

Integrations

Overview

It is possible and even encouraged to use Lava with popular web3 development libraries such as [ethers.js](#), [web3.js](#), [cosmjs](#) and [viem](#). Lava integrations allow for developers to build decentralized applications (dApps) with their favorite development tools while harnessing Lava's decentralized peer-to-peer network of high quality RPC providers. There is no need to sacrifice the functionality of frequently used APIs, simply install and use the specified Lava integration while building!

Usage

Taking advantage of integrations is an important way to multi-chain and decentralize your application. Due to differences in architecture, each Lava integration has its idiosyncracies. However, all of them follow the same general flow:

1. Install the supported web3 development package (i.e. `npm i viem`)
2. `import` in your project.
3. Install the Lava integration package into your project (i.e. `npm i @lavanet/lava-viem`)
4. `import`
5. Initialize the respective SDK integration object using the proper [parameters](#)
6. `const` A chart is provided below for your reference. You must provide either a `subscribedPrivateKey`
7. `or` a `validBadge`
8. `,` but not both.

✓ Required Parameters

The two primary fields that are necessary are `abadge` / `privateKey` to cover the cost of API calls and `chainIds` to let Lava know which chains to target. If you haven't already, you should sign up from the [Lava Gateway](#), where you can acquire both of these with ease!

`badge` :

```
{ badgeServerAddress :
```

```
"https://badges.lavanet.xyz"
```

```
//or your own URL projectId :
```

```
"" }
```

```
// OR
```

```
privateKey : process . env . PRIVATE_KEY
```

```
// AND
```

`chainIds` :

```
"ETH1"
```

All Options

There are numerous other options that can be configured while creating and initializing the Lava integration object. An exhaustive list follows:

Option	Required/Optional	Description
<code>privateKey</code>	Required	The private key with active subscription to be used in lieu of <code>abadge</code> .
<code>badge</code>	Required	Public URL of badge server and ID of the project you want to connect. Remove <code>privateKey</code> if <code>badge</code> is provided.
<code>chainIds</code>	Required	The ID of the chain you want to query or an array of chain IDs (e.g., "ETH1" or ["ETH1", "LAV1"])
<code>pairingListConfig</code>	Optional	The Lava pairing list config used for communicating with the Lava network. Default is Lava-providers
<code>network</code>	Optional	The network from <code>pairingListConfig</code> to be used (e.g., ["mainnet", "testnet"])
<code>geolocation</code>	Optional	The geolocation to be used (e.g., "1" for North America, "2" for Europe -> "1" is default)
<code>lavaChainId</code>	Optional	The Lava chain ID (default value is 'lava-testnet-2' for Lava Testnet)
<code>secure</code>	Optional	Communicates through HTTPS (temporary flag that will be disabled once the chain uses HTTPS by default)
<code>allowInsecureTransport</code>	Optional	Indicates to use an insecure transport when connecting the provider, for testing purposes only
<code>logLevel</code>	Optional	Log level settings (e.g., "debug", "info", "warn", "error", "success", "NoPrints")
<code>transport</code>	Optional	Transport settings for changing the method of transport provider
<code>optimizerStrategy</code>	Optional	The strategy to use to pick providers (default: balanced)
<code>maxConcurrentProviders</code>	Optional	The maximum number of providers to use concurrently (default: 3)
<code>tip</code>	Optional	You can optionally pull in an object containing all of these fields with the following code:

```
import
```

```
{
```

```
LavaSDKOptions
```

```
}
```

```
from
```

```
"@lavanet/lava-sdk" This will require installing @lavanet/lava-sdk as a dependency!
```

Repository

[github](#) :

lavanet/lava-sdk-providers [Edit this page](#) [Previous](#) [All Blogs](#) [Next \(Memid App\)](#)