

# Package: rllama (via r-universe)

March 1, 2025

**Type** Package

**Title** Access and Analyze Data from 'DeFiLlama'

**Version** 0.4.4

**Description** Provides an interface to access and analyze data from 'DeFiLlama' <<https://defillama.com>>. This package simplifies the process of fetching and manipulating 'DeFiLlama' data for further analysis and visualization.

**License** MIT + file LICENSE

**URL** <https://github.com/AlexTwoR/rllama>

**BugReports** <https://github.com/AlexTwoR/rllama/issues>

**Imports** data.table, httr

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Alexey Ivanov [aut, cre, cph]

**Maintainer** Alexey Ivanov <[twor.2r@gmail.com](mailto:twor.2r@gmail.com)>

**Date/Publication** 2024-02-02 21:00:02 UTC

**Additional\_repositories** <https://cranhaven.r-universe.dev>

**Config/pak/sysreqs** libssl-dev

**Repository** <https://cranhaven.r-universe.dev>

**RemoteUrl** <https://github.com/cranhaven/cranhaven.r-universe.dev>

**RemoteRef** package/rllama

**RemoteSha** b72b7eda5b2e50c08e8923464aacf7d842b761ac

**RemoteSubdir** rllama

## Contents

get_defillama_chain_hist_tv1 . . . . .	2
get_defillama_chain_tv1 . . . . .	3
get_defillama_protocol_hist_tv1 . . . . .	3
get_defillama_protocol_tv1 . . . . .	4
get_defillama_stablecoin_data . . . . .	5
get_defillama_stablecoin_hist_data . . . . .	5
<b>Index</b>	<b>7</b>

---

get\_defillama\_chain\_hist\_tv1  
*Chain History Data*

---

### Description

Fetches historical TVL data for a specified Blockchain in USD. The data is returned as a table sorted by date.

### Usage

```
get_defillama_chain_hist_tv1(name)
```

### Arguments

name	A string representing the unique name (name) of the protocol. This name can be obtained from the result of the <a href="#">get_defillama_chain_tv1</a> function.
------	--

### Value

A 'data.table' object containing historical TVL data of the specified chain The table includes columns for date, TVL in USD. It is sorted by date.

### Examples

```
# total tvl by date
chain_tv1 = get_defillama_chain_hist_tv1( "Ethereum" )
tail( chain_tv1 )
```

---

```
get_defillama_chain_tvl
```

*Chain TVL Data*

---

**Description**

This function retrieves TVL data for various chains from a specified API and formats it into a `data.table` for analysis.

**Usage**

```
get_defillama_chain_tvl()
```

**Value**

A `'data.table'` containing the TVL data with columns: `name`, `symbol`, `tvl`, `gecko_id`, `cmcId`, and `chainId`. The data is sorted by TVL in descending order.

**Examples**

```
chain_tvl = get_defillama_chain_tvl()
head( chain_tvl )
```

---

```
get_defillama_protocol_hist_tvl
```

*Protocol History Data*

---

**Description**

Fetches historical TVL data for a specified DeFi protocol, including TVL in USD and token amount. The data is returned as a table sorted by date.

**Usage**

```
get_defillama_protocol_hist_tvl(slug, by_token = FALSE)
```

**Arguments**

<code>slug</code>	A string representing the unique name (slug) of the protocol. This slug can be obtained from the result of the <code>get_defillama_protocol_tvl</code> function.
<code>by_token</code>	A boolean indicating whether to include token-specific data. Defaults to <code>FALSE</code> .

**Value**

A `'data.table'` object containing historical TVL data of the specified protocol. The table includes columns for `date`, TVL in USD. It is sorted by date.

**Examples**

```
# total tv1 by date
protocol_tv1 = get_defillama_protocol_hist_tv1( "lido" )
tail( protocol_tv1 )

# tv1 by token
protocol_tv1 = get_defillama_protocol_hist_tv1( "lido", by_token = TRUE )
tail( protocol_tv1 )
```

---

```
get_defillama_protocol_tv1
Protocol TVL Data
```

---

**Description**

This function retrieves Total Value Locked (TVL) data for various protocols from a specified API and formats it into a table for analysis.

**Usage**

```
get_defillama_protocol_tv1(is_cex_include = F)
```

**Arguments**

`is_cex_include` Logical; if FALSE (default), data for CEX will be excluded from the results. Set to TRUE to include CEX data.

**Value**

A 'data.table' containing TVL data with columns: symbol, tv1, slug, name, chain, and category. The data is sorted by TVL in descending order.

**Examples**

```
protocol_tv1 = get_defillama_protocol_tv1()
head( protocol_tv1 )
```

---

`get_defillama_stablecoin_data`  
*Stablecoins Data*

---

### **Description**

Retrieves stablecoin data, including details like name, symbol, price, and market cap. The data is returned as a sorted table by market cap.

### **Usage**

```
get_defillama_stablecoin_data()
```

### **Value**

A 'data.table' with stablecoin information, sorted by market capitalization.

### **Examples**

```
stablecoin_data = get_defillama_stablecoin_data()
head( stablecoin_data )
```

---

`get_defillama_stablecoin_hist_data`  
*Stablecoins History Data*

---

### **Description**

Retrieves stablecoin data, including details like name, symbol, price, and market cap. The data is returned as a sorted table.

### **Usage**

```
get_defillama_stablecoin_hist_data(stablecoin_id)
```

### **Arguments**

`stablecoin_id` An integer representing the ID of the stablecoin. This ID can be obtained from the result of the [get\\_defillama\\_stablecoin\\_data](#) function.

### **Value**

A 'data.table' object containing historical data of the specified stablecoin. The table is sorted by date.

**Examples**

```
stablecoin_hist_data = get_defillama_stablecoin_hist_data( 5 )  
head( stablecoin_hist_data )
```

# Index

`get_defillama_chain_hist_tvl`, 2  
`get_defillama_chain_tvl`, 2, 3  
`get_defillama_protocol_hist_tvl`, 3  
`get_defillama_protocol_tvl`, 3, 4  
`get_defillama_stablecoin_data`, 5, 5  
`get_defillama_stablecoin_hist_data`, 5