API Reference

The Worldcoin Developer Portal offers an API to enable easy verification of World IDs and relevant credentials.

Base URL

https://developer.worldcoin.org Copy Copied!

All requests to the Developer Portal API must include aContent-Type: application/json header, aUser-Agent header, and a valid JSON body.

Verify Proof

POST

/api/v1/verify/{app_id}

Enables you to verify a World ID proof for aCloud action. To ensure sybil-resistance, by default, a single person can only verify once for every action. The number of allowed verifications for a single user can be configured in the Developer Portal.

Request Body

- Name
- nullifier_hash
- Type
- string
- Required
- REQUIRED
- Description
- The unique user identifier (called the nullifier hash in the ZKP), as provided by IDKit. See IDKit response
- for details.
- Name
- proof
- Type
- string
- Required
- REQUIRED
- Description
- The zero-knowledge proof, as provided by IDKit. See IDKit response
- for details.
- Name
- merkle_root
- Type
- string
- Required
- REQUIRED
- Description
- Part of the ZKP, the hash of the Merkle root that proves membership to the set of credentials. As provided by IDKit. See IDKit response
- for details.
- Name
- verification_level
- Type
- string
- Required
- REQUIRED
- Description
- The verification level, as provided by IDKit. See IDKit response
- for details.
- Not required if acredential_type
- is provided. If both are provided,verification_level
- will take precedence.
- Name
- credential_type
- Type
- string
- Deprecated
- DEPRECATED
- Description
- The credential type, as provided by IDKit. See IDKit response
- for details.
- This parameter is deprecated and will be removed in a future release, but is still supported for versions of IDKit <v1.0.0.
- Name
- action
- Type
- string
- Required
- REQUIRED
- Description
- Same action identifier as passed to IDKit.
- Name
- signal
- Type
- string:""
- Description
- Same signal as passed to IDKit. If not provided, defaults to an empty string.

Possible Responses

- 200 OK
-
  - The proof was successfully verified.
- 400 Bad Request
-
  - The proof was invalid or the user has already verified for this action.

Request

cURL JavaScript POST

/api/v1/verify/{app_id} curl -X

POST

"/api/v1/verify/{app_id}" \ -H

"Content-Type: application/json" \ -d

'{ "nullifier_hash": "0x2bf8406809dcefb1486dadc96c0a897db9bab002053054cf64272db512c6fbd8", "merkle_root": "0x2264a66d162d7893e12ea8e3c072c51e785bc085ad655f64c10c1a61e00f0bc2", "proof": "0x1aa8b8f3b2d2de5ff452c0e1a83e29d6bf46fb83ef35dc5957121ff3d3698a1119090fb...", "verification_level": "orb", "action": "my_action", "signal": "my_signal" }' Copy Copied!

Response

200 OK 400 Invalid Proof 400 Invalid Merkle Root 400 Invalid Credential Type 400 User Exceeded Maximum Verifications 400 User Already Verified { "success" :

true , "action" :

"my_action" , "nullifier_hash" :

"0x2bf8406809dcefb1486dadc96c0a897db9bab002053054cf64272db512c6fbd8" , "created_at" :

"2023-02-18T11:20:39.530041+00:00" } Copy Copied!

Get Action Metadata

GET

/api/v1/precheck/{app_id}

Enables fetching information about a particular app to determine eligibility for verification. This endpoint is also used by the World ID Kiosk, Sign in with World ID, and World App to show metadata about the action being verified. This endpoint will only return information for active actions. Primarily intended for internal use.

Request Body

- Name
- action
- Type
- string:""
- Required
- REQUIRED
- Description
- The action to check. This is the same action that was passed to IDKit. Defaults to an empty string for Sign in with World ID.
- Name
- nullifier_hash
- Type
- string:""
- Description
- The ZKP's nullifier hash, as provided by IDKit. See IDKit response
- for details. Defaults to an empty string.
- Name
- external_nullifier
- Type
- string
- Deprecated
- DEPRECATED
- Description
- The ZKP's external nullifier. This parameter is deprecated and will be removed in a future release, and should no longer be used. Instead, pass the action
- parameter.

Request

cURL JavaScript GET

/api/v1/precheck/{app_id} curl -X

POST

"/api/v1/precheck/{app_id}" \ -H

"Content-Type: application/json" \ -d

'{ "action": "my_custom_action", "nullifier_hash": "0x2bf8406809dcefb1486dadc96c0a897db9bab002053054cf64272db512c6fbd8", "external_nullifier": "0x00949dd9a8c5b600304d010ce3a3cf012352070ae4b77504e17af77ee894cda" }' Copy Copied!

Response

- Name
- __typename
- Type
- string:app
- Description
- Always returns app
- .
- Name
- id
- Type
- string
- Description
- The app_id
- from the Developer Portal.
- Name
- engine
- Type
- "cloud" | "on-chain"
- Description
- Whether the app is configured for use with Cloud
- or On-Chain
- proof verifications.
- Name
- is_staging
- Type
- boolean
- Description
- Whether the app is a staging app. If true
- , the Worldcoin Simulator
- must be used to verify.
- Name
- is_verified
- Type
- string
- Description
- Whether the app has been verified by Worldcoin.
- Name
- name
- Type
- string
- Description
- The app's name as configured in the Developer Portal.
- Name
- verified_app_logo
- Type
- string

- Description
- The URL of the app's logo. Only returned if the app has been verified, otherwise returns an empty string.
- Name
- is_sign_in
- Type
- boolean
- Description
- Whether the action request is for Sign in with World ID.
- Name
- can_user_verify
- Type
- "yes" | "no" | "on-chain" | "undetermined"
- Description
- If the user is eligible to verify for this action.
-
  - "yes"
-
  - : The user has not reached the maximum number of verifications for this action based on thenullifier_hash
-
  - provided. Always returned whenis_sign_in
-
  - istrue
-
  - .
-
  - "no"
-
  - : The user has reached the maximum number of verifications for this action based on thenullifier_hash
-
  - provided.
-
  - "undetermined"
-
  - : A nullifier_hash was not provided, so the user's eligibility cannot be determined.
-
  - "on-chain"
-
  - : The app is configured forOn-Chain
-
  - verifications, so the user's eligibility cannot be determined. Verification eligibility is determined solely by the application's smart contract.
- Name
- action
- Type
- JSON
- Description
- Information about the action being verified.
-
  - name
-
  - : The action's human-readable name as configured in the Developer Portal.
-
  - action
-
  - : The action identifier as configured in the Developer Portal. Typically a slugified version of the action's name.
-
  - description
-
  - : The action's description as configured in the Developer Portal.
-
  - max_verifications
-
  - : The maximum number of verifications allowed for this action. Will be0
-
  - if the action is configured for unlimited verifications.
-
  - max_accounts_per_user
-
  - : The maximum number of accounts allowed per user for Sign in with World ID. Will always be1
-
  - .
-
  - external_nullifier
-
  - : The action's external nullifier.Intended for internal use only.
-
  - status
-
  - : Will returnactive
-
  - if the action is active, otherwise will returninactive
-
  - .
-
  - __typename
-
  - : Always returnsaction
-
  - .

Response

200 OK 400 Required 400 Action Inactive 404 Not Found 405 Method Not Allowed { "id" :

"app_staging_4cfd049031b0da1e8b62084b09a9f430" , "is_staging" :

true , "is_verified" :

false , "name" :

"Default App" , "verified_app_logo" :

"" , "engine" :

"cloud" , "__typename" :

"app" , "sign_in_with_world_id" :

true , "can_user_verify" :

"yes" , "action" : { "external_nullifier" :

"0x00949dd9a8c5b600304d010ce3a3cf012352070ae4b77504e17af77ee894cda" , "name" :

"My action" , "action" :

"my_custom_action" , "description" :

"My action" , "max_verifications" :

0 , "max_accounts_per_user" :

1 , "__typename" :

"action" } } Copy Copied!

Get JWK Keys

GET

/api/v1/jwks

This endpoint lets you retrieve the JWKs (public keys) used to verify the signature on JSON web tokens that authenticate a verification request from the Developer Portal. This verification method is only used if you are using theHosted page user interface. cURL JavaScript curl /api/v1/jwks Copy Copied!

Response

{ "keys" : [{ "e" :

"AQAB" , "n" :

"09ETz2k4_9IbDBYK_Tcr6DzbDdJPeqIgvoeUvXNVjNU8mYzFbhdqh8jRH80FwtuoFqyw5oyuG9ILHxfGaG_SeutPWSxBsquIXhxTnTAx2i8HtF0i2toMuvsEtiAjQ3hD4_w2xInBVOO98WAGcNA_UgWAG2D Au8m1V3U_6404m4RV_wIWGPnhHfG57VTkkqjgrnFKGUDniG-VMJs-WFX4VlQRvy2z1A5nQsmYpobK_clGyV0D0i5P1A_lmWGDEXBLSjEW9zH_o0d2DQ" , "kty" :

"RSA" , "kid" :

"jwk_8934bcc47ec5b86dd490cc2a46f18a5e" }] } Copy Copied!

Authenticated Endpoints

You can generate API Keys on theMy Team page of the Developer Portal.

- API keys are intended for use by third-party applications and for server-to-server communication. They are long-lived and can be revoked at any time.
- API keys are scoped to a team and have full permissions for any actions as any user of the Developer Portal.
- Once you have an API key, use it as the value of theAuthorization
- header in your requests. For example:

Authorization: Bearer API_KEY Copy Copied!

GraphQL

POST

/v1/graphql

Interaction with the Developer Portal API is mostly done through a GraphQL endpoint, which enables retrieving information and interacting with any of the API objects. You can read more aboutqueries andmutations for GraphQL to help you construct your query.

Example Queries

Listing Apps Listing nullifiers for action query

MyApps { app { id name } } Copy Copied!

Request

cURL JavaScript POST

/api/v1/graphql curl -X

POST

"/v1/graphql" \ -H

"Content-Type: application/json" \ -H

"Authorization: Bearer WORLD_ID_TOKEN" \ -d

'{ "query": "YOUR_GRAPHQL_QUERY_HERE" }' Copy Copied!