New issue

Jump to bottom

Ipv4Addr: Incorrect Parsing for Octal format IP string #83648

Closed xu-cheng opened this issue on Mar 29, 2021 · 3 comments · Fixed by #83652

Labels C-bug T-libs-api

xu-cheng commented on Mar 29, 2021 • edited Contributor

This issue is inspired by this blog.

Due to the specification, leading zero in IP string is interpreted as octal literals. So a IP address `0127.0.0.1` actually means `87.0.0.1`. As shown in the following example:

```
❯ ping 0127.0.0.1
PING 0127.0.0.1 (87.0.0.1): 56 data bytes
```

However, the `Ipv4Addr` from the std library will recognize it as `127.0.0.1` instead. A simple code to demo the situation (playground link):

```
use std::net::Ipv4Addr;

fn parse(input: &str) {
    let ip: Option<Ipv4Addr> = input.parse().ok();
    println!("{} -> {:?}", input, ip);

}

fn main() {
    parse("127.0.0.1");
    parse("0127.0.0.1");
}
```

I expected to see this happen:

```
127.0.0.1 -> Some(127.0.0.1)
0127.0.0.1 -> Some(87.0.0.1)
```

Instead, this happened:

```
127.0.0.1 -> Some(127.0.0.1)
0127.0.0.1 -> Some(127.0.0.1)
```

Noted this bug may cause security vulnerabilities in certain cases. For example, a Rust program uses `Ipv4Addr` doing some sanity check then passing the user string to other library or program.

Furthermore, the specification actually also allows hex format in IP string.

Meta

`rustc --version --verbose`:

```
rustc 1.51.0 (2fd73fabe 2021-03-23)
```

👍 3

xu-cheng added the **C-bug** label on Mar 29, 2021

jonas-schievink added the **T-libs-api** label on Mar 29, 2021

xu-cheng commented on Mar 29, 2021 Contributor Author

While the fix should be quite straightforward, there are two possible solutions:

- Make the parsing conform the standard.
- OR only accept IP string in decimal format. Certain software chooses this approach because it may improve the security. See https://tools.ietf.org/html/rfc6943#section-3.1.1

joshtriplett commented on Mar 29, 2021 Member

It's not at all obvious that we *should* parse octal IP addresses. Seems exceedingly unlikely to come up outside of security advisories. I would venture a guess that if we added this, far more people would be tripped up by it happening unexpectedly than would ever use it intentionally.

👍 1

xu-cheng commented on Mar 29, 2021 Contributor Author

> I would venture a guess that if we added this, far more people would be tripped up by it happening unexpectedly than would ever use it intentionally.

I agree. So I think disallowing octal string like `inet_pton` should be better approach.

Nevertheless, the current implementation in Rust std library should be considered as a (low-risk?) security vulnerability.

👍 1

xu-cheng added a commit to xu-cheng/rust that referenced this issue on Mar 29, 2021

`Disallow octal format in Ipv4 string` … f72679b

xu-cheng mentioned this issue on Mar 29, 2021

Disallow octal format in Ipv4 string #83652

Merged

xu-cheng added a commit to xu-cheng/rust that referenced this issue on Mar 29, 2021

`Disallow octal format in Ipv4 string` … f375592

bors added a commit to rust-lang-ci/rust that referenced this issue on Mar 30, 2021

`Auto merge of rust-lang#83652 - xu-cheng:ipv4-octal, r=sfackler` … ✓ 74874a6

bors closed this as completed in `974192c` on Mar 30, 2021

Assignees

No one assigned

Labels

C-bug T-libs-api

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

Disallow octal format in Ipv4 string
xu-cheng/rust

3 participants