

# Connecting to the COFFE Smart Wallet API

**Protocol:** https **Server:** coffe.io **Port:** 8888

**Chain ID:** 1ca925bc8fbc1cec262dedd10fd19d9357a1cc8de0bd92e5b61577740af9a3f2

## History server

**Protocol:** https **Server:** resurces.com **Port:** 17555

## JavaScript API

### Sending messages to the bot

#### Client Session ID Generation

```
function getId(length) {
    const validChars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789';
    let array = new Uint8Array(length);
    window.crypto.getRandomValues(array);
    array = array.map(x => validChars.charCodeAt(x % validChars.length));
    const randomState = String.fromCharCode.apply(null, array);
    return randomState;
}
```

#### Connect to COFFE Network Socket

```
var CLIENT = null;
CLIENT = io.connect('https://coffe.io:28000/');
CLIENT.emit( 'bot_module', { action: 'confirm_action', client_id:client_id, dapp:dapp_
account, data: string_data});
/** Receive responses from a socket **/
CLIENT.on(client_id , function (receive) {
    CLIENT.removeListener(client_id);
    if(receive.action == 'confirm_action'){
        /** Operation execution code: 1 - operation completed **/
        if(receive.code == 1){
            /** Operation response code: false - operation error, true - successful
operation **/
            if(receive.result==false){
                /** resolve - return function **/
                /** receive.message - Error message **/
                return resolve( { result:false, message:receive.message } );
            }else{
                /** If the operation completed successfully, no message **/
                return resolve( { result:true, message:'' } );
            }
        }else if(receive.code == 2){ //
            return resolve( { result:false, message:'No confirmation' } );
        }else if(receive.code == 3){
            return resolve( { result:false, message:'Authorisation Error' } );
        }else if(receive.code == 4){
            return resolve( {
                result:false, message:'Repeated authorization in telegram
is required!'
            } );
        }else{
            /** Other errors **/
            return resolve( { result:false, message:'Error' } );
        }
    }
});
```

## Transfer

```
CLIENT.emit( 'bot_module', { action: 'confirm_action', client_id:client_id,
dapp:'crheroesgame', data: JSON.stringify({
  «account»: «eosio.token»,
  «name»: «transfer»,
  «data»: {
    «from»: «rionass1ffor»,
    «to»: «crheroesgame»,
    «quantity»: «1.0000 CFF»,
    «memo»: «my memo»,
  }
})
});
```

## Action

```
CLIENT.emit( 'bot_module', { action: 'confirm_action', client_id:client_id,
dapp:'crheroesgame', data: JSON.stringify({
  «account»: «crheroesgame»,
  «name»: «battle»,
  «data»: {
    «hero»: «rionass1ffor»
  }
})
});
```