Equivalent to C's `void` type when used as a [pointer].

In essence, `*const c_void` is equivalent to C's `const void*` and `*mut c_void` is equivalent to C's `void*`. That said, this is *not* the same as C's `void` return type, which is Rust's `()` type.

To model pointers to opaque types in FFI, until `extern type` is stabilized, it is recommended to use a newtype wrapper around an empty byte array. See the Nomicon for details.

One could use `std::os::raw::c_void` if they want to support old Rust compiler down to 1.1.0. After Rust 1.30.0, it was re-exported by this definition. For more information, please read RFC 2521.