method and continue to authentication (XML API or SOAP only in return EnrollmentRequest(initial)).
- 51 = No 3DS method and continue to authentication (XML API or SOAP only in return).
- 60 = Action completed, not sent only used internally for PReq result updates.
- 80 = Skip device case (no 3DS performed as device known not well capable)
- 81 = Skip challenge because requestor challenge ind (02 = No challenge requested)
- 88 = Skip 3DS because low risk (used only if with FSS and scores configured)
- 91 = Network error
- 92 = Directory error (read timeout)
- 93 = Configuration error
- 94 = Merchant input errors (also if merchant posted CRes not matching current tx state by RReq)
- 95 = No directory found for PAN/cardtype
- 96 = No version 2 directory found for PAN/cardtype
- 97 = If transaction not found on continue or service query
- 99 = System error (mpi unexpected error)

- eci field is relevant in Visa infrastructure, because there are certain business rules of how to fill the field into the authorization messages. However, in Modirum MPI, eci field is filled only if the field is provided from the ACS in the PARes message (as specified in the protocol spec). The value should then be passed to the Visa authorization infrastructure as such.

Note the following recommendations:

- Firstly, check mdStatus to determine the final status of the transaction.

- The protocol specification requires passing the eci field as such to the authorization infrastructure if it is provided in the PARes message (2.x ARes or PReq). Modirum MPI has value in eci field only if it has been provided in the PARes message (2.x ARes or PReq).

Based on the current understanding of the Visa infrastructure business rules (note that this information is not necessarily accurate and needs to be checked with Visa), the following eci values are recommended to be used in authorization infrastructure:

- 6 = Non-participation and attempts (issuer liability).
- 9x = Service unavailable or system errors (transaction can proceed with the TLS/SSL-secured liability rules, merchant has the liability).
- No value, if authentication or signature check fails

Fully Authenticated Transactions

In fully authenticated transactions, the issuer and cardholder are participating in the 3-D Secure protocol, cardholder authentication has been successful and signature has been validated.

Scenarios:

- Valid PARes or CPRS message has been received with status Y and the signature has been validated successfully. Or in 2.x ARes or PReq has transStatus Y.

Values:

- mdStatus = 1
- MPI ECI is copied as such from the PARes or 2.x ARes or PReq (often 05 for Visa and 01 for MasterCard).
- Visa authorization ECI value 5 (note that this must be verified from Visa).

Non-participation and Not Enrolled Transactions

In the case, where the card issuer or cardholder is not participating to the 3-D Secure, the transaction can still be completed. Business rules of this case depend on the card scheme. Also, the MPI may determine the non-participation based on the card range missing from the cache.

Scenarios:

- VERes message has been received with status N (there is no difference with the issuer not participating compared with the cardholder not participating).
- MPI uses cache and cannot find the card range in the cache.

Values:

- mdStatus = 2 if card not participating, or mdStatus = 3 if range not in cache.
- MPI ECI is empty.
- Visa authorization ECI value 6 (note that this must be verified from Visa).

Attempts

In 3-D Secure version 1.0.2, an attempt functionality was introduced. ACS implementations may produce signed attempt-messages even in the cases where the issuer or cardholder is not participating.

Scenarios:

- ACS has produced PARes with A status and signature validation has been successful (can be issuer specific ACS or centralized attempt-server). In 2.x in case ARes or PReq transStatus is A.

Values:

- mdStatus = 4
- MPI ECI is copied as such from the PARes

Authentication Errors

If authentication is not successful, the transaction should not be continued in those cases where the failure is fatal.

Scenarios:

- ACS has provided PARes with status N (e.g too many false passwords or signature check has failed)
- PARes signature validation has failed.
- In case of 2.x ARes or PReq transStatus is N, R

Values:

- mdStatus = 0
- MPI ECI is empty.
- Visa authorization ECI value is not relevant, because transaction cannot be continued (note that this must be verified from Visa).

Authentication Unavailable

In some cases, the authentication may not be available. The unavailable status may come from the directory server or ACS or it may be related to some data communication problems.

Scenarios:

- Directory server produces VERes message with U status or 2.x ARes or PReq transStatus is U.
- Directory server produces VERes or ACS PARes message with U status (e.g mobile protocol not supported).
- No connection to the directory server.

Values:

- mdStatus = 5
- MPI ECI is empty.
- Visa authorization ECI value 7 (note that this must be verified from Visa).

3-D Secure Errors

3-D Secure related error situations are managed separately.

Scenarios:

- Invalid input field in 3-D Secure message generation
- Merchant authentication in directory server fails
- Error message received from DS or ACS

Values:

- mdStatus = 6
- MPI ECI is empty
- Visa authorization ECI value 7 (note that this must be verified from Visa)

Merchant data errors

If merchant request was invalid or incomplete then:

Values:

- mdStatus = 94
- MPI ECI is empty
- Visa authorization ECI value 7 (note that this must be verified from Visa)

System, networks or directory errors

Sometimes Modirum MPI may produce some error messages related to MPI internal errors. In most cases, it is still recommendable to proceed with the transaction if risk is small.

Scenarios:

- File system full or system misconfigurations.
- Database connection lost.
- Directory connection fails or response times out.

Values:

- mdStatus = 99, mdStatus = 92 or mdStatus = 93
- MPI ECI is empty
- Visa authorization ECI value 7 (note that this must be verified from Visa)

Fraud Score Block

If Modirum MPI is configured to use a fraud scoring service, the score is calculated prior to attempting 3-D Secure authentication. If the score returned exceeds the allowable threshold, then the transaction is not submitted for authentication.

Scenario:

- Fraud score received from scoring service exceeds configured threshold

Values:

- mdStatus = 8
- MPI ECI is empty
- Visa authorization ECI value is not relevant, because transaction cannot be continued with Visa infrastructure (note that this must be verified from Visa)

Pending Transactions

If MPI has passed the PAReq or 2.x CReq to the ACS, the browser session has also moved away from the MPI. Sometimes, the control may never come back to the MPI, which means that there will be a pending transaction in the database for the time beeing.

Modirum MPI does not include a timer to react to the pending transactions, but this functionality needs to be implemented at the merchant site.

The background direct post feature in MPI Server can be used to avoid using the user browser for redirection in return. (This won't help any if user does not return from ACS).

The control flow using background direct post feature is:

1. ACS Posts the PARes (2.x CRes) back to MPI Server via the user browser using the redirection
2. MPI Server determines the status of the transaction and post the status values to merchant payment page using a background session directly from MPI Server to the Merchant payment page server.
3. Merchant server continues with authorization and after the authorization displays the confirmation page to the session opened by the Merchant Server. The Merchant Server passes the page unchanged to the user browser as a response page to the post in step 1.

Summary

The following table summarizes the end status scenarios of the Modirum MPI. The description of ECI values should be placed into the authorization message in the case of Visa.

MPI provides ECI value only if it receives it from the ACS. If merchant receives the ECI from the MPI, that is the value that should be placed in the authorization message. If the ECI value received from the merchant is empty, the merchant should fill out the value prior to sending the authorization message. The suggested values are listed in Visa ECI column. Since these values are schema and Visa region specific, please verify the rules with your local Visa representative. MasterCard uses only ECI values 0,1,2 etc.Especially U cases (mdStatus=5) check schema published rules, what if the best action, default recommended to not make authorization.

Explanation	**Log**	**mdStatus**	**Eci returned**	**Visa ECI**	**Other notes**
Authenticated					
Fully authenticated transaction	green	1	From PARes/ ARes/PReq	Use MPI value	CAVV provided
Non-participation					
Issuer or cardholder not enrolled	green	2	empty	6	VERes with N (3DS1 only)
Not in cache	green	3	empty	6	Deprecated functionality. Not used.
Authentication Attempt					

Attempt receipt received and signature valid	green	4	From PARes/ ARes/PReq	Use MPI value	CAVV provided, PARes/ARes/RReq with A
Authentication Unavailable					
Authentication unavailable	red	5	empty	7	VERes or PARes or ARes or RReq status normally “U”
3-D Secure Error	red	6	empty	7	Invalid field in 3-D Secure message generation, Error message received or directory server fails to validate the merchant
Network error	red	91	empty	7	If connection to directory times out
Directory error	red	92	empty	7	Directory response read timeout or other failure
Configuration errors	red	93	empty	7	Service disabled, invalid configuration directory deleted before processing ended etc
Input error	red	94	empty	7	Merchant request had errors
Error no directory	red	95	empty	7	No directory server found configured for PAN/card type
Error no version 2 directory	red	96	empty	7	No version2 directory server found configured for PAN/card type and flow requires version 2 processing (eg APP)
System error	red	99	empty	7	
Authentication Not Attempted					
Fraud Score blocked	red	8	empty		No Visa or MasterCard processing
Authentication Failed					
Authentication failed	red	0	empty	Not allowed	PARes status normally “N”, ,ARes,RReq in N, R
Pares Signature not valid	red	0	empty	Not allowed	
Pending					
Pending transaction	yellow	9			PARes or RReq yet not received
Perform 3DS method and continue to authentication		50			XML API Only: Perform 3DS method in user browser and sent referencing continue enrollment request after. (fill xid txid in from resp)
Skipped Low risk		88			If MPI with FSS ans score below configured value then authentication skipped. (merchant own risk)
Skipped on request		81			if MPI setting TDS2.doChallenge.02=false and requestor challenge ind=02 and challenge requested by issuer. (merchant own risk)

Table 7: Status descriptions

One example of the merchant implementation is as follows:

- Process transactions normally for mdStatus 1, 2(3DS1 only) and 4
- Process transactions for mdStatus 5,6, and 9x, if risk manageable
- Never shall process transactions for mdStatus 0 and 8

3 XML Interface

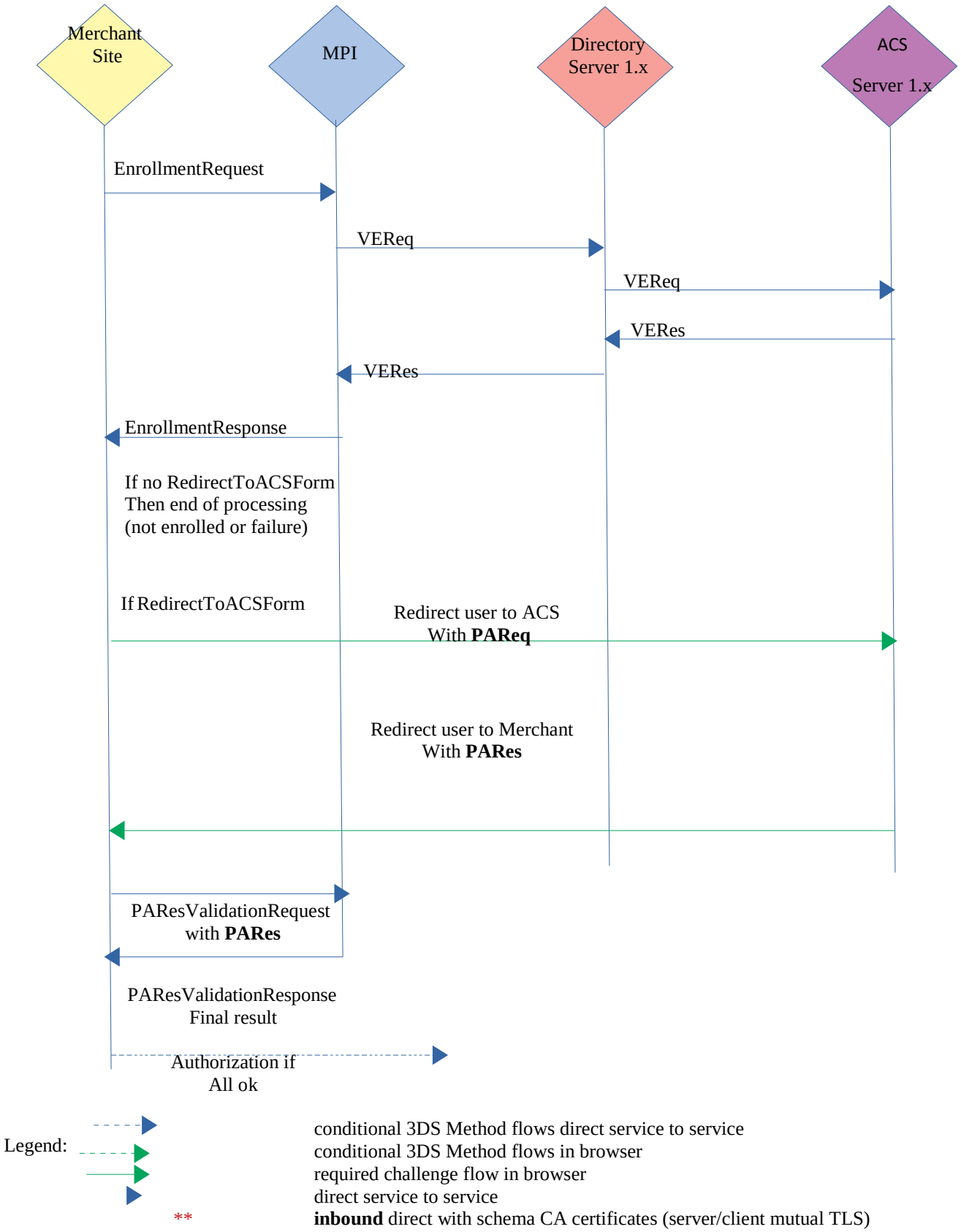
Another way to add 3-D Secure support to existing Merchant site is to use XML interface of the Modirum MPI.

3.1 3D Secure protocol 1.0.2 control flow

The control flow using XML interface is the following ThreeDSecure 1.0:

1. Merchant payment page asks the user all the relevant payment data, such as card number, expiry, etc.
2. Merchant payment page calculates digest of Message element (interface v3 or older) to be posted to MDpay MPI utilizing the shared secret or the digital signature (in case of interface v4), sets the digest/signature to ModirumMPI message and posts the ModirumMPI XML to Modirum MPI with Content-Type=application/xml. Note that Merchant should set termUrl parameter in the request message. It should represent Merchant page url, where ACS will post PARes message after user authentication.
3. Modirum MPI checks the card participation from the 3-D Secure directory and if card is enrolled returns the ACS redirection page to Merchant.
4. Merchant redirects user browser using redirection template. After the ACS is finished with the user authentication, ACS returns the control to the Merchant. Merchant receives PARes and sends PAResValidationRequest to the MPI. Message digest should also be uncluded.
5. The MPI Server verifies the signature of PARes and responds to Merchant.
6. Based on returned parameters Merchant proceeds with transaction authorization or stops the transaction in case of error or authentication failure.

Diagram to illustrate 3DS1 XML API flow. Blue lines server to server communications, green lines user browser redirects:

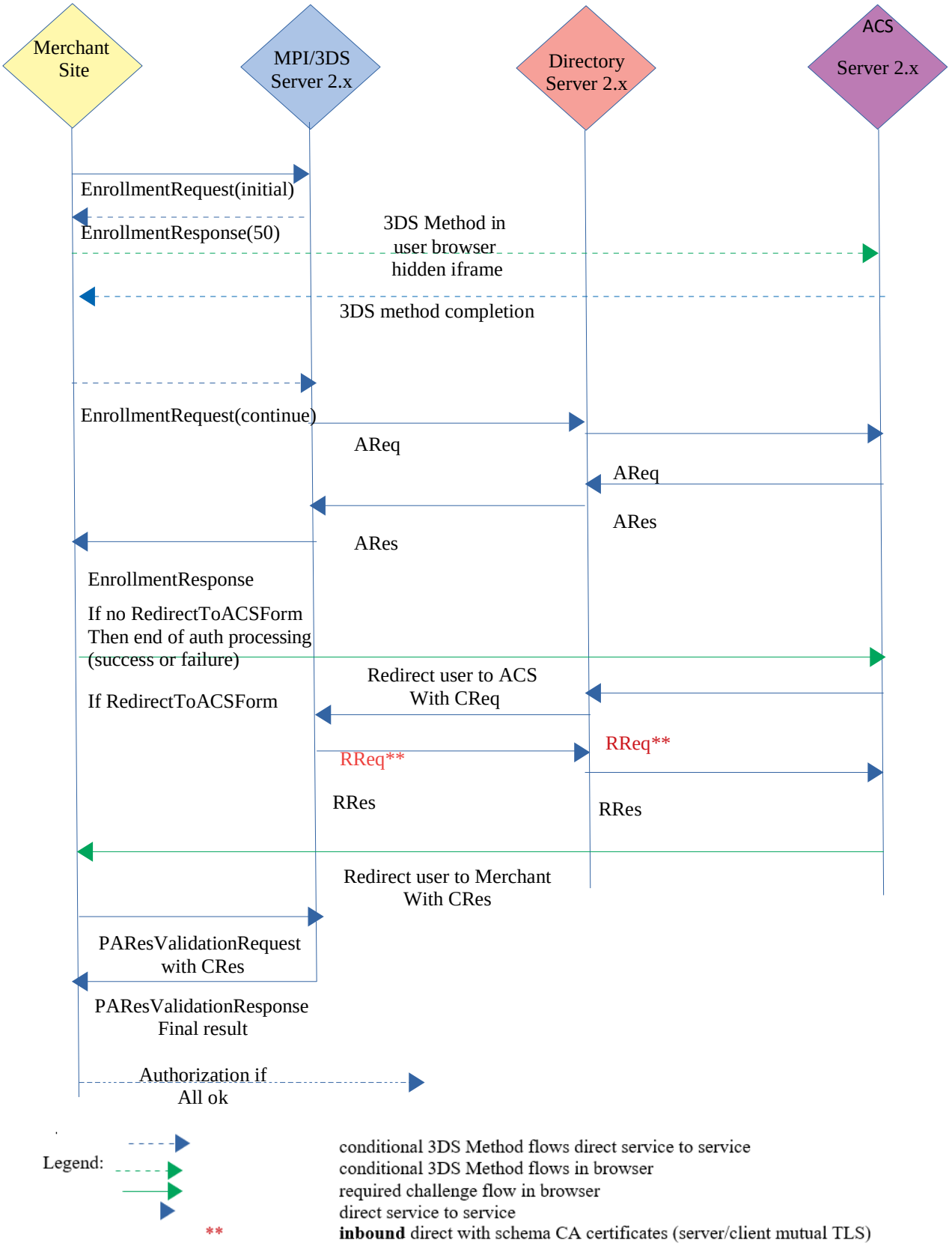


3.2 EMVCo 3DS 2.x protocol control flow

Browser based control flow:

1. Merchant payment page asks the user all the relevant payment data, such as card number, expiry, etc.
2. Merchant payment page creates **EnrollementRequest (initial)** calculates signature of Message element to be posted to Modirum MPI, sets the signature to ModirumMPI message and posts the ModirumMPI XML to Modirum MPI with Content-Type=application/xml. Note that Merchant should set **termUrl** parameter in the request message. It should represent Merchant page url, where ACS will post (via user browser) CRes message after user authentication.
3. Modirum MPI checks the card participation from the 3-D Secure directory published ranges and then:
    a. If range participates in 3DS Method merchant is returned EnrollmentResponse with **mdStatus=50** (perform 3DS method and continue with enrollment request). Upon receiving such response merchant shall take html content from field TDSMethodContent and render user an intermediate page where this content is placed into an invisible iframe and wait about 2..3 seconds (or until 3DSMethodcompletion but normally not longer than 5 seconds (Note: EMVCo spec says up to 10 seconds). The merchant supplied threeDSMethodNotificationURL is notified.
    b. If merchant used TDS2.threeDSMethodNotificationURL in initial request then send new EnrollmentRequest (continue) filling in **only txId, xid, transientData (cloud mode only)** and TDS2.threeDSCompInd attribute if merchant used TDS2.threeDSMethodNotificationURL on its own.
    c. If range does not participate in 3DS Method then continues directly to step 4
4. MPI returns
    a. If no errors and challenge was requested then response message contains field RedirectToACSForm filled with content. It has to be sent to user browser so that user browser posts CReq to ACS for authentication.
    b. With error or final (frictionless) authentication results, **3DS flow ends**. Based on results merchant determines if to proceed authorization or not.
5. After ACS is finished with the user authentication, ACS returns the control to the Merchant. Merchant receives CRes to **termUrl** (set in initial request) and sends PAResValidationRequest to the MPI with CRes field filled with CRes content received. The MPI Server verifies the CRes and matches it with appropriate RReq from directory and responds to Merchant with final outcome.
6. Based on returned parameters Merchant proceeds with transaction authorization or stops the transaction in case of error or authentication failure.

Diagram to illustrate 3DS2 XML API browser flow. Blue lines server to server communications, green lines user browser redirects. Dashed lines when prior authentication 3DS method is required by ACS.



3.3 XML Parameters to be posted to MPI/3DS Server

The following table describes the XML parameters to be posted from the payment page to Modirum MPI. Interface version has been upgraded to 4.0 which utilizes a signature element instead of a digest. The software still supports 3.0, 2.0, but it is not recommended to use versions before 3.0.

Parameters	Type	Required / Optional/Conditional	Description
ModirumMPI	Root element	R	
Message	element	R	
version	attribute, xsi:string	R	Version **4.0** of Modirum MPI protocol.
messageId	attribute, xsi:string	R	Unique alphanumeric message identifier for debugging purposes
md	attribute, xsi:string	R	The MD ("Merchant Data") field: merchant state data that must be returned to the merchant. The content of this field is passed unchanged and without assumptions about its content to the return Message. This field is used to accommodate the different ways merchant systems handle session state. If the merchant system does not maintain state for a given shopping session, the MD can carry whatever data the merchant needs to continue the session. This field must contain only ASCII characters in the range 0x20 to 0x7E; **Since version 4.0.2.15 characters '<' and '>' are not allowed in this parameter.** If other data is needed, the field must be Base64 encoded. The size of the field (after Base64 encoding, if applicable) is limited to 254 bytes. If MD includes confidential data (such as the PAN), it must be encrypted.
merchantId	attribute, xsi:string	R	ID to identify the merchant to Modirum MPI.
lang	attribute, xsi:string	O	Message attribute to specify context language (ISO 639-1 language code en, fi, sv, el, etc..)
RequestMessage	element	R/O	either RequestMessage or ResponseMessage should be present
EnrollmentRequest	element	R/O	either EnrollmentRequest or PAResValidationRequest should be present
Parameters	element	R	
cardType	xsi:string	O	Card scheme can be defined explicitly. Values with 1..2 digits are valid. The value is matched to directory server entry that is mapped to the merchant with that particular cardType The field can also be left empty, which means that the schema/directory server is determined from the card number (see Directory profile settings, cardtypes.xml for card type mappings, and merchant settings in mpimngr to map merchants with particular DS/Acquirer based on card type).
pan	xsi:string	R/-	Card number. 13-19 digit account number. The value may be: • the account number on the card Must be missing if cardEncData field is present

expiry	xsi:string	R/-	Expiration Date supplied to the merchant by cardholder (YYMM). Note that this field is not checked in most of the production ACS installations. Must be missing if cardEncData field is present
cardEncData	xsi:string	C	A base64 encoded value, from VPOS supported Client-side encryption library. If this field is present pan and expiry must be missing.
deviceCategory	xsi:string	O	Integer length 1, Indicates the type of cardholder device. Supported values are:0 = www, 1 = mobile, 4 dtv. Most cases this value shall be allways 0 as wap phones are no longer mainstream and may not be supported by ACSes. Default value is 0
purchAmount	xsi:string	R	Max. 12-digit numeric amount in minor units of currency with all punctuation removed. Examples: Display Amount USD 123.45 Purchase Amount 12345
exponent	xsi:string	R	Exponent number length 1 The minor units of currency specified in ISO 4217. For example, US Dollars has a value of 2; Japanese Yen has a value of 0. Provided
description	xsi:string	O	Purchase description. Brief description of items purchased, determined by the merchant. Maximum size is 125 characters, but merchant should consider the characteristics of the cardholder's device when creating the field. Note: Most Issuers do not display the description to the user.
currency	xsi:string	R	Currency Determined by merchant. ISO 4217, 3 digit numeric.
merchantName	xsi:string	O	Length: 1-25 characters Format: any characters Send this parameter if you want to override PAReq.Merchant.name. By default it is taken from mpi database
xid	xsi:string	R	Unique transaction identifier determined by merchant. Contains a 20 byte statistically unique value that has been Base64 encoded, giving a 28-byte result.
recurFreq	xsi:string	O	Recurring frequency for PAReq Purchase.Recur.frequency (integer days, 28 means monthly)
recurEnd	xsi:string	O/R	Recurring end date for PAReq format YYYYMMDD, If recurFreq is present then recurEnd is required
installments	xsi:string	O	Number of Installments for PAReq.Purchase.install integer value >1 and <=999. I**nstall and recurring parameters can not be present at the same time.**
termUrl	xsi:string	R	URL of the merchant that MPI will insert as termUrl in ACS redirection template (also refferred as notificationURL in 3DS2). As a result after cardholder authentication ACS will return PARes/CRes to Merchant instead of MPI. **Warning: some schemes may reject that URL if contains parameters (?...)**

panMode	xsi:string	O	Possible values: **VPOSToken** – to indicate that instead of real pan pan parameter will contain VPOS token.
txId	xsi:long	C	Required if continue from 3DS method
transientData	Xsi:long	C	Required if continue from 3DS method and present in response to perform 3ds method.
TDS2Attributes	element		3DS 2 related attributes.
			3DS 2.x Merchant Risk fields TDS2.mriShipIndicator TDS2.mriDeliveryTimeframe TDS2.mriDeliveryEmailAddress TDS2.mriReorderItemsInd TDS2.mriPreOrderPurchaseInd TDS2.mriPreOrderDate TDS2.mriGiftCardAmount TDS2.mriGiftCardCurr TDS2.mriGiftCardCount 3DS 2.x extra account info fields TDS2.chAccAgeInd TDS2.chAccDate TDS2.chAccChangeInd TDS2.chAccChange TDS2.chAccPwChangeInd
		R R R O O R R	TDS2.chAccPwChange TDS2.nbPurchaseAccount TDS2.provisionAttemptsDay TDS2.txnActivityDay TDS2.txnActivityYear TDS2.shipAddressUsageInd TDS2.shipAddressUsage TDS2.shipNameIndicator TDS2.paymentAccInd TDS2.paymentAccAge TDS2.suspiciousAccActivity 3DS 2.x extra billing address fields TDS2.billAddrCity TDS2.billAddrCountry TDS2.billAddrLine1 TDS2.billAddrLine2 TDS2.billAddrLine3 TDS2.billAddrPostCode TDS2.billAddrState 3DS 2.x extra shipping address fields TDS2.shipAddrCity TDS2.shipAddrCountry TDS2.shipAddrLine1 TDS2.shipAddrLine2 TDS2.shipAddrLine3 TDS2.shipAddrPostCode TDS2.shipAddrState 3DS 2.x extra authentication info Information about how the 3DS Requestor authenticated he cardholder before or during the transaction. TDS2.AIAuthMethod TDS2.AIAuthTimestamp TDS2.AIAuthData 3DS 2.x extra authentication info Information about how the 3DS Requestor authenticated the cardholder as part of a previous 3DS transaction. TDS2.PAIRef TDS2.PAIAuthMethod TDS2.PAIAuthTimestamp TDS2.PAIAuthData See formats in redirection section or emvco spec 3DS 2.x other TDS2.threeDSMethodNotificationURL – optional send only if You want to detect 3DS method completion on Your pages, else set by mpi. Request to url shall be capable update to TDS2.threeDSCompInd to Y for transaction in question. Since 4.0.2.54 TDS2.threeDSCompInd – **Required in continue to enrollment if** TDS2.threeDSMethodNotificationURL was set by merchant in initial request, else determined by MPI. Since 4.0.2.54
			Service Options
			SEOPT.redirectToACSFormat , optional values HTML or DATA if missing then both

PAResValidationRequest	element	O/R	either EnrollmentRequest or PAResValidationRequest should be present. This element is sent to MPI during pares/cres/acpres validation step.
pares	xsi:string	O/R	pares message, that is returned from ACS should be passed to this parameter as is. Either pares or c64s should be present. **3DS1**
c64s	xsi:string	O/R	c64s message, that can be returned from ACS should be passed to this parameter as is. Either pares or c64s should be present **3DS1**
cres	Xsi:string	O/R	CRes base64 url encoded in case of **3DS2.**
ds:Signature	ds:SignatureType	O	The xml signature as defined https://www.w3.org/TR/xmldsig-core/ Canonicalization http://www.w3.org/TR/2001/REC-xml-c14n-20010315 SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" Requests are signed by merchant private key and validated with merchant Certificate **If missing or mismatching, error is displayed and the request is not further processed.** **Requests are signed by merchant private key and validated with merchant Certificate (merchant certificate generation is referred to section 2.2) Responses are signed by processor private key and validated with Processor certificate (processor certificate is referred to Section 5. page 51)**

Table 8: XML request parameters

3.4 Calculation of the Signature

Signatures shall be calculated and verified according to documentation https://www.w3.org/TR/xmldsig-core/ Canonicalization method to be used is http://www.w3.org/TR/2001/REC-xml-c14n-20010315 SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"

DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"

The signed element is Message element referenced with its ID attribute named messageId. ID attribute is an attribute which type in schema is defined as xsd:ID.

Messages sent by the merchant will be signed by the merchant's private key and verified by the MPI using the merchant's certificate.

Note that the XML documents should be handled with namespace aware xml libraries (parser/serializer). When the Message element is serialized and canonicalized it should also contain xmlns namespace attribute. See from next section XML message with digest example. Canonicalized message starts for example as: <Message xmlns="http://www.modirum.com/schemas/mpiapi" md="C62C2F592F1BCA32A24EA877CFF7A461" merchantId="0000001" messageId="M1523010167533" version="4.0">...

Example code for java:

```
  public static byte[] sign(ModirumMPI root, Marshaller m, DocumentBuilderFactory dbf, PrivateKey
prik, java.security.cert.X509Certificate[] crts) throws Exception
 {
  org.apache.xml.security.Init.init();
  org.w3c.dom.Document document = dbf.newDocumentBuilder().newDocument();
  m.marshal(root, document);// apis.normalizeDOM(dom); dom nomralization is very slow uing instead
msg.setIdAttribute("messageId", true);
```

```
   Element modirumMPI = document.getDocumentElement();
   org.apache.xml.security.signature.XMLSignature xmlsigAp =new XMLSignature(document,
null,"http://www.w3.org/2001/04/xmldsig-more#rsa-sha256", "http://www.w3.org/TR/2001/REC-xml-c14n-
20010315");
   Element sigel = xmlsigAp.getElement();
   modirumMPI.appendChild(sigel);
   Element msg = (Element)modirumMPI.getFirstChild();
   msg.setIdAttribute("messageId", true); // setting id attribute instead of dom normalization
   xmlsigAp.addDocument("#" + msg.getAttribute("messageId"),
   null, "http://www.w3.org/2001/04/xmlenc#sha256", null, null);
   for(int i = 0; crts != null && i < crts.length; i++)
   {
    xmlsigAp.addKeyInfo(crts[i]);
   }
   xmlsigAp.sign(prik);
   ByteArrayOutputStream bos = new ByteArrayOutputStream(4096);
   TransformerFactory transfac = TransformerFactory.newInstance();
   Transformer trans = transfac.newTransformer();
   trans.setOutputProperty(OutputKeys.OMIT_XML_DECLARATION, "no");
   trans.setOutputProperty(OutputKeys.INDENT, "no");
   trans.setOutputProperty(OutputKeys.ENCODING, "utf-8");
   DOMSource source = new DOMSource(document);
   trans.transform(source, new StreamResult(bos));
   return bos.toByteArray();
  }
```

Notes: Never calculate signature in user browser using JavaScript or other script. Or You will expose Your private key to the world. Before passing data to SHA-256 RSA signature function ensure string data is converted to bytes using UTF- 8 character encoding. Signatures should be calculated and checked during any communication with MPI be it request or response.

3.5 XML parameters to be returned to Merchant

Parameters	Type	Required / Optional	Description
ModirumMPI	element	R	
ds:Signature	ds:SignatureType xmlns:ds="http://www.w3.org/2000/09/xmldsig#"	O	The xml signature as defined https://www.w3.org/TR/xmldsig-core/ Canonicalization http://www.w3.org/TR/2001/REC-xml-c14n-20010315 SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
Message	element	R	
version	attribute, xsi:string	R	Version **4.0** of Modirum MPI protocol.
messageId	attribute, xsi:string	R	Unique alphanumeric message identifier for debugging purposes

md	attribute, xsi:string	R	The MD (“Merchant Data”) field: merchant state data that must be returned to the merchant. The content of this field is passed unchanged and without assumptions about its content to the return Message. This field is used to accommodate the different ways merchant systems handle session state. If the merchant system can associate the final post with the original shopping session without any further assistance, the MD field may be empty. If the merchant system does not maintain state for a given shopping session, the MD can carry whatever data the merchant needs to continue the session. This field must contain only ASCII characters in the range 0x20 to 0x7E; **Since version 4.0.2.15 characters '<' and '>' are not allowed in this parameter.** If other data is needed, the field must be Base64 encoded. The size of the field (after Base64 encoding, if applicable) is limited to 254 bytes. If MD includes confidential data (such as the PAN), it must be encrypted.
merchantId	attribute, xsi:string	R	ID to identify the merchant to Modirum MPI.
lang	attribute, xsi:string	O	Message attribute to specify context language (ISO 639-1 language code en, fi, sv, el, etc..)
ResponseMessage	element	O/R	either RequestMessage or ResponseMessage should be present
Parameters	element	R	
xid	xsi:string	R	xid parameter from request message
mdStatus	xsi:string	R	End status of the transaction mdStatus field provides all the information that is needed to determine how to manage the transaction in the merchant system. See section 2.6 Modirum MPI Transaction mdStatus
mdErrorMsg	xsi:string	R	Up to 128 bytes alphanumeric description of the error.
enrollmenStatus	xsi:string	O	The actual value of veres enrollement status like "Y", "N", "U" or "-" if value is not available due errors etc. (since 4.0.2.31). In case of 3DS 2 this is set to Y if connection to directory is established.
authenticationStatus	xsi:string	O	The actual value of PARes tx status or ARes/RReq transStatus "Y", "N", "U", "A","R" or "-" if value is not available due errors or not enrolled. (since 4.0.2.31)
eci	xsi:string	O	Electronic Commerce Indicator With Visa cards, the value to be passed in Authorization Message (exactly 2 decimal digits). ECI fields determine the final status of the transaction See section 2.6 Modirum MPI Transaction mdStatus
cavv	xsi:string	O	Cardholder Authentication Verification Value Determined by ACS. Contains a 20 byte value that has been Base64 encoded, giving a 28 byte result. Some Visa regions may require that this value be included in the VIP authorization message.
cavvAlgorithm	xsi:string	O	A positive integer indicating the algorithm used to generate the Cardholder Authentication Verification Value. Current defined values are: 0 = HMAC (as per SET™ TransStain) 1 = CVV 2 = CVV with ATN 3 = MasterCard AAV Note that the value is copied directly from the PARes message.
PAResVerified	xsi:string	O	If signature validation of the return message is successful, the value is true. If PARes message is not received or signature validation fails, the value is false. (Compliance testing facility requirement). If signature validation is omitted the value will be empty

PAResSyntaxOK	xsi:string	O	If PARes validation is syntactically correct, the value is true. Otherwise value is false. (Compliance testing facility requirement)
iReqCode	xsi:string	O	Two digit numeric code provided by ACS indicating data that is formatted correctly, but which invalidates the request. This element is included when business processing cannot be performed for some reason. Never provided if mdStatus=0. 3-D Secure iReqCode field.
iReqDetail	xsi:string	O	May identify the specific data elements that caused the Invalid Request Code (so never supplied if Invalid Request Code is omitted). See Table 20 on page 60 for details.
vendorCode	xsi:string	O	Error message describing iReqDetail error.
txId	xsi:long	O	MPI internal transaction id if available
transientData	xsi:string	O	Returned only if cloud mode and mdStatus=50
protocol	xsi:string	O	Authentication protocol used in processing. Such as 3DS1.0.2, 3DS2.1.0 or SP5 (Since 4.0.2.54)
cardType	xsi:string	O	Same value as in request or value resolved by MPI. (Since 4.0.2.54)
fssScore	xsi:int	O	Fraud score calculated by the Modirum FSS. Present if the MPI is configured for use with the FSS, if setting fss.includeScoreInResponse has been set to true and if the score is available.
redirectToACSForm	xsi:string	O	ACS redirection form raw HTML, containing all the required parameters to be posted to ACS or SecurePlus. PAReq, termUrl, MD and ACS url itself. Form should be presented to user as is. MPI does not set this parameter in response to PAResValidationRequest step.
redirectToACSFormData	Element	O	ACS redirection form data in elements, containing all the required parameters to be posted to ACS or SecurePlus. PAReq, termUrl, MD and ACS url itself.
TDSMethodContent	Element	O	A complete html fragment that need to be rendered in user browser if 3DS method requested for a card (in initial enrollement response
TDS2RespAttributes	element		3DS 2 related response attributes.
Attribute	Element [0..n]	O	<Attribute name=”aname”>Value</Attribute> Valid attribute names provided if available and applicable basis: TDS2.transStatus TDS2.transStatusReason TDS2.threeDSServerTransID TDS2.dsTransID TDS2.acsTransID TDS2.acsRenderingType (only App) TDS2.acsReferenceNumber TDS2.acsSignedContent (only App) TDS2.authTimestamp TDS2.messageVersion TDS2.acsChallengeMandated TDS2.authenticationType TDS2.acsOperatorID TDS2.cardholderInfo TDS2.acsUrl TDS2.challengeCancel **(since 4.0.2.56)** TDS2.ARresExtensions (in Ares, json as is) TDS2.RReqExtensions (in Rreq, json as is) TDS2.AReqToResMillis (**since 4.0.2.56** time measurement of before sending AReq to Response received)

Table 9: XML response parameters

3.6 XML interface example for 3D Secure protocol 1.0.2

3.6.1 Initial request to MPI

```
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi"
xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
<Message md="k5C0IuwfsDK9yubdQYJnh/ck8GHJYI7pEZbSVkk1MoA=" merchantId="0000001"
messageId="MPIER1222501-0" timeStamp="2017-05-17T09:47:58.997+03:00" version="4.0">
  <Request>
    <EnrollmentRequest>
      <Parameters>
        <cardType>1</cardType>
        <pan>##############</pan>
        <expiry>1801</expiry>
        <deviceCategory>0</deviceCategory>
        <purchAmount>000000000112</purchAmount>
        <exponent>2</exponent>
        <description>Order O170517094737</description>
        <currency>978</currency>
        <termUrl>https://vposadmin.modirum.com/vpos/PaymentHandler?acsResult=k5C0IuwfsDK9yubdQYJnh
        %2Fck8GHJYI7pEZbSVkk1Mo A%3D</termUrl>
        <TDS2Attributes>
          <Attribute name="TDS2_BrowserIP">88.196.25.166</Attribute>
          <Attribute name="TDS2_Navigator_language">en-US</Attribute>
          <Attribute name="TDS2_Navigator_javaEnabled">false</Attribute>
          <Attribute name="TDS2_Navigator_jsEnabled">true</Attribute>
          <Attribute name="TDS2_Screen_colorDepth">24</Attribute>
          <Attribute name="TDS2_Screen_height">1200</Attribute>
          <Attribute name="TDS2_Screen_width">1920</Attribute>
          <Attribute name="TDS2_TimezoneOffset">-180</Attribute>
          <Attribute name="TDS2_UserAgent">Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36
          (KHTML, like Gecko) Chrome/58.0.3029.96 Safari/537.36 </Attribute>
        </TDS2Attributes>
        <xid>MjUwMS1PMTcwNTE3MDk0NzM3LTA=</xid>
      </Parameters>
    </EnrollmentRequest>
  </Request>
</Message>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
    <ds:Reference URI="#MPIER1222501-0">
      <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
      <ds:DigestValue>g6ckv8JZ8lGbpEuEDLGbPntKLBjzP+QcyjEMwTy9EmY=</ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>Ef4cfgN+vvfSt/8ewDdYirqYsyVuGM/ooZCJIpKFYf/wRA1/vXKvJRmfjNgHxMDYitCeNquS
    X3j5 TNk5cO7ZxGOQS9IJKPLORCzTIHXWBD0MH16UZnbU08mnfhR+lJzd6kX6qnel+eVrypUaYMviQutG
    mzKAm31AuC4DUEqA9PKRQXdXvGsW5I76sJcTawpNZh7oZf8aK/D1N+FQOUCC+KmtNl5VdgSwuQj9
    Frtz2oGG9nlc/xFtZssJ2gg5/1wLFh4kQkFP3Pg/v5dSOCDzv3PjySS3rR3kKIopU7t7eyQ6muC3
    cjQvJyUZW81u4HEEONWxBzNmTkWajOv4EnQjw6OkcMIvi5UGoQkZcgBAhI0BeyEz9hBgDfFXuSS6
    GSMM2WHbBQh1utnUJEZLfXkPpQvZ3pzzThnYj5v0/hP9aHc=</ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>
MIID5TCCAo0CBFjeXq8wDQYJKoZIhvcNAQELBQAwdzEoMCYGA1UEAxMfVlBPUyBERU1PIHZwb3Nh
...K1tddzVPdH+QK8q3EKBNt0H3KwbRPk9qRmH4xuoX4XA=
      </ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
</ModirumMPI>
```

3.6.2 MPI response with ACS redirect template

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
<Message md="k5C0IuwfsDK9yubdQYJnh/ck8GHJYI7pEZbSVkk1MoA=" merchantId="0000001"
messageId="MPIER1222501-0" version="4.0">
  <Response>
    <Parameters>
      <xid>MjUwMS1PMTcwNTE3MDk0NzM3LTA=</xid>
      <mdStatus>9</mdStatus>
      <mdErrorMsg>To be redirected to acs</mdErrorMsg>
      <enrollmenStatus>Y</enrollmenStatus>
    </Parameters>
    <redirectToACSForm>&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd"&gt;
      &lt;html&gt;
      &lt;head&gt;
      &lt;title&gt;MDpay default response template for web&lt;/title&gt;
      &lt;script type="text/javascript"
      language="javascript"&gt;
      function moveWindow() {
      var form = document.getElementById("downloadForm");
      if (form != null) {
      form.submit();
      }
      }
      &lt;/script&gt;
      &lt;/head&gt;
      &lt;body&gt;
      &lt;form id="downloadForm"
      action="https://3ds-acs.test.modirum.com/mdpayacs/pareq"
      method="POST"&gt;
      &lt;input type="hidden"
      name="PaReq"
value="eJxVUttuwjAM/ZWKp+2lSXrLikwkBkhDWgFx+YAqNWu39ULarrCvX1LK2PLkc07s2MeBfaoQ5zuUrUIBEdZ1/I
ZWlkxGYeBTXo0EbKZbPAn4QlVnZSGYTW0HyA3qHCXTuGgExPL0vFwJrz9ABgg5quVcMMf1/ADIFUER5yhm5fGIm
JZtjdZDgnn5CKQXQJZt0aiLcFwXyA1Aqz5F2jTVmJCu6+y8TDLV5rYscyBGA3JvZtOaqNa1zlkiovdDF+3YJtrLbrVfuNH8
g66+I/d1P50AMTcgiRsUDmWc+oxbNBh7fOyHQHoe4tw0IRaHrcVspucfCKjMO9MrYEb4S4C2VWEhLyLkT3qQGwI8V2
WB+oZO+I0hwVqKtUpQWetrHzT0uMt1E0YBch9q9mL8lo320nG5EzicGcd7wlTPtF/sibK+vAFATAoZlkmGTevo3w/4AW
3urPk="&gt;
      &lt;input type="hidden"
      name="TermUrl"
value="https://vposadmin.modirum.com/vpos/PaymentHandler?acsResult=k5C0IuwfsDK9yubdQYJnh%2Fck8GHJYI7pEZb
SVkk1MoA%3D"&gt;
      &lt;input
      type="hidden"
      name="MD"
      value="k5C0IuwfsDK9yubdQYJnh/ck8GHJYI7pEZbSVkk1MoA="&gt;
      &lt;!-- To support javascript unaware/disabled browsers
      --&gt;
      &lt;div style="text-align: center;"&gt;
      &lt;img src="preloader.gif"/&gt;&lt;br/&gt;
      &lt;noscript&gt;
      &lt;center&gt;Please click the
      submit button below.&lt;br&gt;
      &lt;input type="submit" name="submit" value="Submit"&gt;&lt;/center&gt;
      &lt;/noscript&gt;
      &lt;/div&gt;
      &lt;/form&gt;

      &lt;script type="text/javascript"&gt;
      moveWindow();
```

```
        &lt;/script&gt;
        &lt;/body&gt;
        &lt;/html&gt;</redirectToACSForm>
    </Response>
</Message>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
      <ds:Reference URI="#MPIER1222501-0">
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
        <ds:DigestValue>gNJwiQO3qpKAHybkru+31B1Y8V9haD8Q0mfFwu2eo/k=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>
      PzQiFtORzYRqB9wk0PBe2Ki0tvg9R2S5YVK95BKsK9hdRrUc+uNGXbvcEY+POtcTAj8zQPkf8Ztn
      rUB81kWsNhBvLSytfZslmxYXS/XM7gvYOPINa0kOcqT0e/F7egq4M7ZhYxFM6SebmlKkXPySATfv
      qZCoaNJh0DUTxB9l/PvAjr1UISv5K2xmm9XqZd75oE6Zf+QFFWppNhaDBGUiFHw9iG1VqwFw9nqI
      sfw8kIpu8kIhqp4zu3oibXaLgVcqcpXNf8PQixFafuTHGhzbb/R3hd2sTUMSrt/M5PQmhufKi++O
      FIV34mRBmkit5KCrovEE/ACO9vBgG6w5RJUabzLJkhkOv5TZwufzxX12OxWVcXQRKcwzsNNBzhaz
      gPG7tMPkfBhIDXdeQPliKghGvRqpCWuPX9mqc1lbHoucdgM=
    </ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
MIID5TCCAo0CBFjeXq8wDQYJKoZIhvcNAQELBQAwdzEoMCYGA1UEAxMfVlBPUyBERU1PIHZwb3Nh
...
K1tddzVPdH+QK8q3EKBNt0H3KwbRPk9qRmH4xuoX4XA=
        </ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
</ds:Signature>
</ModirumMPI>
```

3.6.3 PARes validation request to MPI

```
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
<Message md="StndCjQrZpcH8XTvAmFrGX2bm9wZrmYszyiym7k1OhA=" merchantId="0000001"
messageId="MPIPVR1222501-0" timeStamp="2017-05-17T09:48:04.825+03:00"
      version="4.0">
  <Request>
    <PAResValidationRequest>
      <pares>

eJzVmFmzoziygP9KRc2jo5vFYEOH60SIHRsw+/bGZsBsZjP

…

TSheWEEAv0Ez9Pd5H/pxB/D37cD7reX7denrsu3na9T/Af5LBqc=</pares>
    </PAResValidationRequest>
  </Request>
</Message>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
    <ds:Reference URI="#MPIPVR1222501-0">
      <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
      <ds:DigestValue>+NGJpXUDc2M3+vG8c8hccR081PSAC5aPzWlYcUvDXec=</ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>
    b/NklbvW/pyCeCu+ZLt5rqpUsClVr3gu4EWbYz935rlZpvX/Oq0NssZD5SiH06jT5WZx5Sq9QL7J
    JCySHsa819q5TCxkKaNgQS2+NUw4q2SEAEBHfkJIZ4dDmeJnH4AuebAryQxa3NVW2Lmz4SZfx8MG
    4o7CSBKqueNsJP1u3SMsQl8YEVOptwC23e1yfiPYip6KBDtIibcpk5r/GCHZJw3maWAYZ5YZXjig
    3cVfGroxWuVCBaXKNquZQgy5/uhwBHpFoKBPqiy/CInkSpm1kTtNdkSCNmOzzqqcW3i3DHF44bJq
    UEbpOtSeLRhOxHOHnfzpFaG1+tEcdEXSj5JcffVpg+6YNsGrfCIE2Ays295s6OPWAGQuekcCcUbM
    zLXIVDBvtXo7vqf09v4NBBjqydnyFwvBuZcNWxk64Pp+qRc=
  </ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>
MIID5TCCAo0CBFjeXq8wDQYJKoZIhvcNAQELBQAwdzEoMCYGA1UEAxMfVlBPUyBERU1PIHZwb3Nh
...
K1tddzVPdH+QK8q3EKBNt0H3KwbRPk9qRmH4xuoX4XA=
      </ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
</ModirumMPI>
```

3.6.4 PARes validation response from MPI

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
<Message md="StndCjQrZpcH8XTvAmFrGX2bm9wZrmYszyiym7k1OhA=" merchantId="0000001"
messageId="MPIPVR1222501-0" version="4.0">
  <Response>
    <Parameters>
      <xid>MjUwMS1PMTcwNTE3MDk0NzM3LTA=</xid>
      <mdStatus>1</mdStatus>
      <mdErrorMsg>Authenticated</mdErrorMsg>
      <enrollmenStatus>Y</enrollmenStatus>
      <authenticationStatus>Y</authenticationStatus>
      <eci>05</eci>
      <cavv>AAABAEVicQAAAAAjcmJxAAAAAAA=</cavv>
      <cavvAlgorithm>2</cavvAlgorithm>
      <PAResVerified>true</PAResVerified>
      <PAResSyntaxOK>true</PAResSyntaxOK>
      <txId>96507</txId>
    </Parameters>
  </Response>
</Message>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
    <ds:Reference URI="#MPIPVR1222501-0">
      <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
      <ds:DigestValue>eXmv0YqTgw3StXqaj/PsvZ6KJAzlb05dzCPPuoieo3c=</ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>
    CdudW1IP7LkYxcl1JQJRt8WwtGgyRC7q+6MuKisOQk/avISLKN9mzu7UsYsrAXQEtx3KwZVVhV/y
    PF4j5qwAHw7+Uib7tfLjO3+vZ8bOgl3I8KVWbXjgsLRbghidqGvMr3ieRO4SKwrDJQ8jQ/7ptjnd
    rmCOwa1hKhSP3FFpwiRUZAVYyGU2Ji32h3VrU4BLp/0lqZF8Kju+P861K36vgeNxt9PTgdyCiTln
    E3yo5kPczop2EvLg92MZ4xBfO995Nv+d5BKCw1kl4GaskEVOG/hRlD5gCS6Ehw4l5yKEwK5Nb2qS
    k/WEMj9A9NQVZv8+MyRxQKTFB/9wPB7yvXP6rRraEvREgTk0VMvZFZPFM/BKsnoN88JBYGQ6UUA0
    py+HAjm1HcRIyyh7BfznkmdWggxEgP0mysscfNYdir02yTM=
  </ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>
MIID5TCCAo0CBFjeXq8wDQYJKoZIhvcNAQELBQAwdzEoMCYGA1UEAxMfVlBPUyBERU1PIHZwb3Nh
...

K1tddzVPdH+QK8q3EKBNt0H3KwbRPk9qRmH4xuoX4XA=
      </ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
</ModirumMPI>
```

3.7 Example XML messages for EMVCo protocol 2.x browser flow

3.7.1 Initial request to MPI

```
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi">
   <Message md="50F2156E03083CA665BCB42774614B6A" merchantId="sys1331720664552"
messageId="M1544784608946"
         version="4.0">
      <Request>
         <EnrollmentRequest>
            <Parameters>
               <pan>4016000000051</pan>
               <expiry>1812</expiry>
               <deviceCategory>0</deviceCategory>
               <purchAmount>1100</purchAmount>
               <exponent>2</exponent>
               <description>DVD Movies</description>
               <currency>840</currency>
               <termUrl>

https://localhost.modirum.com:8543/coffeehouse/MerchantHandler2;jsessionid=50F2156E03083CA665BCB42774614B6A
               </termUrl>
               <xid>EnT7echNxbjyNJSEIQiIsogpnUg=</xid>
            </Parameters>
         </EnrollmentRequest>
      </Request>
   </Message>
   <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
         <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
         <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
         <ds:Reference URI="#M1544784608946">
            <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <ds:DigestValue>25Pkv287ypp/7dFL5OkZB8TpX98adSX4waWKzwt+IIU=</ds:DigestValue>
         </ds:Reference>
      </ds:SignedInfo>
      <ds:SignatureValue>
         8BN83M86tInv1oMdzU4NcHfMxU0PW1x6vrx7pzIocv1i/QK3sdS/kDeP6HnRR1rwPqCp/xeWLqRq
         sFmoDNz46VNKUw6bsFFC5nYFDNcrUAWVaKtVF6xqPK3n5gkm0xdpodZMRXmvBviSEUsnfENepKXL
         fn+jptfLmQSum/afL0eVbSJ1+5Ai12BF167m0H1k4qoOvI9VnfD3GydqUUUEEeVQMS87FbYcVSAX
         7gYAspf3qBAoCEfKy+LJ+/kxkCH3a2Z9Ri6Zut16hu4YQPz8ORrWBfFbImpoaYORmA6tibhOxQ2I
         H5XHO2WNJ7KUWob+944EKeMZRkg3+sVkCqnxhA==
      </ds:SignatureValue>
      <ds:KeyInfo>
         <ds:X509Data>
            <ds:X509Certificate>
               MIIESzCCAzOgAwIBAgIJANq6Drpt8SB5MA0GCSqGSIb3DQEBCwUAMHYxCzAJBgNVBAYTAkVFMREw
               ...
               RLXZUNmvy+zFSB+QFWEW2nlfYI8=
            </ds:X509Certificate>
         </ds:X509Data>
      </ds:KeyInfo>
   </ds:Signature>
</ModirumMPI>
```

3.7.2 Response from MPI with 3DS Method redirection form

```
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi">
   <Message md="50F2156E03083CA665BCB42774614B6A" merchantId="sys1331720664552" messageId="M1544784608946"
         version="4.0">
      <Response>
         <Parameters>
            <xid>EnT7echNxbjyNJSEIQiIsogpnUg=</xid>
            <mdStatus>50</mdStatus>
            <mdErrorMsg>3DS method requested before enrollment</mdErrorMsg>
            <enrollmenStatus>-</enrollmenStatus>
            <authenticationStatus>-</authenticationStatus>
            <txId>16809</txId>
            <protocol>3DS2.1.0</protocol>
            <cardType>1</cardType>
         </Parameters>
         <TDSMethodContent>&lt;!DOCTYPE iframe SYSTEM "about:legacy-compat"&gt;
            &lt;iframe id="tdsMmethodTgtFrame" name="tdsMmethodTgtFrame" style="width: 1px; height: 1px; display:
            none;" src="javascript:false;" xmlns="http://www.w3.org/1999/xhtml"&gt;
            &lt;!--.--&gt;
            &lt;/iframe&gt;&lt;form id="tdsMmethodForm" name="tdsMmethodForm"
            action="https://localhost.modirum.com:8543/dstests/ACSEmu2" method="post" target="tdsMmethodTgtFrame"
            xmlns="http://www.w3.org/1999/xhtml"&gt;
            &lt;input type="hidden" name="3DSMethodData"

value="eyAidGhyZWVEU1NlcnZlclRyYW5zSUQiIDogIjAwMDAwMDAwLTU2NzYtNTY2My04MDAwLTAwMDAw&amp;#10;MD
AwNDFhOSIsICJ0aHJlZURTTWV0aG9kTm90aWZpY2F0aW9uVVJMIiA6ICJodHRwczovL2xvY2Fs&amp;#10;aG9zdC5tb2RpcnVtLm
NvbTo4NTQzL21kcGF5bXBpL01lcmNoYW50U2VydmVyP21uPVkmdHhpZD0x&amp;#10;NjgwOSZkaWdlc3Q9aSUyQnhhUEF5NWF
OcVJRbllqNmozbWFDZlFJbTdFdjJYTmkwNnh6YmZNJTJG&amp;#10;R3MlM0QiIH0"/&gt;
            &lt;input type="hidden" name="threeDSMethodData"

value="eyAidGhyZWVEU1NlcnZlclRyYW5zSUQiIDogIjAwMDAwMDAwLTU2NzYtNTY2My04MDAwLTAwMDAw&amp;#10;MD
AwNDFhOSIsICJ0aHJlZURTTWV0aG9kTm90aWZpY2F0aW9uVVJMIiA6ICJodHRwczovL2xvY2Fs&amp;#10;aG9zdC5tb2RpcnVtLm
NvbTo4NTQzL21kcGF5bXBpL01lcmNoYW50U2VydmVyP21uPVkmdHhpZD0x&amp;#10;NjgwOSZkaWdlc3Q9aSUyQnhhUEF5NWF
OcVJRbllqNmozbWFDZlFJbTdFdjJYTmkwNnh6YmZNJTJG&amp;#10;R3MlM0QiIH0"/&gt;
            &lt;/form&gt;&lt;script type="text/javascript" xmlns="http://www.w3.org/1999/xhtml"&gt;
            document.getElementById("tdsMmethodForm").submit();
            &lt;/script&gt;
         </TDSMethodContent>
      </Response>
   </Message>
   <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
         <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
         <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
         <ds:Reference URI="#M1544784608946">
            <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <ds:DigestValue>/HJiYwWCim06yW6VTXPPVbicxpXg8jVJaXDXPk5vpMA=</ds:DigestValue>
         </ds:Reference>
      </ds:SignedInfo>
      <ds:SignatureValue>
         mfAI9Z+C+JrVxIakEM6ZHCm+h9cWBwpghDE3MZKY4CeRiWKyb837lSznA3uhj8CYF6JdzUM7Ndrn
         2jNRUxzkyn+FDvj+4q1v4scwFF298nS9/7lt5eU+k4MgUs+nqlBtjQqYBnnicKOtnDuDrICIPjo8
         gxwrebBTDtGIexEKwpybpNgb1FRlwulKUEk03TTHRPAWGBd9/xDG6HsATtKTwb7GQK8e+xd2oK7+
         YGpMS2sm6GB1fb3LWklMGk5cF1xqakCUFsG3FPm2PEqpX0bACvSIKuOtP9zMMHCmjDAA8V1n25/U
         O2rW2lHtB2D68pkSYp0jsVSPORpp1kGG56QhlQ==
      </ds:SignatureValue>
      <ds:KeyInfo>
         <ds:X509Data>
            <ds:X509Certificate>
               MIIDYDCCAkgCAQEwDQYJKoZIhvcNAQELBQAwdjEaMBgGA1UEAxMRcHJvY0ludGVyZmFjZVNpZ24x
               ...
               9PDtopCdwc7k0pzUiQ==
            </ds:X509Certificate>
         </ds:X509Data>
      </ds:KeyInfo>
   </ds:Signature>
</ModirumMPI>
```

3.7.3 Request to MPI after 3DS Method

```
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi">
  <Message merchantId="sys1331720664552" messageId="M1544784619256" version="4.0">
    <Request>
      <EnrollmentRequest>
        <Parameters>
          <txId>16809</txId>
          <xid>EnT7echNxbjyNJSEIQiIsogpnUg=</xid>
        </Parameters>
      </EnrollmentRequest>
    </Request>
  </Message>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
      <ds:Reference URI="#M1544784619256">
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
        <ds:DigestValue>oWYO8KIJaEuo9Bf7k/ORD1bPpxOMMorPmypfivp5SBw=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>
      lexfHtEYbi45Nxh4kcZmwxGaYrgePEU6mIdOqXZ5RVWmWbLi5iRaNfdjI0/FPiTCz7o+aQ6944Zk
      aO4ATJCEGYNjfz+vAONls8IMZA6L+KV+DztDhMocWv7acwp7FiwKwcvdxDsg7Kqc14GT9owjtZXN
      ZGv8Ucxvt17Mm8DB+FBvIiKGdVzwE5i1KsEDbAzkYM63DMjl713BAKf/cFStlir4l+2A4g6HNB+p
      PlT6ME2mL4ZBbTFAcWQsQPN1Go3wk7QvtV8tUXvJ5WCZrzMaAU37fvF8POfbsj8fw01cDIw/2ATR
      Ohl2XkR24M+7pBitN4/Zu3GLCfGIF/u0sKnK+Q==
    </ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
          MIIESzCCAzOgAwIBAgIJANq6Drpt8SB5MA0GCSqGSIb3DQEBCwUAMHYxCzAJBgNVBAYTAkVFMREw
          ...
          RLXZUNmvy+zFSB+QFWEW2nlfYI8=
        </ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
</ModirumMPI>
```

3.7.4 Response from MPI with CReq redirection form to ACS

```
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi">
  <Message md="50F2156E03083CA665BCB42774614B6A" merchantId="sys1331720664552" messageId="M1544784608946"
      version="4.0">
    <Response>
      <Parameters>
        <xid>EnT7echNxbjyNJSEIQiIsogpnUg=</xid>
        <mdStatus>9</mdStatus>
        <mdErrorMsg>To be redirected to acs</mdErrorMsg>
        <enrollmenStatus>Y</enrollmenStatus>
        <protocol>3DS2.1.0</protocol>
        <cardType>1</cardType>
      </Parameters>
      <redirectToACSForm>&lt;!DOCTYPE html SYSTEM "about:legacy-compat"&gt;
        &lt;html class="no-js" lang="en" xmlns="http://www.w3.org/1999/xhtml"&gt;
        &lt;head&gt;
        &lt;meta http-equiv="Content-Type" content="text/html; charset=utf-8"/&gt;
        &lt;meta charset="utf-8"/&gt;
        &lt;title&gt;3D Secure Processing&lt;/title&gt;
        &lt;link href="https://localhost.modirum.com:8543/mdpaympi/mpi.css" rel="stylesheet" type="text/css"/&gt;
        &lt;/head&gt;
        &lt;body&gt;
        &lt;div id="main"&gt;
        &lt;div id="content"&gt;
        &lt;div id="order"&gt;
        &lt;h2&gt;3D Secure Processing&lt;/h2&gt;
        &lt;img src="https://localhost.modirum.com:8543/mdpaympi/preloader.gif" alt="Please wait.."/&gt;
        &lt;div id="formdiv"&gt;
        &lt;script type="text/javascript"&gt;
        function hideAndSubmitTimed(formid)
        {
        var timer=setTimeout("hideAndSubmit('"+formid+"');",10);
        }
        function hideAndSubmit(formid)
        {
        var formx=document.getElementById(formid);
        if (formx!=null)
        {
```

formx.style.visibility="hidden";

formx.submit();

}

}

</script>

<div>

<form id="webform0" name="" method="POST" action="https://localhost.modirum.com:8543/dstests/ACSEmu2"

accept_charset="UTF-8">

<input type="hidden" name="_charset_" value="UTF-8"/>

<input type="hidden" name="creq"

value="ewogICAiYWNzVHJhbnNJRCIgOiAiMDAwMDAwMDAtMDAwNS01YTVhLTgwMDAtMDE2N2FjNTZiNTgx
IiwKICAgImNoYWxsZW5nZVdpbmRvd1NpemUiIDogIjAzIiwKICAgIm1lc3NhZ2VUeXBlIiA6ICJD
UmVxIiwKICAgIm1lc3NhZ2VWZXJzaW9uIiA6ICIyLjEuMCIsCiAgICJ0aHJlZURTU2VydmVyVHJh
bnNJRCIgOiAiMDAwMDAwMDAtNTY3Ni01NjYzLTgwMDAtMDAwMDA0MWE5Igp9"/>

<input type="hidden" name="threeDSSessionData" value="50F2156E03083CA665BCB42774614B6A"/>

<input type="hidden" name="TermUrl"

value="https://localhost.modirum.com:8543/coffeehouse/MerchantHandler2;jsessionid=50F2156E03083CA665BCB42774614B6A"/>

<input type="submit" name="submitBtn" value="Please click here to continue"/>

</form>

</div>

</div>

<script type="text/javascript">

hideAndSubmitTimed('webform0');

</script>

<noscript>

<div align="center">

Javascript is turned off or not supported!

</div>

</noscript>

</div>

<div id="content-footer">

<img height="20" src="https://localhost.modirum.com:8543/mdpaympi/powered-by-modirum.svg"

alt="Powered by Modirum"/>

</div>

</div>

</div>

</body>

</html>

```
        </redirectToACSForm>
      </Response>
    </Message>
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
        <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
        <ds:Reference URI="#M1544784608946">
          <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
          <ds:DigestValue>Lm0bG314K60fFnGXtQ3xMvoYIXfJxxSNLvJ1Y5FxLuY=</ds:DigestValue>
        </ds:Reference>
      </ds:SignedInfo>
      <ds:SignatureValue>
        EpCv425wDg0pT1alRd2n27RIvHituiZqdQCrrCesXeeqTJ1UMprk7wuFvx9iFDzt9fKKjniJ5oND
        Vb4IhYoV9u3DneU62xfnkDy/0E4TfdcA2sw8kQ5lCkcci2Mf55CNtmCILi0jvo92P7LpmHpi2vAb
        j4WJDMOjsLneaPIGSIxdmzHlM2vbRnDboxc3Qlm+Le+hfglhwDd0acQeT7xtfeiRMYCI4r2HIYWl
        64Xxbv1ErnOAh4PO/WNTRuaCxiIAcqvxerBmhz5DUpTzjwwpNiSYKDgp+v0F0UaSyHoSgjDXmuPQ
        CI9slK5Vgh2dcVYbcByVNzxA96OB+wnuLCJVow==
      </ds:SignatureValue>
      <ds:KeyInfo>
        <ds:X509Data>
          <ds:X509Certificate>
            MIIDYDCCAkgCAQEwDQYJKoZIhvcNAQELBQAwdjEaMBgGA1UEAxMRcHJvY0ludGVyZmFjZVNpZ24x
            ...
            9PDtopCdwc7k0pzUiQ==
          </ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </ds:Signature>
</ModirumMPI>
```

3.7.5 Request to MPI with CRes

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi">
  <Message merchantId="sys1331720664552" messageId="M1544784619256" version="4.0">
    <Request>
      <PAResValidationRequest>
        <cres>ewogICAiYWNzUmVmZXJlbmNlTnVtYmVyIiA6ICJBQ1NFbXUyIiwKICAgImFjc1RyYW5zSUQiIDog&#13;
          IjAwMDAwMDAwLTAwMDUtNWE1YS04MDAwLTAxNjdhYzU2YjU4MSIsCiAgICJtZXNzYWdlVHlwZSIg&#13;
          OiAiQ1JlcyIsCiAgICJtZXNzYWdlVmVyc2lvbiIgOiAiMi4xLjAiLAogICAidGhyZWVEU1NlcnZl&#13;
          clRyYW5zSUQiIDogIjAwMDAwMDAwLTU2NzYtNTY2My04MDAwLTAwMDAwMDAwNDFhOSIsCiAgICJ0&#13;
          cmFuc1N0YXR1cyIgOiAiWSIKfQ==
        </cres>
      </PAResValidationRequest>
    </Request>
  </Message>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
      <ds:Reference URI="#M1544784619256">
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
        <ds:DigestValue>8EHq3bJC1zCTZBoY6uvh2VL4//iDmvasHc7EldS350Q=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>
      MH64sMitMnxpyuEi4mAEjP/u6HoVsxCBKhnxjgaUCGmKiuBq5h+yRYsGYCuD8vHlpZsBdFmUfTQN
      ToKM0MaiOENdOvMwnzehYmSEN077rqZOfWJWd09VNNwcG0h5yR9Vej8n/PknL3IXach67KrBB7+e
      6am963x3XVIWhsQCh+youJspnYrhj2TiObgAxWvVUU6WAmxrAFOZDKNYj1Y745TjAztRFIOof8u/
      hx9uwfwUUfwzbQKxMTjL8DoBeBKzP6AW1K5cfjP45SNd31vGhXe9LKqdJrYNxOX8tCk7/MkxypYl
      yi6Eg1n7CNZ5+7ZKmXZPJ1yns915g5up9EoBpQ==
    </ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
          MIIESzCCAzOgAwIBAgIJANq6Drpt8SB5MA0GCSqGSIb3DQEBCwUAMHYxCzAJBgNVBAYTAkVFMREw
          ...
          RLXZUNmvy+zFSB+QFWEW2nlfYI8=
        </ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
</ModirumMPI>
```

3.7.6 Final response from MPI

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi">
  <Message merchantId="sys1331720664552" messageId="M1544784619256" version="4.0">
    <Response>
      <Parameters>
        <xid>EnT7echNxbjyNJSEIQiIsogpnUg=</xid>
        <mdStatus>1</mdStatus>
        <mdErrorMsg>Y-status/Challenge authentication via ACS:
          https://localhost.modirum.com:8543/dstests/ACSEmu2
        </mdErrorMsg>
        <enrollmenStatus>C</enrollmenStatus>
        <authenticationStatus>Y</authenticationStatus>
        <eci>05</eci>
        <cavv>QUNTRU1VUDYlLGl/eTtSLiQ8Ync=</cavv>
        <PAResVerified>true</PAResVerified>
        <PAResSyntaxOK>true</PAResSyntaxOK>
        <txId>16809</txId>
        <protocol>3DS2.1.0</protocol>
        <cardType>1</cardType>
      </Parameters>
      <TDS2RespAttributes>
        <Attribute name="TDS2.threeDSServerTransID">00000000-5676-5663-8000-0000000041a9</Attribute>
        <Attribute name="TDS2.dsTransID">dffc1e22-b02d-5af6-8000-0000000028a9</Attribute>
        <Attribute name="TDS2.acsTransID">00000000-0005-5a5a-8000-0167ac56b581</Attribute>
        <Attribute name="TDS2.authTimestamp">201812141050</Attribute>
      </TDS2RespAttributes>
    </Response>
  </Message>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
      <ds:Reference URI="#M1544784619256">
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
        <ds:DigestValue>B3VqtWHO9MFFmjVmEY1xHGWvyaAjiI6RmDMO1cUisvo=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>
      kCqt9Pj5+siFHCRkSET5215alFVwXa5wYJHaZkcl+JApFsCaF2E7Ic4deagJBcy4JPMdtiIAF1H0
      EF3yNZSHc6+Pjsz2slnmq8FvXjZUixBzxEfSqqDSp1GWZisjsx/jHVZ9lWH9JcBXqPFrxzTKYeX9
      oGF+BS+0IM5K3Q6Aw59VNELszFaKjkDDI4Un5qCnaJZCloIoTILz4cPUdpBBuQYHnZN1PplNxMgb
```

```
        gbLrHIpbhghvzKzSNwxNpLoLwJEA8cl3OZ5zp+WsTsPhSYcYI8eDQb4nOt9nF/9mZifiU9DzXHj9
        dBrL5BlNqn9Z76+n3Hf3N5xda9b8N7jYnV0UWw==
      </ds:SignatureValue>
      <ds:KeyInfo>
        <ds:X509Data>
          <ds:X509Certificate>
            MIIDYDCCAkgCAQEwDQYJKoZIhvcNAQELBQAwdjEaMBgGA1UEAxMRcHJvY0ludGVyZmFjZVNpZ24x
            ...
            9PDtopCdwc7k0pzUiQ==
          </ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </ds:Signature>
</ModirumMPI>
```

4 SOAP interface

As of MPI 4.0.2.42 there is an opportunity to send SOAP 1.2 messages to MPI. Interface reflects format and behavior of XML interface. There are a few differences:

1. XML messages should be sent with SOAP envelope. Examples below.
2. Messages should be sent with ContentType=application/soap+xml

WSDL schema is located here mdpaympi.war/ModirumMPI.wsdl or here http(s)://address/mdpaympi/ModirumMPI.wsdl. Message contents of SOAP are equal to XML API only wrapped to SOAP envelopes.

4.1 SOAP interface example

4.1.1 Initial request to MPI:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header/>
  <env:Body>
    <ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
      <Message md="535CF439C39D63F978C1B58845C2005D" merchantId="sys1331720664552"
messageId="M1494972873437" version="4.0">
        <Request>
          <EnrollmentRequest>
            <Parameters>
              <pan>4016000000002</pan>
              <expiry>1812</expiry>
              <deviceCategory>0</deviceCategory>
              <purchAmount>1100</purchAmount>

              <exponent>2</exponent>
              <description>DVD Movies</description>
              <currency>840</currency>

<termUrl>https://bank1.modirum.com:443/coffeehouse/MerchantHandler2;jsessionid=535CF439C39D63F978C1B58845C20
05D
              </termUrl>
              <xid>VGnMqa+1jNBTSnCnoh3K8PuiVZQ=</xid>
            </Parameters>
          </EnrollmentRequest>
        </Request>
      </Message>
      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
          <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
          <ds:Reference URI="#M1494972873437">
            <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <ds:DigestValue>ngY4nXCbtb00r8Uwb7ElwhMshoX3a1H/JkonU6yIzqI=</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>

<ds:SignatureValue>AuausHQxQuVWIE7V/ZMoT8PeDSTJhpUTImyPGS+EWOUqMkJMAogZQh/xiM9H+0g9cZ9fXa9zk
dX8j83AK0pnfy768CZU7MZ4kLIn50W/qaf62NNvS2o2BTQSpfnFjz4DX11hcKZzProdY63C/wqJHwOO
bN9al96mKm0kFkqU0batWnCRRnzD8blJZOZ35j+hdI5bTZjnXYsrft3FVhUlb0WbQccVfULmksnR
Ozf/DeKxo0vjvJCLCWYfu4plMrUKT6xCzsALWR8ycx4/kpWi9ICpemVIyOeXGZantH+p/s8RTk0r
+F1iVyTxOVGgmryHGxFizYA79g9ZAHmC0gugrg==
        </ds:SignatureValue>
        <ds:KeyInfo>
          <ds:X509Data>
            <ds:X509Certificate>
MIIERTCCAy2gAwIBAgIJAKcoY+m8gQgZMA0GCSqGSIb3
…
pAhfSIiDh2jcOvPq1F4=
```

```
            </ds:X509Certificate>
          </ds:X509Data>
        </ds:KeyInfo>
      </ds:Signature>
    </ModirumMPI>
  </env:Body>
</env:Envelope>
```

4.1.2 MPI response with ACS redirect template:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header/>
  <env:Body>
    <ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
      <Message md="535CF439C39D63F978C1B58845C2005D" merchantId="sys1331720664552"
messageId="M1494972873437" version="4.0">
        <Response>
          <Parameters>
            <xid>VGnMqa+1jNBTSnCnoh3K8PuiVZQ=</xid>
            <mdStatus>9</mdStatus>
            <mdErrorMsg>To be redirected to acs</mdErrorMsg>
            <enrollmenStatus>Y</enrollmenStatus>
          </Parameters>
          <redirectToACSForm>&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd"&gt;
            &lt;html&gt;
            &lt;head&gt;
            &lt;title&gt;MDpay default response template for web&lt;/title&gt;
            &lt;script type="text/javascript" language="javascript"&gt;
            function moveWindow() {
            var form = document.getElementById("downloadForm");
            if (form != null) {
            form.submit();
            }
            }
            &lt;/script&gt;
            &lt;/head&gt;
            &lt;body&gt;
            &lt;form id="downloadForm"
            action="https://3ds-acs.test.modirum.com/mdpayacs/pareq"
            method="POST"&gt;
            &lt;input type="hidden"
            name="PaReq"

value="eJxVUcluwjAQ/ZWIa6U4TsJSNIwERCqoAtFCOfQWOUOTqrGDnbD8fe2wtT7Ne+PZ3oNNromSNYlGE8KCjEm/
yCuyUYdH3SCqOgir8TvtEQ6kTaEkcj/wQ2A3aGu0yFNZI6RiP5kvMW4fsCuEkvQ8QR5GcbcH7IJApiXhVO12RDPVGPIy
KpVnclUBa3MgVCNrfcYwtlU3AI3+wbyuqyFjx+PRL1VW6Kb0hSqBuRywxz6rxkXG9joVGW5f5GKfPvHv5WSzllOp8uh
1sGqK7efbCJj7AVlaE4YB7wdd3vPCcMjjYRQBa3lIS7cEfqwTj1sRAnvhhYHKDRpfAOcu85cBq60mKc44iG3qjoBOlZJkf1
g57zFkZAQm28RbqENBxk53BLDHNdOZ01rUVscw6lufnrlTuyVc08IKxQf80tUBYK6EXY1kV5dt9M/9X/zRraA="&gt;
            &lt;input type="hidden"
            name="TermUrl"

value="https://bank1.modirum.com:443/coffeehouse/MerchantHandler2;jsessionid=535CF439C39D63F978C1B58845C2005
D"&gt;
            &lt;input type="hidden"
            name="MD"

            value="535CF439C39D63F978C1B58845C2005D"&gt;
            &lt;!-- To support javascript unaware/disabled browsers --&gt;
```

```
            &lt;div style="text-align: center;"&gt;
            &lt;img src="preloader.gif"/&gt;&lt;br/&gt;
            &lt;noscript&gt;
            &lt;center&gt;Please click the submit button below.&lt;br&gt;
            &lt;input type="submit" name="submit" value="Submit"&gt;&lt;/center&gt;
            &lt;/noscript&gt;
            &lt;/div&gt;
            &lt;/form&gt;
            &lt;script type="text/javascript"&gt;
            moveWindow();
            &lt;/script&gt;
            &lt;/body&gt;
            &lt;/html&gt;
          </redirectToACSForm>
        </Response>
      </Message>
      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
          <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
          <ds:Reference URI="#M1494972873437">
            <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <ds:DigestValue>4ACquN4SO0Dg9DZp7QG2GacukQkwuicEbexzlOKcjrM=</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>
rXYl+kWPqma87EYT5SLgSMILL56FlVsGrVVKqhDuc91WfwXXf0fNaB3RbPPg0BOflsTDTLo04UJ/
e5VaioL4JOO1reBeg+qglkGmESeFRpSNtpAnOTrRJycvIVoUkc1blmhErBCIiB6CH1CX9aaCtsdk
xalhEBBH0YhC4uUHzVvzmLEvcZIHvxW42Zlq1fWRilTQCkoxADv+MrV9LrTVnrOTX80gNcWe6eFB
Yla1t7fSl2cyD0M7/HtJ5KMKtP2AFwqaJjTkIIIY1n49GU45SJSUqxNKwOXyTmfh9N+qwVlEp0yS
y8Ty3HQ1u/f76EkrDyDPKrrb6jkTjS9mjQc+9A==
        </ds:SignatureValue>
        <ds:KeyInfo>
          <ds:X509Data>
            <ds:X509Certificate>
MIIEDDCCAvSgAwIBAgIJAPmMWIp6hdlRMA0GCSqGSIb3DQEBCwUAMGExCzAJBgNVBAYTAkVFMQ4w
…
orvXz6ca1a62exlyoQw=
            </ds:X509Certificate>
          </ds:X509Data>
        </ds:KeyInfo>
      </ds:Signature>
    </ModirumMPI>
  </env:Body>
</env:Envelope>
```

4.1.3 PARes validation request to MPI:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header/>
  <env:Body>
    <ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
      <Message md="535CF439C39D63F978C1B58845C2005D" merchantId="sys1331720664552"
messageId="M1494972873437" version="4.0">
        <Request>
          <PAResValidationRequest>
            <pares>
eJzVWNmyozqy/ZWK6kfHOQwGG064doSYMQabeXhjMsbM8/D1jfeuXae6um5E3/vUlxejJZFKKVeulHUyHm0c
…
fh37cAfx9O/B+bfl+X/q6bPv5HvWfn+sHWA==
            </pares>
          </PAResValidationRequest>
```

```
        </Request>
      </Message>
      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
          <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
          <ds:Reference URI="#M1494972873437">
            <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <ds:DigestValue>mOZ4WMPNRTfY/Uj4SWKRNBdKvrt+X7xgPpdxz7PBFMI=</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>

<ds:SignatureValue>lU6U6+SThF1697DBGJ8cEf7kJIH6R9NzVriTtr8Tq9dvbTKDT4BeBGHyTRRbhPiGG05whfRDAXpy
+JwgwhRKRjgScd1uudCpPdkOW0IPvCdT6/BGOEMy+OSBLhUGpoykSb3hS7xyQ2mVuFLJ3MSY8/o6a3W61IRSnNPvi1
ENpqudgAU1Qg8Opy8aHqy6nzquAFnjQlYbrJmB/5UCZkwMpUiK1F6fWz27yY8N1l5hJ/P/bE1SYfOjUQSQqd1ZXJ4czAY
8S1uzrNCbjtcxtoev6/TtUI46mnQmCfEXyxtK3Qfzkff YWqVdZ+Rxd/ZQ7zL5DD69IUvo2jRzOdO6+n6uw==
        </ds:SignatureValue>
        <ds:KeyInfo>
          <ds:X509Data>
            <ds:X509Certificate>
MIIERTCCAy2gAwIBAgIJAKcoY+m8gQgZMA0GCSqGSIb3DQEBCwUAMHQxCzAJBgN
…
pAhfSIiDh2jcOvPq1F4=
            </ds:X509Certificate>
          </ds:X509Data>
        </ds:KeyInfo>
      </ds:Signature>
    </ModirumMPI>
  </env:Body>
</env:Envelope>
```

4.1.4 PARes validation response from MPI:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header/>
  <env:Body>
    <ModirumMPI xmlns="http://www.modirum.com/schemas/mpiapi" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#">
      <Message md="535CF439C39D63F978C1B58845C2005D" merchantId="sys1331720664552"
messageId="M1494972873437" version="4.0">
        <Response>
          <Parameters>
            <xid>VGnMqa+1jNBTSnCnoh3K8PuiVZQ=</xid>
            <mdStatus>1</mdStatus>
            <mdErrorMsg>Authenticated</mdErrorMsg>
            <enrollmenStatus>Y</enrollmenStatus>
            <authenticationStatus>Y</authenticationStatus>
            <eci>05</eci>
            <cavv>AAABAGI1kQAAAAAjcTWRAAAAAAA=</cavv>
            <cavvAlgorithm>2</cavvAlgorithm>
            <PAResVerified>true</PAResVerified>
            <PAResSyntaxOK>true</PAResSyntaxOK>
            <txId>13503</txId>
          </Parameters>
        </Response>
      </Message>

      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">

        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
          <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
          <ds:Reference URI="#M1494972873437">
```

```
            <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <ds:DigestValue>sYLd56IRpCQN4GdMOrJpGCOhbAzP90u+Ew/8f/kPuf4=</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>

        <ds:SignatureValue>
qtFKBS+MrOeL1DKyOg8LYkM260DC4UViXutKKFhKto3A7kKfE0Ww1Zh+Ri4NNzehsDgi59NAINMD
EEQGh2Mlpjvjl35PO1GylLRyOOZMxsUjIFlxwDb/nHlNuQWPPADwBbtmQz7wHG60Boc9l5Z6IBI2
zGEtas6ZxPcbD8VLeOgZYg34TvjIY5K9giFyEdUYUyKw2HHaABXrjlndCFTr/3NeenqRAuI/vThh
uZyNLw99baE1qIZcMeHcq4CaBNM8EG4dgKETtda96PBUy/E6ykBp+VZooSwRm6Ts8JuRQVjPwH2S
BbxWIXwp15qSlw0wRwSCQugRBz2UC/REk/dVkA==

        </ds:SignatureValue>
        <ds:KeyInfo>
          <ds:X509Data>
            <ds:X509Certificate>
MIIEDDCCAvSgAwIBAgIJAPmMWIp6hdlRMA0GCSqGSIb3DQEBCwUAMGExCzAJBgNVBAYTAkVFMQ4w
...
orvXz6ca1a62exlyoQw=
            </ds:X509Certificate>
          </ds:X509Data>
        </ds:KeyInfo>
      </ds:Signature>
    </ModirumMPI>
  </env:Body>
</env:Envelope>
```

5 Processor Certificate

Processor certificate is used by merchant to calculate the signature value for the response messages.

For testing purposes, merchant can use the following processor certificate:

-----BEGIN CERTIFICATE-----
MIIEXjCCAsYCAQEwDQYJKoZIhvcNAQELBQAwdTElMCMGA1UEAxMcQ2FyZGxpbmsgVUFUIFNpZ25p
bmcgYW5kIENTRTENMAsGA1UECxMERUNPTTERMA8GA1UEChMIQ2FyZGxpbmsxDzANBgNVBAcTBkF0
aGVuczEMMAoGA1UECBMDQVRIMQswCQYDVQQGEwJHUjAeFw0xODA2MjEyMTAwMDBaFw0yNTA2MjIy
MDU5NTlaMHUxJTAjBgNVBAMTHENhcmRsaW5rIFVBVCBTaWduaW5nIGFuZCBDU0UxDTALBgNVBAsT
BEVDT00xETAPBgNVBAoTCENhcmRsaW5rMQ8wDQYDVQQHEwZBdGhlbnMxDDAKBgNVBAgTA0FUSDEL
MAkGA1UEBhMCR1IwggGiMA0GCSqGSIb3DQEBAQUAA4IBjwAwggGKAoIBgQDlZIj4eMY2hU7ot4kk
gB1e7xJniAe07ntRVwPZdJ1cxevLvSoQMvgd8070RrT7cPDXp6iJIl0RKBnCwZspwoO5evUngdfo
AleyLSVUXljKp2G/e6Kt22RMCLtYsqNv4qFW5nW8XwB88wvqziSMPu9Mo1gGhOxWpS4Viy3NvrtE
VOWXvssx+ZLPolb3AW93w7BOfzEpt7LM3GwrSYZuPoPHcwdkBs0nF+htIEOq/2T7GDcZPNIUmllu
4nQt6u7T1SJ0/TpdHta/p55xptE7QLZlNdphIxvu4Zc9U7mwvlCN8MqMNQnQSFlqnBdOgtQ5gxfE
8x/cSWOVLzTh6dWOc2o7aiAhk8sVopl7N4jeL4U4Nvp0GyDodoWgUJeweDookIb9DL2fgQeBLKn8
ZFDPOyoBQSNr8AAm3p0bgTDY4XkTuav919LGgCjR5k389CW256zXCgsj5Dnn8gcTrf0mwziUbjlG
t/UIy7CA7kmpELwna4NNo7Lt6laILqletJi1rlECAwEAATANBgkqhkiG9w0BAQsFAAOCAYEAVkOF
bVwxj/pbnTH8Z2y/17P1yzv4H6vKB2RdG60CMSou0X/WNybBgaMSf6qJJs3osUC68qx27Q3pYp4i
7onsTlNedhSsUVZVabRHXkjLxGLx9saZNiZ9turIyxzfC7VdeGaogvmcFPZAFgkGSFy4tAZz8fIk
L7XI9pp5NTrjP9AL1ETVgwoHFKoeEKU1ewgQGRXpsM2sQnanMrTOgfVWz+qmaMmCcgeuQnYDPkZX
X3jo456N0IDcGhJRmzkO8x0ge3DGyTc2mdS+38c61VEDd2TQHDHJuGsjCSVMjYh83JF7Ut3imFYh
v3jgmHNkEDsp7XU81UMaV1nD0WzwNTbuMlyuvUQltLtQ0lciDl+yT7zciHZr3JkL3am9lCtny/DR
Oyw7pZnDCbWHaUKl4pV5UtwCIT/o5v7yo3av1z5o6Ufial+kemeyhcU7PtMXZ6mgW9Hcq4htX1BT
l/LsTN/42XxvrdzystkmvJeSlrNLPbeASi8MC3j/xQdUjc6mWQ/t
-----END CERTIFICATE-----

For production purposes, please contact us via email at ecommerce_support@cardlink.gr