Loading…

AI Assist is now on Stack Overflow. Start a chat to get instant answers from across the network. Sign up to save and share your chats.

Communities for your favorite technologies. Explore all Collectives

Stack Internal

Stack Overflow for Teams is now called **Stack Internal**. Bring the best of human thought and AI automation together at your work.

Try for free Learn more

Stack Internal

Bring the best of human thought and AI automation together at your work. Learn more

Collectives™ on Stack Overflow

Find centralized, trusted content and collaborate around the technologies you use most.

Learn more about Collectives

Stack Internal

Knowledge at work

Bring the best of human thought and AI automation together at your work.

Explore Stack Internal

How to define &mut [u8] in Rust

Asked 5 years, 3 months ago

Modified 5 years, 3 months ago

Viewed 3k times

1

I want to call `openssl::sign::Signer::sign`, which has the signature:

```
pub fn sign(&self, buf: &mut [u8]) -> Result<usize, ErrorStack>
```

I'm trying to hand over `buf`. How to I make it `&mut [u8]`?

cargo.toml

```
[dependencies]
openssl = { version = "0.10", features = ["vendored"] }
```

src/main.rs

```
use openssl::{base64, sign::Signer};

fn main() {
    let mut buffer = [];
    let str = base64::encode_block(signer.sign(&mut buffer));
}
```

But get an error:

```
openssl::base64::encode_block(signer.sign(&mut buffer));
                              ^^^^^^^^^^^^^^^^^^^^ expected `&[u8]`, found enum `std::result::Result`
```

- rust

Share

Improve this question

Follow

edited Aug 20, 2020 at 19:10

Jason

5,685 22 gold badges 36 36 silver badges 44 44 bronze badges

asked Aug 20, 2020 at 5:01

Nick

29 11 silver badge 3 3 bronze badges

3

> `sign` doesn't return the modified `buffer`, it modifies `buffer` over buffer's mutable reference, `signer.sign(&mut buffer);//handle result in this line` then use it like this `let encoded_str= openssl::base64::encode_block(&buffer);`
>
> Ömer Erden – Ömer Erden
>
> 2020-08-20 06:07:09 +00:00
>
> Commented Aug 20, 2020 at 6:07

> 1
>
> Could you tell us how `Signer` is created? It's currently not included in the post.
>
> Jason – Jason
>
> 2020-08-20 06:29:52 +00:00
>
> Commented Aug 20, 2020 at 6:29

> Please edit your question to include a minimal reproducible example. The code that you have here does not reproduce the error you have shown, as you instead get a different error due to `signer` not existing in your code. (playground)
>
> Herohtar – Herohtar
>
> 2020-08-20 21:06:00 +00:00
>
> Commented Aug 20, 2020 at 21:06

Add a comment |

1 Answer

Sorted by: Reset to default

Highest score (default) Trending (recent votes count more) Date modified (newest first) Date created (oldest first)

3

I'm not familiar with this crate, but <u>base64::encode_block</u> expects `&[u8]`, while the return value of <u>Signer::sign</u> is a `Result` that returns the *number of bytes written* as `usize`.

I'm not sure if the *number of bytes written* is what you want to encode?

If you want to encode `buffer` to `base64`, you could do:

```
use openssl;

fn main() {
  let mut buf = [104, 101, 108, 108, 111];
  let encoded = openssl::base64::encode_block(&buf);

  println!("{}", encoded);
}
```

That should give you:

```
aGVsbG8=
```

Which is `hello` as `base64`.

Or, as an example, if you want to `sign` the given `buf` and encode that as `base64`:

```
use openssl::{base64, hash::MessageDigest, pkey::PKey, rsa::Rsa, sign::Signer};

fn main() {
    let keypair = Rsa::generate(2048).unwrap();
    let keypair = PKey::from_rsa(keypair).unwrap();
    let signer = Signer::new(MessageDigest::sha256(), &keypair).unwrap();

    let mut buf = [104, 101, 108, 108, 111].repeat(52);

    let written = signer.sign(&mut buf).unwrap();
    println!("bytes written: {}", written);

    let encoded = base64::encode_block(&buf);
    println!("base64: {}", encoded);
}
```

Which returns:

```
bytes written: 256
base64: wB4lBbyzpfRPInxhwm0XVKI3dQYqcUZWNdKyb4RTGDUmwq1DDDbQjMRmFBobRse3pNOxoMy+QQNSXsgI46b75hPfkar9TUowrIGk4Y+ZWWX/cwNJCJazC1
```

Share
Improve this answer
Follow
edited Aug 20, 2020 at 6:30
answered Aug 20, 2020 at 5:54
Jason
5,68522 gold badges3636 silver badges4444 bronze badges
Sign up to request clarification or add additional context in comments.

2 Comments

Add a comment

Nick

Nick Over a year ago

Thanks, and my apologies for totally noob.I feel understand how to question at this website.I solved this problem by your help!

2020-08-21T05:04:16.083Z+00:00

0

Reply

- Copy link

Jason

Jason Over a year ago

@Nick, I'm glad you managed to solve it! Have a great day.

2020-08-21T06:55:26.153Z+00:00

0

Reply

- Copy link

Your Answer

Thanks for contributing an answer to Stack Overflow!

- Please be sure to *answer the question.* Provide details and share your research!

But *avoid* …

- Asking for help, clarification, or responding to other answers.
- Making statements based on opinion; back them up with references or personal experience.

To learn more, see our .

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

- rust

See similar questions with these tags.

Related

[2](#)
[Convert "*mut *mut f32" into "&[&[f32]]"](#)
[0](#)
[Can I get rid of "mut" here?](#)
[7](#)
[casting *mut u8 to &[u8] without std](#)
[3](#)
[Unable to pass a Vec<u8> into a funtction taking &mut Read](#)
[2](#)
[How do I convert from a &mut T to a &mut U type?](#)
[2](#)
[Usage of 'mut' keyword in Rust](#)
[3](#)
[How to create *mut u8 from Rust string?](#)
[2](#)
[Pin::as_mut(&mut self)](#)
[7](#)
[Why can't I convert &&mut to &&?](#)
[14](#)
[Trouble understanding &mut &mut reference](#)

[Hot Network Questions](#)

- [Why are left and right cosets different in the non-normal case?](#)
- [TV program where real actors interact with 2-Dimensional characters](#)
- [Proportional odds logistic regression for ordered category outcome - how to convert odds ratios to probabilities in this case?](#)
- [Installing root certificates by government](#)
- [Is there a website where we can find the average stipend/salary for the PhD and Postdoc for each country?](#)
- [Is my homebrew Species "Attuned" balanced for D&D 2024?](#)
- [Where in the world is the chicken?](#)
- [Looking for a proper formal substitute for "quick fix" for formal letter](#)
- [How to use Bézier curves in `TikZ` to draw a quotation mark?](#)
- [Displaying features in a layer of geometrytype = GeometryCollection25D](#)
- [What does Felix mean by telling Bond to pick a contact point while "standing up" in Diamonds Are Forever?](#)
- [variable name change causes segfault](#)
- [Statistical testing for bimodal/multimodal sample](#)
- [Does short range teleporting end floating disk?](#)
- [Can enemies damage each other in Breath of the Wild?](#)
- [I can't add a loop cut on one face](#)
- [Obscure method of supression of transient voltage spike in CLC-circuit](#)
- [How does supporting "One China" work without also supporting Taiwan-China reunification?](#)
- [A very weird integral equation](#)
- [Is there any point in sharing a scientific discovery as a "nobody"?](#)
- [When the fuel governor fails, why is autorotation the only option for landing?](#)
- [How to prevent a batch file call from escaping a caret character?](#)
- [The state of icy worlds during differentiation and accretion, and how they survive with their water](#)
- [How to protect 100-year-old wooden floors for the next 100 years?](#)

[more hot questions](#)

[Question feed](#)

Subscribe to RSS

Question feed

To subscribe to this RSS feed, copy and paste this URL into your RSS reader.

Stack Overflow

- [Questions](#)
- [Help](#)
- [Chat](#)

Business

- [Stack Internal](#)
- [Stack Data Licensing](#)
- [Stack Ads](#)

Company

- [About](#)
- [Press](#)
- [Work Here](#)
- [Legal](#)
- [Privacy Policy](#)
- [Terms of Service](#)
- [Contact Us](#)
- Cookie Settings
- [Cookie Policy](#)

Stack Exchange Network

- [Technology](#)
- [Culture & recreation](#)
- [Life & arts](#)
- [Science](#)
- [Professional](#)
- [Business](#)
- [API](#)
- [Data](#)
- [Blog](#)
- [Facebook](#)
- [Twitter](#)
- [LinkedIn](#)
- [Instagram](#)