

# Configuração do ambiente Demo

---

1. Faça o download do arquivo do link abaixo e extraia seu conteúdo:

`hamsa-msft-demo.zip`

2. Pelo prompt de comando, acesse o diretório:

```
cd hamsa-msft-demo
```

## Configuração do arquivo hardhat.config.js

1. Abra o arquivo hardhat.config.js em seu software de edição de preferência e atualize-o com os IPs/URI:

1. server\_L1\_besu: IP/URI do Hyperledger Besu (Layer 1);
2. server\_L2\_1: IP/URI referente ao Node Layer 2 do Bacen/SELIC;
3. server\_L2\_2: IP/URI referente à instituição A;
4. server\_L2\_3: IP/URI referente à instituição B;

Para os testes desta demo é necessário manter os valores referentes às portas. Exemplo: Besu:8545, Nodes: 8123, 8124 e 8125, imagem abaixo:

```
hamsa-privacy > hardhat.config.js > <unknown> > networks > server_L2_3
1  require("@nomicfoundation/hardhat-toolbox");
2
3  /** @type import('hardhat/config').HardhatUserConfig */
4  module.exports = {
5    solidity: {
6      version: "0.8.20",
7      settings: {
8        optimizer: {
9          enabled: true,
10         runs: 77,
11       },
12     },
13   },
14   ignition: {
15     requiredConfirmations: 1,
16   },
17   networks: {
18     server_L1_besu: {
19       url: "http://192.168.1.6:8545",
20     },
21     accounts: [...],
22   },
23   },
24   server_L2_1: {
25     url: "http://192.168.1.6:8123",
26     accounts: [...],
27   },
28   },
29   server_L2_2: {
30     url: "http://192.168.1.6:8124",
31     accounts: [...],
32   },
33   },
34   server_L2_3: {
35     url: "http://192.168.1.6:8125",
36     accounts: [...],
37   },
38   },
39   },
40   };
41
```

2. Para testes já existentes nesta demo, as chaves privadas das contas na Layer 1 não precisam ser modificadas. Caso você esteja usando sua própria blockchain Layer 1, atualize o arquivo com as suas próprias chaves.

## Compilação dos smart contracts, Implantação do DVP-Match e Rollup na Layer 1

1. Execute os seguintes comandos para realizar instalação das dependências e compilar os smart contracts

instalação das dependências

```
npm i
```

compilação dos smart contracts

```
npx hardhat compile
```

2. Implantação do servidor DVP-Match, execute o comando abaixo:

```
npm run deploy-dvp-match-server-L1
```

O resultado esperado será seguindo o seguinte exemplo abaixo:

```
▶ → hamsa-msft-demo npm run deploy-dvp-match-server-L1

> deploy-dvp-match-server-L1
> npx hardhat run ./script/deploy_dvp_match.js --network server_L1_besu

PoseidonHasher is deployed at 0x86Df37EC5b6FAe973eB891b7Bb09F8E9238cf8FE
poseidonT4 is deployed at 0x40aaa9363bb810b40a01E36E99E455Cce9D7a6A2
L1MatchScAddress: 0xAa1e9B9E4FF1200F8101c6bB7C380E6C808d8190
▶ → hamsa-msft-demo █

7 # L1 blockchain matchsc address, execute the deployment command to get the address: npm run deploy-dvp-match-server-L1
8 DVP_L1MATCHSCADDRESS=0xd96b353D16bee554b5a5c7C52eAA14a4E407203
```

3. Um endereço **L1MatchScAddress** será gerado e será necessário atualizar o arquivo **.env** existente em cada Node (instituição).

- Localize no arquivo **./server/node{1,2 ou 3}/.env** a variável de ambiente: **DVP\_L1MATCHSCADDRESS**;
- Atualize o valor existente pelo valor de **L1MatchScAddress**, destacado acima;

4. Implantação do servidor de Rollup, execute o comando abaixo:

```
npm run deploy-rollup-server-L1
```

```
▶ → hamsa-msft-demo npm run deploy-rollup-server-L1

> deploy-rollup-server-L1
> npx hardhat run ./script/deploy_verifier_l1.js --network server_L1_besu

verifier deployed to: 0xFBe654AF8E3C473D5E2ab3fbE8C86d452D67328b
▶ → hamsa-msft-demo █

11 # prover verify contract address
12 ROLLUP_VERIFYCONFIG_VERIFYCONTRACT=0xf82479A43c51f66EDf4952aE86246d79Ed2fb933
```

- Localize no arquivo **./server/node{1,2 ou 3}/.env** a variável de ambiente: **ROLLUP\_VERIFYCONFIG\_VERIFYCONTRACT**;
- Atualize o valor existente pelo valor destacado acima;

5. No mesmo arquivo **./server/node{1,2 ou 3}/.env** do Node, localize a variável **L1\_URL** e atualize com o endereço de IP/URI do servidor Layer 1 (Hyperledger Besu utilizado).

```
1 # L1 blockchain address, the local IP address of the computer
2 L1_URL=http://123.456.789.00:8545
```

---

[Implantação dos Nodes](#)

[Node privado - Layer 2](#)

[Inicio](#)