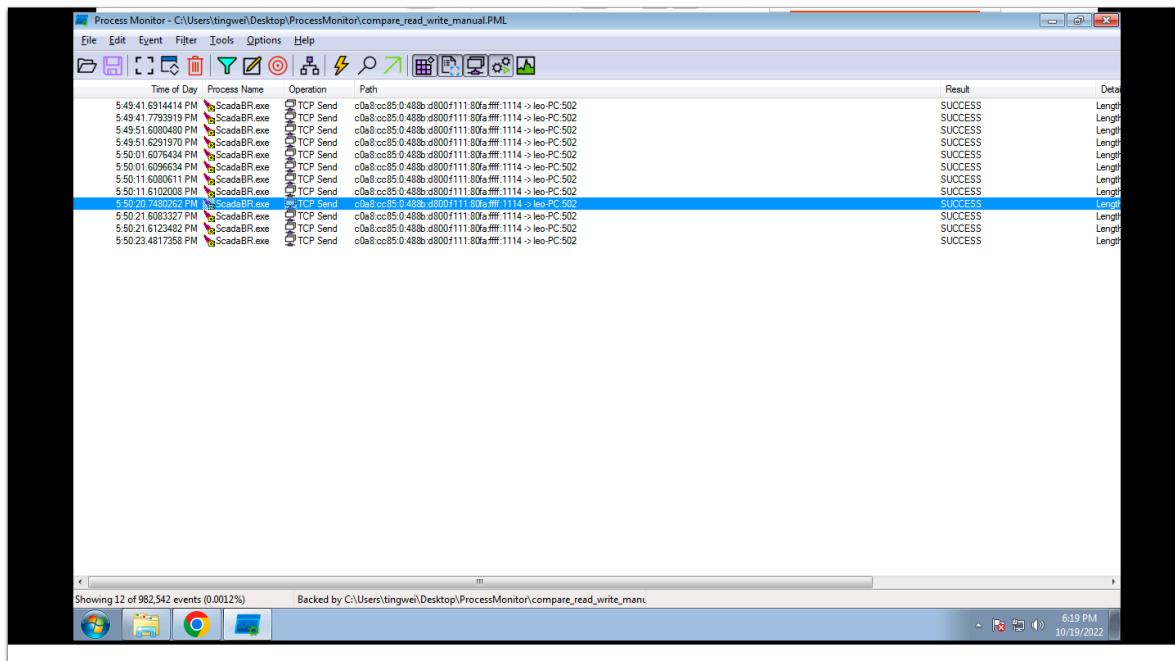


Identify a WRITE API call

Argument Analysis

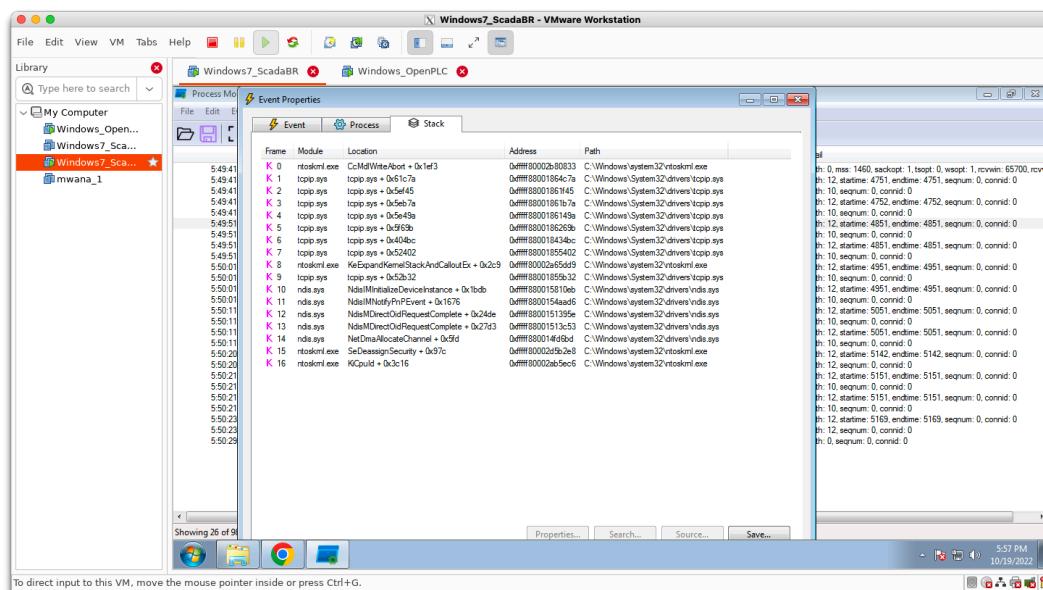
First, I used Process Monitor to see if there is any difference between READ API calls and WRITE API calls (compare_read_write_manual.pcapng, compare_read_write_manual.PML)

Here's all the TCP Send API calls.

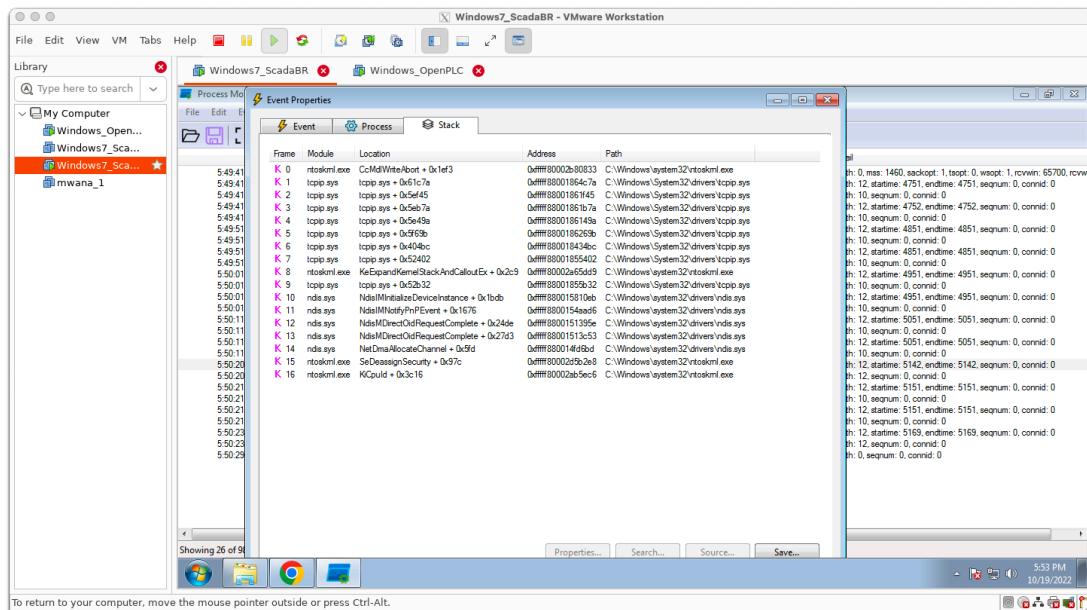


I compare every column and stack. There is no information about WRITE and READ, and there's no arguments information. There is no clear identifier to differentiate the two, most of them are the same.

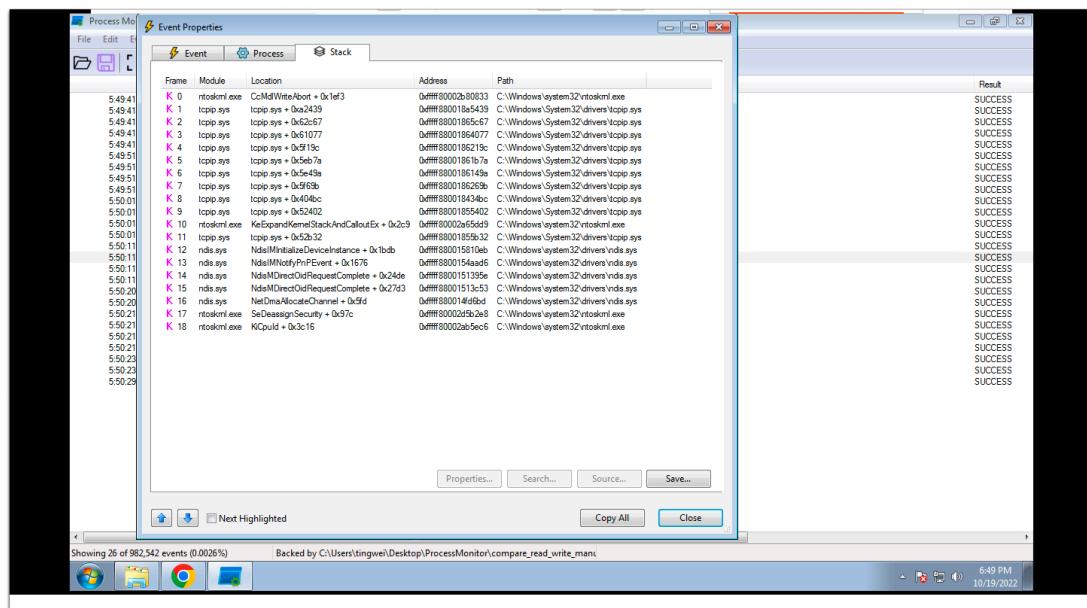
READ TCP Send



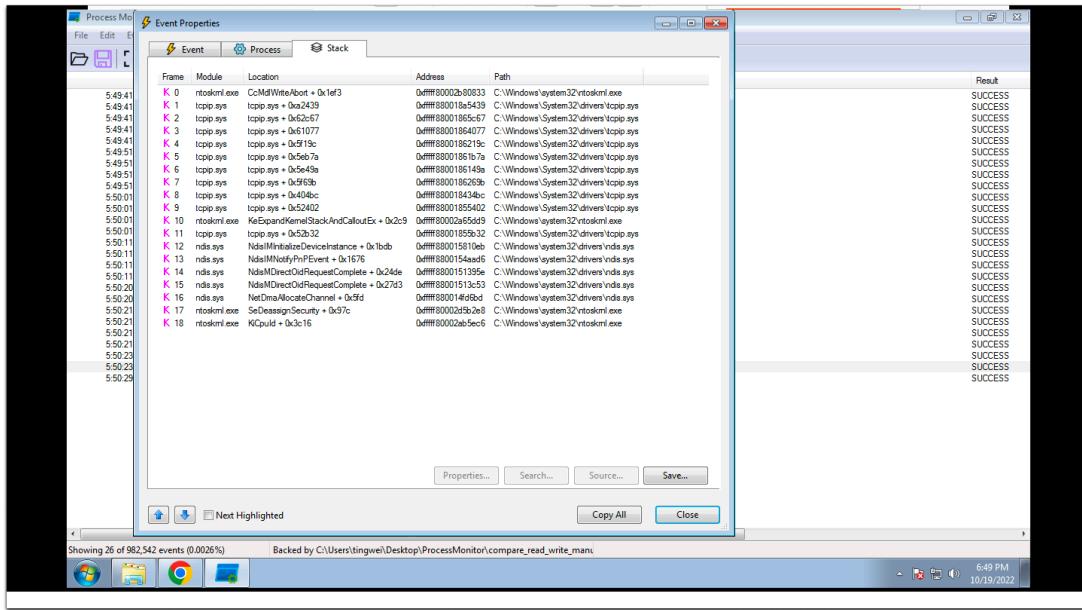
WRITE TCP Send



READ TCP Recv



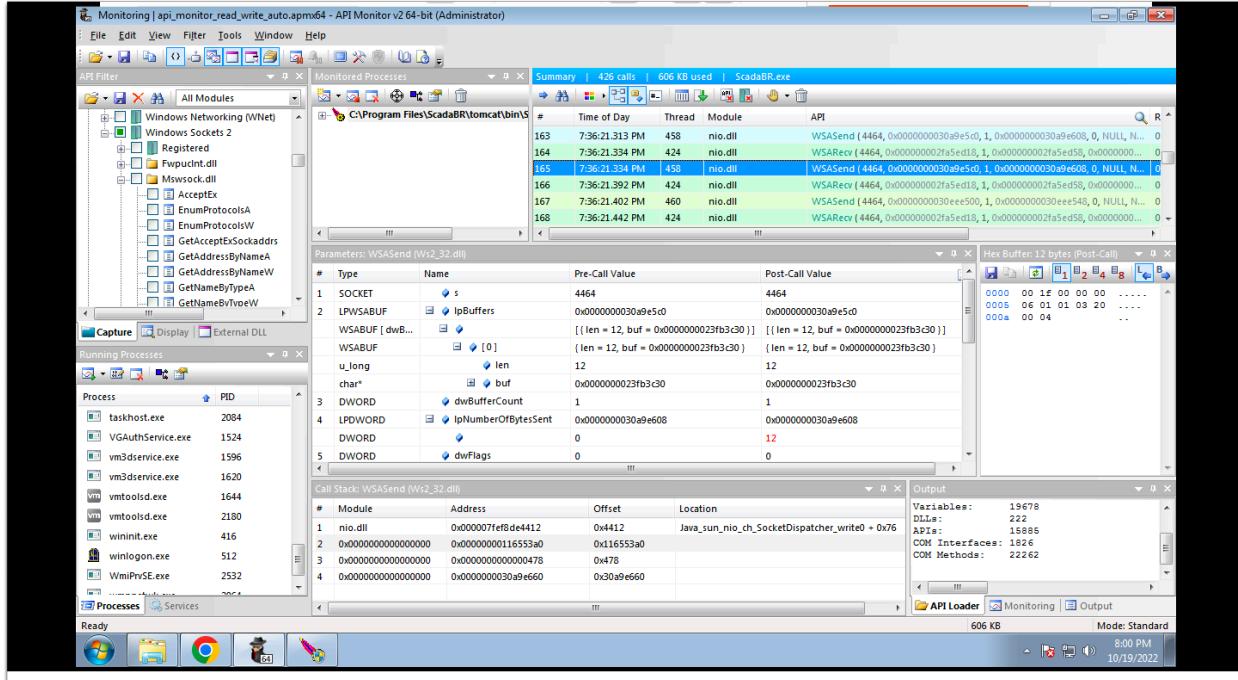
WRITE TCP Recv



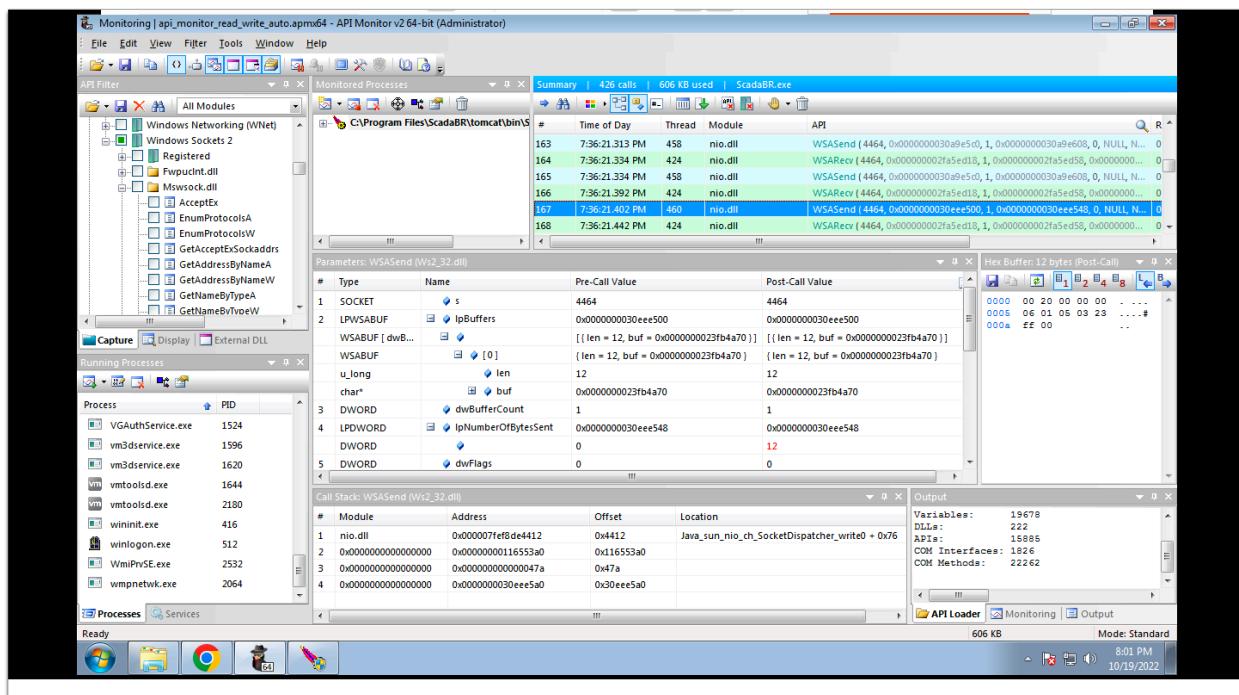
After that, I tried API Monitor (api_monitor_read_write_auto.apmx64), and it shows that the arguments for sending the network traffic is exactly the same. The only difference is the content in the buffer. The content in the buffer decides if this is a READ call or a WRITE call.

Here are some examples.

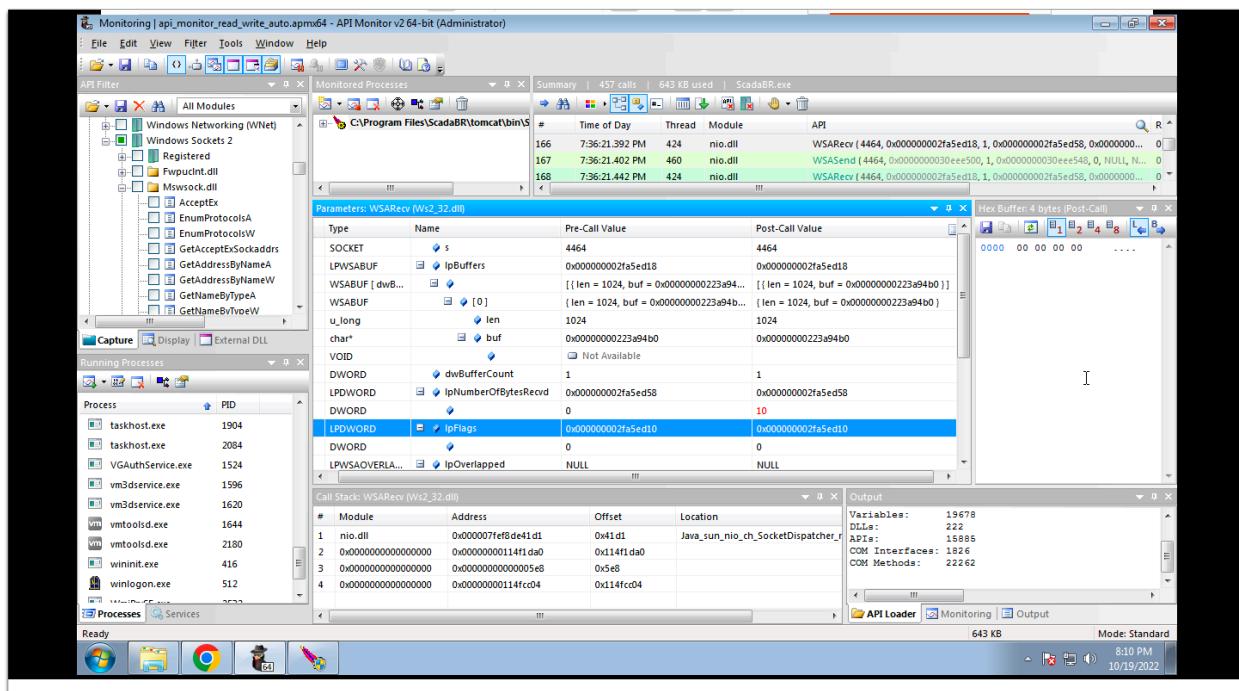
WSASend for READ



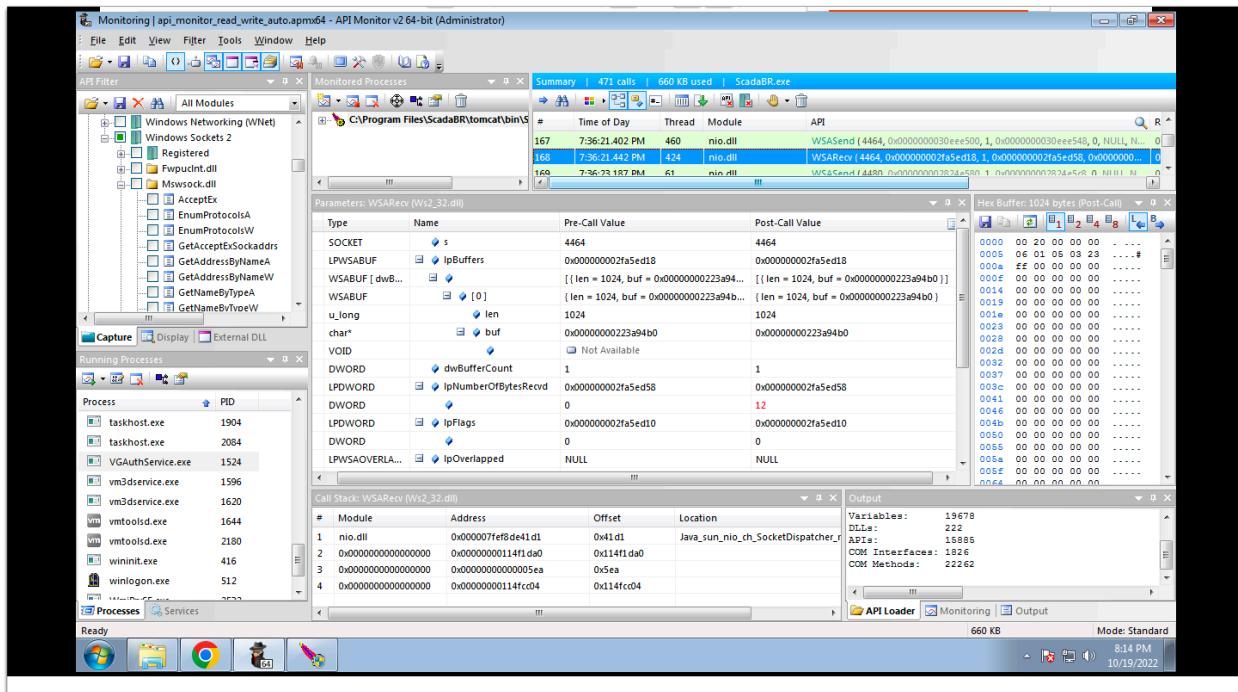
WSASend for WRITE



WSARecv for READ



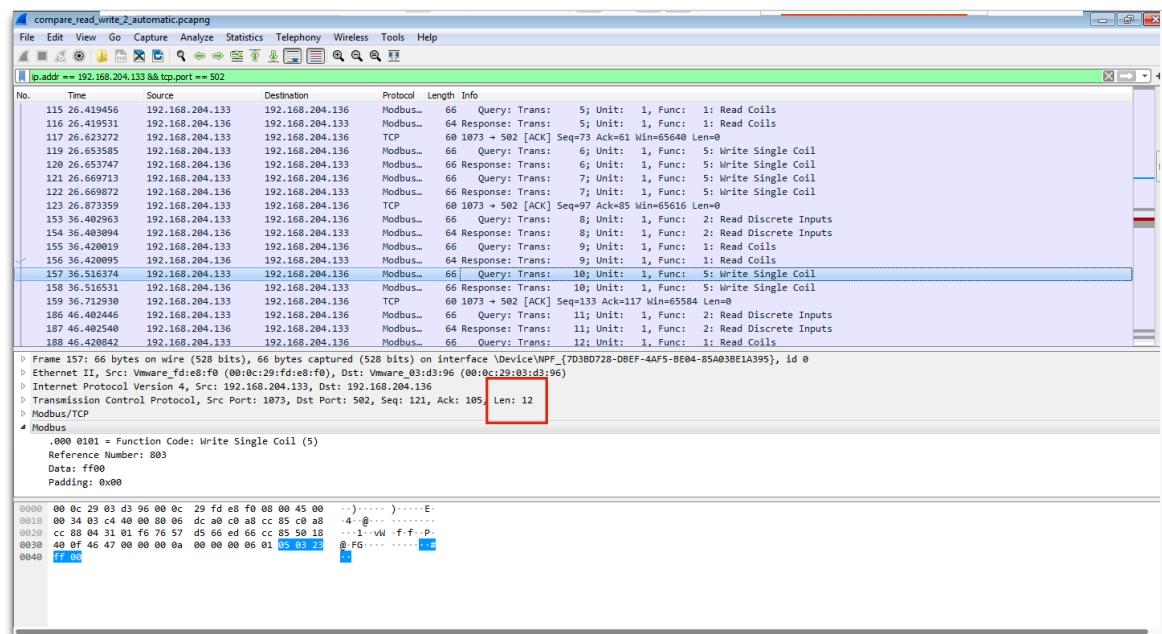
WSARecv for WRITE



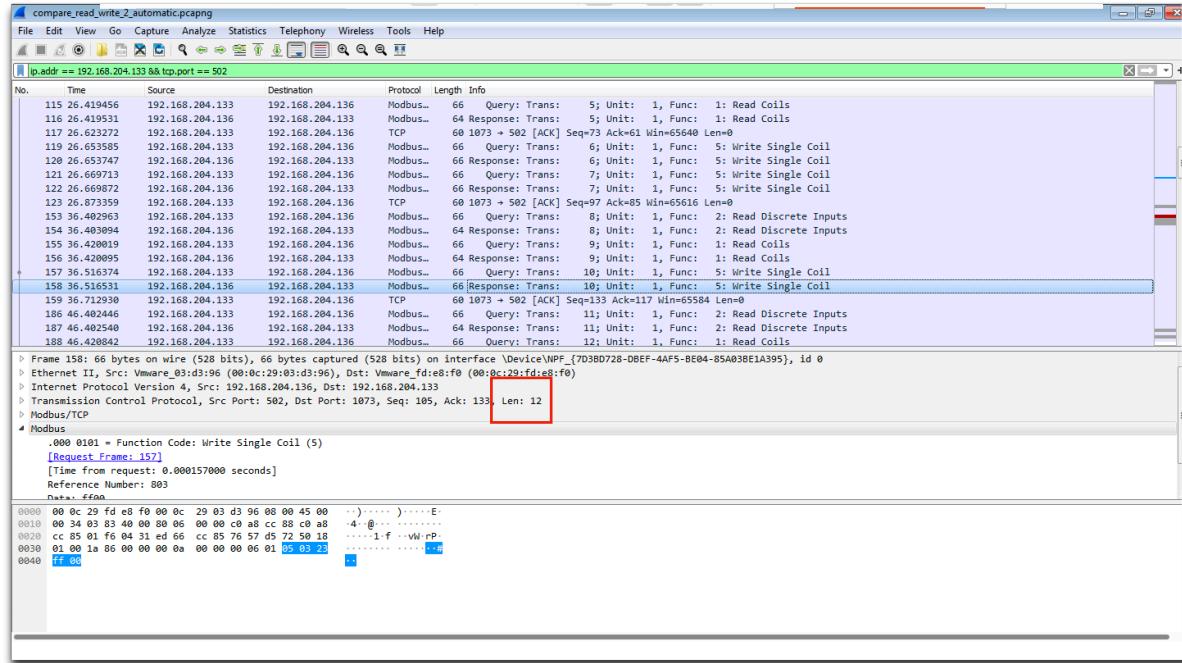
I also compared READ calls with READ calls, and still there's no single identifier that can be used to differentiate.

Length Analysis

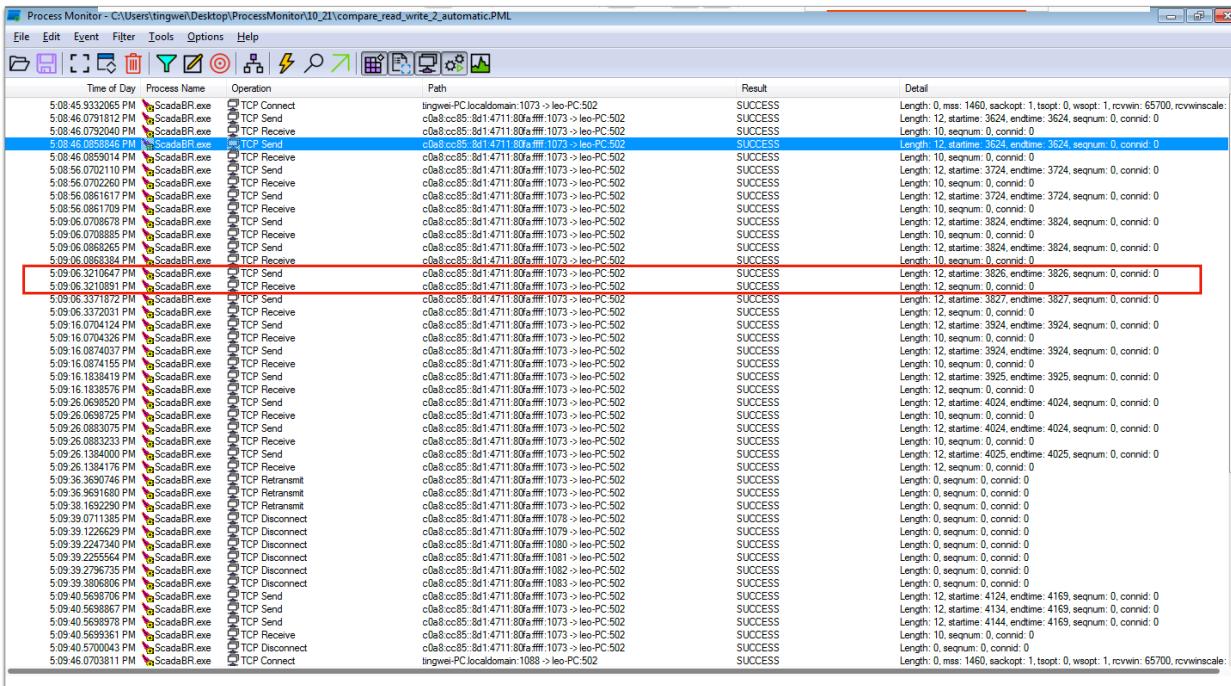
In Process Monitor, it shows the length of the packet in TCP Send and TCP Recv. And I found that when it's a WRITE call, the length of the outbound packet will always be the same as the length of the inbound packet.



The reason for that is OpenPLC simply replies the same data to ScadaBR when it receives a WRITE call. In my case, it's always 12.



In above screenshots, we see the data in write request and response is exactly the same.



The red block above shows a pair of WRITE call request and response based on the same TCP length.

Behavior Analysis

From my observation, I found that before ScadaBR sends a WRITE call, it always check the database. It totally makes sense because data is stored in the database, and it checks the database to see if it needs to send a WRITE call.

On the other hand, READ calls are executed regularly. It does not check the database before sending the request.

Here's how it looks before sending a WRITE call, we see it's checking scadabrDB

Time of Day	Process Name	Operation	Path	Result	Detail
5:09:56.1605069 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR	SUCCESS	
5:09:56.1606477 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db	SUCCESS	
5:09:56.1606810 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db\scadabrDB	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1607142 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: scadabrDB, 2, scadabrD
5:09:56.1607357 PM	ScadaBR.exe	CreateFile	C:\	SUCCESS	
5:09:56.1607597 PM	ScadaBR.exe	QueryDirectory	C:\Program Files	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1608065 PM	ScadaBR.exe	Close File	C:\	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: Program Files, 2, Program
5:09:56.1610413 PM	ScadaBR.exe	CreateFile	C:\Program Files	SUCCESS	
5:09:56.1610738 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1611263 PM	ScadaBR.exe	Close File	C:\Program Files	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: ScadaBR, 2, ScadaBR
5:09:56.1611488 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR	SUCCESS	
5:09:56.1612788 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1613112 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: tomcat, 2, tomcat
5:09:56.1614206 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat	SUCCESS	
5:09:56.1614625 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat\webapps	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1614956 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: webapps, 2, webapps
5:09:56.1615152 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat\webapps	SUCCESS	
5:09:56.1616471 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1616797 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat\webapps	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: ScadaBR, 2, ScadaBR
5:09:56.1617396 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR	SUCCESS	
5:09:56.1618316 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1618434 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: db, 2, db
5:09:56.1620397 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db\scadabrDB	SUCCESS	
5:09:56.1620397 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db\scadabrDB	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1620436 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db\scadabrDB	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: scadabrDB, 2, scadabrD
5:09:56.1620710 PM	ScadaBR.exe	CreateFile	C:\	SUCCESS	
5:09:56.1620955 PM	ScadaBR.exe	Thread Create	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR	SUCCESS	
5:09:56.1640740 PM	ScadaBR.exe	TCP Send	c0a8cc85-8d14-7111-80ff-ffff.1088->leo-PC-502	SUCCESS	Thread ID: 3616
5:09:56.1640888 PM	ScadaBR.exe	TCP Receive	c0a8cc85-8d14-7111-80ff-ffff.1088->leo-PC-502	SUCCESS	Length: 12, starttime: 4325, endtime: 4325, seqnum: 0, connid: 0
5:09:56.1717575 PM	ScadaBR.exe	CreateFile	C:\	SUCCESS	Length: 12, seqnum: 0, connid: 0
5:09:56.1718441 PM	ScadaBR.exe	QueryDirectory	C:\Program Files	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1719263 PM	ScadaBR.exe	Close File	C:\	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: Program Files, 2, Program
5:09:56.1720796 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR	SUCCESS	
5:09:56.1722740 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1723251 PM	ScadaBR.exe	Close File	C:\Program Files	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: ScadaBR, 2, ScadaBR
5:09:56.1723749 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR	SUCCESS	
5:09:56.1725616 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1726310 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: tomcat, 2, tomcat
5:09:56.1727395 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat	SUCCESS	
5:09:56.1728439 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat\webapps	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1728926 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: webapps, 2, webapps
5:09:56.1730759 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat\webapps	SUCCESS	
5:09:56.1731267 PM	ScadaBR.exe	QueryDirectory	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Optio
5:09:56.1731763 PM	ScadaBR.exe	Close File	C:\Program Files\ScadaBR\tomcat\webapps	SUCCESS	FileInformationClass: FileBothDirectoryInformation, Filter: ScadaBR, 2, ScadaBR
5:09:56.1733746 PM	ScadaBR.exe	CreateFile	C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR	SUCCESS	

I collect data twice.

Network traffic and Process Monitor data is in the folder “Process Monitor/automatic_write/1st Try” and “Process Monitor/automatic_write/2nd Try”

I also created a script and use those data to look for WRITE API calls, and it successfully identified every WRITE calls.

Here's how my script looks.

```

EXPLORER OPEN EDITORS ... find_write.py
> OPEN EDITORS
10_21 [SSH: GATECH_CS6727] find_write.py
  compare_read_write_2_auto...
  compare_read_write_auto...
  find_write.py

24 # Identify WRITE Commands
25 write_time = []
26 for index, row in scada_tcp_send_df.iterrows():
27     #print(row['Time of Day'])
28     mask = ((scada_df['Time of Day'] > (row['Time of Day'] - timedelta(seconds = 0.01))) &
29             (scada_df['Time of Day'] < (row['Time of Day']))) &&
30             (scada_df['Path'] == 'C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db\scadabrDB'))
31     tmp = scada_df[mask]
32     if tmp.shape[0] > 2:
33         write_time.append(row['Time of Day'])
34
35 else:
36     # Check if there is Thread Exit before TCP Send
37     mask = ((scada_df.index < index) &&
38             (scada_df.index > (index - 10)) &&
39             (scada_df['Operation'] == 'Thread Exit'))
40
41     tmp = scada_df[mask]
42
43     if tmp.shape[0] > 0:
44         mask = ((scada_df['Time of Day'] > (row['Time of Day'] - timedelta(seconds = 0.06))) &&
45                 (scada_df['Time of Day'] < (row['Time of Day']))) &&
46                 (scada_df['Path'] == 'C:\Program Files\ScadaBR\tomcat\webapps\ScadaBR\db\scadabrDB'))
47         tmp = scada_df[mask]
48         if tmp.shape[0] > 2:
49             write_time.append(row['Time of Day'])
50
51
52 print(f"\nThere are {len(write_time)} WRITE commands in total\n")
53 for t in write_time:
54     print(t)
55
56 # Extract APIs and timecode between 2 WRITES
57 for i in range(len(write_time) - 1):
58     mask = ((scada_df['Time of Day'] > (write_time[i])) &&
59             (scada_df['Time of Day'] < (write_time[i + 1])))
60     tmp = scada_df[mask]
61     result = tmp.drop(['Process Name', 'Result', 'Detail', 'Path', 'index'], axis=1)
62     print(f"\n\n[+] Part of API Sequence {i+1}")
63
64     print(result.to_dict('records')[0:10])

```

Ln 8, Col 37 Spaces: 4 UTF-8 LF Python ⚙

And the output. In the output, I only keep the timestamp and the API call.

```

..esktop/CS6727 (-zsh) twang626@bird:~/Desktop/10_21$ python3 find_write.py
twang626@bird:~/Desktop/10_21$ python3 find_write.py

There are 5 WRITE commands in total

2022-10-21 11:58:30.732544
2022-10-21 11:58:30.748723
2022-10-21 11:58:40.734190
2022-10-21 11:58:50.645582
2022-10-21 11:59:20.736315

[+] Part of API Sequence 1
[{'Time of Day': Timestamp('2022-10-21 11:58:30.732560'), 'Operation': 'TCP Receive'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.732597'), 'Operation': 'CreateFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.732631'), 'Operation': 'QueryDirectory'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.732666'), 'Operation': 'CloseFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.732790'), 'Operation': 'CreateFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.732823'), 'Operation': 'QueryDirectory'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.732857'), 'Operation': 'CloseFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.733012'), 'Operation': 'CreateFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.733046'), 'Operation': 'QueryDirectory'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.733046'), 'Operation': 'CloseFile'}]

[+] Part of API Sequence 2
[{'Time of Day': Timestamp('2022-10-21 11:58:30.748742'), 'Operation': 'TCP Receive'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.749707'), 'Operation': 'Thread Exit'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.750083'), 'Operation': 'WriteFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.750105'), 'Operation': 'WriteFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.750176'), 'Operation': 'WriteFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.750608'), 'Operation': 'QueryDirectory'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.750650'), 'Operation': 'CloseFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.750785'), 'Operation': 'CreateFile'}, {'Time of Day': Timestamp('2022-10-21 11:58:30.750819'), 'Operation': 'QueryDirectory'}]

```