

Forum:[Hack Forums](#)

Thread: [ntdll!ntquerydirectoryfile, how initialize buffer?](#)

poker - January 8, 2021, 3:29 pm

Hi to all,

I'm using NtQueryDirectoryFile from ntdll (userland mode) to get all items in a specific path.

This is the official documentation

```
Code __kernel_entry NTSYSCALLAPI NTSTATUS NtQueryDirectoryFile(
    HANDLE          FileHandle,
    HANDLE          Event,
    PIO_APC_ROUTINE ApcRoutine,
    PVOID           ApcContext,
    PIO_STATUS_BLOCK IoStatusBlock,
    PVOID           FileInformation,
    ULONG           Length,
    FILE_INFORMATION_CLASS FileInformationClass,
    BOOLEAN         ReturnSingleEntry,
    PUNICODE_STRING FileName,
    BOOLEAN         RestartScan
);
```

I dont know whats is the correct way to initialize FileInformation param, because, if i create a variable in stack, like this:

```
Code char buffer[100000];
NtQueryDirectoryFile(hFile, NULL, NULL, NULL, &ioStatusBlock, (void*)buffer, 100000 *
sizeof(char), FileBothDirectoryInformation, FALSE, NULL, FALSE);
```

It's working, but when i call this function in a recursive way(for example, if i want to delete all item in a desired path), like this example i have a bad_alloc exception ???

```
Code function RemoveInPath(path){
    array_items = GetAllItems(path);
    for(int i = 0; i < array_items.length(); i++){
        if(array_items[i].isDirectory()) RemoveInPath(array_items[i].path);
    }
    ....
    ...
    ...
    delete array_items;
}
```

In this code i can have multiples array_items and this Code char buffer[100000]; doesn't help.

I was looking myself, but i cant find the solution (i cannot post links):

- ExAllocatePool
- I think this only can be used in kernel mode

- WCHAR Buffer[8192]
- why 8192 ?? There are a maximum size for this param?

In other sites i saw something like this:

if you want, for example _FILE_BOTH_DIR_INFORMATION:

- _FILE_BOTH_DIR_INFORMATION* buff = 512 * sizeof(_FILE_BOTH_DIR_INFORMATION);
- Why 512?

Thanks for reading