



Why I hated Rust as a malware developer

RUMP - Pass the Salt 2025

02/07/2025

Don't try to convince me

First :

- This presentation is just about my feelings, don't take it personally
- Doesn't reflect the views of my employer

Second :

- Just to present some problems I had when switching from C to Rust

I'm easily offended

I don't like people judging my code.

I'm easily offended

In C :

```
HANDLE handle = OpenProcess(PROCESS_TERMINATE, FALSE, pid as u32);
```

In Rust :

```
let handle = unsafe {};
```

I'm easily offended

In C :

```
HANDLE handle = OpenProcess(PROCESS_TERMINATE, FALSE, pid as u32);
```

In Rust :

```
let handle = unsafe {OpenProcess(PROCESS_TERMINATE, FALSE, pid as u32)}?;
```

Let me mess with memory !!!

In C :

```
importDescriptor = (PIMAGE_IMPORT_DESCRIPTOR)(ntHeaders->OptionalHeader.DataDirectory[IMAGE_DIRECTORY_ENTRY_IMPORT].VirtualAddress + (DWORD_PTR)dllBase);
```

In Rust :

```
UNSAFE {} (!!!!)
```

Let me mess with memory !!!

In C :

```
importDescriptor = (PIMAGE_IMPORT_DESCRIPTOR)(ntHeaders->OptionalHeader.DataDirectory[IMAGE_DIRECTORY_ENTRY_IMPORT].VirtualAddress + (DWORD_PTR)dllBase);
```

In Rust :

```
UNSAFE {  
    let mut importDirectory = (dllBase as usize + (*nt_headers).OptionalHeader.DataDirectory[IMAGE_DIRECTORY_ENTRY_IMPORT as usize].VirtualAddress as usize) as *mut IMAGE_IMPORT_DESCRIPTOR;  
}
```

Handling imports

In Rust :

```
use core::{ffi::c_void, ptr::null_mut, slice::from_raw_parts, mem::{transmute, size_of}, arch::asm};
use ntapi::{ntpebteb::PEB, ntpsapi::PEB_LDR_DATA, ntldr::LDR_DATA_TABLE_ENTRY};
use windows_sys::{
    core::PCSTR,
    Win32::{
        Foundation::{BOOL, FARPROC, HANDLE, HINSTANCE},
        System::{
            Diagnostics::{Debug::{
                IMAGE_NT_HEADERS64, IMAGE_SCN_MEM_EXECUTE,
                IMAGE_SCN_MEM_READ, IMAGE_SCN_MEM_WRITE, IMAGE_SECTION_HEADER, IMAGE_DIRECTORY_ENTRY_EXPORT, IMAGE_DIRECTORY_ENTRY_BASERELOC, IMAGE_DIRECTORY_ENTRY_IMPORT,
            }},
            Memory::{
                MEM_COMMIT, MEM_RESERVE, PAGE_EXECUTE, PAGE_EXECUTE_READ,
                PAGE_EXECUTE_READWRITE, PAGE_EXECUTE_WRITECOPY, PAGE_PROTECTION_FLAGS,
                PAGE_READONLY, PAGE_READWRITE, PAGE_WRITECOPY, VIRTUAL_ALLOCATION_TYPE,
                VIRTUAL_FREE_TYPE,
            },
            SystemServices::{
                DLL_PROCESS_ATTACH, IMAGE_DOS_HEADER, IMAGE_DOS_SIGNATURE, IMAGE_NT_SIGNATURE, IMAGE_EXPORT_DIRECTORY, IMAGE_BASE_RELOCATION, IMAGE_REL_BASED_HIGHLOW, IMAGE_REL_BASED_DIR64, IMAGE_IMPORT_DESCRIPTOR, IMAGE_ORDINAL_FLAG64, IMAGE_IMPORT_BY_NAME,
            },
            WindowsProgramming::IMAGE_THUNK_DATA64,
        },
    },
};
```

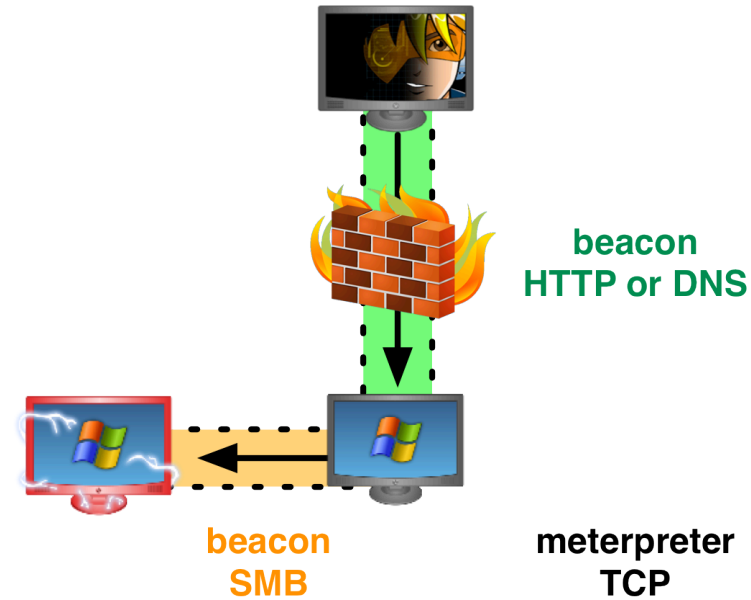
Handling imports

In C :

```
#include <windows.h>
```

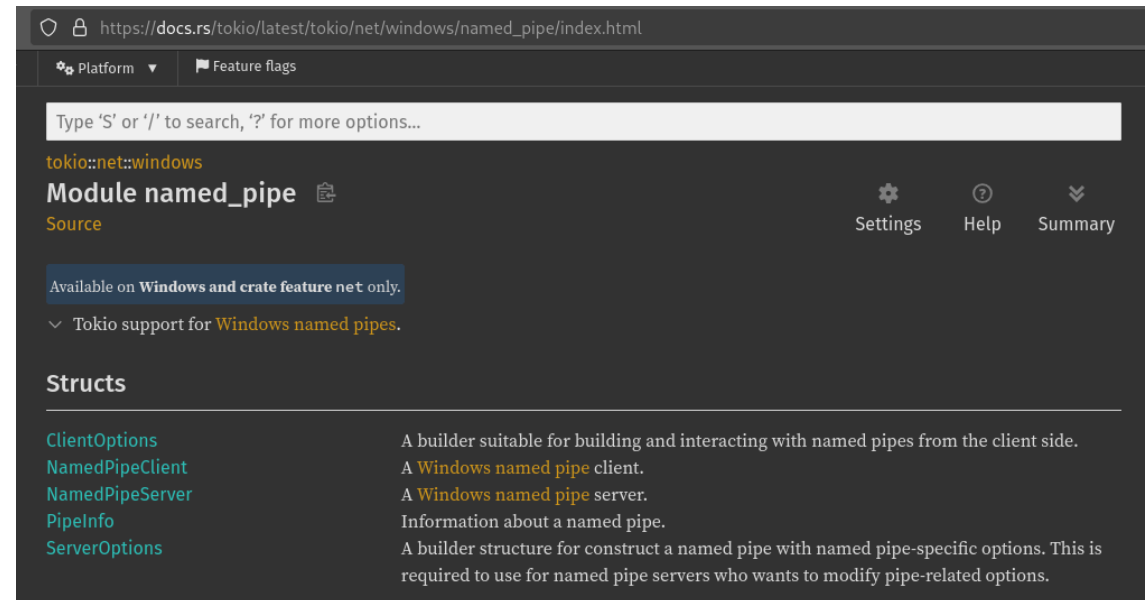
SMB nightmare

A wanted feature for C2 devs : SMB linking



Reach a PC that cannot beacon via HTTP

SMB nightmare



The screenshot shows a web browser displaying the Tokio documentation for the `named_pipe` module. The URL is `https://docs.rs/tokio/latest/tokio/net/windows/named_pipe/index.html`. The page title is `Module named_pipe` under the `tokio::net::windows` namespace. A search bar is present at the top. Below the title, there are links for `Source`, `Settings`, `Help`, and `Summary`. A blue box indicates that the module is available on Windows and the `crate feature net` only. A dropdown menu shows 'Tokio support for Windows named pipes'. The 'Structs' section lists several structs with their descriptions:

Struct	Description
ClientOptions	A builder suitable for building and interacting with named pipes from the client side.
NamedPipeClient	A <code>Windows named pipe</code> client.
NamedPipeServer	A <code>Windows named pipe</code> server.
PipeInfo	Information about a named pipe.
ServerOptions	A builder structure for construct a named pipe with named pipe-specific options. This is required to use for named pipe servers who wants to modify pipe-related options.

Cool ! A library !

SMB nightmare

```
panic assert failed left: 4096 right: 0 - when sending > 4096 bytes  
in windows pipe message mode #6460
```

Open

[tokio-rs/mio#1778](#)

Not cool !

SMB nightmare

In the end, I copied code from C code and wrapped it in ...

UNSAFE

Thank you for listening !

That's all folks !