Why do STANDARD_RIGHTS_READ, STANDARD_RIGHTS_WRITE, and STANDARD_RIGHTS_EXECUTE have the same values?

devblogs.microsoft.com/oldnewthing/20240411-00

April 11, 2024

Raymond Chen

Windows defines values for the access rights `STANDARD_RIGHTS_READ`, `STANDARD_RIGHTS_WRITE`, and `STANDARD_RIGHTS_EXECUTE`. But if you look at their definitions, they are all defined to be the same thing, namely, `READ_CONTROL`. How can this possibly make sense? Certainly read, write, and execute rights should be different, shouldn't they?

They should, but that's not what the `STANDARD_RIGHTS_*` values are for.

What these values are trying to say is "Every securable object should include `STANDARD_RIGHTS_READ` in their generic read access mask, `STANDARD_RIGHTS_WRITE` in their generic write access mask, and `STANDARD_RIGHTS_EXECUTE` in their generic execute access mask." Confusingly, `STANDARD_RIGHTS_ALL` is just a mask of all the standard rights, not the "mask that must be present in the generic all access mask." It is `STANDARD_RIGHTS_REQUIRED` that you have to put in your generic all access mask.

For example, the file access rights are defined as

```
#define FILE_GENERIC_READ       (STANDARD_RIGHTS_READ     |\
                                 FILE_READ_DATA           |\
                                 FILE_READ_ATTRIBUTES     |\
                                 FILE_READ_EA             |\
                                 SYNCHRONIZE)


#define FILE_GENERIC_WRITE      (STANDARD_RIGHTS_WRITE    |\
                                 FILE_WRITE_DATA          |\
                                 FILE_WRITE_ATTRIBUTES    |\
                                 FILE_WRITE_EA            |\
                                 FILE_APPEND_DATA         |\
                                 SYNCHRONIZE)


#define FILE_GENERIC_EXECUTE    (STANDARD_RIGHTS_EXECUTE  |\
                                 FILE_READ_ATTRIBUTES     |\
                                 FILE_EXECUTE             |\
                                 SYNCHRONIZE)

#define FILE_ALL_ACCESS         (STANDARD_RIGHTS_REQUIRED |\
                                 SYNCHRONIZE              |\
                                 0x1FF)

const GENERIC_MAPPING FileGenericMapping =
{
    FILE_GENERIC_READ,
    FILE_GENERIC_WRITE,
    FILE_GENERIC_EXECUTE,
    FILE_ALL_ACCESS,
};
```

The "generic read" file access includes `STANDARD_RIGHTS_READ`, plus any other read rights specific to files. Similarly for "write", "execute", and "all" access.

Now, it so happens that the only mandatory access right for read, write, and execute is `READ_CONTROL`, so that's why all three of the macros expand to the same underlying value.

But you weren't supposed to care about that. Just include the corresponding standard rights in each of the four levels of access, and you're all set.