

Enable API request forwarding

For Avalanche (C-Chain) JSON-RPC methods, you can request failover protection by adding the failover header to your API request using `cURL`, `Web3.js`, `Ethers.js`, or any other language of your choice.

For more information about this feature, including our partner and their privacy information, see [Failover protection](#).

info Avalanche (C-Chain) failover support is available on Mainnet only.

Request

In the code tabs, the `eth_blockNumber` method is used as an example.

- `cURL`
- `Web3.js`
- `Ethers.js`

```
curl https://avalanche-mainnet.infura.io/v3/YOUR-API-KEY \-X POST \-H "Enable-Failover: true" \-d '{"jsonrpc":"2.0","method":"eth_blockNumber","params":[],"id":1}' const { Web3 } = require('web3'); const https = require('https');
```

```
const options = { headers: { "Enable-Failover": "true", }, };
```

```
const provider = new https.Agent(options);
```

```
const web3 = new Web3(new Web3.providers.HttpProvider('https://avalanche-mainnet.infura.io/v3/YOUR-API-KEY', { agent: provider }));
```

```
web3.eth.getBlockNumber().then(console.log); const ethers = require('ethers'); const fetch = require('node-fetch');
```

```
class InfuraJsonRpcProvider extends ethers.providers.JsonRpcProvider { constructor(network, apiKey) { super(network, apiKey); this.fetchFunc = async (url, json, processFunc) => { const response = await fetch(url, { method: 'POST', body: json.body, headers: { 'Content-Type': 'application/json', 'Enable-Failover': 'true' } });
```

```
const text = await response.text(); const fetchJsonResponse = { jsonrpc: json.jsonrpc, id: json.id, result: JSON.parse(text).result, error: JSON.parse(text).error };
```

```
return processFunc(fetchJsonResponse); } }
```

```
const provider = new InfuraJsonRpcProvider('https://avalanche-mainnet.infura.io/v3/YOUR-API-KEY');
```

```
provider.getBlockNumber().then((blockNumber) => { console.log(blockNumber); });
```

Last updated on Apr 19, 2024 [Previous](#) [How to Next](#) [Get testnet AVAX tokens](#)