Travel Rule On-Chain

Avoid errors when performing 3000+ transfers by complying with the FinCEN Travel Rule. Suggest Edits

Due to FinCEN guidance regarding transfers of 3,000 or more, Circle APIs may require you to share information about your end-user to prevent the transfer from failing. According to FinCEN guidance regarding applicability of the Travel Rule to ERC-20 transfers (including USDC) on Ethereum of 3,000 or more (or matching cryptocurrency), the originating institution must transmit KYC information about the originator of a transaction to the beneficiary institution.

- For more information on these requirements and how data is securely collected and shared, please see ourFAQ
- .

Sending Transfers (Non-Business Account)

When sending non-business account transfers on the Ethereum or Bitcoin chain of 3,000 in value or more, the originator's identity will be required. If you are sending a transfer on behalf of your end-user, you need to provide the identity of your end-user. Otherwise, you need to provide your company's identity.

In technical terms, the originator's identity is required for POST /v1/transfers when:

1. destination.type: "blockchain"
2. destination.chain: "ETH" or "BTC"
3. amount.amount
4. = 3,000 in value

If the POST /v1/transfers request requires an originator's identity and the transfer does not contain an originator's identity, the transfer will fail. You will receive a webhook notification with data.status: fail , riskEvaluation.decision: denied , and riskEvaluation.reason: 3220 .

Sending Transfers (Business Account)

Although first-party transfers must comply with the Travel Rule, you do not need to provide your company's identity each time you make a business account transfer POST /v1/businessAccount/transfers . Instead, Circle will use your company's identity stored on file and provide it to the receiving institution as required.

Receiving Transfers

If yours is a certain type of regulated financial institution, you are subject to additional Travel Rule compliance requirements when receiving transfers. In this case, we will provide you with the originator's identity for webhook notifications of type transfer in addition to GET /v1/transfers and GET /v1/transfers/{id} . However, originator data will not be associated with a transfer until it is marked with a status of complete.

- At this time, not all sending institutions are participating in the on-chain Travel Rule requirement. Therefore, not all transfers valued at 3,000 or more will have the originating person associated.
- If you are a financial institution and do not see the originator's identity, please reach out to a Circle representative.

Non-bank financial institutions will not see the originator's identity for received transfers.

Identities Schema

Field Type Definition Identities.type [REQUIRED] String [enum]: individual , business individual - A uniquely distinguishable individual.

business - Any entity other than a natural person that can establish a permanent customer relationship with an affected entity or otherwise own property. This can include companies, foundations, anstalt, partnerships, associations and other relevantly similar entities Identities.name [REQUIRED] String [max length 1024] Full name of the originator. Identities.addresses.line1 [REQUIRED] String [max length 1024] Line one of the street address.

Note: If the originator is a business, this is the principal place of business address. If the originator is an individual, this is the current residential address. identities.addresses.line2 String [max length 1024] Line two of the street address.

Identities.addresses.city [REQUIRED] String [max length 1024] City portion of the address. identities.addresses.district String [max length 16] State / County / Province / Region portion of the address. If the country is US or Canada district is required and should use the two-letter code for the subdivision. Identities.addresses.country [REQUIRED] String [max length 2] Country portion of the address. Formatted as a two-letter country code specified in ISO 3166-1 alpha-2. identities.addresses.postalCode [REQUIRED] String [max length 16] Postal / ZIP code of the address.

Example: POST /v1/transfers

Request { "idempotencyKey": "ba943ff1-ca16-49b2-ba55-1057e70ca5c7", "source": { "type": "wallet", "id": "12345", "identities": [{ "type": "individual", "name": "Satoshi Nakamoto", "addresses": [{ "line1": "100 Money Street", "line2": "Suite 1", "city": "Boston", "district": "MA", "country": "US", "postalCode": "01234" }] }] }, "destination": { "type": "blockchain", "address": "0x8381470ED67C3802402dbbFa0058E8871F017A6F", "chain": "ETH" }, "amount": { "amount": "3000.00", "currency": "USD" } } Response { "data": [{ "id": "b36cbf12-6ed1-47ed-9eb9-5874f8991ca8", "source": { "type": "wallet", "id": "12345", "identities": [{ "type": "individual", "name": "Satoshi Nakamoto", "addresses": [{ "line1": "100 Money Street", "line2": "Suite 1", "city": "Boston", "district": "MA", "country": "US", "postalCode": "01234" }] }] }, "destination": { "type": "blockchain", "address": "0x8381470ED67C3802402dbbFa0058E8871F017A6F", "chain": "ETH" }, "amount": { "amount": "3000.00", "currency": "USD" }, "transactionHash": "0x4cebf8f90c9243a23c77e4ae20df691469e4b933b295a73376292843968f7a63", "status": "pending", "createDate": "2020-04-10T02:13:30.000Z" }] }

Example: GET /v1/transfers?returnIdentities=true

returnIdentities[boolean]: Specify if you would like to see identities in the response. Restricts maximum returned items to 5. By default returnIdentities is false, resulting in the response not returning data.source.identities .

Response { "data": [{ "id": "b36cbf12-6ed1-47ed-9eb9-5874f8991ca8", "source": { "type": "wallet", "id": "12345", "identities": [{ "type": "individual", "name": "Satoshi Nakamoto", "addresses": [{ "line1": "100 Money Street", "line2": "Suite 1", "city": "Boston", "district": "MA", "country": "US", "postalCode": "01234" }] }] }, "destination": { "type": "blockchain", "address": "0x8381470ED67C3802402dbbFa0058E8871F017A6F", "chain": "ETH" }, "amount": { "amount": "3000.00", "currency": "USD" }, "transactionHash": "0x4cebf8f90c9243a23c77e4ae20df691469e4b933b295a73376292843968f7a63", "status": "pending", "createDate": "2020-04-10T02:13:30.000Z" }] }

Example: GET /v1/transfers/{id}?returnIdentities=true

returnIdentities[boolean]: Specify if you would like to see identities in the response. Restricts maximum returned items to 5. By default returnIdentities is false, resulting in the response not returning data.source.identities .

Response { "data": { "id": "b36cbf12-6ed1-47ed-9eb9-5874f8991ca8", "source": { "type": "wallet", "id": "12345", "identities": [{ "type": "individual", "name": "Satoshi Nakamoto", "addresses": [{ "line1": "100 Money Street", "line2": "Suite 1", "city": "Boston", "district": "MA", "country": "US", "postalCode": "01234" }] }] }, "destination": { "type": "blockchain", "address": "0x8381470ED67C3802402dbbFa0058E8871F017A6F", "chain": "ETH" }, "amount": { "amount": "3000.00", "currency": "USD" }, "transactionHash": "0x4cebf8f90c9243a23c77e4ae20df691469e4b933b295a73376292843968f7a63", "status": "pending", "createDate": "2020-04-10T02:13:30.000Z" } }

Example: Webhook Notification

Payload { "clientId": "c60d2d5b-203c-45bb-9f6e-93641d40a599", "notificationType": "transfers", "transfer": { "id": "b36cbf12-6ed1-47ed-9eb9-5874f8991ca8", "source": { "type": "wallet", "id": "12345", "identities": [{ "type": "individual", "name": "Satoshi Nakamoto", "addresses": [{ "line1": "100 Money Street", "line2": "Suite 1", "city": "Boston", "district": "MA", "country": "US", "postalCode": "01234" }] }] }, "destination": { "type": "blockchain", "address": "0x8381470ED67C3802402dbbFa0058E8871F017A6F", "chain": "ETH" }, "amount": { "amount": "3000.00", "currency": "USD" }, "transactionHash": "0x4cebf8f90c9243a23c77e4ae20df691469e4b933b295a73376292843968f7a63", "status": "pending", "createDate": "2020-04-10T02:13:30.000Z" } } Updated5 months ago