

Installing dependencies using npm

```
(base) PS D:\VS CODE\Python\CS5572\dtip-project> npm install --save-dev @openzeppelin/contracts hardhat-deploy ethers @nomicfoundation/hardhat-ethers dotenv
npm WARN idealTree Removing dependencies.dotenv in favor of devDependencies.dotenv

added 199 packages, removed 1 package, changed 6 packages, and audited 263 packages in 21s

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

14 vulnerabilities (4 low, 9 high, 1 critical)

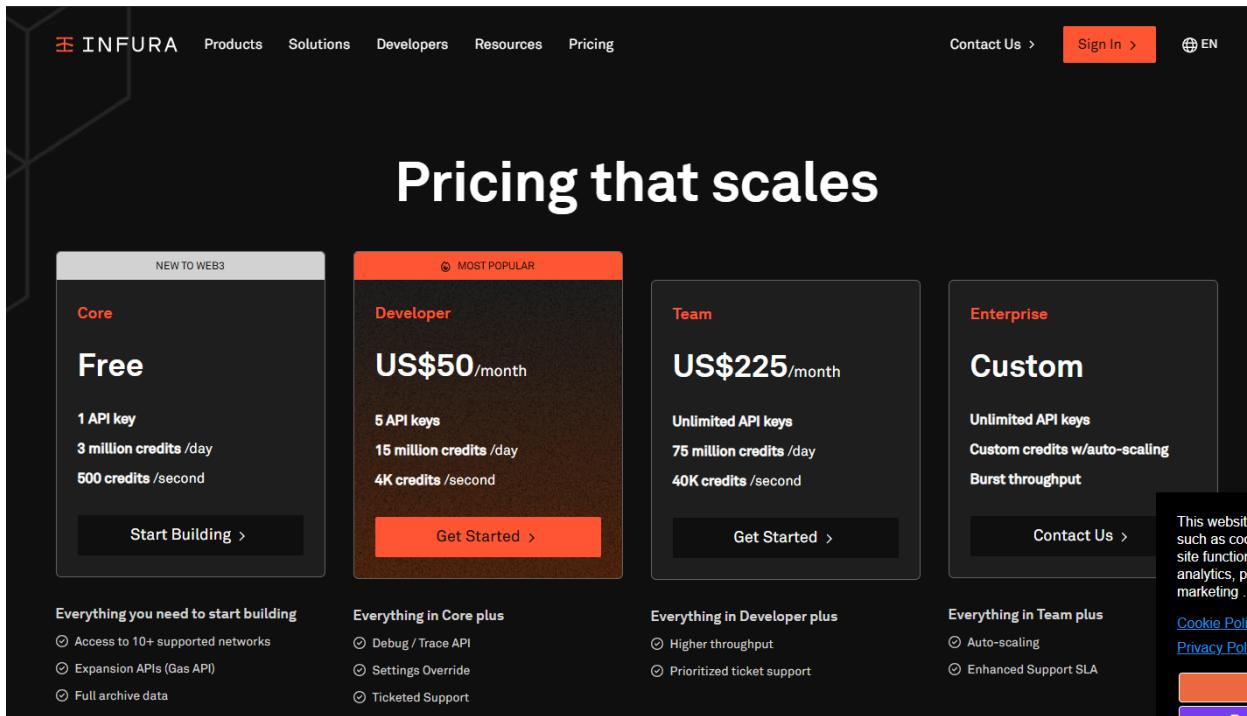
To address issues that do not require attention, run:
  npm audit fix

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

Run `npm audit` for details.

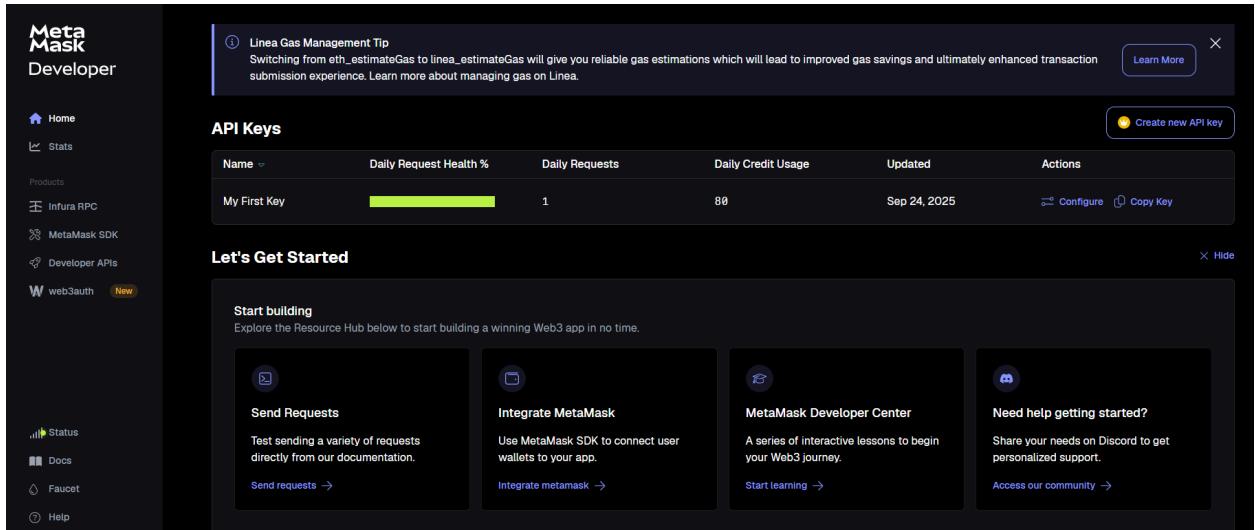
(base) PS D:\VS CODE\Python\CS5572\dtip-project>
```

To get the SEPOLIA_RPC_URL API key, head to <https://www.infura.io/> to sign up and choose the free plan.



The screenshot shows the Infura Pricing page. At the top, there's a navigation bar with links for Products, Solutions, Developers, Resources, Pricing, Contact Us, Sign In, and a language switcher (EN). The main heading is "Pricing that scales". Below it, four pricing plans are listed: "NEW TO WEB3 Core Free", "MOST POPULAR Developer US\$50/month", "Team US\$225/month", and "Enterprise Custom". Each plan includes a brief description of its features and a "Get Started" button. At the bottom, there are sections for "Everything you need to start building", "Everything in Core plus", "Everything in Developer plus", and "Everything in Team plus", each with a list of included features. A note on the right side states: "This website uses cookies and other tracking technologies to provide you with a better user experience, analyze site function, perform analytics, personalization and marketing." It also links to "Cookie Policy" and "Privacy Policy".

It supports 1 API key with 3 million credits /day and 500 credits /second
Create an API key to get the SEPOLIA_RPC_URL



The screenshot shows the MetaMask Developer dashboard. On the left is a sidebar with links: Home, Stats, Products, Infura RPC, MetaMask SDK, Developer APIs, web3auth (marked as New), Status, Docs, Faucet, and Help. The main area has a dark header with a 'Linea Gas Management Tip' about switching from eth_estimateGas to linea_estimateGas. Below is an 'API Keys' section with a table showing one key: 'My First Key'. The 'Actions' column for this key includes 'Configure' and 'Copy Key'. A 'Create new API key' button is at the top right of the table. Underneath is a 'Let's Get Started' section titled 'Start building'. It contains four cards: 'Send Requests' (Test sending a variety of requests directly from our documentation. [Send requests](#)), 'Integrate MetaMask' (Use MetaMask SDK to connect user wallets to your app. [Integrate metamask](#)), 'MetaMask Developer Center' (A series of interactive lessons to begin your Web3 journey. [Start learning](#)), and 'Need help getting started?' (Share your needs on Discord to get personalized support. [Access our community](#)). A 'Hide' link is at the top right of this section.

Npm hardhit init error:

```
⑤ (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat init

WARNING: You are using Node.js 20.11.1 which is not supported by Hardhat.
Please upgrade to 22.10.0 or a later LTS version (even major version number

Hardhat only supports ESM projects.

Please make sure you have `"type": "module"` in your package.json.

You can set it automatically by running:

npm pkg set type="module"

● (base) PS D:\VS CODE\Python\CS572\dtip-project> npm pkg set type="module"
◆ (base) PS D:\VS CODE\Python\CS572\dtip-project>
```

```
○ (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat init
 888   888           888 888           888
 888   888           888 888           888
 888   888           888 888           888
 888888888888 8888b. 888d888 .d88888 888888b. 8888b. 8888888
 888   888     "88b 888P" d88" 888 888 "88b     "88b 888
 888   888 .d888888 888   888 888 888 .d888888 888
 888   888 888 888 888 Y88b 888 888 888 888 888 Y88b.
 888   888 "Y888888 888     "Y88888 888 888 "Y888888  "Y888

Welcome to Hardhat v2.26.3

? What do you want to do? ...
> Create a JavaScript project
Create a TypeScript project
Create a TypeScript project (with Viem)
Create an empty hardhat.config.js
Quit
```

Success!

After initiating the Hardhat project, we can see the menu of options
Choose 'Create an empty [hardhat.config.js](#)'

```
(base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat init
✓ What do you want to do? - Create an empty hardhat.config.js
Config file created

You need to install hardhat locally to use it. Please run:

npm install --save-dev "hardhat@^2.26.3"

Give Hardhat a star on Github if you're enjoying it!

https://github.com/NomicFoundation/hardhat
● (base) PS D:\VS CODE\Python\CS572\dtip-project> npm install --save-dev "hardhat@^2.26.3"

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 glob@8.1.0: Glob versions prior to v9 are no longer supported

added 216 packages, and audited 217 packages in 20s

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

5 low severity vulnerabilities

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
○ (base) PS D:\VS CODE\Python\CS572\dtip-project>
```

Import contracts layout from OpenZeppelin blockchain library (@openzeppelin/contracts)
Command: npm install --save-dev @openzeppelin/contracts hardhat-deploy ethers

```
● (base) PS D:\VS CODE\Python\CS572\dtip-project> npm install @openzeppelin/contracts
added 1 package, and audited 218 packages in 2s

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

5 low severity vulnerabilities

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Import @nomicfoundation/hardhat-toolbox library

```
(base) PS D:\VS CODE\Python\CS572\dtip-project> npm install @nomicfoundation/hardhat-toolbox
npm WARN deprecated @types/minimatch@6.0.0: This is a stub types definition. minimatch provides its own type definitions,
o you do not need this installed.
npm WARN deprecated lodash.isEqual@4.5.0: This package is deprecated. Use require('node:util').isDeepStrictEqual instead.
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@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

added 361 packages, and audited 579 packages in 57s

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

15 low severity vulnerabilities

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

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

Run `npm audit` for details.
(base) PS D:\VS CODE\Python\CS572\dtip-project> []
```

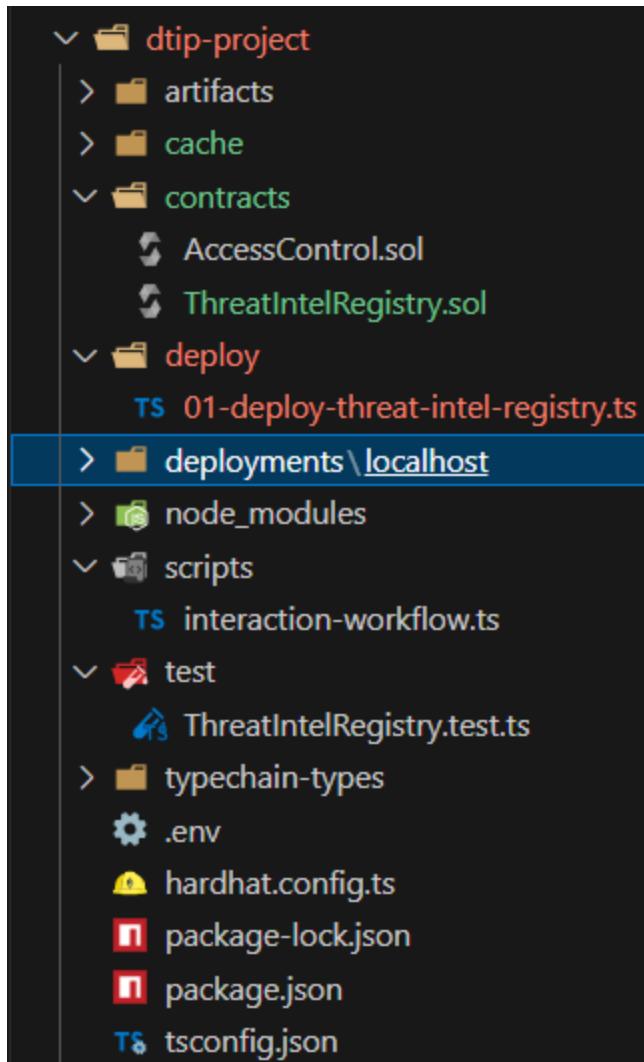
Create tsconfig.json to handle the TypeScript configuration file

```
dtip-project > ts tsconfig.json > ...
1  {
2    "compilerOptions": {
3      "target": "es2020",
4      "module": "NodeNext",
5      "moduleResolution": "NodeNext",
6      "strict": true,
7      "esModuleInterop": true,
8      "outDir": "dist",
9      "forceConsistentCasingInFileNames": true,
10     "skipLibCheck": true
11   },
12   "include": ["./scripts", "./test", "./typechain-types"],
13   "files": ["./hardhat.config.ts"]
14 }
```

Create [hardhat.config.ts](#) and compile with **npx hardhat compile**

```
● (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat compile
  ✘ Help us improve Hardhat with anonymous crash reports & basic usage data? (Y/n) • y
Nothing to compile
No need to generate any newer typings.
```

Add files and folders (/contracts, /deploy, /test)



Then run `npx hardhat compile`

```
● (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat compile
✖ Help us improve Hardhat with anonymous crash reports & basic usage data? (Y/n) · y
Generating typings for: 18 artifacts in dir: typechain-types for target: ethers-v6
Successfully generated 46 typings!
Compiled 18 Solidity files successfully (evm target: paris).
❖ (base) PS D:\VS CODE\Python\CS572\dtip-project> █
```

On Terminal 1, run `npx hardhat node` to start the node

```

❖ (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat node
  ✘ Help us improve Hardhat with anonymous crash reports & basic usage data? (Y/n) · y
Nothing to compile
No need to generate any newer typings.
Started HTTP and WebSocket JSON-RPC server at http://127.0.0.1:8545/

Accounts
=====
WARNING: These accounts, and their private keys, are publicly known.
Any funds sent to them on Mainnet or any other live network WILL BE LOST.

Account #0: 0xf39Fd6e51aad88F6F4ce6aB8827279cffFb92266 (10000 ETH)
Private Key: 0xac0974bec39a17e36ba4a6b4d238ff944bacb478cbed5efcae784d7bf4f2ff80

Account #1: 0x70997970C51812dc3A010C7d01b50e0d17dc79C8 (10000 ETH)
Private Key: 0x59c6995e998f97a5a0044966f0945389dc9e86dae88c7a8412f4603b6b78690d

Account #2: 0x3C44CdDdB6a900fa2b585dd299e03d12FA4293BC (10000 ETH)
Private Key: 0x5de4111afa1a4b94908f83103eb1f1706367c2e68ca870fc3fb9a804cdab365a

Account #3: 0x90F79bf6EB2c4f870365E785982E1f101E93b906 (10000 ETH)
Private Key: 0x7c852118294e51e653712a81e05800f419141751be58f605c371e15141b007a6

Account #4: 0x15d34AAf54267DB7D7c367839AAf71A00a2C6A65 (10000 ETH)
Private Key: 0x47e179ec197488593b187f80a00eb0da91f1b9d0b13f8733639f19c30a34926a

Account #5: 0x9965507D1a55bcC2695C58ba16FB37d819B0A4dc (10000 ETH)
Private Key: 0x8b3a350cf5c34c9194ca85829a2df0ec3153be0318b5e2d3348e872092edffba

Account #6: 0x976EA74026E726554dB657fA54763abd0C3a0aa9 (10000 ETH)
Private Key: 0x92db14e403b83dfe3df233f83dfa3a0d7096f21ca9b0d6d6b8d88b2b4ec1564e

Account #7: 0x14dC79964da2C08b23698B3D3cc7Ca32193d9955 (10000 ETH)
Private Key: 0x4bbbff85ce3377467afe5d46f804f221813b2bb87f24d81f60f1fcdbf7cbf4356

Account #8: 0x23618e81E3f5cdF7f54C3d65f7FBc0aBf5B21E8f (10000 ETH)
Private Key: 0xdbda1821b80551c9d65939329250298aa3472ba22feea921c0cf5d620ea67b97

Account #9: 0xa0Ee7A142d267C1f36714E4a8F75612F20a79720 (10000 ETH)
Private Key: 0x2a871d0798f97d79848a013d4936a73bf4cc922c825d33c1cf7073dff6d409c6

Account #10: 0xBcd4042DE499D14e55001CcbB24a551F3b954096 (10000 ETH)
Private Key: 0xf214f2b2cd398c806f84e317254e0f0b801d0643303237d97a22a48e01628897

Account #11: 0x71bE63f3384f5fb98995898A86B02Fb2426c5788 (10000 ETH)

```

On Terminal 2, run **npx hardhat deploy --network localhost**

In case of failing the test, reset the deployment with **npx hardhat deploy --network localhost --reset**

```

● (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat deploy --network localhost --reset
  ✘ Help us improve Hardhat with anonymous crash reports & basic usage data? (Y/n) · y
    Nothing to compile
    No need to generate any newer typings.

    Deploying ThreatIntelRegistry...
    deploying "ThreatIntelRegistry" (tx: 0x312f013e5bd6a2045978f36dcf689ee5935f963833561fdd4295235f86352d0d)...: deployed at 0xe7f1725E7734CE288F8367e1Bb143E90bb3F0512 with 2540095 gas
    ThreatIntelRegistry Deployed at: 0xe7f1725E7734CE288F8367e1Bb143E90bb3F0512

    Granting RESEARCHER_ROLE to researcher1...
    RESEARCHER_ROLE granted to: 0x70997970C51812dc3A010C7d01b50e0d17dc79C8

◆ (base) PS D:\VS CODE\Python\CS572\dtip-project>

```

Then, run this command when done **npx hardhat run scripts/interaction-workflow.ts --network localhost**

```

● (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat run scripts/interaction-workflow.ts --network localhost
  ✘ Help us improve Hardhat with anonymous crash reports & basic usage data? (Y/n) · y
    --- Interacting with Deployed ThreatIntelRegistry ---
    Contract instance attached at: 0xCf7Ed3AccA5a467e9e704C703E8D87F634fB0Fc9

    --- Step 1: Researcher is preparing and signing a report off-chain ---
    Current nonce for researcher 0x70997970C51812dc3A010C7d01b50e0d17dc79C8 is: 0
    EIP-712 Digest to be signed: 0x8c33d37c49733396dfd5829e4f5b0dbe45d0b6e843fb461ccb201e960ba40611
    Generated Signature: 0xda53b50e7bd61e38b2e65388123f477013d28c6d652c8a8a5e8c4bcba86b0f9039b226b47b346ca24c3e2fa92c4c4a28
    86b24e7cd8a7968ce9954b57ae8276411c

    --- Step 2: Submitting the signed report to the blockchain ---
    Transaction sent. Waiting for confirmation... Hash: 0xd969150d4543b81009ba07f079cee58a195e77a41aa1967a4359ede7a5100cdc
    Transaction confirmed. Report has been submitted.
    New nonce for researcher 0x70997970C51812dc3A010C7d01b50e0d17dc79C8 is: 1

    --- Step 3: A consumer queries the registry for the submitted report ---
    Calculated Report ID for query: 0x77669e7fa8c5a39057a6917c61ac8af7c3b474148897fc821e3f9e2bb73ce9b8

    --- Retrieved Report Details ---
    Reporter: 0x70997970C51812dc3A010C7d01b50e0d17dc79C8
    IOC Type: 1 (DomainName)
    IOC Value: evil-phish.net
    Severity: 2 (High)
    Description: Credential harvesting site
    Timestamp: Sat, 27 Sep 2025 20:01:19 GMT
    Is Deprecated: false
◆ (base) PS D:\VS CODE\Python\CS572\dtip-project>

```

We can see that the test is running successfully!

Unit Testing

Inside the ‘test’ folder, we can see [ThreatIntelRegistry.test.ts](#)

Now, let’s test with the command **npx hardhat test**

```
(base) PS D:\VS CODE\Python\CS5572\dtip-project> npx hardhat test
x Help us improve Hardhat with anonymous crash reports & basic usage data? (y/n) · y

ThreatIntelRegistry
✓ Should set the right admin role
✓ Should allow a researcher to submit a report with a valid signature (44ms)
✓ Should prevent submission with an invalid signature
✓ Should prevent submission from an account without the RESEARCHER_ROLE
1) Should prevent a replay attack using the same signature

4 passing (536ms)
1 failing

1) ThreatIntelRegistry
    Should prevent a replay attack using the same signature:
AssertionError: Expected transaction to be reverted with reason 'ThreatIntel: Invalid signature.', but it reverted with reason 'ThreatIntel: Signer mismatch.'
at async Context.<anonymous> (test\ThreatIntelRegistry.test.ts:121:9)

-----| Solidity and Network Configuration |-----|-----|-----|-----|-----|
| Solidity: 0.8.20 | Optim: false | Runs: 200 | viaIR: false | Block: 30,000,000 gas |
-----| Methods |-----|-----|-----|-----|-----|
| Contracts / Methods | Min | Max | Avg | # calls | usd (avg) |
-----| ThreatIntelRegistry |-----|-----|-----|-----|-----|
| grantRole | 29,750 | 52,058 | 36,124 | 7 | - |
| submitReportBySignature | 182,124 | 204,499 | 189,582 | 3 | - |
-----| Deployments |-----|-----|-----|-----|-----|
| ThreatIntelRegistry | - | - | - | 2,583,701 | 8.6 % |
-----| Key |-----|-----|-----|-----|-----|
| 0 Execution gas for this method does not include intrinsic gas overhead |
| ^ Cost was non-zero but below the precision setting for the currency display (see options) |
| Toolchain: hardhat |
-----
```

We can see that the ThreatIntelRegistry passed 4 tests and failed 1 last test.

I fixed the code in the [ThreatIntelRegistry.test.ts](#) as below:

```
await expect(  
  
threatIntelRegistry.connect(researcher1).submitReportBySignature(report,  
signature)  
).to.be.revertedWith("ThreatIntel: Signer mismatch.");
```

Run again, and we can see the result:

```

● (base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat test
✖ Help us improve Hardhat with anonymous crash reports & basic usage data? (Y/n) · y

ThreatIntelRegistry
  ✓ Should set the right admin role
  ✓ Should allow a researcher to submit a report with a valid signature
  ✓ Should prevent submission with an invalid signature
  ✓ Should prevent submission from an account without the RESEARCHER_ROLE
  ✓ Should prevent a replay attack using the same signature

  5 passing (597ms)

-----
| Solidity and Network Configuration
| Solidity: 0.8.20      Optim: false   Runs: 200    viaIR: false   Block: 30,000,000 gas
| Methods
| Contracts / Methods   Min          Max          Avg          # calls     usd (avg)
| ThreatIntelRegistry
|   grantRole           29,750       52,058       36,124       7          -
|   submitReportBySignature 182,124     204,499     189,582       3          -
| Deployments
| ThreatIntelRegistry          -          -          -          2,583,701     8.6 %        -
| Key
| ⓘ Execution gas for this method does not include intrinsic gas overhead
| ⚡ Cost was non-zero but below the precision setting for the currency display (see options)
| Toolchain: hardhat
◆ (base) PS D:\VS CODE\Python\CS572\dtip-project>

```

I added some more tests to check inside [ThreatIntelRegistry.test.ts](#). It should test the contracts' conditions in ThreatIntelRegistry.sol file

To reset all and start the hardhat from beginning (especially after **npx hardhat compile**):

- ***npx hardhat node***
- ***npx hardhat deploy --network localhost --reset***
- ***npx hardhat run scripts/interaction-workflow.ts --network localhost***
- ***npx hardhat test***

Option 1: Restart the Blockchain (The Quick Fix) [🔗](#)

This wipes the blockchain's memory, letting you start from a clean slate.

1. Go to the terminal where `npx hardhat node` is running and press `Ctrl + C` to stop it.
2. Restart the node: `npx hardhat node`
3. In your other terminal, re-deploy your contracts to the fresh node: `npx hardhat deploy --network localhost --reset`
4. Now you can run your script again successfully: `npx hardhat run scripts/interaction-workflow.ts --network localhost`

Restart the blockchain again, and see the result on the tests:

```
(base) PS D:\VS CODE\Python\CS572\dtip-project> npx hardhat test
✖ Help us improve Hardhat with anonymous crash reports & basic usage data? (Y/n) · y
```

ThreatIntelRegistry

- ✓ Should set the right admin role
- ✓ Should allow a researcher to submit a report with a valid signature (62ms)
- ✓ Should prevent submission with an invalid signature (41ms)
- ✓ Should prevent submission from an account without the RESEARCHER_ROLE
- ✓ Should prevent a replay attack using the same signature (38ms)
- ✓ Should prevent duplicate submission overwriting a valid report (46ms)
- ✓ Should prevent someone not original reporter or an admin from deprecating a report (97ms)
- ✓ Should allow the original reporter to deprecate their own report
- ✓ Should allow an admin to deprecate any report
- ✓ Should prevent deprecating a non-existent report
- ✓ Should prevent deprecating an already deprecated report
- ✓ Should prevent non-admins from granting roles
- ✓ Should prevent overwriting a deprecated report

13 passing (1s)

Solidity and Network Configuration									
Solidity	0.8.20	Optim	false	Runs	200	viaIR	false	Block	30,000,000 gas
Methods									
Contracts / Methods									
Contract	Method	Min	Max	Avg	# calls	Gas	Gas	USD	USD (avg)
ThreatIntelRegistry									
	deprecateReport	48,448	50,802	50,013	6	-	-	-	-
	grantRole	29,750	52,058	32,724	15	-	-	-	-
	submitReportBySignature	181,960	249,116	206,892	9	-	-	-	-
Deployments									
ThreatIntelRegistry									
		-	-	2,704,217	9 %	-	-	-	-
Key									
Execution gas for this method does not include intrinsic gas overhead									
Cost was non-zero but below the precision setting for the currency display (see options)									
Toolchain: hardhat									

```
❖ (base) PS D:\VS CODE\Python\CS572\dtip-project> []
```