

eth_getStorageAt

Returns the value from a storage position at a given address.

Parameters

- address
 - : [Required]
 -] A string representing the address (20 bytes) of the storage.
- storage position
 - : [Required]
 -] A hexadecimal code of the position in the storage.
- block parameter
 - : [Required]
 -] A hexadecimal block number, or one of the string tags latest, earliest, pending, safe, or finalized
 - . See the [default block parameter](#)
 - .

Returns

storage value : A hexadecimal equivalent of the integer indicating the value of the storage position at the provided address.

Example

Replace YOUR-API-KEY with an API key from your [Infura dashboard](#) .

Request

- cURL

```
curl https://palm-mainnet.infura.io/v3/YOUR-API-KEY \-X POST \-H "Content-Type: application/json" \-d '{"jsonrpc":"2.0","method":"eth_getStorageAt","params":["0x295a70b2de5e3953354a6a8344e616ed314d7251", "0x6661e9d6d8b923d5bbaab1b96e1dd51ff6ea2a93520fdc9eb75d059238b8c5e9", "0x65a8db"],"id":1}'
```

Response

- JSON

```
{ "jsonrpc": "2.0", "id": 1, "result": "0x0000000000000000000000000000000000000000000000000000000000000000" }
```

Additional notes

Calculating the correct position depends on the storage to retrieve. Consider the following contract deployed at 0x295a70b2de5e3953354a6a8344e616ed314d7251 by address 0x391694e7e0b0cce554cb130d723a9d27458f9298 ,

contract

Storage

```
{ uint pos0 ; mapping ( address
```

```
=>
```

```
uint ) pos1 ;
```

```
function
```

```
Storage ( )
```

```
{ pos0 =
```

```
1234 ; pos1 [ msg . sender ]
```

5678 ; } } Retrieving the value of pos0 is straight forward:

```
curl https://mainnet.infura.io/v3/YOUR-API-KEY \-X POST \-H "Content-Type: application/json" \-d '{"jsonrpc":"2.0",
"method": "eth_getStorageAt", "params": ["0x295a70b2de5e3953354a6a8344e616ed314d7251", "0x0", "0x65a8db"], "id":
1}' { "jsonrpc": "2.0", "id": 1, "result": "0x000000000000000000000000000000000000000000000000000000000000004d2" }
```

Retrieving an element of the map is harder. The position of an element in the map is calculated with:

keccak (LeftPad32 (key ,

0),

LeftPad32 (map position ,

0)) This means to retrieve the storage onpos1["0x391694e7e0b0cce554cb130d723a9d27458f9298"] we need to calculate the position with:

[illegible][illegible]

```
var key =
```

[illegible]

+

[illegible]

```
web3 . sha3 ( key ,
```

```
{ "encoding" :
```

```
"hex" } ) "0x6661e9d6d8b923d5bbaab1b96e1dd51ff6ea2a93520fdc9eb75d059238b8c5e9" Now to fetch the storage:
```

[illegible]

Last updated on Apr 19, 2024 [Previous eth_getLogs](#) [Next eth_getTransactionByHash](#)