1. IdentityHub API

1.1 Register Hub User by Public Key

Interface Address	/hub/regiterByIdPublicKey
Description	Register by Public key. This function allows users define the ID by themselves. If successful, returns the user's ID in the Hub.

Interface request parameter

No.	Parameter	Type	Required	Description
1		RegisterHubReq	Y	Wrapper class

RegisterHubReq

No.	Parameter	Type	Required	Description
1	id	String	N	Self-defined Hub ID
2	publicKey	String	Y	Public key
3	cryptoType	String	Y	Encryption Algorithm: ECDSA

Interface response parameter

No.	Parameter	Type	Description
1		RegisterHubResult	Wrapper class of the returned value

RegisterHubResult

No.	Parameter	Type	Description
1	success	Boolean	Whether it is successful
2	uid	String	ID in the Hub
3	message	String	Result description

1.2 Save Resource

Interface Address	/hub/saveResource
Description	Store the resource to the Hub. If the user stores it himself, the uid and the ownerUid should be the same. At this time, there is no need to create permissions to call directly; if the issuer stores it for the user after issuing the credentials, the uid should be the issuer, and the ownerUid should be the user. At this time, the user must have created "WRITE" permissions for it, otherwise the storage will fail.

Interface request parameter

No.	Parameter	Type	Required	Description
1		SaveResourceReq	Y	Wrapper class

SaveResourceReq

No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	content	String	Y	Resource information
3	url	String	N	The path of the resource storage. Null when it is

				user self-storage; required when the issuer stores it for the user
4	ownerUid	String	Y	Resource owner

5	grant	String	Y	WRITE means add;
				UPDATE means update

6	key	String	Y	Key
7	sign	String	Y	Signature

Interface response parameter

No.	Parameter	Type	Description
1		SaveResourceResp	Wrapper class

SaveResourceResp

No.	Parameter	Type	Description
1	url	String	The path of the stored resource
2	encryptKey	String	Null when the issuer stores it for users; if it is the user self-storage, returns the KeyA encrypted by the user's Public key, and the user's private key decrypts KeyA to derive the plaintext Key. The resource stored in the Hub is encrypted by Key with AES-ECB algorithm.

1.3 Get Resource

Interface Address	/hub/getResource
Description	Access the Hub and read the specified resource. Users have direct access, third parties need to obtain authorization from the user to access.

Interface request parameter

No.	Parameter	Type	Required	Description
1		QueryResourceReq	Y	Wrapper class

QueryResourceReq

No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	url	String	Y	The path of the stored resource
3	sign	String	Y	Signature

Interface response parameter

No.	Parameter	Type	Description
1	ResourceInfo	ResourceInfo	Wrapper class

ResourceInfo

No.	Parameter	Type	Description
1	content	String	Content of cryptographic resource
2	key	String	For the ciphertext key, you need to use your own

			private key to decrypt it first, and then decrypt the content.

1.4 Delete Resource

Interface Address	/hub/deleteResource			
Description	The owner of the resource can call this function to delete certain resource within the Hub.			
Interface request parameter				
No.	Parameter	Type	Required	Description
1		DeleteResourceReq	Y	Wrapper class
DeleteResourceReq				
No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	url	String	Y	The path of the stored resource
3	sign	String	Y	Signature
Interface response parameter				
No.	Parameter	Type	Description	
1		Boolean	Return true if success, return false if failed	

1.5 Create Permissions

Interface Address	/hub/createPerm			
Description	Resource owner creates the permissions of accessing to the resource in the Hub for third parties. Permissions WRITE means store resource, UPDATE means update resource, READ means read resource. An authorization can only be accessed once, and an authorization with the same uid and the same permission cannot be created again without access. However, the permissions of UPDATE and READ from the server side will return the same result for each call.			
Interface request parameter				
No.	Parameter	Type	Required	Description
1		CreatePermissionReq	Y	Wrapper class
CreatePermissionReq				
No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	grant	String	Y	Permissions
3	grantUid	String	Y	Authorized ID in the Hub
4	grantPublicKey	String	Y	Authorized public key
5	grantEncryptKey	String	N	Encryption key. READ/UPDATE permission is required
6	url	String	N	The path of the stored resource. READ/UPDATE

				permission is required
7	sign	String	Y	Signature

Interface response parameter

No.	Parameter	Type	Description
1		AddPermissionResult	Response data

AddPermissionResult

No.	Parameter	Type	Description
1	url	String	The path of the stored resource.
2	Key	String	Ciphertext key to encrypt the resource (encrypted using the authorized public key)

1.6 Delete Permissions

Interface Address	/hub/deletePermission
Description	Resource owner calls this function for permissions that have been created but not yet accessed by third parties.

Interface request parameter

No.	Parameter	Type	Required	Description
1		DeletePermissionReq	Y	Wrapper class

DeletePermissionReq

No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	grant	String	Y	Permissions
3	grantUid	String	Y	Authorized ID in the Hub
4	url	String	Y	The path of the stored resource.
5	sign	String	Y	Signature

Interface response parameter

No.	Parameter	Type	Description
1		DeletePermissionResult	Response data

DeletePermissionResult

No.	Parameter	Type	Description
1	succes	boolean	Whether deleted successfully
2	message	String	Result description

1.7 Query Permissions

Interface Address	/hub/queryPermission
Description	Resource owner calls this function to query Permissions that have already been created.

Interface request parameter

No.	Parameter	Type	Required	Description

1		QueryPermissionReq	Y	Wrapper class
QueryPermissionReq				
1	uid	String	Y	ID in the Hub
2	flag	String	N	YES means accessed; NO means not accessed.
3	grantUid	String	N	Authorized ID in the Hub
4	sign	String	Y	Signature
Interface response parameter				

No.	Parameter	Type	Description
1		List<PermissionInfo>	Response data
PermissionInfo			
No.	Parameter	Type	Description
1	url	String	The path of the stored resource
2	grant	String	Permissions
3	grantUid	String	Authorized ID in the Hub
4	status	Integer	0 means deleted; 1 means not deleted
5	createTime	LocalDateTime	Created date
6	readTime	LocalDateTime	Read time
7	flag	UsedFlag	YES means accessed; NO means not accessed.
8	uid	String	ID in the Hub
9	key	String	Ciphertext key
10	ownerKey	String	Owner's key

1.8 Query Granted Permissions

Interface Address	/hub/queryGrantedPermission
Description	Users can look up all or part of the permission records authorized to them in three dimensions: the uid of the resource owner, whether it has been accessed and the permission type.

Interface request parameter				
No.	Parameter	Type	Required	Description
1		QueryGrantedPermissionReq	Y	Wrapper class
QueryGrantedPermissionReq				
No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	ownerUid	String	N	Resource owner's ID in the Hub
3	flag	String	N	Access status. 0: query accessed permissions; 1: query unaccessed records; if null, query both

4	grant	Sting	N	Authorization type. READ: Read; WRITE: Add; UPDATE: Update
5	sign	String	Y	Signature

Interface response parameter

No.	Parameter	Type	Description
1		List<GrantPermissionInfo>	List of authorized records

GrantPermissionInfo

No.	Parameter	Type	Description
1	url	String	The path of the resource storage.
2	grant	String	Authorization type. READ: Read; WRITE: Add; UPDATE: Update
3	status	Integer	0: deleted; 1: not deleted
4	createTime	Date	Time to create authorization.
5	readTime	Date	Accessed time. Null if not accessed.
6	flag	Integer	0: accessed; 1: not accessed.
7	ownerUid	String	ID in the Hub
8	OwnerKey	String	Ciphertext key
9	key	String	Owner's key

1.9 Query Operation Record of the Resource

Interface Address	/hub/getResourceHistory
Description	Resource owner calls this function to query the operation record of the resource.

Interface request parameter

No.	Parameter	Type	Required	Description
1		QueryResourceHistoryReq	Y	Wrapper class

QueryPermissionReq

No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	url	String	N	The path of the stored resource.
3	operation	String	N	Authorization type. WRITE: Add; UPDATE: Update; DELETE: delete
4	sign	String	Y	Signature

Interface response parameter

No.	Parameter	Type	Description
1		List<ResourceHistoryInfo>	Resource operation record list

PermissionInfo

No.	Parameter	Type	Description
1	OperationUid	String	Operator's ID in the Hub

2	ownerUid	String	Owner’s ID in the Hub
3	operation	String	Authorization type. WRITE: Add; UPDATE: Update; DELETE: delete
4	OperationTime	LocalDateTime	Operation time
5	url	String	The path of the stored resource.
6	key	String	Ciphertext key
7	content	String	Ciphertext resource content

1.10 Change Resource Owner

Interface Address	/hub/transferowner			
Description	Change the resource owner in the Hub.			
Interface request parameter				
No.	Parameter	Type	Required	Description
1		TransferOwnerReq	Y	Wrapper class
QueryPermissionReq				
No.	Parameter	Type	Required	Description
1	uid	String	Y	ID in the Hub
2	url	String	Y	The path of the stored resource.
3	newOwnerUid	String	Y	New owner’s ID in the Hub
4	newOwnerPublicKey	String	Y	New owner’s public key
5	newKey	String	N	New ciphertext key
6	sign	String	Y	Signature
Interface response parameter				
No.	Parameter	Type		Description
1		Boolean	Y	Return true if success, return false if failure

2 .Response Code

Response code	Description
1303	Private key is empty
1304	Private key format is invalid
1305	Public key is empty
1306	recoveryKey.publicKey format is invalid
1307	Public key and private key do not match
1309	The content is empty
1310	The URL is empty
1321	The URL of the Identity Hub cannot be empty
1322	The public key of the Identity Hub cannot be empty
1327	Failed to send the request
1328	The format of the grant is invalid

1329	Grant cannot be empty
1335	Resource does not exist
1336	The key is empty
1337	Failed to encrypt the key
1338	Failed to sign the data
1339	Failed to add permission
1341	Failed to delete permission
1342	Failed to query permission
1343	Grant is empty
1344	Failed to check permission
1347	Failed to query publicKey
1350	Config file does not exist
1351	The public key is empty
1352	You cannot add permissions to yourself
1354	Illegal flag
1361	The user ID is empty
1362	The granted user ID is empty
1363	The public key of the granted user is empty
1364	Failed to generate the user ID
1365	Failed to decrypt the data
1366	Failed to encrypt the data
1367	Failed to decrypt Identity Hub's private key
1368	The private key of Identity Hub is empty
1369	The public key of Identity Hub is empty
1370	No permission to save resource
1371	No permission to update resource
1372	The resource has been saved, cannot be saved again
1373	The granted user already has an unused permission
1400	Permission does not exist
1402	Failed to delete permission
1403	Failed to query permission
1406	Failed to get the public key
1407	Failed to update the public key
1408	Failed to save the resource
1409	Failed to query the resource
1410	Failed to delete the resource
1411	Failed to update the resource
1412	Failed to get the HTTP request
1413	Failed to get the private key of the Identity Hub

1414	Missing request data
1415	Failed to convert the request parameter
1416	Unknown client data or not connected
1417	Parse return code error
1418	Resource does not exist
1419	The URL format is invalid
1422	Signature verification failed
1423	The user is not registered
1424	User registration failed, the format of the public key is invalid
1425	The user is already registered
1426	The granted user does not exist
1427	The resource owner does not exist
1428	Failed to close the permission
1429	Only the resource owner can delete it
1430	Failed to add operation record
1431	Failed to register the user
1432	The user ID is empty
1433	Failed to parse the request parameter
1434	Failed to add the operation record
1435	Failed to update the resource
1436	The resource does not exist
1437	Failed to save the resource
1438	Failed to save the user information to the database
1441	The current permission has been used or the user does not have permission
1442	Failed to check the permission
1455	decrypt encptyKey failed
1456	decrypt content failed
1460	New owner's user ID is empty
1461	Failed to change the owner
1462	The uid and the ownerUid cannot be the same
1463	The ownerUid is not exist
1465	The new owner and new owner's public key do not match
1466	Crypto type format is invalid
1501	Failed to query the encryption key of the granted resource
1502	Failed to decrypt the key of the granted resource
1503	The resource does not exist
1504	recoveryKey.privateKey and recoveryKey.publicKey mismatch
1505	The primary private key and public key do not match

Note: {attribute} is a dynamic parameter.