

```

avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ ls
Caddyfile
EchidnaLHILEcceNFT.sol
LHILEcceNFT-finale
LHILEcceNFT_flat.sol
LHI_Lecce_NFT_anagrafe_internazionale_OK.png
LHI_NFT_Mobile
index_with_full_abi.html
LHI_NFT_Mobile_backend_Heroku
index_with_full_abi.html:Zone.Identifier
'LHI_NFT_conferma definitiva_LHI_NFT_6_maggio_2025.png'
Polygon_indirizzo_codice_de_contratto_smart.txt
archivio
artifacts
not_verifyPrivateKey.cjs
backend.js
backend_access.log
cache
'console.log("Node.js'
pinata_ipfs_database.txt
contract-address.json
contracts
crytic-export
decripted_images
echidna_config.yaml
encrypted_images
env.production
estraiCID.js
frontend_package.zip
frontend_package.zip:Zone.Identifier
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ cd scripts
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ ls
'API_KEY QuickNode.txt'
checkRpcConnection.cjs
uploadToIPFS.js
FlattenAPI.sol
verifyCoinMarketAPI.cjs
LHILEcceNFT.test.cjs
createMetadata.js
not_validatePrivateKey.cjs
verifyCoinMarketAPI.js
checkBalance.js
decryptFolder.js
not_validatePrivateKey.js
verifyPinataAPI.js
checkBalance1.cjs
deploy.cjs
not_verifyPrivateKey.cjs
verifyPrivateKey.cjs
checkEndpoin_AMOY_Polygon.sh
deploy1.cjs
not_verifyPrivateKey.js
verifyPrivateKey.js
checkEndpoint1base_mainnet.sh
encryptFile.js
not_verifyWallet.js
verifyWallet.cjs
checkEndpoint2basesepolia.sh
encryptFolder.js
request.json
encryptFolderWithUniqueKeys.cjs
testConnection.cjs
checkEndpoint_Polygon_Mainet.sh
getEthereumPrice.js
testConnection1.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ nano testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ cat testConnection.cjs
getWalletAddress.cjs
git
hardhat.config.cjs
history_output.txt
https-server.js
keys_decryptor.json
moneta_LHI_roadmap.txt
node_modules
not_verifyWallet.js
package-lock.json
package.json
script.cjs
script.js
scripts
server.cert
server.js
server.key
sintesi_moneta_LHI.txt
test
testDecrypt.js
tsconfig.json
mintNFT.cjs
mintNFT.js

```

```

// scripts/testConnection.cjs
const hre = require("hardhat"); // Usato per accedere alla configurazione di rete
const { ethers } = require("ethers"); // IMPORTANTE: Importa ethers direttamente
require("dotenv").config(); // Carica le variabili da .env; Hardhat lo fa anche, ma per sicurezza.

async function main() {
  const contractAddress = "0x6a6d5c29ad8f23209186775873e123b31c26e9"; // L'indirizzo del tuo contratto

  console.log(`Rete target specificata: ${hre.network.name}`);

  // Ottieni l'URL RPC dalla configurazione di Hardhat per la rete selezionata.
  // Questo assicura che stiamo usando l'URL che Hardhat userebbe per --network.
  const networkConfig = hre.config.networks[hre.network.name];
  let providerUrl;

  if (networkConfig && networkConfig.url) {
    providerUrl = networkConfig.url;
  } else if (hre.network.name === 'polygon' && process.env.QUICKNODE_MATIC_URL) {
    // Fallback specifico per la rete 'polygon' se l'URL non è in hardhat.config ma QUICKNODE_MATIC_URL è nel .env
    providerUrl = process.env.QUICKNODE_MATIC_URL;
    console.log(`URL RPC per la rete '${hre.network.name}' non trovato in hardhat.config.cjs, utilizzo QUICKNODE_MATIC_URL da .env.`);
  } else {
    console.error(`URL RPC per la rete "${hre.network.name}" non trovato in hardhat.config.cjs e QUICKNODE_MATIC_URL non disponibile per fallback.`);
    return;
  }
  console.log(`Utilizzo dell'URL RPC: ${providerUrl}`);

  // 1. Crea un provider ethers standard DIRETTO
  const provider = new ethers.JsonRpcProvider(providerUrl);

  // All'inizio della funzione main() in testConnection.cjs, dopo aver creato il provider
  try {
    console.log("Tentativo di lookup ENS per 'unlikely-to-exist.eth'...");
    const ensAddress = await provider.resolveName("unlikely-to-exist.eth");
    console.log("Risultato lookup ENS (dovrebbe essere null o fallire):", ensAddress);
  } catch (e) {
    console.log("Errore durante il lookup ENS (questo è OK/atteso se il nome non esiste):", e.message);
  }

  try {
    console.log("Tentativo di reverse lookup ENS per un indirizzo noto (es.

```

```

Vitalik)...");
    const ensName = await
provider.lookupAddress("0xd8dA6BF26964aF9D7eEd9e03E53415D37aA96045");
    console.log("Risultato reverse lookup ENS:", ensName); // Potrebbe essere
null su Polygon se non c'è un reverse record
    } catch (e) {
        console.log("Errore durante il reverse lookup ENS (potrebbe essere OK):",
e.message);
    }

    // 2. Crea un wallet/signer DIRETTO dalla chiave privata
    // Assicurati che PRIVATE_KEY sia definito nel tuo file .env
    if (!process.env.PRIVATE_KEY) {
        console.error("ERRORE: La variabile d'ambiente PRIVATE_KEY non è
definita nel file .env.");
        return;
    }

    const privateKeyRaw = process.env.PRIVATE_KEY; // <--- MODIFICA QUI
    const privateKey = privateKeyRaw.startsWith("0x") ? privateKeyRaw :
`0x${privateKeyRaw}`; // Aggiungi '0x' se manca
    const wallet = new ethers.Wallet(privateKey, provider);
    const signer = wallet; // L'istanza Wallet è già un Signer

    console.log(`Tentativo di connessione con l'account: ${signer.address}`);

    try {
        const balanceWei = await provider.getBalance(signer.address);
        console.log(`Saldo dell'account: ${ethers.formatEther(balanceWei)}
MATIC`);
    } catch (e) {
        console.error(`Errore nel recuperare il saldo per l'account
${signer.address}: ${e.message}`);
        // Non terminiamo qui, proviamo comunque a interagire con il contratto
    }

    try {
        // 3. Carica l'ABI del contratto.
        // Il percorso è relativo alla root del tuo progetto Hardhat.
        console.log("Caricamento ABI del contratto da artifacts...");
        const contractArtifact =
require("../artifacts/contracts/LHILecceNFT.sol/LHILecceNFT.json");
        const contractABI = contractArtifact.abi;
        console.log("ABI del contratto caricato con successo.");

        // 4. Crea un'istanza del contratto usando il provider e il signer
ethers.js diretti
        const nftContract = new ethers.Contract(contractAddress, contractABI,
signer);
        // In ethers v6, nftContract.getAddress() restituisce una
Promise<string>
        console.log(`Collegato all'istanza del contratto all'indirizzo: ${await
nftContract.getAddress()}`);

```

```

        console.log("Tentativo di leggere il nome del contratto (funzione
view)...");
        const name = await nftContract.name(); // Chiama la funzione pubblica
'name()'
        console.log(`Nome del contratto letto con successo: ${name}`);

        console.log("Test di connessione e lettura semplice (con provider e
signer ethers.js diretti) completato con successo.");

    } catch (error) {
        console.error("--- ERRORE DURANTE IL TEST DI CONNESSIONE O LETTURA (CON
PROVIDER E SIGNER ethers.js DIRETTI) ---");
        console.error(`Messaggio: ${error.message}`);
        if (error.stack) {
            console.error("Stack trace:", error.stack);
        }

        console.error("-----");
        process.exitCode = 1;
    }
}

main()
    .then(() => process.exit(0))
    .catch((error) => {
        console.error(error);
        process.exit(1);
    });
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ > testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ nano testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ cd ..
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ npx hardhat run
scripts/testConnection.cjs --network polygon --show-stack-traces
Rete target specificata: polygon
Utilizzo dell'URL RPC:
https://proportionate-dry-brook.matic.quiknode.pro/752b1e43a682209ddc19a49978c10
acac9458739
Tentativo di lookup ENS per 'unlikely-to-exist.eth'...
Errore (o nessun risultato) durante il lookup ENS per 'unlikely-to-exist.eth':
could not decode result data (value="0x", info={ "method": "resolver",
"signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.14.1)
Tentativo di reverse lookup ENS per un indirizzo noto (es. Polygon Bridge)...
Errore (o nessun risultato) durante il reverse lookup ENS: bad address checksum
(argument="address", value="0xA0c68C638235ee32657e8f720a23cec1bFc77C77",
code=INVALID_ARGUMENT, version=6.14.1)
Tentativo di connessione con l'account:
0xf9909c6CD90566BD56621EE0cAc42986ae334Ea3
Saldo dell'account: 365.977862946638599478 MATIC
Caricamento ABI del contratto da artifacts...
ABI del contratto caricato con successo.
Collegato all'istanza del contratto all'indirizzo:
0x6a6d5c29ad8f23209186775873e123b31c26e9
Tentativo di leggere il nome del contratto (funzione view)...

```

```

--- ERRORE DURANTE IL TEST DI CONNESSIONE O LETTURA (CON PROVIDER E SIGNER
ethers.js DIRETTI) ---
Messaggio: could not decode result data (value="0x", info={ "method":
"resolver", "signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.14.1)
Stack trace: Error: could not decode result data (value="0x", info={ "method":
"resolver", "signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.14.1)
    at makeError
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/utils/errors.ts:694:
21)
    at assert
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/utils/errors.ts:715:
25)
    at Interface.decodeFunctionResult
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/abi/interface.ts:916
:15)
    at staticCallResult
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:346:35)
    at processTicksAndRejections (node:internal/process/task_queues:105:5)
    at staticCall
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:303:24)
    at Proxy.resolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:351:41)
    at Function.#getResolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/ens-resolv
er.ts:559:26)
    at Function.fromName
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/ens-resolv
er.ts:590:26)
    at JsonRpcProvider.getResolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/abstract-p
rovider.ts:1193:16)
-----
-----

```

```

avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ nano package.json
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ rm -rf node_modules
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ rm package-lock.json
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ npm install
npm warn EBADENGINE Unsupported engine {
npm warn EBADENGINE   package: 'compile@0.0.2',
npm warn EBADENGINE   required: { node: '0.x' },
npm warn EBADENGINE   current: { node: 'v22.15.1', npm: '10.9.2' }
npm warn EBADENGINE }
npm warn deprecated safe-event-emitter@1.0.1: Renamed to
@metamask/safe-event-emitter
npm warn deprecated @walletconnect/types@1.8.0: WalletConnect's v1 SDKs are now
deprecated. Please upgrade to a v2 SDK. For details see:
https://docs.walletconnect.com/
npm warn deprecated inflight@1.0.6: This module is not supported, and leaks
memory. Do not use it. Check out lru-cache if you want a good and tested way to
coalesce async requests by a key value, which is much more comprehensive and
powerful.

```

npm warn deprecated har-validator@5.1.5: this library is no longer supported  
npm warn deprecated level-errors@1.0.5: Superseded by abstract-level  
(<https://github.com/Level/community#faq>)  
npm warn deprecated memdown@1.4.1: Superseded by memory-level  
(<https://github.com/Level/community#faq>)  
npm warn deprecated eth-json-rpc-infura@5.1.0: Package no longer supported.  
Contact Support at <https://www.npmjs.com/support> for more info.  
npm warn deprecated lodash.isequal@4.5.0: This package is deprecated. Use  
require('node:util').isDeepStrictEqual instead.  
npm warn deprecated eth-sig-util@1.4.2: Deprecated in favor of  
'@metamask/eth-sig-util'  
npm warn deprecated rimraf@3.0.2: Rimraf versions prior to v4 are no longer  
supported  
npm warn deprecated @walletconnect/mobile-registry@1.4.0: Deprecated in favor of  
dynamic registry available from:  
<https://github.com/walletconnect/walletconnect-registry>  
npm warn deprecated deferred-leveldown@1.2.2: Superseded by abstract-level  
(<https://github.com/Level/community#faq>)  
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@5.0.15: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@8.1.0: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated glob@7.1.7: Glob versions prior to v9 are no longer  
supported  
npm warn deprecated ethereumjs-block@2.2.2: New package name format for new  
versions: @ethereumjs/block. Please update.  
npm warn deprecated level-codec@7.0.1: Superseded by level-transcoder  
(<https://github.com/Level/community#faq>)  
npm warn deprecated ethereumjs-tx@2.1.2: New package name format for new  
versions: @ethereumjs/tx. Please update.  
npm warn deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older  
versions may use Math.random() in certain circumstances, which is known to be  
problematic. See <https://v8.dev/blog/math-random> for details.  
npm warn deprecated request@2.88.2: request has been deprecated, see  
<https://github.com/request/request/issues/3142>  
npm warn deprecated @walletconnect/client@1.8.0: WalletConnect's v1 SDKs are now  
deprecated. Please upgrade to a v2 SDK. For details see:  
<https://docs.walletconnect.com/>  
npm warn deprecated ethereumjs-tx@1.3.7: New package name format for new  
versions: @ethereumjs/tx. Please update.  
npm warn deprecated abstract-leveldown@2.6.3: Superseded by abstract-level  
(<https://github.com/Level/community#faq>)

```
npm warn deprecated abstract-leveldown@2.7.2: Superseded by abstract-level
(https://github.com/Level/community#faq)
npm warn deprecated ethereumjs-vm@2.6.0: New package name format for new
versions: @ethereumjs/vm. Please update.
npm warn deprecated levelup@1.3.9: Superseded by abstract-level
(https://github.com/Level/community#faq)
npm warn deprecated ethereumjs-block@1.7.1: New package name format for new
versions: @ethereumjs/block. Please update.
npm warn deprecated @walletconnect/web3-provider@1.8.0: WalletConnect's v1 SDKs
are now deprecated. Please upgrade to a v2 SDK. For details see:
https://docs.walletconnect.com/
npm warn deprecated @walletconnect/qrcode-modal@1.8.0: WalletConnect's v1 SDKs
are now deprecated. Please upgrade to a v2 SDK. For details see:
https://docs.walletconnect.com/
npm warn deprecated ethereumjs-common@1.5.2: New package name format for new
versions: @ethereumjs/common. Please update.
npm warn deprecated @nomiclabs/hardhat-etherscan@3.1.8: The
@nomiclabs/hardhat-etherscan package is deprecated, please use
@nomicfoundation/hardhat-verify instead
npm warn deprecated web3-provider-engine@16.0.1: This package has been
deprecated, see the README for details:
https://github.com/MetaMask/web3-provider-engine
```

added 1404 packages, and audited 1405 packages in 2m

160 packages are looking for funding  
run `npm fund` for details

28 vulnerabilities (14 low, 2 moderate, 12 high)

To address issues that do not require attention, run:  
npm audit fix

To address all issues possible, run:  
npm audit fix --force

Some issues need review, and may require choosing  
a different dependency.

```
Run `npm audit` for details.
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ npm audit fix
npm warn EBADENGINE Unsupported engine {
npm warn EBADENGINE   package: 'compile@0.0.2',
npm warn EBADENGINE   required: { node: '0.x' },
npm warn EBADENGINE   current: { node: 'v22.15.1', npm: '10.9.2' }
npm warn EBADENGINE }
```

added 1 package, and audited 1406 packages in 20s

161 packages are looking for funding  
run `npm fund` for details

# npm audit report

cookie <0.7.0

cookie accepts cookie name, path, and domain with out of bounds characters -

<https://github.com/advisories/GHSA-pxg6-pf52-xh8x>

No fix available

node\_modules/@sentry/node/node\_modules/cookie

@sentry/node 4.0.0-beta.0 - 7.74.2-alpha.1

Depends on vulnerable versions of cookie

node\_modules/@sentry/node

hardhat 0.1.0-rc.0 - 3.0.0-next.1

Depends on vulnerable versions of @sentry/node

node\_modules/hardhat

@nomicfoundation/hardhat-chai-matchers \*

Depends on vulnerable versions of @nomicfoundation/hardhat-ethers

Depends on vulnerable versions of hardhat

node\_modules/@nomicfoundation/hardhat-chai-matchers

@nomicfoundation/hardhat-ethers <=3.0.8

Depends on vulnerable versions of hardhat

node\_modules/@nomicfoundation/hardhat-ethers

@nomicfoundation/hardhat-toolbox \*

Depends on vulnerable versions of @nomicfoundation/hardhat-chai-matchers

Depends on vulnerable versions of @nomicfoundation/hardhat-ethers

Depends on vulnerable versions of

@nomicfoundation/hardhat-ignition-ethers

Depends on vulnerable versions of

@nomicfoundation/hardhat-network-helpers

Depends on vulnerable versions of @nomicfoundation/hardhat-verify

Depends on vulnerable versions of @typechain/hardhat

Depends on vulnerable versions of hardhat

Depends on vulnerable versions of hardhat-gas-reporter

Depends on vulnerable versions of solidity-coverage

node\_modules/@nomicfoundation/hardhat-toolbox

@nomicfoundation/hardhat-ignition <=0.15.11

Depends on vulnerable versions of @nomicfoundation/hardhat-verify

Depends on vulnerable versions of hardhat

node\_modules/@nomicfoundation/hardhat-ignition

@nomicfoundation/hardhat-ignition-ethers <=0.15.11

Depends on vulnerable versions of @nomicfoundation/hardhat-ethers

Depends on vulnerable versions of @nomicfoundation/hardhat-ignition

Depends on vulnerable versions of hardhat

node\_modules/@nomicfoundation/hardhat-ignition-ethers

@nomicfoundation/hardhat-network-helpers <=1.0.12

Depends on vulnerable versions of hardhat

node\_modules/@nomicfoundation/hardhat-network-helpers

@nomicfoundation/hardhat-verify \*

Depends on vulnerable versions of hardhat

node\_modules/@nomicfoundation/hardhat-verify

@nomiclabs/hardhat-etherscan \*

Depends on vulnerable versions of hardhat

node\_modules/@nomiclabs/hardhat-etherscan

@typechain/hardhat \*

Depends on vulnerable versions of hardhat

node\_modules/@typechain/hardhat

hardhat-gas-reporter \*

Depends on vulnerable versions of hardhat



node\_modules/hardhat-gas-reporter  
solidity-coverage >=0.8.0-beta.0  
Depends on vulnerable versions of hardhat  
node\_modules/solidity-coverage

request \*

Severity: moderate

Server-Side Request Forgery in Request -  
<https://github.com/advisories/GHSA-p8p7-x288-28g6>  
Depends on vulnerable versions of tough-cookie  
No fix available

node\_modules/request

web3-provider-engine <=16.0.8  
Depends on vulnerable versions of ethereumjs-block  
Depends on vulnerable versions of ethereumjs-vm  
Depends on vulnerable versions of request

node\_modules/web3-provider-engine

@walletconnect/web3-provider \*

Depends on vulnerable versions of @walletconnect/client  
Depends on vulnerable versions of web3-provider-engine

node\_modules/@walletconnect/web3-provider

semver <5.7.2

Severity: high

semver vulnerable to Regular Expression Denial of Service -  
<https://github.com/advisories/GHSA-c2qf-rxjj-qqgw>  
No fix available

node\_modules/levelup/node\_modules/semver

levelup 0.9.0 - 1.3.9  
Depends on vulnerable versions of semver

node\_modules/levelup

merkle-patricia-tree 0.1.22 - 2.3.2  
Depends on vulnerable versions of levelup

node\_modules/merkle-patricia-tree

ethereumjs-block >=0.0.3  
Depends on vulnerable versions of merkle-patricia-tree

node\_modules/ethereumjs-block

node\_modules/ethereumjs-vm/node\_modules/ethereumjs-block

ethereumjs-vm >=0.1.1  
Depends on vulnerable versions of ethereumjs-block  
Depends on vulnerable versions of merkle-patricia-tree

node\_modules/ethereumjs-vm

tough-cookie <4.1.3

Severity: moderate

tough-cookie Prototype Pollution vulnerability -  
<https://github.com/advisories/GHSA-72xf-g2v4-qvf3>  
No fix available

node\_modules/tough-cookie

ws 7.0.0 - 7.5.9 || 8.0.0 - 8.17.0

Severity: high

ws affected by a DoS when handling a request with many HTTP headers -  
<https://github.com/advisories/GHSA-3h5v-q93c-6h6q>

```

ws affected by a DoS when handling a request with many HTTP headers -
https://github.com/advisories/GHSA-3h5v-q93c-6h6q
fix available via `npm audit fix --force`
Will install ethers@6.14.1, which is outside the stated dependency range
node_modules/ethers/node_modules/ws
node_modules/ws
  @walletconnect/socket-transport *
  Depends on vulnerable versions of ws
  node_modules/@walletconnect/socket-transport
    @walletconnect/core <=1.8.0
    Depends on vulnerable versions of @walletconnect/socket-transport
    node_modules/@walletconnect/core
      @walletconnect/client <=1.8.0
      Depends on vulnerable versions of @walletconnect/core
      node_modules/@walletconnect/client
    ethers 6.0.0-beta.1 - 6.13.0
    Depends on vulnerable versions of ws
    node_modules/ethers

```

28 vulnerabilities (14 low, 2 moderate, 12 high)

To address issues that do not require attention, run:  
 npm audit fix

To address all issues possible, run:  
 npm audit fix --force

Some issues need review, and may require choosing  
 a different dependency.

```

avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ npx hardhat run
scripts/testConnection.cjs --network polygon --show-stack-traces
Rete target specificata: polygon
Utilizzo dell'URL RPC:
https://proportionate-dry-brook.matic.quiknode.pro/752b1e43a682209ddc19a49978c10
acac9458739
Tentativo di lookup ENS per 'unlikely-to-exist.eth'...
Errore (o nessun risultato) durante il lookup ENS per 'unlikely-to-exist.eth':
could not decode result data (value="0x", info={ "method": "resolver",
"signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.11.1)
Tentativo di reverse lookup ENS per un indirizzo noto (es. Polygon Bridge)...
Errore (o nessun risultato) durante il reverse lookup ENS: bad address checksum
(argument="address", value="0xA0c68C638235ee32657e8f720a23cec1bFc77C77",
code=INVALID_ARGUMENT, version=6.11.1)
Tentativo di connessione con l'account:
0xf9909c6CD90566BD56621EE0cAc42986ae334Ea3
Saldo dell'account: 365.977862946638599478 MATIC
Caricamento ABI del contratto da artifacts...
ABI del contratto caricato con successo.
Collegato all'istanza del contratto all'indirizzo:
0x6a6d5c29ad8f23209186775873e123b31c26e9
Tentativo di leggere il nome del contratto (funzione view)...
--- ERRORE DURANTE IL TEST DI CONNESSIONE O LETTURA (CON PROVIDER E SIGNER
ethers.js DIRETTI) ---
Messaggio: could not decode result data (value="0x", info={ "method":

```

```

"resolver", "signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.11.1)
Stack trace: Error: could not decode result data (value="0x", info={ "method":
"resolver", "signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.11.1)
  at makeError
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/utils/errors.ts:694:
21)
    at assert
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/utils/errors.ts:715:
25)
    at Interface.decodeFunctionResult
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/abi/interface.ts:916
:15)
    at staticCallResult
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:346:35)
    at processTicksAndRejections (node:internal/process/task_queues:105:5)
    at staticCall
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:303:24)
    at Proxy.resolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:351:41)
    at Function.#getResolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/ens-resolv
er.ts:559:26)
    at Function.fromName
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/ens-resolv
er.ts:590:26)
    at JsonRpcProvider.getResolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/abstract-p
rovider.ts:1182:16)
-----
-----

```

```

avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ cd scripts
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ nano testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ > testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ nano testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ cd ..
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ npx hardhat run
scripts/testConnection.cjs --network polygon --show-stack-traces
Rete target specificata: polygon
Utilizzo dell'URL RPC:
https://proportionate-dry-brook.matic.quiknode.pro/752b1e43a682209ddc19a49978c10
acac9458739
Tentativo di chiamare provider.getBlockNumber()...
Numero blocco attuale: 71632465
Tentativo di ottenere il resolver ENS per 'unlikelyname.eth'...
Errore (o nessun risultato) durante provider.getResolver('unlikelyname.eth'):
could not decode result data (value="0x", info={ "method": "resolver",
"signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.11.1)
Tentativo di connessione con l'account:
0xf9909c6CD90566BD56621EE0cAc42986ae334Ea3
Saldo dell'account: 365.977862946638599478 MATIC
Caricamento ABI del contratto da artifacts...

```

```

ABI del contratto caricato con successo.
Collegato all'istanza del contratto all'indirizzo:
0x6a6d5c29ad8f23209186775873e123b31c26e9
Tentativo di leggere il nome del contratto (funzione view)...
--- ERRORE DURANTE IL TEST DI CONNESSIONE O LETTURA (CON PROVIDER E SIGNER
ethers.js DIRETTI) ---
Messaggio: could not decode result data (value="0x", info={ "method":
"resolver", "signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.11.1)
Stack trace: Error: could not decode result data (value="0x", info={ "method":
"resolver", "signature": "resolver(bytes32)" }, code=BAD_DATA, version=6.11.1)
    at makeError
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/utils/errors.ts:694:
21)
    at assert
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/utils/errors.ts:715:
25)
    at Interface.decodeFunctionResult
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/abi/interface.ts:916
:15)
    at staticCallResult
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:346:35)
    at processTicksAndRejections (node:internal/process/task_queues:105:5)
    at staticCall
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:303:24)
    at Proxy.resolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/contract/contract.ts
:351:41)
    at Function.#getResolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/ens-resolv
er.ts:559:26)
    at Function.fromName
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/ens-resolv
er.ts:590:26)
    at JsonRpcProvider.getResolver
(/home/avvocato/LHI-NFT-CORRETTO/node_modules/ethers/src.ts/providers/abstract-p
rovider.ts:1182:16)
-----
-----
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ nano testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO$ cd scripts
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$ nano testConnection.cjs
avvocato@LAPTOP-7E2BDF4K:~/LHI-NFT-CORRETTO/scripts$

```