

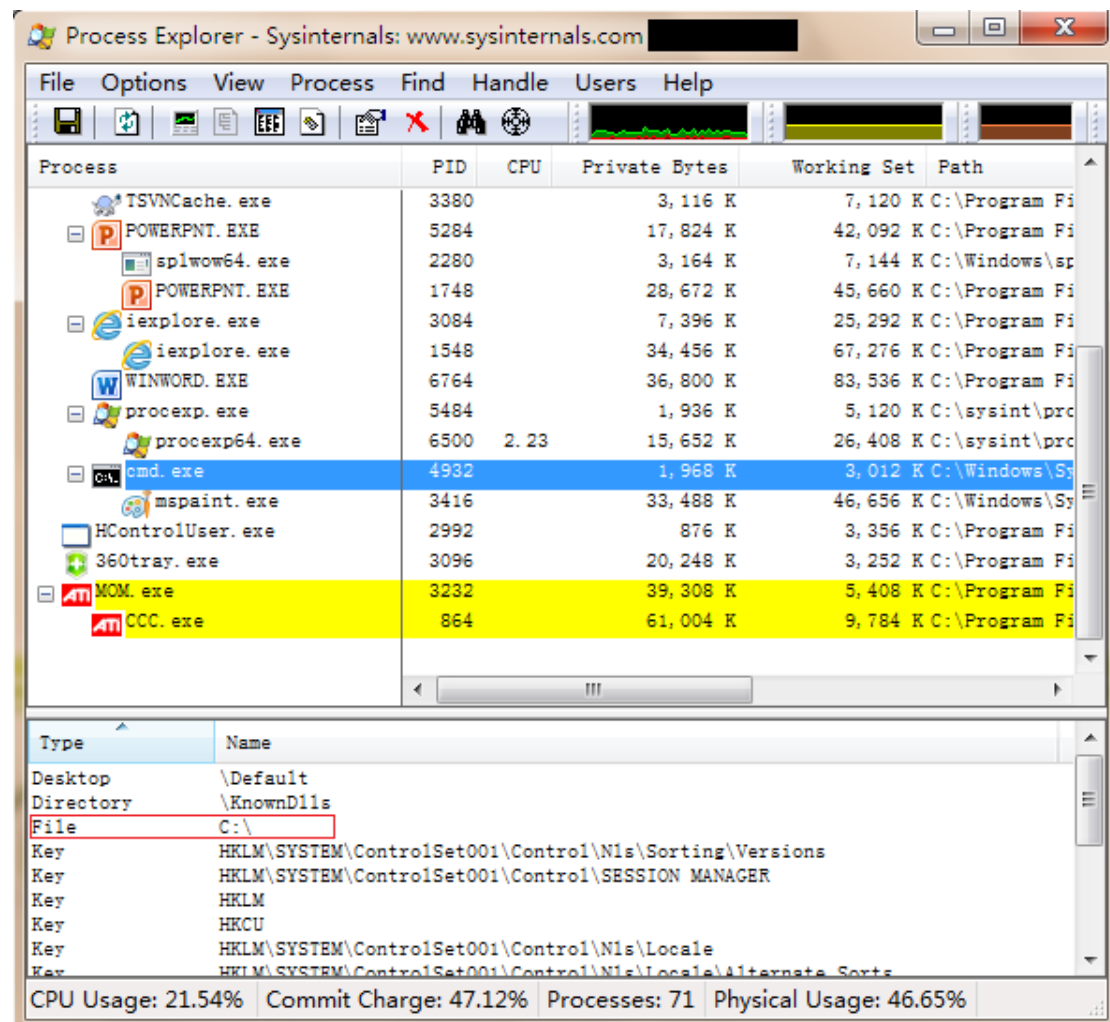
CS490 Windows Internals Lab

Sep 21, 2012

1. Viewing Handles

Use Process Explorer to view handles

Run Process Explorer, and make sure the lower pane is enabled and configured to show open handles. (Click on View, Lower Pane View, and then Handles). Then open a command prompt and view the handle table for the new Cmd.exe process. You should see an open file handle to the current directory.



If you then change the current directory with the CD command, you will see in Process Explorer that the handle to the previous current directory is closed and a new handle is opened to the new current directory. The previous handle is highlighted briefly in red, and the new handle is

highlighted in green. The duration of the highlight can be adjusted by clicking Options and then Difference Highlight Duration.

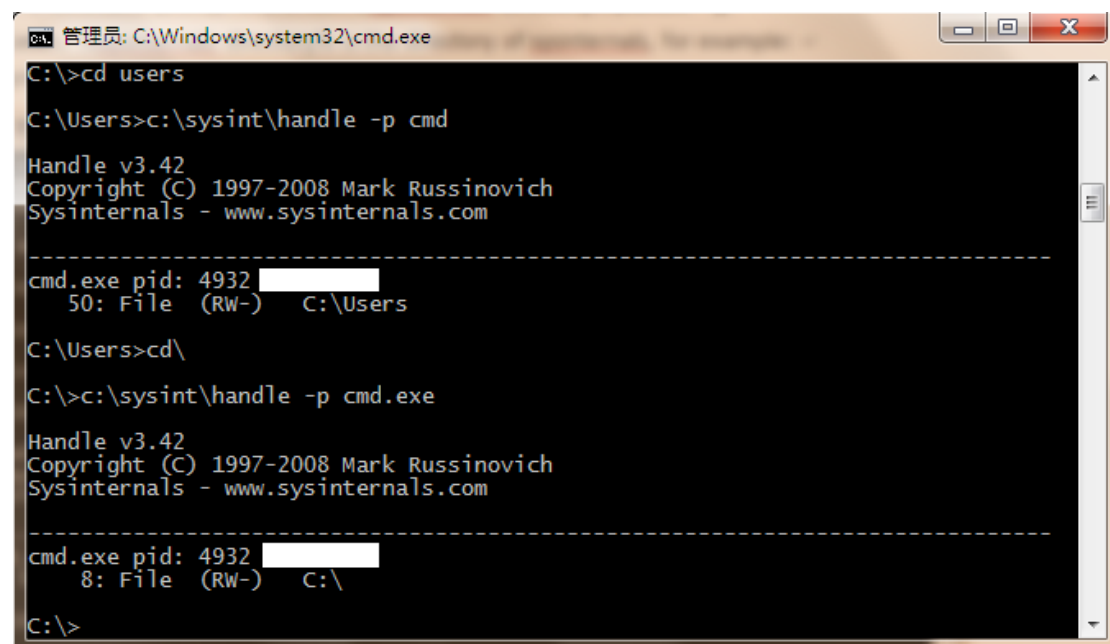
Process Explorer's differences highlighting feature makes it easy to see changes in the handle table. For example, if a process is leaking handles, viewing the handle table with Process Explorer can quickly show what handle or handles are being opened but not closed. This information can assist the programmer to find the handle leak.

Use Handle-tool to view handles

You can also display the open handle table by using the command line Handle tool from www.sysinternals.com.

1. Open a command prompt and enter: **%sysinternals directory%\handle -p cmd.exe**, %sysinternals directory% is the directory of sysinternals, for example:
C:\sysint\handle -p cmd.exe
2. Change the current directory by using command **CD**, and repeat step 1 to see the current file path of process cmd.exe.

Here is an example for handle-tool. After step 1, you can see the current file path is **c:\users**. Because we change the path by using **cd**, the path changed to **c:**, as shown in the following picture.



```
管理员: C:\Windows\system32\cmd.exe
C:\>cd users
C:\Users>c:\sysint\handle -p cmd
Handle v3.42
Copyright (C) 1997-2008 Mark Russinovich
Sysinternals - www.sysinternals.com
-----
cmd.exe pid: 4932 [REDACTED]
50: File (RW-) C:\Users
C:\Users>cd\
C:\>c:\sysint\handle -p cmd.exe
Handle v3.42
Copyright (C) 1997-2008 Mark Russinovich
Sysinternals - www.sysinternals.com
-----
cmd.exe pid: 4932 [REDACTED]
8: File (RW-) C:\
C:\>
```

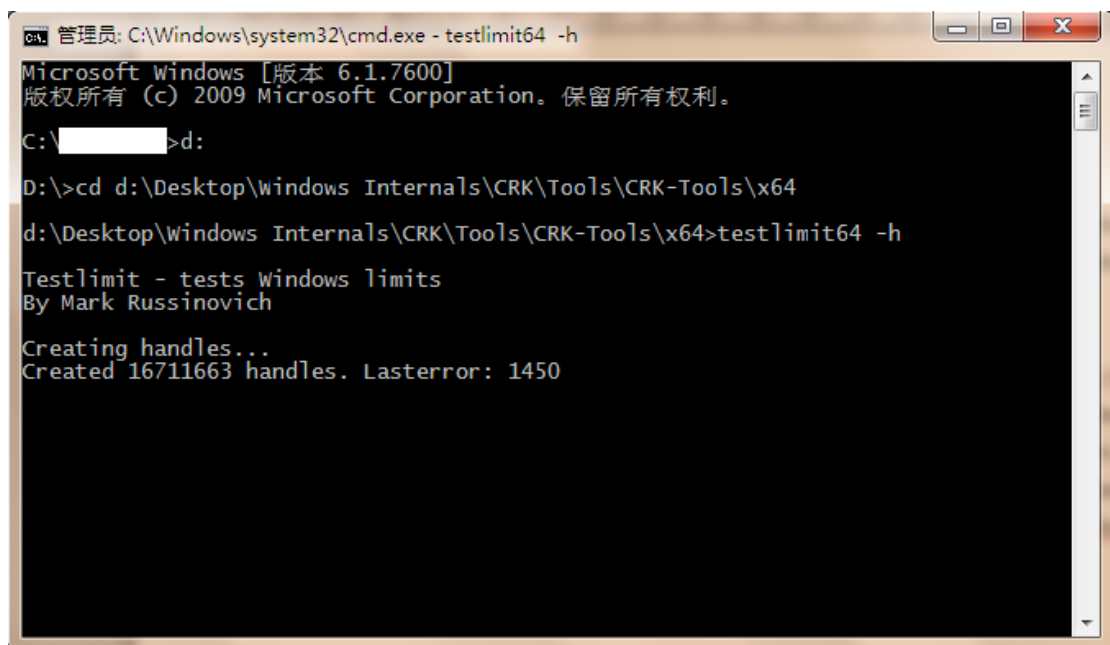
Viewing Maximum Number of handles

The test program Testlimit from Sysinternals (in CRK-Tools\testlimit.exe for 32-bit Windows or CRK-Tools\x64\testlimit64.exe for 64-bit Windows) has an option to open handles to an object until it cannot open any more handles. You can use this to see how many handles can be created in a single process on your system. Because handle tables are allocated from paged pool, you might run out of paged pool before you hit the maximum number of handles that can be created

in a single process.

To see how many handles you can create on your system, follow these steps:

1. Run Process Explorer, and click View and then System Information. Notice the current and maximum size of paged pool. (To display the maximum pool size values, Process Explorer must be configured properly to access the symbols for the kernel image, Ntoskrnl.exe.) Leave this system information display running so that you can see pool utilization when you run the Testlimit program.
2. Open a command prompt.
3. Run the Testlimit program with the "-h" switch (do this by typing `testlimit -h`). When Testlimit fails to open a new handle, it will display the total number of handles it was able to create. If the number is less than approximately 16 million, you are probably running out of paged pool before hitting the theoretical per-process handle limit.
4. Close the command-prompt window; doing this will kill the Testlimit process, thus closing all the open handles.



```
管理员: C:\Windows\system32\cmd.exe - testlimit64 -h
Microsoft Windows [版本 6.1.7600]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\>>d:

D:\>>cd d:\Desktop\Windows Internals\CRK\Tools\CRK-Tools\x64
d:\Desktop\Windows Internals\CRK\Tools\CRK-Tools\x64>testlimit64 -h

Testlimit - tests Windows limits
By Mark Russinovich

Creating handles...
Created 16711663 handles. Lasterror: 1450
```

2. Troubleshooting a Pool Leak

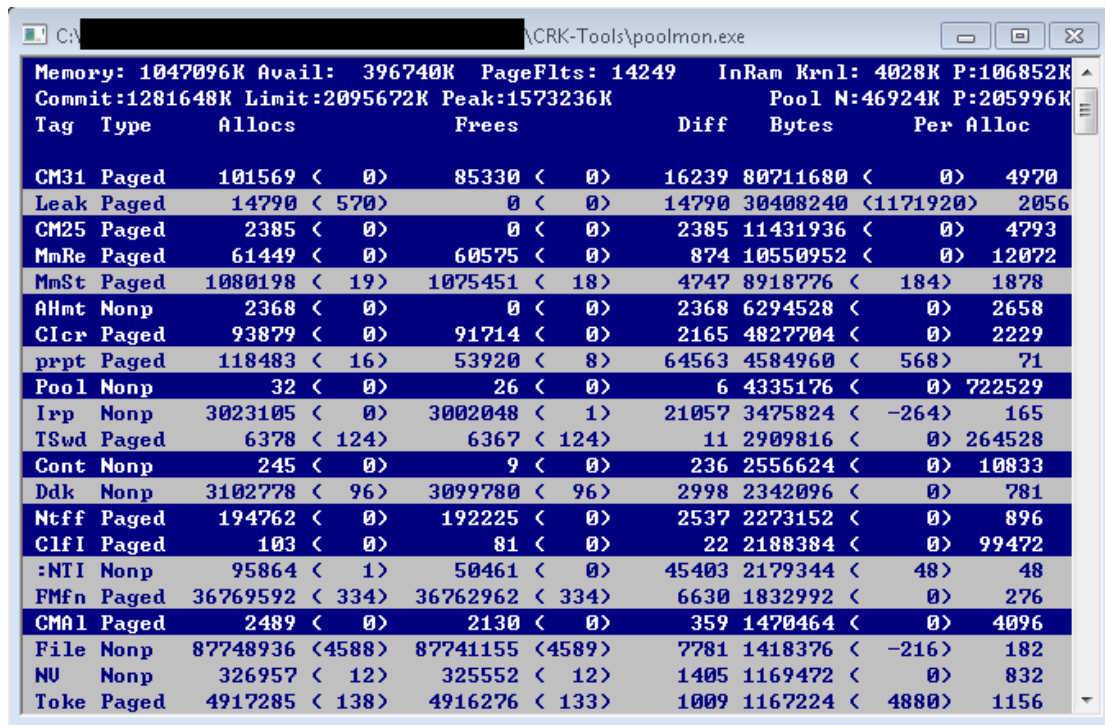
In this experiment, you will fix a real paged pool leak on your system so that you can put to use the techniques described in the previous section to track down the leak. The leak will be generated by the NotMyFault tool, which you can download from CRK-Tools package.

When you run NotMyFault.exe, it loads a device driver Myfault.sys and presents the following dialog box:

1. Click the Leak Pool button. This causes NotMyFault to begin sending requests to the Myfault device driver to allocate paged pool. NotMyFault will continue to do this until you click the Stop Leaking button. Note that the paged pool is not released even when

you close the program; the pool is permanently leaked until you reboot the system.

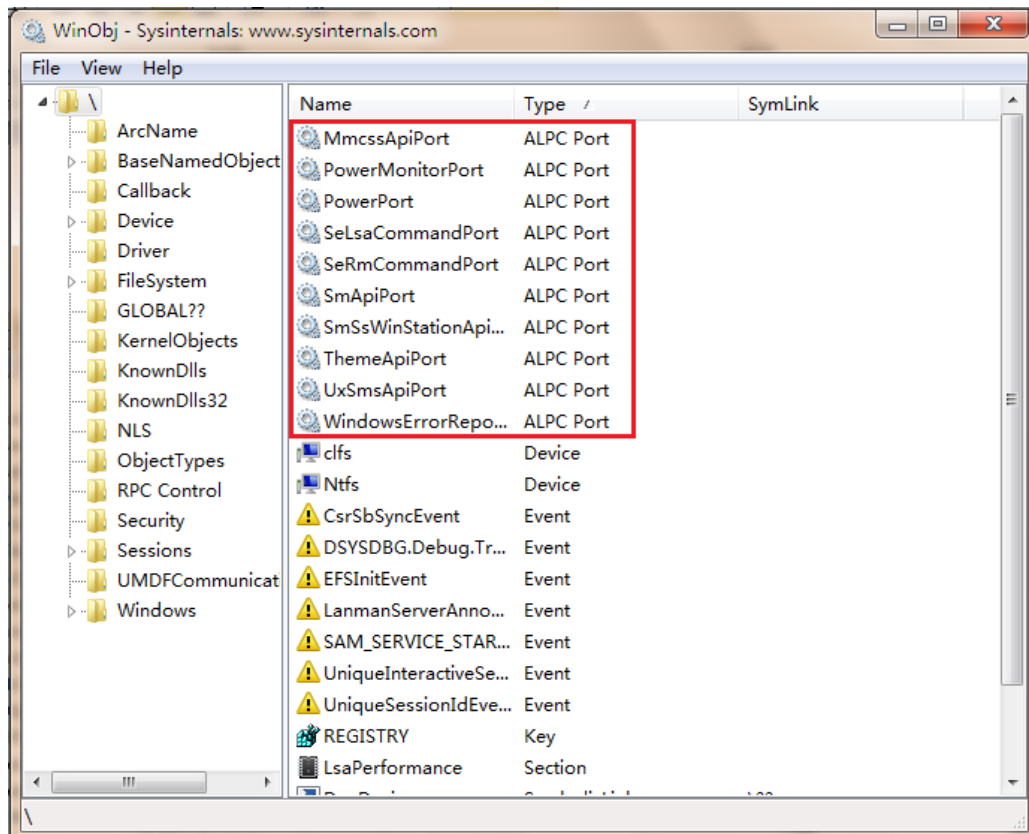
2. While the pool is leaking, first open Task Manager and click on the Performance tab. You should notice Paged Pool climbing. You can also check this with Process Explorer's System Information display. (Click on Show and then System Information.)
3. To determine the pool tag that is leaking, run Poolmon and press the "b" key to sort by the number of bytes. Press "p" twice so that Poolmon is showing only paged pool. You should notice the pool tag "Leak" climbing to the top of the list. -- Press Stop Leaking!



Tag	Type	Allocs	Free	Diff	Bytes	Per	Alloc
CM31	Paged	101569 < 0>	85330 < 0>	16239	80711680 < 0>	4970	
Leak	Paged	14790 < 570>	0 < 0>	14790	30408240 < 1171920>	2056	
CM25	Paged	2385 < 0>	0 < 0>	2385	11431936 < 0>	4793	
MmRe	Paged	61449 < 0>	60575 < 0>	874	10550952 < 0>	12072	
MmSt	Paged	1080198 < 19>	1075451 < 18>	4747	8918776 < 184>	1878	
AHnt	Nonp	2368 < 0>	0 < 0>	2368	6294528 < 0>	2658	
Clcr	Paged	93879 < 0>	91714 < 0>	2165	4827704 < 0>	2229	
prpt	Paged	118483 < 16>	53920 < 8>	64563	4584960 < 568>	71	
Pool	Nonp	32 < 0>	26 < 0>	6	4335176 < 0>	722529	
Irp	Nonp	3023105 < 0>	3002048 < 1>	21057	3475824 < -264>	165	
TSwd	Paged	6378 < 124>	6367 < 124>	11	2909816 < 0>	264528	
Cont	Nonp	245 < 0>	9 < 0>	236	2556624 < 0>	10833	
Ddk	Nonp	3102778 < 96>	3099780 < 96>	2998	2342096 < 0>	781	
Ntff	Paged	194762 < 0>	192225 < 0>	2537	2273152 < 0>	896	
Clfi	Paged	103 < 0>	81 < 0>	22	2188384 < 0>	99472	
:NTI	Nonp	95864 < 1>	50461 < 0>	45403	2179344 < 48>	48	
FMfn	Paged	36769592 < 334>	36762962 < 334>	6630	1832992 < 0>	276	
CMAl	Paged	2489 < 0>	2130 < 0>	359	1470464 < 0>	4096	
File	Nonp	87748936 < 4588>	87741155 < 4589>	7781	1418376 < -216>	182	
NU	Nonp	326957 < 12>	325552 < 12>	1405	1169472 < 0>	832	
Token	Paged	4917285 < 138>	4916276 < 133>	1009	1167224 < 4880>	1156	

3. Viewing ALPC Port Objects

You can see named ALPC port objects with the WinObj tool from Sysinternals. Run WinObj.exe and select the root directory. You can see different kinds of objects. By sorting with "Type", you can see the ALPC Port objects on the top of the list, as objects in the red box in the following picture:



If you want to see ALPC Port in PRC(Remote Procedure Calls), select "PRC Control".

