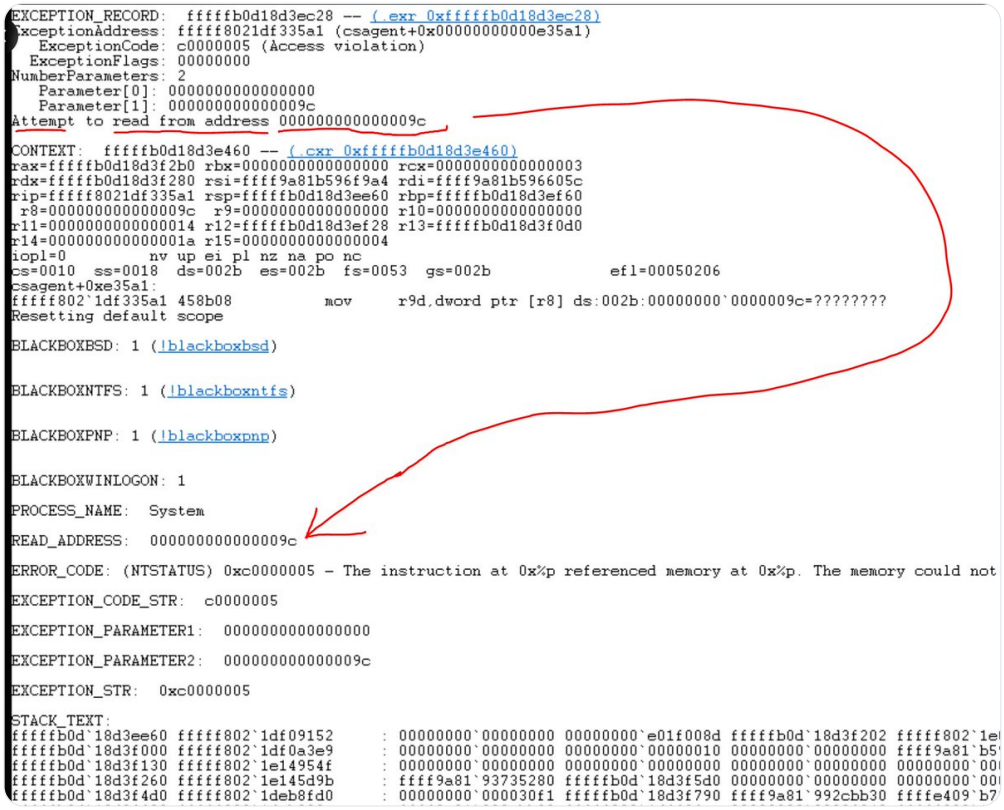
Zach Vorhies / Google Whistleblower @Perpetualmaniac

Jul 19, 2024 • 17 tweets • Perpetualmaniac/status/1814376668095754753

Crowdstrike Analysis:

It was a NULL pointer from the memory unsafe C++ language.

Since I am a professional C++ programmer, let me decode this stack trace dump for you.



Memory in your computer is laid out as one giant array of numbers. We represent these numbers here as hexadecimal, which is base 16 (hexadecimal) because it's easier to work with... for reasons.

The problem area? The computer tried to read memory address 0x9c (aka 156).

Why is this bad?

This is an invalid region of memory for any program. Any program that tries to read from this region WILL IMMEDIATELY GET KILLED BY WINDOWS.

That is what you see here with this stack dump.

So why is memory address 0x9c trying to be read from? Well because... programmer error.

It turns out that C++, the language crowdstrike is using, likes to use address 0x0 as a special value to mean "there's nothing here", don't try to access it or you'll die.
Programmers in C++ are supposed to check for this when they pass objects around by "checking full null".

Usually you'll see something like this:

```
string* p = get_name();

if (p == NULL) { print("Could not get name"); }
```

The string* part means we have a "pointer" to the start of the string value. If it's null, then there's nothing there, don't try to access it.

So let's take a generic object with stuff in it:

```
struct Obj {
int a;
int b;
};
```

if we create a pointer to it:

```
Obj* obj = new Obj();
```

We can get it's start address, let's say its something random like 0x9030=36912 (I'm using small numbers)

Then the address of:
obj is 0x9030
obj->a is 0x9030 + 0x4
obj->b is 0x9030 + 0x8

Each member is an offset from the start address.

Now let's assume the following:

```
Obj* obj = NULL;
```

Then the address of:
obj is 0
obj->a is 0 + 4
obj->b is 0 + 8

So if I do this on a NULL pointer:

```
print(obj->a);
```

The program stack dump like what you'll see above. It will cannot read value 0x000000004

In this stack dump you see that it's trying to read memory value 0x9c. In human numbers, this is the value 156.

So what happened is that the programmer forgot to check that the object it's working with isn't valid, it tried to access one of the objects member variables...

NULL + 0x9C = 0x9C = 156.

That's an invalid region of memory.

And what's bad about this is that this is a special program called a system driver, which has PRIVLIDGED access to the computer. So the operating system is forced to, out of an abundance of caution, crash immediately

This is what is causing the blue screen of death. A computer can recover from a crash in non-privileged code by simply terminating the program, but not a system driver. When your computer crashes, 95% of the time it's because it's a crash in the system drivers.

If the programmer had done a check for NULL, or if they used modern tooling that checks these sorts of things, it could have been caught. But somehow it made it into production and then got pushed as a forced update by Crowdstrike... OOPS!

The fix going forward is that Microsoft needs to have better policies to roll back defective drivers and not just raw dog risky updates to customers.

Crowdstrike will likely promote their code safety officer to put in code sanitization tools that will catch this automatically.

And Crowdstrike will likely take a hard look at rewriting their system driver from what it currently is, C++ to a more modern language like Rust, which doesn't have this problem.

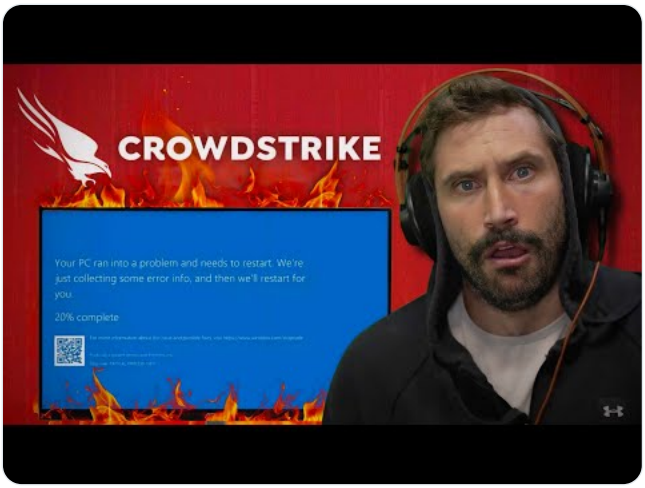
For people looking for a conspiracy, the replacement language for C++, Rust, is compromised by a cabal of woke tards that are doing strange things.

It's possible this could be a plot to move mission-critical code to Rust. It's the only other language Linux is allowing, other than C. But who knows.

https://x.com/Perpetualmaniac/status/1814405221738786984

Hat tip to @ThePrimeagen who first posted this stack trace and mentioned how few could actually understand it.

IMHO: He has the best daily youtube for talking about software. Give his channel a sub, you won’t be disappointed.



https://www.youtube.com/embed/3N4m5k9GAW0

• • •