If you ask for STANDARD_RIGHTS_REQUIRED, you may as well ask for the moon

devblogs.microsoft.com/oldnewthing/20080227-00

February 27, 2008

Raymond Chen

One of the predefined security access masks is `STANDARD_RIGHTS_REQUIRED`. You see it used in defining the `_ALL_ACCESS` masks for various objects. Here are just a few examples:

```
#define PROCESS_ALL_ACCESS        (STANDARD_RIGHTS_REQUIRED | SYNCHRONIZE | \
                                   0xFFF)
#define EVENT_ALL_ACCESS (STANDARD_RIGHTS_REQUIRED|SYNCHRONIZE|0x3)
#define FILE_ALL_ACCESS (STANDARD_RIGHTS_REQUIRED | SYNCHRONIZE | 0x1FF)
```

The `STANDARD_RIGHTS_REQUIRED` mask is meant to be used when defining access masks for object types. I'm guessing it's called `STANDARD_RIGHTS_REQUIRED` because it's the set of access masks that all securable objects must support. Look at the documentation or just at the definition:

```
#define DELETE                           (0x00010000L)
#define READ_CONTROL                     (0x00020000L)
#define WRITE_DAC                        (0x00040000L)
#define WRITE_OWNER                      (0x00080000L)
#define STANDARD_RIGHTS_REQUIRED         (0x000F0000L)
```

Notice that `STANDARD_RIGHTS_REQUIRED` is just an abbreviation for the union of the four access bits `DELETE | READ_CONTROL | WRITE_DAC | WRITE_OWNER`.

Now that you see what it's for, you can also see what it's **not** for: You're not expected to pass it as the mask of **requested** access bits when you attempt to open an object. In other words, the following is wrong:

```
// wrong!
HANDLE hProcess =
    OpenProcess(dwProcessId, FALSE,
                STANDARD_RIGHTS_REQUIRED | PROCESS_QUERY_INFORMATION);
```

The person writing this code probably thought, "Well, I just want to be able to query information, so I need to pass `PROCESS_QUERY_INFORMATION`. There's this other thing here called `STANDARD_RIGHTS_REQUIRED`; since it's required, I'll pass that too."

The “required”ness of `STANDARD_RIGHTS_REQUIRED` doesn’t apply to you, the program opening the object. It applies to the person who is designing the object.

Your attempt to be a “good security citizen” and ask only for the access you need (namely, `PROCESS_QUERY_INFORMATION`) has backfired due to the addition of `STANDARD_RIGHTS_REQUIRED`. If you ask for `STANDARD_RIGHTS_REQUIRED`, you are asking for **everything**.

Why is that? Notice that `STANDARD_RIGHTS_REQUIRED` includes `WRITE_DAC`. If you have `WRITE_DAC` permission, that means that you have permission to change the security descriptor on the object, at which point you totally ownzor it. You want `PROCESS_VM_WRITE` access but the security descriptor doesn’t let you? No problem. Just set a new security descriptor that grants you `PROCESS_ALL_ACCESS` to the process object. Tada! You now have all the access in the world.

Moral of the story: Don’t ask for `STANDARD_RIGHTS_REQUIRED`, because only somebody with full control will be able to get it. Ask for what you actually want.

Raymond Chen

Follow