



API Reference

# AWS Payment Cryptography Data Plane



**API Version 2022-02-03**

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# AWS Payment Cryptography Data Plane: API Reference

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

# Table of Contents

<b>Welcome</b> .....	<b>1</b>
<b>Actions</b> .....	<b>2</b>
<b>DecryptData</b> .....	<b>3</b>
Request Syntax .....	3
URI Request Parameters .....	4
Request Body .....	4
Response Syntax .....	5
Response Elements .....	5
Errors .....	6
See Also .....	7
<b>EncryptData</b> .....	<b>8</b>
Request Syntax .....	9
URI Request Parameters .....	9
Request Body .....	9
Response Syntax .....	10
Response Elements .....	10
Errors .....	11
See Also .....	12
<b>GenerateCardValidationData</b> .....	<b>13</b>
Request Syntax .....	13
URI Request Parameters .....	13
Request Body .....	14
Response Syntax .....	15
Response Elements .....	15
Errors .....	16
See Also .....	17
<b>GenerateMac</b> .....	<b>18</b>
Request Syntax .....	18
URI Request Parameters .....	18
Request Body .....	19
Response Syntax .....	20
Response Elements .....	20
Errors .....	21
See Also .....	22

GeneratePinData .....	23
Request Syntax .....	23
URI Request Parameters .....	23
Request Body .....	24
Response Syntax .....	25
Response Elements .....	26
Errors .....	27
See Also .....	28
ReEncryptData .....	29
Request Syntax .....	29
URI Request Parameters .....	29
Request Body .....	30
Response Syntax .....	31
Response Elements .....	31
Errors .....	32
See Also .....	33
TranslatePinData .....	34
Request Syntax .....	34
URI Request Parameters .....	35
Request Body .....	35
Response Syntax .....	37
Response Elements .....	37
Errors .....	38
See Also .....	39
VerifyAuthRequestCryptogram .....	40
Request Syntax .....	40
URI Request Parameters .....	41
Request Body .....	41
Response Syntax .....	42
Response Elements .....	43
Errors .....	44
See Also .....	44
VerifyCardValidationData .....	46
Request Syntax .....	46
URI Request Parameters .....	46
Request Body .....	47

---

Response Syntax .....	48
Response Elements .....	48
Errors .....	49
See Also .....	50
VerifyMac .....	51
Request Syntax .....	51
URI Request Parameters .....	51
Request Body .....	51
Response Syntax .....	53
Response Elements .....	53
Errors .....	54
See Also .....	55
VerifyPinData .....	56
Request Syntax .....	56
URI Request Parameters .....	57
Request Body .....	57
Response Syntax .....	59
Response Elements .....	59
Errors .....	61
See Also .....	61
<b>Data Types .....</b>	<b>63</b>
AmexCardSecurityCodeVersion1 .....	65
Contents .....	65
See Also .....	65
AmexCardSecurityCodeVersion2 .....	66
Contents .....	66
See Also .....	66
AsymmetricEncryptionAttributes .....	67
Contents .....	67
See Also .....	67
CardGenerationAttributes .....	68
Contents .....	68
See Also .....	69
CardHolderVerificationValue .....	70
Contents .....	70
See Also .....	71

CardVerificationAttributes .....	72
Contents .....	72
See Also .....	73
CardVerificationValue1 .....	75
Contents .....	75
See Also .....	75
CardVerificationValue2 .....	76
Contents .....	76
See Also .....	76
CryptogramAuthResponse .....	77
Contents .....	77
See Also .....	77
CryptogramVerificationArpcMethod1 .....	79
Contents .....	79
See Also .....	79
CryptogramVerificationArpcMethod2 .....	80
Contents .....	80
See Also .....	80
DiscoverDynamicCardVerificationCode .....	82
Contents .....	82
See Also .....	83
DukptAttributes .....	84
Contents .....	84
See Also .....	84
DukptDerivationAttributes .....	86
Contents .....	86
See Also .....	87
DukptEncryptionAttributes .....	88
Contents .....	88
See Also .....	89
DynamicCardVerificationCode .....	90
Contents .....	90
See Also .....	91
DynamicCardVerificationValue .....	92
Contents .....	92
See Also .....	93

EmvEncryptionAttributes .....	94
Contents .....	94
See Also .....	95
EncryptionDecryptionAttributes .....	97
Contents .....	97
See Also .....	98
Ibm3624NaturalPin .....	99
Contents .....	99
See Also .....	100
Ibm3624PinFromOffset .....	101
Contents .....	101
See Also .....	102
Ibm3624PinOffset .....	103
Contents .....	103
See Also .....	104
Ibm3624PinVerification .....	105
Contents .....	105
See Also .....	106
Ibm3624RandomPin .....	107
Contents .....	107
See Also .....	108
MacAlgorithmDukpt .....	109
Contents .....	109
See Also .....	110
MacAlgorithmEmv .....	111
Contents .....	111
See Also .....	112
MacAttributes .....	113
Contents .....	113
See Also .....	114
PinData .....	115
Contents .....	115
See Also .....	115
PinGenerationAttributes .....	117
Contents .....	117
See Also .....	118

PinVerificationAttributes .....	119
Contents .....	119
See Also .....	119
ReEncryptionAttributes .....	120
Contents .....	120
See Also .....	120
SessionKeyAmex .....	121
Contents .....	121
See Also .....	121
SessionKeyDerivation .....	122
Contents .....	122
See Also .....	123
SessionKeyDerivationValue .....	124
Contents .....	124
See Also .....	124
SessionKeyEmv2000 .....	126
Contents .....	126
See Also .....	127
SessionKeyEmvCommon .....	128
Contents .....	128
See Also .....	129
SessionKeyMastercard .....	130
Contents .....	130
See Also .....	131
SessionKeyVisa .....	132
Contents .....	132
See Also .....	132
SymmetricEncryptionAttributes .....	133
Contents .....	133
See Also .....	133
TranslationIsoFormats .....	135
Contents .....	135
See Also .....	136
TranslationPinDataIsoFormat034 .....	137
Contents .....	137
See Also .....	137



---

TranslationPinDataAlsoFormat1 .....	138
Contents .....	138
See Also .....	138
ValidationExceptionField .....	139
Contents .....	139
See Also .....	139
VisaPin .....	140
Contents .....	140
See Also .....	140
VisaPinVerification .....	141
Contents .....	141
See Also .....	141
VisaPinVerificationValue .....	142
Contents .....	142
See Also .....	142

# Welcome

You use the AWS Payment Cryptography Data Plane to manage how encryption keys are used for payment-related transaction processing and associated cryptographic operations. You can encrypt, decrypt, generate, verify, and translate payment-related cryptographic operations in AWS Payment Cryptography. For more information, see [Data operations](#) in the *AWS Payment Cryptography User Guide*.

To manage your encryption keys, you use the [AWS Payment Cryptography Control Plane](#). You can create, import, export, share, manage, and delete keys. You can also manage AWS Identity and Access Management (IAM) policies for keys.

This document was last published on July 2, 2024.

# Actions

The following actions are supported:

- [DecryptData](#)
- [EncryptData](#)
- [GenerateCardValidationData](#)
- [GenerateMac](#)
- [GeneratePinData](#)
- [ReEncryptData](#)
- [TranslatePinData](#)
- [VerifyAuthRequestCryptogram](#)
- [VerifyCardValidationData](#)
- [VerifyMac](#)
- [VerifyPinData](#)

# DecryptData

Decrypts ciphertext data to plaintext using a symmetric (TDES, AES), asymmetric (RSA), or derived (DUKPT or EMV) encryption key scheme. For more information, see [Decrypt data](#) in the *AWS Payment Cryptography User Guide*.

You can use an encryption key generated within AWS Payment Cryptography, or you can import your own encryption key by calling [ImportKey](#). For this operation, the key must have `KeyModesOfUse` set to `Decrypt`. In asymmetric decryption, AWS Payment Cryptography decrypts the ciphertext using the private component of the asymmetric encryption key pair. For data encryption outside of AWS Payment Cryptography, you can export the public component of the asymmetric key pair by calling [GetPublicCertificate](#).

For symmetric and DUKPT decryption, AWS Payment Cryptography supports TDES and AES algorithms. For EMV decryption, AWS Payment Cryptography supports TDES algorithms. For asymmetric decryption, AWS Payment Cryptography supports RSA.

When you use TDES or TDES DUKPT, the ciphertext data length must be a multiple of 8 bytes. For AES or AES DUKPT, the ciphertext data length must be a multiple of 16 bytes. For RSA, it could be equal to the key size unless padding is enabled.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [EncryptData](#)
- [GetPublicCertificate](#)
- [ImportKey](#)

## Request Syntax

```
POST /keys/KeyIdentifier/decrypt HTTP/1.1
Content-type: application/json

{
  "CipherText": "string",
```

```
"DecryptionAttributes": { ... }  
}
```

## URI Request Parameters

The request uses the following URI parameters.

### [KeyIdentifier](#)

The keyARN of the encryption key that AWS Payment Cryptography uses for ciphertext decryption.

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### [CipherText](#)

The ciphertext to decrypt.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 4096.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

Required: Yes

### [DecryptionAttributes](#)

The encryption key type and attributes for ciphertext decryption.

Type: [EncryptionDecryptionAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "KeyArn": "string",
  "KeyCheckValue": "string",
  "PlainText": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### KeyArn

The keyARN of the encryption key that AWS Payment Cryptography uses for ciphertext decryption.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

### **PlainText**

The decrypted plaintext data in hexBinary format.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 4096.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

## **Errors**

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 429

### **ValidationException**

The request was denied due to an invalid request error.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# EncryptData

Encrypts plaintext data to ciphertext using a symmetric (TDES, AES), asymmetric (RSA), or derived (DUKPT or EMV) encryption key scheme. For more information, see [Encrypt data](#) in the *AWS Payment Cryptography User Guide*.

You can generate an encryption key within AWS Payment Cryptography by calling [CreateKey](#). You can import your own encryption key by calling [ImportKey](#). For this operation, the key must have `KeyModesOfUse` set to `Encrypt`. In asymmetric encryption, plaintext is encrypted using public component. You can import the public component of an asymmetric key pair created outside AWS Payment Cryptography by calling [ImportKey](#).

For symmetric and DUKPT encryption, AWS Payment Cryptography supports TDES and AES algorithms. For EMV encryption, AWS Payment Cryptography supports TDES algorithms. For asymmetric encryption, AWS Payment Cryptography supports RSA.

When you use TDES or TDES DUKPT, the plaintext data length must be a multiple of 8 bytes. For AES or AES DUKPT, the plaintext data length must be a multiple of 16 bytes. For RSA, it could be equal to the key size unless padding is enabled.

To encrypt using DUKPT, you must already have a BDK (Base Derivation Key) key in your account with `KeyModesOfUse` set to `DeriveKey`, or you can generate a new DUKPT key by calling [CreateKey](#). To encrypt using EMV, you must already have an IMK (Issuer Master Key) key in your account with `KeyModesOfUse` set to `DeriveKey`.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [DecryptData](#)
- [GetPublicCertificate](#)
- [ImportKey](#)
- [ReEncryptData](#)

## Request Syntax

```
POST /keys/KeyIdentifier/encrypt HTTP/1.1
Content-type: application/json

{
  "EncryptionAttributes": { ... },
  "PlainText": "string"
}
```

## URI Request Parameters

The request uses the following URI parameters.

### [KeyIdentifier](#)

The keyARN of the encryption key that AWS Payment Cryptography uses for plaintext encryption.

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### [EncryptionAttributes](#)

The encryption key type and attributes for plaintext encryption.

Type: [EncryptionDecryptionAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

## PlainText

The plaintext to be encrypted.

### Note

For encryption using asymmetric keys, plaintext data length is constrained by encryption key strength that you define in `KeyAlgorithm` and padding type that you define in `AsymmetricEncryptionAttributes`. For more information, see [Encrypt data](#) in the *AWS Payment Cryptography User Guide*.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 4064.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "CipherText": "string",
  "KeyArn": "string",
  "KeyCheckValue": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### CipherText

The encrypted ciphertext.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 4096.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

### KeyArn

The keyARN of the encryption key that AWS Payment Cryptography uses for plaintext encryption.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

## Errors

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

## ResourceNotFoundException

The request was denied due to an invalid resource error.

HTTP Status Code: 404

## ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 429

## ValidationException

The request was denied due to an invalid request error.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GenerateCardValidationData

Generates card-related validation data using algorithms such as Card Verification Values (CVV/CVV2), Dynamic Card Verification Values (dCVV/dCVV2), or Card Security Codes (CSC). For more information, see [Generate card data](#) in the *AWS Payment Cryptography User Guide*.

This operation generates a CVV or CSC value that is printed on a payment credit or debit card during card production. The CVV or CSC, PAN (Primary Account Number) and expiration date of the card are required to check its validity during transaction processing. To begin this operation, a CVK (Card Verification Key) encryption key is required. You can use [CreateKey](#) or [ImportKey](#) to establish a CVK within AWS Payment Cryptography. The `KeyModesOfUse` should be set to `Generate` and `Verify` for a CVK encryption key.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [ImportKey](#)
- [VerifyCardValidationData](#)

## Request Syntax

```
POST /cardvalidationdata/generate HTTP/1.1
Content-type: application/json
```

```
{
  "GenerationAttributes": { ... },
  "KeyIdIdentifier": "string",
  "PrimaryAccountNumber": "string",
  "ValidationDataLength": number
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### GenerationAttributes

The algorithm for generating CVV or CSC values for the card within AWS Payment Cryptography.

Type: [CardGenerationAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### KeyIdIdentifier

The keyARN of the CVK encryption key that AWS Payment Cryptography uses to generate card data.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN), a unique identifier for a payment credit or debit card that associates the card with a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

### ValidationDataLength

The length of the CVV or CSC to be generated. The default value is 3.

Type: Integer

Valid Range: Minimum value of 3. Maximum value of 5.

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "KeyArn": "string",
  "KeyCheckValue": "string",
  "ValidationData": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### KeyArn

The keyARN of the CVK encryption key that AWS Payment Cryptography uses to generate CVV or CSC.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.



Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

### ValidationData

The CVV or CSC value that AWS Payment Cryptography generates for the card.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 5.

Pattern: `[0-9]+`

## Errors

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 429

### **ValidationException**

The request was denied due to an invalid request error.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GenerateMac

Generates a Message Authentication Code (MAC) cryptogram within AWS Payment Cryptography.

You can use this operation to authenticate card-related data by using known data values to generate MAC for data validation between the sending and receiving parties. This operation uses message data, a secret encryption key and MAC algorithm to generate a unique MAC value for transmission. The receiving party of the MAC must use the same message data, secret encryption key and MAC algorithm to reproduce another MAC value for comparison.

You can use this operation to generate a DUPKT, CMAC, HMAC or EMV MAC by setting generation attributes and algorithm to the associated values. The MAC generation encryption key must have valid values for KeyUsage such as TR31\_M7\_HMAC\_KEY for HMAC generation, and they key must have KeyModesOfUse set to Generate and Verify.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [VerifyMac](#)

## Request Syntax

```
POST /mac/generate HTTP/1.1
Content-type: application/json

{
  "GenerationAttributes": { ... },
  "KeyIdentifier": "string",
  "MacLength": number,
  "MessageData": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### GenerationAttributes

The attributes and data values to use for MAC generation within AWS Payment Cryptography.

Type: [MacAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### KeyIdIdentifier

The keyARN of the MAC generation encryption key.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

### MacLength

The length of a MAC under generation.

Type: Integer

Valid Range: Minimum value of 4. Maximum value of 16.

Required: No

### MessageData

The data for which a MAC is under generation. This value must be hexBinary.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4096.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "KeyArn": "string",
  "KeyCheckValue": "string",
  "Mac": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### [KeyArn](#)

The keyARN of the encryption key that AWS Payment Cryptography uses for MAC generation.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### [KeyCheckValue](#)

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

## **Mac**

The MAC cryptogram generated within AWS Payment Cryptography.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 128.

Pattern: `[0-9a-fA-F]+`

## **Errors**

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 429

### **ValidationException**

The request was denied due to an invalid request error.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GeneratePinData

Generates pin-related data such as PIN, PIN Verification Value (PVV), PIN Block, and PIN Offset during new card issuance or reissuance. For more information, see [Generate PIN data](#) in the *AWS Payment Cryptography User Guide*.

PIN data is never transmitted in clear to or from AWS Payment Cryptography. This operation generates PIN, PVV, or PIN Offset and then encrypts it using Pin Encryption Key (PEK) to create an EncryptedPinBlock for transmission from AWS Payment Cryptography. This operation uses a separate Pin Verification Key (PVK) for VISA PVV generation.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [GenerateCardValidationData](#)
- [TranslatePinData](#)
- [VerifyPinData](#)

## Request Syntax

```
POST /pindata/generate HTTP/1.1
Content-type: application/json

{
  "EncryptionKeyIdentifier": "string",
  "GenerationAttributes": { ... },
  "GenerationKeyIdentifier": "string",
  "PinBlockFormat": "string",
  "PinDataLength": number,
  "PrimaryAccountNumber": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.



## Request Body

The request accepts the following data in JSON format.

### EncryptionKeyIdentifier

The keyARN of the PEK that AWS Payment Cryptography uses to encrypt the PIN Block.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

### GenerationAttributes

The attributes and values to use for PIN, PVV, or PIN Offset generation.

Type: [PinGenerationAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### GenerationKeyIdentifier

The keyARN of the PEK that AWS Payment Cryptography uses for pin data generation.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

### PinBlockFormat

The PIN encoding format for pin data generation as specified in ISO 9564. AWS Payment Cryptography supports `ISO_Format_0` and `ISO_Format_3`.

The `ISO_Format_0` PIN block format is equivalent to the ANSI X9.8, VISA-1, and ECI-1 PIN block formats. It is similar to a VISA-4 PIN block format. It supports a PIN from 4 to 12 digits in length.

The `ISO_Format_3` PIN block format is the same as `ISO_Format_0` except that the fill digits are random values from 10 to 15.

Type: String

Valid Values: `ISO_FORMAT_0` | `ISO_FORMAT_3`

Required: Yes

### PinDataLength

The length of PIN under generation.

Type: Integer

Valid Range: Minimum value of 4. Maximum value of 12.

Required: No

### PrimaryAccountNumber

The Primary Account Number (PAN), a unique identifier for a payment credit or debit card that associates the card with a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "EncryptedPinBlock": "string",
```

```
"EncryptionKeyArn": "string",  
"EncryptionKeyCheckValue": "string",  
"GenerationKeyArn": "string",  
"GenerationKeyCheckValue": "string",  
"PinData": { ... }  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### EncryptedPinBlock

The PIN block encrypted under PEK from AWS Payment Cryptography. The encrypted PIN block is a composite of PAN (Primary Account Number) and PIN (Personal Identification Number), generated in accordance with ISO 9564 standard.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: `[0-9a-fA-F]+`

### EncryptionKeyArn

The keyARN of the PEK that AWS Payment Cryptography uses for encrypted pin block generation.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### EncryptionKeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

### GenerationKeyArn

The keyARN of the pin data generation key that AWS Payment Cryptography uses for PIN, PVV or PIN Offset generation.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### GenerationKeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

### PinData

The attributes and values AWS Payment Cryptography uses for pin data generation.

Type: [PinData](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

## Errors

### AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 429

### **ValidationException**

The request was denied due to an invalid request error.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ReEncryptData

Re-encrypt ciphertext using DUKPT or Symmetric data encryption keys.

You can either generate an encryption key within AWS Payment Cryptography by calling [CreateKey](#) or import your own encryption key by calling [ImportKey](#). The KeyArn for use with this operation must be in a compatible key state with KeyModesOfUse set to Encrypt.

For symmetric and DUKPT encryption, AWS Payment Cryptography supports TDES and AES algorithms. To encrypt using DUKPT, a DUKPT key must already exist within your account with KeyModesOfUse set to DeriveKey or a new DUKPT can be generated by calling [CreateKey](#).

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [DecryptData](#)
- [EncryptData](#)
- [GetPublicCertificate](#)
- [ImportKey](#)

## Request Syntax

```
POST /keys/IncomingKeyIdentifier/reencrypt HTTP/1.1
Content-type: application/json
```

```
{
  "CipherText": "string",
  "IncomingEncryptionAttributes": { ... },
  "OutgoingEncryptionAttributes": { ... },
  "OutgoingKeyIdentifier": "string"
}
```

## URI Request Parameters

The request uses the following URI parameters.

## IncomingKeyIdentifier

The keyARN of the encryption key of incoming ciphertext data.

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### CipherText

Ciphertext to be encrypted. The minimum allowed length is 16 bytes and maximum allowed length is 4096 bytes.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 4096.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

Required: Yes

### IncomingEncryptionAttributes

The attributes and values for incoming ciphertext.

Type: [ReEncryptionAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### OutgoingEncryptionAttributes

The attributes and values for outgoing ciphertext data after encryption by AWS Payment Cryptography.

Type: [ReEncryptionAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### [OutgoingKeyIdentifier](#)

The keyARN of the encryption key of outgoing ciphertext data after encryption by AWS Payment Cryptography.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "CipherText": "string",
  "KeyArn": "string",
  "KeyCheckValue": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### [CipherText](#)

The encrypted ciphertext.



Type: String

Length Constraints: Minimum length of 16. Maximum length of 4096.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

### KeyArn

The keyARN (Amazon Resource Name) of the encryption key that AWS Payment Cryptography uses for plaintext encryption.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

## Errors

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

## ResourceNotFoundException

The request was denied due to an invalid resource error.

HTTP Status Code: 404

## ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 429

## ValidationException

The request was denied due to an invalid request error.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# TranslatePinData

Translates encrypted PIN block from and to ISO 9564 formats 0,1,3,4. For more information, see [Translate PIN data](#) in the *AWS Payment Cryptography User Guide*.

PIN block translation involves changing the encryption of PIN block from one encryption key to another encryption key and changing PIN block format from one to another without PIN block data leaving AWS Payment Cryptography. The encryption key transformation can be from PEK (Pin Encryption Key) to BDK (Base Derivation Key) for DUKPT or from BDK for DUKPT to PEK. AWS Payment Cryptography supports TDES and AES key derivation type for DUKPT translations.

The allowed combinations of PIN block format translations are guided by PCI. It is important to note that not all encrypted PIN block formats (example, format 1) require PAN (Primary Account Number) as input. And as such, PIN block format that requires PAN (example, formats 0,3,4) cannot be translated to a format (format 1) that does not require a PAN for generation.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

## Note

AWS Payment Cryptography currently supports ISO PIN block 4 translation for PIN block built using legacy PAN length. That is, PAN is the right most 12 digits excluding the check digits.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [GeneratePinData](#)
- [VerifyPinData](#)

## Request Syntax

```
POST /pindata/translate HTTP/1.1
Content-type: application/json
```

```
{
  "EncryptedPinBlock": "string",
  "IncomingDukptAttributes": {
    "DukptKeyDerivationType": "string",
    "DukptKeyVariant": "string",
    "KeySerialNumber": "string"
  },
  "IncomingKeyIdentifier": "string",
  "IncomingTranslationAttributes": { ... },
  "OutgoingDukptAttributes": {
    "DukptKeyDerivationType": "string",
    "DukptKeyVariant": "string",
    "KeySerialNumber": "string"
  },
  "OutgoingKeyIdentifier": "string",
  "OutgoingTranslationAttributes": { ... }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### EncryptedPinBlock

The encrypted PIN block data that AWS Payment Cryptography translates.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: (?:[0-9a-fA-F][0-9a-fA-F])+

Required: Yes

### IncomingDukptAttributes

The attributes and values to use for incoming DUKPT encryption key for PIN block translation.

Type: [DukptDerivationAttributes](#) object

Required: No

### IncomingKeyIdentifier

The keyARN of the encryption key under which incoming PIN block data is encrypted. This key type can be PEK or BDK.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

### IncomingTranslationAttributes

The format of the incoming PIN block data for translation within AWS Payment Cryptography.

Type: [TranslationIsoFormats](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### OutgoingDukptAttributes

The attributes and values to use for outgoing DUKPT encryption key after PIN block translation.

Type: [DukptDerivationAttributes](#) object

Required: No

### OutgoingKeyIdentifier

The keyARN of the encryption key for encrypting outgoing PIN block data. This key type can be PEK or BDK.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## OutgoingTranslationAttributes

The format of the outgoing PIN block data after translation by AWS Payment Cryptography.

Type: [TranslationIsoFormats](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "KeyArn": "string",
  "KeyCheckValue": "string",
  "PinBlock": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### KeyArn

The keyARN of the encryption key that AWS Payment Cryptography uses to encrypt outgoing PIN block data after translation.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

### PinBlock

The outgoing encrypted PIN block data after translation.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: `[0-9a-fA-F]+`

## Errors

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

## ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 429

## ValidationException

The request was denied due to an invalid request error.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# VerifyAuthRequestCryptogram

Verifies Authorization Request Cryptogram (ARQC) for a EMV chip payment card authorization. For more information, see [Verify auth request cryptogram](#) in the *AWS Payment Cryptography User Guide*.

ARQC generation is done outside of AWS Payment Cryptography and is typically generated on a point of sale terminal for an EMV chip card to obtain payment authorization during transaction time. For ARQC verification, you must first import the ARQC generated outside of AWS Payment Cryptography by calling [ImportKey](#). This operation uses the imported ARQC and an major encryption key (DUKPT) created by calling [CreateKey](#) to either provide a boolean ARQC verification result or provide an APRC (Authorization Response Cryptogram) response using Method 1 or Method 2. The ARPC\_METHOD\_1 uses AuthResponseCode to generate ARPC and ARPC\_METHOD\_2 uses CardStatusUpdate to generate ARPC.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [VerifyCardValidationData](#)
- [VerifyPinData](#)

## Request Syntax

```
POST /cryptogram/verify HTTP/1.1
Content-type: application/json

{
  "AuthRequestCryptogram": "string",
  "AuthResponseAttributes": { ... },
  "KeyIdentifier": "string",
  "MajorKeyDerivationMode": "string",
  "SessionKeyDerivationAttributes": { ... },
  "TransactionData": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### AuthRequestCryptogram

The auth request cryptogram imported into AWS Payment Cryptography for ARQC verification using a major encryption key and transaction data.

Type: String

Length Constraints: Fixed length of 16.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### AuthResponseAttributes

The attributes and values for auth request cryptogram verification. These parameters are required in case using ARPC Method 1 or Method 2 for ARQC verification.

Type: [CryptogramAuthResponse](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: No

### KeyIdentifier

The keyARN of the major encryption key that AWS Payment Cryptography uses for ARQC verification.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

### MajorKeyDerivationMode

The method to use when deriving the major encryption key for ARQC verification within AWS Payment Cryptography. The same key derivation mode was used for ARQC generation outside of AWS Payment Cryptography.

Type: String

Valid Values: EMV\_OPTION\_A | EMV\_OPTION\_B

Required: Yes

### SessionKeyDerivationAttributes

The attributes and values to use for deriving a session key for ARQC verification within AWS Payment Cryptography. The same attributes were used for ARQC generation outside of AWS Payment Cryptography.

Type: [SessionKeyDerivation](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### TransactionData

The transaction data that AWS Payment Cryptography uses for ARQC verification. The same transaction is used for ARQC generation outside of AWS Payment Cryptography.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 1024.

Pattern: [0-9a-fA-F]+

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "AuthResponseValue": "string",
  "KeyArn": "string",
  "KeyCheckValue": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### AuthResponseValue

The result for ARQC verification or ARPC generation within AWS Payment Cryptography.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

### KeyArn

The keyARN of the major encryption key that AWS Payment Cryptography uses for ARQC verification.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

## Errors

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 429

### **ValidationException**

The request was denied due to an invalid request error.

HTTP Status Code: 400

### **VerificationFailedException**

This request failed verification.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# VerifyCardValidationData

Verifies card-related validation data using algorithms such as Card Verification Values (CVV/CVV2), Dynamic Card Verification Values (dCVV/dCVV2) and Card Security Codes (CSC). For more information, see [Verify card data](#) in the *AWS Payment Cryptography User Guide*.

This operation validates the CVV or CSC codes that is printed on a payment credit or debit card during card payment transaction. The input values are typically provided as part of an inbound transaction to an issuer or supporting platform partner. AWS Payment Cryptography uses CVV or CSC, PAN (Primary Account Number) and expiration date of the card to check its validity during transaction processing. In this operation, the CVK (Card Verification Key) encryption key for use with card data verification is same as the one in used for [GenerateCardValidationData](#).

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [GenerateCardValidationData](#)
- [VerifyAuthRequestCryptogram](#)
- [VerifyPinData](#)

## Request Syntax

```
POST /cardvalidationdata/verify HTTP/1.1
Content-type: application/json
```

```
{
  "KeyIdentifier": "string",
  "PrimaryAccountNumber": "string",
  "ValidationData": "string",
  "VerificationAttributes": { ... }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### KeyIdentifier

The keyARN of the CVK encryption key that AWS Payment Cryptography uses to verify card data.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN), a unique identifier for a payment credit or debit card that associates the card with a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

### ValidationData

The CVV or CSC value for use for card data verification within AWS Payment Cryptography.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 5.

Pattern: `[0-9]+`

Required: Yes



## VerificationAttributes

The algorithm to use for verification of card data within AWS Payment Cryptography.

Type: [CardVerificationAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "KeyArn": "string",
  "KeyCheckValue": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### KeyArn

The keyARN of the CVK encryption key that AWS Payment Cryptography uses to verify CVV or CSC.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

## Errors

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 429

### **ValidationException**

The request was denied due to an invalid request error.

HTTP Status Code: 400

### **VerificationFailedException**

This request failed verification.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# VerifyMac

Verifies a Message Authentication Code (MAC).

You can use this operation to verify MAC for message data authentication such as . In this operation, you must use the same message data, secret encryption key and MAC algorithm that was used to generate MAC. You can use this operation to verify a DUPKT, CMAC, HMAC or EMV MAC by setting generation attributes and algorithm to the associated values.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [GenerateMac](#)

## Request Syntax

```
POST /mac/verify HTTP/1.1
Content-type: application/json

{
  "KeyIdentifier": "string",
  "Mac": "string",
  "MacLength": number,
  "MessageData": "string",
  "VerificationAttributes": { ... }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

## KeyIdIdentifier

The keyARN of the encryption key that AWS Payment Cryptography uses to verify MAC data.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## Mac

The MAC being verified.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 128.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

Required: Yes

## MacLength

The length of the MAC.

Type: Integer

Valid Range: Minimum value of 4. Maximum value of 16.

Required: No

## MessageData

The data on for which MAC is under verification. This value must be hexBinary.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4096.

Pattern: `(?:[0-9a-fA-F][0-9a-fA-F])+`

Required: Yes

## VerificationAttributes

The attributes and data values to use for MAC verification within AWS Payment Cryptography.

Type: [MacAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "KeyArn": "string",
  "KeyCheckValue": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### KeyArn

The keyARN of the encryption key that AWS Payment Cryptography uses for MAC verification.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### KeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

## Errors

### **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### **InternalServerErrorException**

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### **ResourceNotFoundException**

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 429

### **ValidationException**

The request was denied due to an invalid request error.

HTTP Status Code: 400

### **VerificationFailedException**

This request failed verification.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# VerifyPinData

Verifies pin-related data such as PIN and PIN Offset using algorithms including VISA PVV and IBM3624. For more information, see [Verify PIN data](#) in the *AWS Payment Cryptography User Guide*.

This operation verifies PIN data for user payment card. A card holder PIN data is never transmitted in clear to or from AWS Payment Cryptography. This operation uses PIN Verification Key (PVK) for PIN or PIN Offset generation and then encrypts it using PIN Encryption Key (PEK) to create an EncryptedPinBlock for transmission from AWS Payment Cryptography.

For information about valid keys for this operation, see [Understanding key attributes](#) and [Key types for specific data operations](#) in the *AWS Payment Cryptography User Guide*.

**Cross-account use:** This operation can't be used across different AWS accounts.

## Related operations:

- [GeneratePinData](#)
- [TranslatePinData](#)

## Request Syntax

```
POST /pindata/verify HTTP/1.1
Content-type: application/json

{
  "DukptAttributes": {
    "DukptDerivationType": "string",
    "KeySerialNumber": "string"
  },
  "EncryptedPinBlock": "string",
  "EncryptionKeyIdentifier": "string",
  "PinBlockFormat": "string",
  "PinDataLength": number,
  "PrimaryAccountNumber": "string",
  "VerificationAttributes": { ... },
  "VerificationKeyIdentifier": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### DukptAttributes

The attributes and values for the DUKPT encrypted PIN block data.

Type: [DukptAttributes](#) object

Required: No

### EncryptedPinBlock

The encrypted PIN block data that AWS Payment Cryptography verifies.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### EncryptionKeyIdentifier

The keyARN of the encryption key under which the PIN block data is encrypted. This key type can be PEK or BDK.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## PinBlockFormat

The PIN encoding format for pin data generation as specified in ISO 9564. AWS Payment Cryptography supports `ISO_Format_0` and `ISO_Format_3`.

The `ISO_Format_0` PIN block format is equivalent to the ANSI X9.8, VISA-1, and ECI-1 PIN block formats. It is similar to a VISA-4 PIN block format. It supports a PIN from 4 to 12 digits in length.

The `ISO_Format_3` PIN block format is the same as `ISO_Format_0` except that the fill digits are random values from 10 to 15.

Type: String

Valid Values: `ISO_FORMAT_0` | `ISO_FORMAT_3`

Required: Yes

## PinDataLength

The length of PIN being verified.

Type: Integer

Valid Range: Minimum value of 4. Maximum value of 12.

Required: No

## PrimaryAccountNumber

The Primary Account Number (PAN), a unique identifier for a payment credit or debit card that associates the card with a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

## VerificationAttributes

The attributes and values for PIN data verification.

Type: [PinVerificationAttributes](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

### [VerificationKeyIdentifier](#)

The keyARN of the PIN verification key.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 322.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:(key/[0-9a-zA-Z]{16,64}|alias/[a-zA-Z0-9/_-]+)$|^alias/[a-zA-Z0-9/_-]+`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "EncryptionKeyArn": "string",
  "EncryptionKeyCheckValue": "string",
  "VerificationKeyArn": "string",
  "VerificationKeyCheckValue": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### [EncryptionKeyArn](#)

The keyARN of the PEK that AWS Payment Cryptography uses for encrypted pin block generation.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### EncryptionKeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

### VerificationKeyArn

The keyARN of the PIN encryption key that AWS Payment Cryptography uses for PIN or PIN Offset verification.

Type: String

Length Constraints: Minimum length of 70. Maximum length of 150.

Pattern: `arn:aws:payment-cryptography:[a-z]{2}-[a-z]{1,16}-[0-9]+:[0-9]{12}:key/[0-9a-zA-Z]{16,64}`

### VerificationKeyCheckValue

The key check value (KCV) of the encryption key. The KCV is used to check if all parties holding a given key have the same key or to detect that a key has changed.

AWS Payment Cryptography computes the KCV according to the CMAC specification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9a-fA-F]+`

## Errors

### AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 403

### InternalServerErrorException

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

### ResourceNotFoundException

The request was denied due to an invalid resource error.

HTTP Status Code: 404

### ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 429

### ValidationException

The request was denied due to an invalid request error.

HTTP Status Code: 400

### VerificationFailedException

This request failed verification.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# Data Types

The Payment Cryptography Data Plane API contains several data types that various actions use. This section describes each data type in detail.

## Note

The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- [AmexCardSecurityCodeVersion1](#)
- [AmexCardSecurityCodeVersion2](#)
- [AsymmetricEncryptionAttributes](#)
- [CardGenerationAttributes](#)
- [CardHolderVerificationValue](#)
- [CardVerificationAttributes](#)
- [CardVerificationValue1](#)
- [CardVerificationValue2](#)
- [CryptogramAuthResponse](#)
- [CryptogramVerificationArpcMethod1](#)
- [CryptogramVerificationArpcMethod2](#)
- [DiscoverDynamicCardVerificationCode](#)
- [DukptAttributes](#)
- [DukptDerivationAttributes](#)
- [DukptEncryptionAttributes](#)
- [DynamicCardVerificationCode](#)
- [DynamicCardVerificationValue](#)
- [EmvEncryptionAttributes](#)
- [EncryptionDecryptionAttributes](#)
- [Ibm3624NaturalPin](#)



- [Ibm3624PinFromOffset](#)
- [Ibm3624PinOffset](#)
- [Ibm3624PinVerification](#)
- [Ibm3624RandomPin](#)
- [MacAlgorithmDukpt](#)
- [MacAlgorithmEmv](#)
- [MacAttributes](#)
- [PinData](#)
- [PinGenerationAttributes](#)
- [PinVerificationAttributes](#)
- [ReEncryptionAttributes](#)
- [SessionKeyAmex](#)
- [SessionKeyDerivation](#)
- [SessionKeyDerivationValue](#)
- [SessionKeyEmv2000](#)
- [SessionKeyEmvCommon](#)
- [SessionKeyMastercard](#)
- [SessionKeyVisa](#)
- [SymmetricEncryptionAttributes](#)
- [TranslationIsoFormats](#)
- [TranslationPinDataIsoFormat034](#)
- [TranslationPinDataIsoFormat1](#)
- [ValidationExceptionField](#)
- [VisaPin](#)
- [VisaPinVerification](#)
- [VisaPinVerificationValue](#)

# AmexCardSecurityCodeVersion1

Card data parameters that are required to generate a Card Security Code (CSC2) for an AMEX payment card.

## Contents

### CardExpiryDate

The expiry date of a payment card.

Type: String

Length Constraints: Fixed length of 4.

Pattern: [0-9]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# AmexCardSecurityCodeVersion2

Card data parameters that are required to generate a Card Security Code (CSC2) for an AMEX payment card.

## Contents

### CardExpiryDate

The expiry date of a payment card.

Type: String

Length Constraints: Fixed length of 4.

Pattern: [0-9]+

Required: Yes

### ServiceCode

The service code of the AMEX payment card. This is different from the Card Security Code (CSC).

Type: String

Length Constraints: Fixed length of 3.

Pattern: [0-9]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# AsymmetricEncryptionAttributes

Parameters for plaintext encryption using asymmetric keys.

## Contents

### PaddingType

The padding to be included with the data.

Type: String

Valid Values: PKCS1 | OAEP\_SHA1 | OAEP\_SHA256 | OAEP\_SHA512

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# CardGenerationAttributes

Card data parameters that are required to generate Card Verification Values (CVV/CVV2), Dynamic Card Verification Values (dCVV/dCVV2), or Card Security Codes (CSC).

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### AmexCardSecurityCodeVersion1

Card data parameters that are required to generate a Card Security Code (CSC2) for an AMEX payment card.

Type: [AmexCardSecurityCodeVersion1](#) object

Required: No

### AmexCardSecurityCodeVersion2

Card data parameters that are required to generate a Card Security Code (CSC2) for an AMEX payment card.

Type: [AmexCardSecurityCodeVersion2](#) object

Required: No

### CardHolderVerificationValue

Card data parameters that are required to generate a cardholder verification value for the payment card.

Type: [CardHolderVerificationValue](#) object

Required: No

### CardVerificationValue1

Card data parameters that are required to generate Card Verification Value (CVV) for the payment card.

Type: [CardVerificationValue1](#) object

Required: No

### **CardVerificationValue2**

Card data parameters that are required to generate Card Verification Value (CVV2) for the payment card.

Type: [CardVerificationValue2](#) object

Required: No

### **DynamicCardVerificationCode**

Card data parameters that are required to generate CDynamic Card Verification Code (dCVC) for the payment card.

Type: [DynamicCardVerificationCode](#) object

Required: No

### **DynamicCardVerificationValue**

Card data parameters that are required to generate CDynamic Card Verification Value (dCVV) for the payment card.

Type: [DynamicCardVerificationValue](#) object

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# CardHolderVerificationValue

Card data parameters that are required to generate a cardholder verification value for the payment card.

## Contents

### ApplicationTransactionCounter

The transaction counter value that comes from a point of sale terminal.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### UnpredictableNumber

A random number generated by the issuer.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 8.

Pattern: `[0-9a-fA-F]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# CardVerificationAttributes

Card data parameters that are required to verify Card Verification Values (CVV/CVV2), Dynamic Card Verification Values (dCVV/dCVV2), or Card Security Codes (CSC).

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### AmexCardSecurityCodeVersion1

Card data parameters that are required to generate a Card Security Code (CSC2) for an AMEX payment card.

Type: [AmexCardSecurityCodeVersion1](#) object

Required: No

### AmexCardSecurityCodeVersion2

Card data parameters that are required to verify a Card Security Code (CSC2) for an AMEX payment card.

Type: [AmexCardSecurityCodeVersion2](#) object

Required: No

### CardHolderVerificationValue

Card data parameters that are required to verify a cardholder verification value for the payment card.

Type: [CardHolderVerificationValue](#) object

Required: No

### CardVerificationValue1

Card data parameters that are required to verify Card Verification Value (CVV) for the payment card.

Type: [CardVerificationValue1](#) object

Required: No

### **CardVerificationValue2**

Card data parameters that are required to verify Card Verification Value (CVV2) for the payment card.

Type: [CardVerificationValue2](#) object

Required: No

### **DiscoverDynamicCardVerificationCode**

Card data parameters that are required to verify CDynamic Card Verification Code (dCVC) for the payment card.

Type: [DiscoverDynamicCardVerificationCode](#) object

Required: No

### **DynamicCardVerificationCode**

Card data parameters that are required to verify CDynamic Card Verification Code (dCVC) for the payment card.

Type: [DynamicCardVerificationCode](#) object

Required: No

### **DynamicCardVerificationValue**

Card data parameters that are required to verify CDynamic Card Verification Value (dCVV) for the payment card.

Type: [DynamicCardVerificationValue](#) object

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# CardVerificationValue1

Card data parameters that are required to verify CVV (Card Verification Value) for the payment card.

## Contents

### CardExpiryDate

The expiry date of a payment card.

Type: String

Length Constraints: Fixed length of 4.

Pattern: [0-9]+

Required: Yes

### ServiceCode

The service code of the payment card. This is different from Card Security Code (CSC).

Type: String

Length Constraints: Fixed length of 3.

Pattern: [0-9]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# CardVerificationValue2

Card data parameters that are required to verify Card Verification Value (CVV2) for the payment card.

## Contents

### CardExpiryDate

The expiry date of a payment card.

Type: String

Length Constraints: Fixed length of 4.

Pattern: [0-9]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# CryptogramAuthResponse

Parameters that are required for Authorization Response Cryptogram (ARPC) generation after Authorization Request Cryptogram (ARQC) verification is successful.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### ArpcMethod1

Parameters that are required for ARPC response generation using method1 after ARQC verification is successful.

Type: [CryptogramVerificationArpcMethod1](#) object

Required: No

### ArpcMethod2

Parameters that are required for ARPC response generation using method2 after ARQC verification is successful.

Type: [CryptogramVerificationArpcMethod2](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# CryptogramVerificationArpcMethod1

Parameters that are required for ARPC response generation using method1 after ARQC verification is successful.

## Contents

### AuthResponseCode

The auth code used to calculate APRC after ARQC verification is successful. This is the same auth code used for ARQC generation outside of AWS Payment Cryptography.

Type: String

Length Constraints: Fixed length of 4.

Pattern: [0-9a-fA-F]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# CryptogramVerificationArpcMethod2

Parameters that are required for ARPC response generation using method2 after ARQC verification is successful.

## Contents

### CardStatusUpdate

The data indicating whether the issuer approves or declines an online transaction using an EMV chip card.

Type: String

Length Constraints: Fixed length of 8.

Pattern: [0-9a-fA-F]+

Required: Yes

### ProprietaryAuthenticationData

The proprietary authentication data used by issuer for communication during online transaction using an EMV chip card.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 16.

Pattern: [0-9a-fA-F]+

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# DiscoverDynamicCardVerificationCode

Parameters that are required to generate or verify dCVC (Dynamic Card Verification Code).

## Contents

### ApplicationTransactionCounter

The transaction counter value that comes from the terminal.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### CardExpiryDate

The expiry date of a payment card.

Type: String

Length Constraints: Fixed length of 4.

Pattern: `[0-9]+`

Required: Yes

### UnpredictableNumber

A random number that is generated by the issuer.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 8.

Pattern: `[0-9a-fA-F]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DukptAttributes

Parameters that are used for Derived Unique Key Per Transaction (DUKPT) derivation algorithm.

## Contents

### DukptDerivationType

The key type derived using DUKPT from a Base Derivation Key (BDK) and Key Serial Number (KSN). This must be less than or equal to the strength of the BDK. For example, you can't use AES\_128 as a derivation type for a BDK of AES\_128 or TDES\_2KEY.

Type: String

Valid Values: TDES\_2KEY | TDES\_3KEY | AES\_128 | AES\_192 | AES\_256

Required: Yes

### KeySerialNumber

The unique identifier known as Key Serial Number (KSN) that comes from an encrypting device using DUKPT encryption method. The KSN is derived from the encrypting device unique identifier and an internal transaction counter.

Type: String

Length Constraints: Minimum length of 10. Maximum length of 24.

Pattern: [0-9a-fA-F]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# DukptDerivationAttributes

Parameters required for encryption or decryption of data using DUKPT.

## Contents

### KeySerialNumber

The unique identifier known as Key Serial Number (KSN) that comes from an encrypting device using DUKPT encryption method. The KSN is derived from the encrypting device unique identifier and an internal transaction counter.

Type: String

Length Constraints: Minimum length of 10. Maximum length of 24.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### DukptKeyDerivationType

The key type derived using DUKPT from a Base Derivation Key (BDK) and Key Serial Number (KSN). This must be less than or equal to the strength of the BDK. For example, you can't use AES\_128 as a derivation type for a BDK of AES\_128 or TDES\_2KEY

Type: String

Valid Values: TDES\_2KEY | TDES\_3KEY | AES\_128 | AES\_192 | AES\_256

Required: No

### DukptKeyVariant

The type of use of DUKPT, which can be for incoming data decryption, outgoing data encryption, or both.

Type: String

Valid Values: BIDIRECTIONAL | REQUEST | RESPONSE

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# DukptEncryptionAttributes

Parameters that are required to encrypt plaintext data using DUKPT.

## Contents

### KeySerialNumber

The unique identifier known as Key Serial Number (KSN) that comes from an encrypting device using DUKPT encryption method. The KSN is derived from the encrypting device unique identifier and an internal transaction counter.

Type: String

Length Constraints: Minimum length of 10. Maximum length of 24.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### DukptKeyDerivationType

The key type encrypted using DUKPT from a Base Derivation Key (BDK) and Key Serial Number (KSN). This must be less than or equal to the strength of the BDK. For example, you can't use AES\_128 as a derivation type for a BDK of AES\_128 or TDES\_2KEY

Type: String

Valid Values: TDES\_2KEY | TDES\_3KEY | AES\_128 | AES\_192 | AES\_256

Required: No

### DukptKeyVariant

The type of use of DUKPT, which can be incoming data decryption, outgoing data encryption, or both.

Type: String

Valid Values: BIDIRECTIONAL | REQUEST | RESPONSE

Required: No

## InitializationVector

An input used to provide the initial state. If no value is provided, AWS Payment Cryptography defaults it to zero.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: `(?:[0-9a-fA-F]{16}|[0-9a-fA-F]{32})`

Required: No

## Mode

The block cipher method to use for encryption.

The default is CBC.

Type: String

Valid Values: ECB | CBC

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DynamicCardVerificationCode

Parameters that are required to generate or verify Dynamic Card Verification Value (dCVV).

## Contents

### ApplicationTransactionCounter

The transaction counter value that comes from the terminal.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### TrackData

The data on the two tracks of magnetic cards used for financial transactions. This includes the cardholder name, PAN, expiration date, bank ID (BIN) and several other numbers the issuing bank uses to validate the data received.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 160.

Pattern: `[0-9a-fA-F]+`

Required: Yes

## UnpredictableNumber

A random number generated by the issuer.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 8.

Pattern: `[0-9a-fA-F]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DynamicCardVerificationValue

Parameters that are required to generate or verify Dynamic Card Verification Value (dCVV).

## Contents

### ApplicationTransactionCounter

The transaction counter value that comes from the terminal.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### CardExpiryDate

The expiry date of a payment card.

Type: String

Length Constraints: Fixed length of 4.

Pattern: `[0-9]+`

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### ServiceCode

The service code of the payment card. This is different from Card Security Code (CSC).

Type: String

Length Constraints: Fixed length of 3.

Pattern: [0-9]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# EmvEncryptionAttributes

Parameters for plaintext encryption using EMV keys.

## Contents

### MajorKeyDerivationMode

The EMV derivation mode to use for ICC master key derivation as per EMV version 4.3 book 2.

Type: String

Valid Values: EMV\_OPTION\_A | EMV\_OPTION\_B

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: [0-9a-fA-F]+

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN), a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: [0-9]+

Required: Yes

## SessionDerivationData

The derivation value used to derive the ICC session key. It is typically the application transaction counter value padded with zeros or previous ARQC value padded with zeros as per EMV version 4.3 book 2.

Type: String

Length Constraints: Fixed length of 16.

Pattern: `[0-9a-fA-F]+`

Required: Yes

## InitializationVector

An input used to provide the initial state. If no value is provided, AWS Payment Cryptography defaults it to zero.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: `(?:[0-9a-fA-F]{16}|[0-9a-fA-F]{32})`

Required: No

## Mode

The block cipher method to use for encryption.

Type: String

Valid Values: ECB | CBC

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)



- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# EncryptionDecryptionAttributes

Parameters that are required to perform encryption and decryption operations.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### Asymmetric

Parameters for plaintext encryption using asymmetric keys.

Type: [AsymmetricEncryptionAttributes](#) object

Required: No

### Dukpt

Parameters that are required to encrypt plaintext data using DUKPT.

Type: [DukptEncryptionAttributes](#) object

Required: No

### Emv

Parameters for plaintext encryption using EMV keys.

Type: [EmvEncryptionAttributes](#) object

Required: No

### Symmetric

Parameters that are required to perform encryption and decryption using symmetric keys.

Type: [SymmetricEncryptionAttributes](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Ibm3624NaturalPin

Parameters that are required to generate or verify Ibm3624 natural PIN.

## Contents

### DecimalizationTable

The decimalization table to use for IBM 3624 PIN algorithm. The table is used to convert the algorithm intermediate result from hexadecimal characters to decimal.

Type: String

Length Constraints: Fixed length of 16.

Pattern: [0-9]+

Required: Yes

### PinValidationData

The unique data for cardholder identification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: [0-9]+

Required: Yes

### PinValidationDataPadCharacter

The padding character for validation data.

Type: String

Length Constraints: Fixed length of 1.

Pattern: [0-9A-F]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Ibm3624PinFromOffset

Parameters that are required to generate or verify Ibm3624 PIN from offset PIN.

## Contents

### DecimalizationTable

The decimalization table to use for IBM 3624 PIN algorithm. The table is used to convert the algorithm intermediate result from hexadecimal characters to decimal.

Type: String

Length Constraints: Fixed length of 16.

Pattern: [0-9]+

Required: Yes

### PinOffset

The PIN offset value.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 12.

Pattern: [0-9]+

Required: Yes

### PinValidationData

The unique data for cardholder identification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: [0-9]+

Required: Yes

### PinValidationDataPadCharacter

The padding character for validation data.

Type: String

Length Constraints: Fixed length of 1.

Pattern: [0-9A-F]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Ibm3624PinOffset

Parameters that are required to generate or verify Ibm3624 PIN offset PIN.

## Contents

### DecimalizationTable

The decimalization table to use for IBM 3624 PIN algorithm. The table is used to convert the algorithm intermediate result from hexadecimal characters to decimal.

Type: String

Length Constraints: Fixed length of 16.

Pattern: `[0-9]+`

Required: Yes

### EncryptedPinBlock

The encrypted PIN block data. According to ISO 9564 standard, a PIN Block is an encoded representation of a payment card Personal Account Number (PAN) and the cardholder Personal Identification Number (PIN).

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PinValidationData

The unique data for cardholder identification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: `[0-9]+`

Required: Yes



## PinValidationDataPadCharacter

The padding character for validation data.

Type: String

Length Constraints: Fixed length of 1.

Pattern: `[0-9A-F]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Ibm3624PinVerification

Parameters that are required to generate or verify Ibm3624 PIN verification PIN.

## Contents

### DecimalizationTable

The decimalization table to use for IBM 3624 PIN algorithm. The table is used to convert the algorithm intermediate result from hexadecimal characters to decimal.

Type: String

Length Constraints: Fixed length of 16.

Pattern: [0-9]+

Required: Yes

### PinOffset

The PIN offset value.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 12.

Pattern: [0-9]+

Required: Yes

### PinValidationData

The unique data for cardholder identification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: [0-9]+

Required: Yes

### PinValidationDataPadCharacter

The padding character for validation data.

Type: String

Length Constraints: Fixed length of 1.

Pattern: [0-9A-F]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Ibm3624RandomPin

Parameters that are required to generate or verify Ibm3624 random PIN.

## Contents

### DecimalizationTable

The decimalization table to use for IBM 3624 PIN algorithm. The table is used to convert the algorithm intermediate result from hexadecimal characters to decimal.

Type: String

Length Constraints: Fixed length of 16.

Pattern: [0-9]+

Required: Yes

### PinValidationData

The unique data for cardholder identification.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 16.

Pattern: [0-9]+

Required: Yes

### PinValidationDataPadCharacter

The padding character for validation data.

Type: String

Length Constraints: Fixed length of 1.

Pattern: [0-9A-F]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# MacAlgorithmDukpt

Parameters required for DUKPT MAC generation and verification.

## Contents

### DukptKeyVariant

The type of use of DUKPT, which can be MAC generation, MAC verification, or both.

Type: String

Valid Values: BIDIRECTIONAL | REQUEST | RESPONSE

Required: Yes

### KeySerialNumber

The unique identifier known as Key Serial Number (KSN) that comes from an encrypting device using DUKPT encryption method. The KSN is derived from the encrypting device unique identifier and an internal transaction counter.

Type: String

Length Constraints: Minimum length of 10. Maximum length of 24.

Pattern: [0-9a-fA-F]+

Required: Yes

### DukptDerivationType

The key type derived using DUKPT from a Base Derivation Key (BDK) and Key Serial Number (KSN). This must be less than or equal to the strength of the BDK. For example, you can't use AES\_128 as a derivation type for a BDK of AES\_128 or TDES\_2KEY.

Type: String

Valid Values: TDES\_2KEY | TDES\_3KEY | AES\_128 | AES\_192 | AES\_256

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# MacAlgorithmEmv

Parameters that are required for EMV MAC generation and verification.

## Contents

### MajorKeyDerivationMode

The method to use when deriving the master key for EMV MAC generation or verification.

Type: String

Valid Values: EMV\_OPTION\_A | EMV\_OPTION\_B

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: [0-9a-fA-F]+

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN), a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: [0-9]+

Required: Yes

### SessionKeyDerivationMode

The method of deriving a session key for EMV MAC generation or verification.



Type: String

Valid Values: EMV\_COMMON\_SESSION\_KEY | EMV2000 | AMEX |  
MASTERCARD\_SESSION\_KEY | VISA

Required: Yes

### SessionKeyDerivationValue

Parameters that are required to generate session key for EMV generation and verification.

Type: [SessionKeyDerivationValue](#) object

**Note:** This object is a Union. Only one member of this object can be specified or returned.

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# MacAttributes

Parameters that are required for DUKPT, HMAC, or EMV MAC generation or verification.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

## Algorithm

The encryption algorithm for MAC generation or verification.

Type: String

Valid Values: ISO9797\_ALGORITHM1 | ISO9797\_ALGORITHM3 | CMAC | HMAC\_SHA224 | HMAC\_SHA256 | HMAC\_SHA384 | HMAC\_SHA512

Required: No

## DukptCmac

Parameters that are required for MAC generation or verification using DUKPT CMAC algorithm.

Type: [MacAlgorithmDukpt](#) object

Required: No

## DukptIso9797Algorithm1

Parameters that are required for MAC generation or verification using DUKPT ISO 9797 algorithm1.

Type: [MacAlgorithmDukpt](#) object

Required: No

## DukptIso9797Algorithm3

Parameters that are required for MAC generation or verification using DUKPT ISO 9797 algorithm3.

Type: [MacAlgorithmDukpt](#) object

Required: No

### **EmvMac**

Parameters that are required for MAC generation or verification using EMV MAC algorithm.

Type: [MacAlgorithmEmv](#) object

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# PinData

Parameters that are required to generate, translate, or verify PIN data.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### PinOffset

The PIN offset value.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 12.

Pattern: [0-9]+

Required: No

### VerificationValue

The unique data to identify a cardholder. In most cases, this is the same as cardholder's Primary Account Number (PAN). If a value is not provided, it defaults to PAN.

Type: String

Length Constraints: Minimum length of 4. Maximum length of 12.

Pattern: [0-9]+

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# PinGenerationAttributes

Parameters that are required for PIN data generation.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### **Ibm3624NaturalPin**

Parameters that are required to generate or verify Ibm3624 natural PIN.

Type: [Ibm3624NaturalPin](#) object

Required: No

### **Ibm3624PinFromOffset**

Parameters that are required to generate or verify Ibm3624 PIN from offset PIN.

Type: [Ibm3624PinFromOffset](#) object

Required: No

### **Ibm3624PinOffset**

Parameters that are required to generate or verify Ibm3624 PIN offset PIN.

Type: [Ibm3624PinOffset](#) object

Required: No

### **Ibm3624RandomPin**

Parameters that are required to generate or verify Ibm3624 random PIN.

Type: [Ibm3624RandomPin](#) object

Required: No

## VisaPin

Parameters that are required to generate or verify Visa PIN.

Type: [VisaPin](#) object

Required: No

## VisaPinVerificationValue

Parameters that are required to generate or verify Visa PIN Verification Value (PVV).

Type: [VisaPinVerificationValue](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# PinVerificationAttributes

Parameters that are required for PIN data verification.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### Ibm3624Pin

Parameters that are required to generate or verify Ibm3624 PIN.

Type: [Ibm3624PinVerification](#) object

Required: No

### VisaPin

Parameters that are required to generate or verify Visa PIN.

Type: [VisaPinVerification](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# ReEncryptionAttributes

Parameters that are required to perform reencryption operation.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### Dukpt

Parameters that are required to encrypt plaintext data using DUKPT.

Type: [DukptEncryptionAttributes](#) object

Required: No

### Symmetric

Parameters that are required to encrypt data using symmetric keys.

Type: [SymmetricEncryptionAttributes](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SessionKeyAmex

Parameters to derive session key for an Amex payment card.

## Contents

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN) of the cardholder. A PAN is a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SessionKeyDerivation

Parameters to derive a session key for Authorization Response Cryptogram (ARQC) verification.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### Amex

Parameters to derive session key for an Amex payment card for ARQC verification.

Type: [SessionKeyAmex](#) object

Required: No

### Emv2000

Parameters to derive session key for an Emv2000 payment card for ARQC verification.

Type: [SessionKeyEmv2000](#) object

Required: No

### EmvCommon

Parameters to derive session key for an Emv common payment card for ARQC verification.

Type: [SessionKeyEmvCommon](#) object

Required: No

### Mastercard

Parameters to derive session key for a Mastercard payment card for ARQC verification.

Type: [SessionKeyMastercard](#) object

Required: No

## Visa

Parameters to derive session key for a Visa payment cardfor ARQC verification.

Type: [SessionKeyVisa](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SessionKeyDerivationValue

Parameters to derive session key value using a MAC EMV algorithm.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### ApplicationCryptogram

The cryptogram provided by the terminal during transaction processing.

Type: String

Length Constraints: Fixed length of 16.

Pattern: [0-9a-fA-F]+

Required: No

### ApplicationTransactionCounter

The transaction counter that is provided by the terminal during transaction processing.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: [0-9a-fA-F]+

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SessionKeyEmv2000

Parameters to derive session key for an Emv2000 payment card for ARQC verification.

## Contents

### ApplicationTransactionCounter

The transaction counter that is provided by the terminal during transaction processing.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN) of the cardholder. A PAN is a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# SessionKeyEmvCommon

Parameters to derive session key for an Emv common payment card for ARQC verification.

## Contents

### ApplicationTransactionCounter

The transaction counter that is provided by the terminal during transaction processing.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN) of the cardholder. A PAN is a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SessionKeyMastercard

Parameters to derive session key for Mastercard payment card for ARQC verification.

## Contents

### ApplicationTransactionCounter

The transaction counter that is provided by the terminal during transaction processing.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 4.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN) of the cardholder. A PAN is a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

## UnpredictableNumber

A random number generated by the issuer.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 8.

Pattern: `[0-9a-fA-F]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SessionKeyVisa

Parameters to derive session key for Visa payment card for ARQC verification.

## Contents

### PanSequenceNumber

A number that identifies and differentiates payment cards with the same Primary Account Number (PAN).

Type: String

Length Constraints: Fixed length of 2.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PrimaryAccountNumber

The Primary Account Number (PAN) of the cardholder. A PAN is a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: `[0-9]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SymmetricEncryptionAttributes

Parameters required to encrypt plaintext data using symmetric keys.

## Contents

### Mode

The block cipher method to use for encryption.

Type: String

Valid Values: ECB | CBC | CFB | CFB1 | CFB8 | CFB64 | CFB128 | OFB

Required: Yes

### InitializationVector

An input used to provide the initial state. If no value is provided, AWS Payment Cryptography defaults it to zero.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: (?:[0-9a-fA-F]{16}|[0-9a-fA-F]{32})

Required: No

### PaddingType

The padding to be included with the data.

Type: String

Valid Values: PKCS1 | OAEP\_SHA1 | OAEP\_SHA256 | OAEP\_SHA512

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# TranslationIsoFormats

Parameters that are required for translation between ISO9564 PIN block formats 0,1,3,4.

## Contents

### Important

This data type is a UNION, so only one of the following members can be specified when used or returned.

### IsoFormat0

Parameters that are required for ISO9564 PIN format 0 translation.

Type: [TranslationPinDataIsoFormat034](#) object

Required: No

### IsoFormat1

Parameters that are required for ISO9564 PIN format 1 translation.

Type: [TranslationPinDataIsoFormat1](#) object

Required: No

### IsoFormat3

Parameters that are required for ISO9564 PIN format 3 translation.

Type: [TranslationPinDataIsoFormat034](#) object

Required: No

### IsoFormat4

Parameters that are required for ISO9564 PIN format 4 translation.

Type: [TranslationPinDataIsoFormat034](#) object

Required: No



## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# TranslationPinDataIsoFormat034

Parameters that are required for translation between ISO9564 PIN format 0,3,4 translation.

## Contents

### PrimaryAccountNumber

The Primary Account Number (PAN) of the cardholder. A PAN is a unique identifier for a payment credit or debit card and associates the card to a specific account holder.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 19.

Pattern: [0-9]+

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# TranslationPinDataIsoFormat1

Parameters that are required for ISO9564 PIN format 1 translation.

## Contents

The members of this exception structure are context-dependent.

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# ValidationExceptionField

The request was denied due to an invalid request error.

## Contents

### message

The request was denied due to an invalid request error.

Type: String

Required: Yes

### path

The request was denied due to an invalid request error.

Type: String

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# VisaPin

Parameters that are required to generate or verify Visa PIN.

## Contents

### PinVerificationKeyIndex

The value for PIN verification index. It is used in the Visa PIN algorithm to calculate the PVV (PIN Verification Value).

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 9.

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# VisaPinVerification

Parameters that are required to generate or verify Visa PIN.

## Contents

### PinVerificationKeyIndex

The value for PIN verification index. It is used in the Visa PIN algorithm to calculate the PVV (PIN Verification Value).

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 9.

Required: Yes

### VerificationValue

Parameters that are required to generate or verify Visa PVV (PIN Verification Value).

Type: String

Length Constraints: Minimum length of 4. Maximum length of 12.

Pattern: `[0-9]+`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# VisaPinVerificationValue

Parameters that are required to generate or verify Visa PVV (PIN Verification Value).

## Contents

### EncryptedPinBlock

The encrypted PIN block data to verify.

Type: String

Length Constraints: Minimum length of 16. Maximum length of 32.

Pattern: `[0-9a-fA-F]+`

Required: Yes

### PinVerificationKeyIndex

The value for PIN verification index. It is used in the Visa PIN algorithm to calculate the PVV (PIN Verification Value).

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 9.

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)