Lab Report 3: Identifying Running Processes

GuYi

October 18, 2025

1 Lab Objectives

This lab will use the TCP/UDP Endpoint Viewer tool from Windows Sysinternals Suite to identify processes running on the computer. The specific objectives are as follows:

- Download Windows Sysinternals Suite
- Launch TCP/UDP Endpoint Viewer
- Explore running processes
- Explore user-initiated processes

2 Lab Background

In this lab, we will explore processes. A process is a program or application that is currently executing. We will use Process Explorer from Windows Sysinternals Suite to explore processes, and we will also launch and observe a new process.

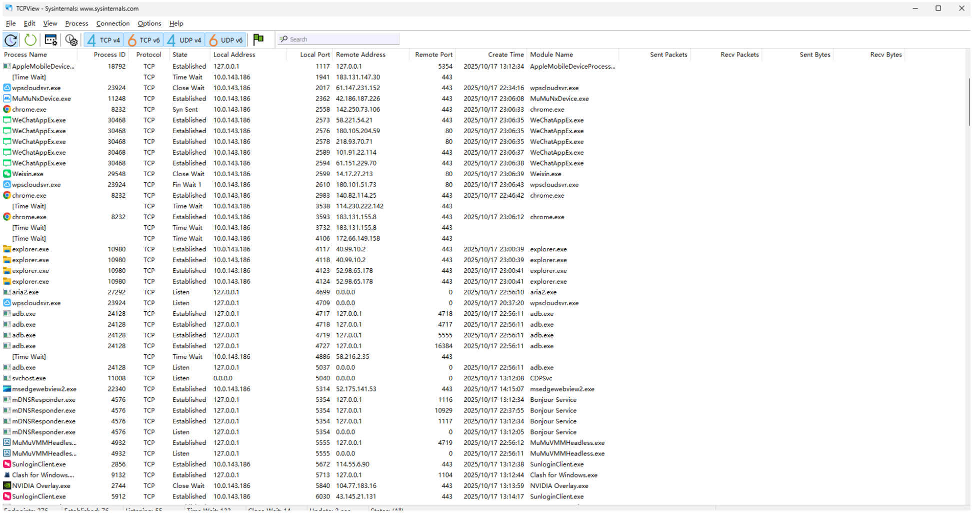
3 Lab Steps

3.1 Part 1: Download Windows Sysinternals Suite

1. Navigate to the following link to download Windows Sysinternals Suite:
   `https://technet.microsoft.com/en-us/sysinternals/bb842062.aspx`
2. After downloading, right-click on the zip file, select ”Extract All...”, and extract the files from the folder.
3. Choose the default name and destination location (Downloads folder), then click ”Extract”.
4. Exit the Web browser.

3.2 Part 2: Launch TCP/UDP Endpoint Viewer

1. Navigate to the SysinternalsSuite folder containing all the extracted files.
2. Open Tcpview.exe. When prompted, accept the Process Explorer license agreement.
3. Click ”Yes” to allow this application to make changes to your device.
4. Exit File Explorer and close all currently running applications.

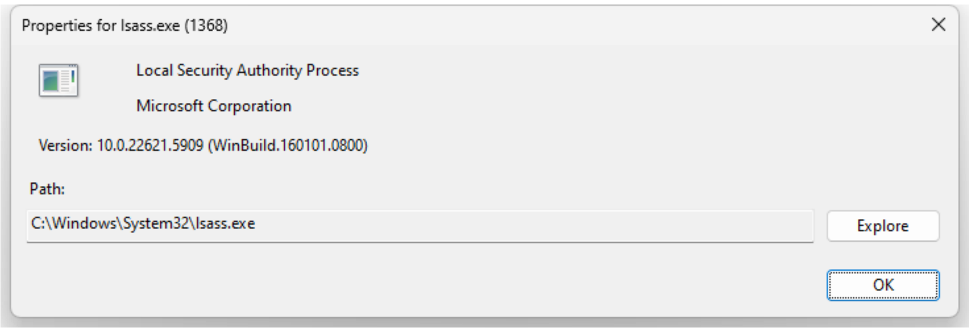


Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port	Create Time	Module Name	Sent Packets	Recv Packets	Sent Bytes	Recv Bytes
AppleMobileDevice...	18792	TCP	Established	127.0.0.1	1117	127.0.0.1	5354	2025/10/17 13:12:34	AppleMobileDeviceProcess...				
[Time Wait]		TCP	Time Wait	10.0.143.186	1941	183.131.147.30	443						
wpscloudsvr.exe	23924	TCP	Close Wait	10.0.143.186	2017	61.147.231.152	443	2025/10/17 22:34:16	wpscloudsvr.exe				
MuMuNxDevice.exe	11248	TCP	Established	10.0.143.186	2362	42.186.187.226	443	2025/10/17 23:06:08	MuMuNxDevice.exe				
chrome.exe	8232	TCP	Syn Sent	10.0.143.186	2558	142.250.73.106	443	2025/10/17 23:06:33	chrome.exe				
WeChatAppEx.exe	30468	TCP	Established	10.0.143.186	2573	58.221.54.21	443	2025/10/17 23:06:35	WeChatAppEx.exe				
WeChatAppEx.exe	30468	TCP	Established	10.0.143.186	2576	180.105.204.59	80	2025/10/17 23:06:35	WeChatAppEx.exe				
WeChatAppEx.exe	30468	TCP	Established	10.0.143.186	2578	218.93.70.71	80	2025/10/17 23:06:35	WeChatAppEx.exe				
WeChatAppEx.exe	30468	TCP	Established	10.0.143.186	2589	101.91.22.114	443	2025/10/17 23:06:37	WeChatAppEx.exe				
WeChatAppEx.exe	30468	TCP	Established	10.0.143.186	2594	61.151.229.70	443	2025/10/17 23:06:38	WeChatAppEx.exe				
Weixin.exe	29548	TCP	Close Wait	10.0.143.186	2599	14.17.27.213	80	2025/10/17 23:06:39	Weixin.exe				
wpscloudsvr.exe	23924	TCP	Fin Wait 1	10.0.143.186	2610	180.101.51.73	80	2025/10/17 23:06:43	wpscloudsvr.exe				
chrome.exe	8232	TCP	Established	10.0.143.186	2983	140.82.114.25	443	2025/10/17 22:46:42	chrome.exe				
[Time Wait]		TCP	Time Wait	10.0.143.186	3538	114.230.222.142	443						
chrome.exe	8232	TCP	Established	10.0.143.186	3593	183.131.155.8	443	2025/10/17 23:06:12	chrome.exe				
[Time Wait]		TCP	Time Wait	10.0.143.186	3732	183.131.155.8	443						
[Time Wait]		TCP	Time Wait	10.0.143.186	4106	172.66.149.158	443						
explorer.exe	10980	TCP	Established	10.0.143.186	4117	40.99.10.2	443	2025/10/17 23:00:39	explorer.exe				
explorer.exe	10980	TCP	Established	10.0.143.186	4118	40.99.10.2	443	2025/10/17 23:00:39	explorer.exe				
explorer.exe	10980	TCP	Established	10.0.143.186	4123	52.98.65.178	443	2025/10/17 23:00:41	explorer.exe				
explorer.exe	10980	TCP	Established	10.0.143.186	4124	52.98.65.178	443	2025/10/17 23:00:41	explorer.exe				
aria2.exe	27292	TCP	Listen	127.0.0.1	4699	0.0.0.0	0	2025/10/17 22:56:10	aria2.exe				
wpscloudsvr.exe	23924	TCP	Listen	127.0.0.1	4709	0.0.0.0	0	2025/10/17 20:37:20	wpscloudsvr.exe				
adb.exe	24128	TCP	Established	127.0.0.1	4717	127.0.0.1	4718	2025/10/17 22:56:11	adb.exe				
adb.exe	24128	TCP	Established	127.0.0.1	4718	127.0.0.1	4717	2025/10/17 22:56:11	adb.exe				
adb.exe	24128	TCP	Established	127.0.0.1	4719	127.0.0.1	5555	2025/10/17 22:56:11	adb.exe				
adb.exe	24128	TCP	Established	127.0.0.1	4727	127.0.0.1	16384	2025/10/17 22:56:11	adb.exe				
[Time Wait]		TCP	Time Wait	10.0.143.186	4886	58.216.2.35	443						
adb.exe	24128	TCP	Listen	127.0.0.1	5037	0.0.0.0	0	2025/10/17 22:56:11	adb.exe				
svchost.exe	11008	TCP	Listen	0.0.0.0	5040	0.0.0.0	0	2025/10/17 13:12:08	CDPSvc				
msedgewebview2.exe	22340	TCP	Established	10.0.143.186	5314	52.175.141.53	443	2025/10/17 14:15:07	msedgewebview2.exe				
mDNSResponder.exe	4576	TCP	Established	127.0.0.1	5354	127.0.0.1	1116	2025/10/17 13:12:34	Bonjour Service				
mDNSResponder.exe	4576	TCP	Established	127.0.0.1	5354	127.0.0.1	10929	2025/10/17 22:37:55	Bonjour Service				
mDNSResponder.exe	4576	TCP	Established	127.0.0.1	5354	127.0.0.1	1117	2025/10/17 13:12:34	Bonjour Service				
mDNSResponder.exe	4576	TCP	Listen	127.0.0.1	5354	0.0.0.0	0	2025/10/17 13:12:05	Bonjour Service				
MuMuVMMHeadles...	4932	TCP	Established	127.0.0.1	5555	127.0.0.1	4719	2025/10/17 22:56:12	MuMuVMMHeadless.exe				
MuMuVMMHeadles...	4932	TCP	Listen	127.0.0.1	5555	0.0.0.0	0	2025/10/17 22:56:11	MuMuVMMHeadless.exe				
SunloginClient.exe	2856	TCP	Established	10.0.143.186	5672	114.55.6.90	443	2025/10/17 13:12:38	SunloginClient.exe				
Clash for Windows....	9132	TCP	Established	127.0.0.1	5713	127.0.0.1	1104	2025/10/17 13:12:44	Clash for Windows.exe				
NVIDIA Overlay.exe	2744	TCP	Close Wait	10.0.143.186	5840	104.77.183.16	443	2025/10/17 13:13:59	NVIDIA Overlay.exe				
SunloginClient.exe	5912	TCP	Established	10.0.143.186	6030	43.145.21.131	443	2025/10/17 13:14:17	SunloginClient.exe				

Figure 1: TCP/UDP Endpoint Viewer Interface

3.3 Part 3: Explore Running Processes

1. TCPView lists the currently running processes on your Windows PC. At this point, only Windows processes are running.
2. Double-click on lsass.exe.



Properties for lsass.exe (1368)

Local Security Authority Process

Microsoft Corporation

Version: 10.0.22621.5909 (WinBuild.160101.0800)

Path:

C:\Windows\System32\lsass.exe

Figure 2: lsass.exe Process Properties

Question:

- What is lsass.exe? In which folder is it located?

Answer: lsass.exe is the executable file for the Windows Local Security Authority Subsystem Service. It is responsible for enforcing security policies, verifying user logins to Windows systems, managing password changes, and creating access tokens. As seen in the image, it is located in the C:/Windows/System32 folder. This is a critical Windows system process, and if terminated, the system will become unstable or force a restart.

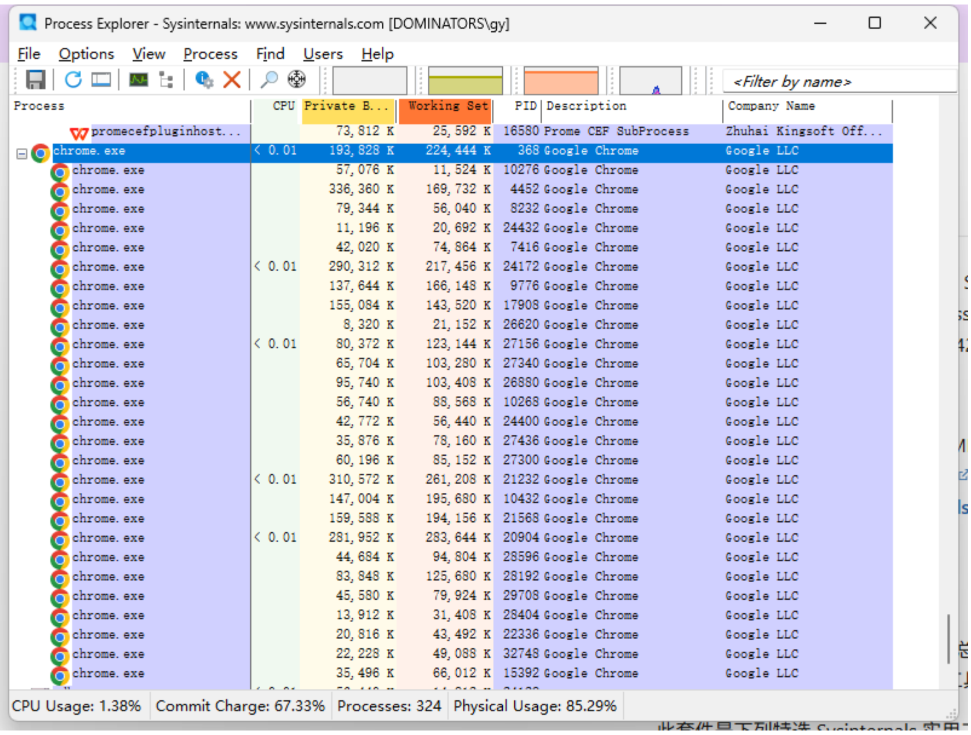
3. Close the lsass.exe properties window when finished.

4. View the properties of other running processes.

Note: Not all processes can have their property information queried.

3.4 Part 4: Explore User-Initiated Processes

1. Open a Web browser, such as Microsoft Edge.



Process Explorer - Sysinternals: www.sysinternals.com [DOMINATORS\gy]

Process	CPU	Private B...	Working Set	PID	Description	Company Name
promecefpluginhost...		73, 812 K	25, 592 K	16580	Prome CEF SubProcess	Zhuhai Kingsoft Off...
chrome. exe	< 0. 01	193, 828 K	224, 444 K	368	Google Chrome	Google LLC
chrome. exe		57, 076 K	11, 524 K	10276	Google Chrome	Google LLC
chrome. exe		336, 360 K	169, 732 K	4452	Google Chrome	Google LLC
chrome. exe		79, 344 K	56, 040 K	8232	Google Chrome	Google LLC
chrome. exe		11, 196 K	20, 692 K	24432	Google Chrome	Google LLC
chrome. exe		42, 020 K	74, 864 K	7416	Google Chrome	Google LLC
chrome. exe	< 0. 01	290, 312 K	217, 456 K	24172	Google Chrome	Google LLC
chrome. exe		137, 644 K	166, 148 K	9776	Google Chrome	Google LLC
chrome. exe		155, 084 K	143, 520 K	17908	Google Chrome	Google LLC
chrome. exe		8, 320 K	21, 152 K	26620	Google Chrome	Google LLC
chrome. exe	< 0. 01	80, 372 K	123, 144 K	27156	Google Chrome	Google LLC
chrome. exe		65, 704 K	103, 280 K	27340	Google Chrome	Google LLC
chrome. exe		95, 740 K	103, 408 K	26880	Google Chrome	Google LLC
chrome. exe		56, 740 K	88, 568 K	10268	Google Chrome	Google LLC
chrome. exe		42, 772 K	56, 440 K	24400	Google Chrome	Google LLC
chrome. exe		35, 876 K	78, 160 K	27436	Google Chrome	Google LLC
chrome. exe		60, 196 K	85, 152 K	27300	Google Chrome	Google LLC
chrome. exe	< 0. 01	310, 572 K	261, 208 K	21232	Google Chrome	Google LLC
chrome. exe		147, 004 K	195, 680 K	10432	Google Chrome	Google LLC
chrome. exe		159, 588 K	194, 156 K	21568	Google Chrome	Google LLC
chrome. exe	< 0. 01	281, 952 K	283, 644 K	20904	Google Chrome	Google LLC
chrome. exe		44, 684 K	94, 804 K	28596	Google Chrome	Google LLC
chrome. exe		83, 848 K	125, 680 K	28192	Google Chrome	Google LLC
chrome. exe		45, 580 K	79, 924 K	29708	Google Chrome	Google LLC
chrome. exe		13, 912 K	31, 408 K	28404	Google Chrome	Google LLC
chrome. exe		20, 816 K	43, 492 K	22336	Google Chrome	Google LLC
chrome. exe		22, 228 K	49, 088 K	32748	Google Chrome	Google LLC
chrome. exe		35, 496 K	66, 012 K	15392	Google Chrome	Google LLC

CPU Usage: 1.38% | Commit Charge: 67.33% | Processes: 324 | Physical Usage: 85.29%

Figure 3: Process Explorer Detecting Browser Process

Question:

- What did you observe in the TCPView window?

Answer: In the TCPView window, when opening a web browser (such as Microsoft Edge), multiple new network connections can be observed being created. These connections show the TCP/UDP connections established between the browser process and various remote servers. The browser process can be seen using different local ports to communicate with remote servers, including DNS queries (for domain name resolution), HTTP/HTTPS connections (for loading web content), and other possible connections (such as WebSocket, content delivery networks, etc.). The status of these connections (such as ESTABLISHED, LISTENING, CLOSE_WAIT, etc.) is also displayed in TCPView.

2. Close the Web browser.

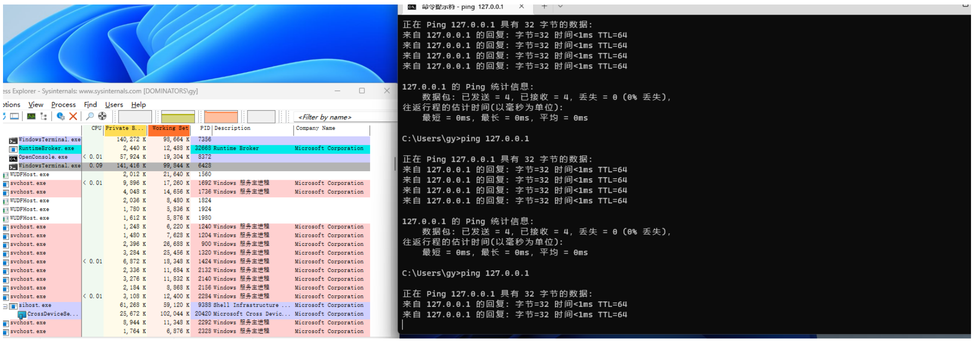
Question:

- What did you observe in the TCPView window?

Answer: After closing the web browser, all connections related to the browser disappear from the TCPView window. This indicates that when an application is closed, its related network connections are also terminated. TCPView briefly displays these closed connections with a color change (usually red) before they completely disappear from the list. The system returns to displaying only the network connection status of basic Windows system processes.

4 Lab 2: Exploring Processes, Threads, Handles, and Windows Registry

4.1 Part 1: Exploring Processes



ess Explorer - Sysinternals: www.sysinternals.com [DOMINATORS\gy]

	CPU	Private B...	Working Set	PID	Description	Company Name
WindowsTerminal.exe		140,272 K	98,664 K	7356		
RuntimeBroker.exe		2,440 K	12,488 K	32668	Runtime Broker	Microsoft Corporation
OpenConsole.exe	< 0.01	57,924 K	19,304 K	8372		
WindowsTerminal.exe	0.09	141,416 K	99,844 K	6428		
WUDFHost.exe		2,012 K	21,640 K	1560		
svchost.exe	< 0.01	9,896 K	17,260 K	1692	Windows 服务主进程	Microsoft Corporation
svchost.exe		4,048 K	14,656 K	1736	Windows 服务主进程	Microsoft Corporation
WUDFHost.exe		2,036 K	8,480 K	1824		
WUDFHost.exe		1,780 K	5,836 K	1924		
WUDFHost.exe		1,612 K	5,876 K	1980		
svchost.exe		1,248 K	6,220 K	1240	Windows 服务主进程	Microsoft Corporation
svchost.exe		1,480 K	7,628 K	1204	Windows 服务主进程	Microsoft Corporation
svchost.exe		2,396 K	26,688 K	900	Windows 服务主进程	Microsoft Corporation
svchost.exe		3,284 K	25,456 K	1320	Windows 服务主进程	Microsoft Corporation
svchost.exe	< 0.01	6,872 K	18,348 K	1424	Windows 服务主进程	Microsoft Corporation
svchost.exe		2,336 K	11,684 K	2132	Windows 服务主进程	Microsoft Corporation
svchost.exe		3,276 K	11,832 K	2140	Windows 服务主进程	Microsoft Corporation
svchost.exe		2,184 K	8,868 K	2156	Windows 服务主进程	Microsoft Corporation
svchost.exe	< 0.01	3,108 K	12,400 K	2284	Windows 服务主进程	Microsoft Corporation
sihost.exe		61,268 K	59,120 K	9388	Shell Infrastructure ...	Microsoft Corporation
CrossDeviceSe...		25,672 K	102,044 K	20420	Microsoft Cross Devic...	Microsoft Corporation
svchost.exe		8,944 K	11,348 K	2292	Windows 服务主进程	Microsoft Corporation
svchost.exe		1,764 K	6,876 K	2328	Windows 服务主进程	Microsoft Corporation

命令提示符 - ping 127.0.0.1

```
正在 Ping 127.0.0.1 具有 32 字节的数据:
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64

127.0.0.1 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 0ms，平均 = 0ms

C:\Users\gy>ping 127.0.0.1

正在 Ping 127.0.0.1 具有 32 字节的数据:
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64

127.0.0.1 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 0ms，平均 = 0ms

C:\Users\gy>ping 127.0.0.1

正在 Ping 127.0.0.1 具有 32 字节的数据:
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
来自 127.0.0.1 的回复: 字节=32 时间<1ms TTL=64
```

Figure 4: Process Explorer Detecting Command Prompt Process

Question:

- What happened to the Web browser window when the process was terminated?

Answer: When terminating the web browser process using Process Explorer, the browser window closes immediately without any warning or save prompts. This is because terminating a process forcibly ends all threads and resources of that process, giving the application no opportunity to execute normal closing procedures (such as saving data or closing connections). This can lead to loss of unsaved data and possibly loss of browsing history or session data in some cases.

Question:

- What happened during the ping process?

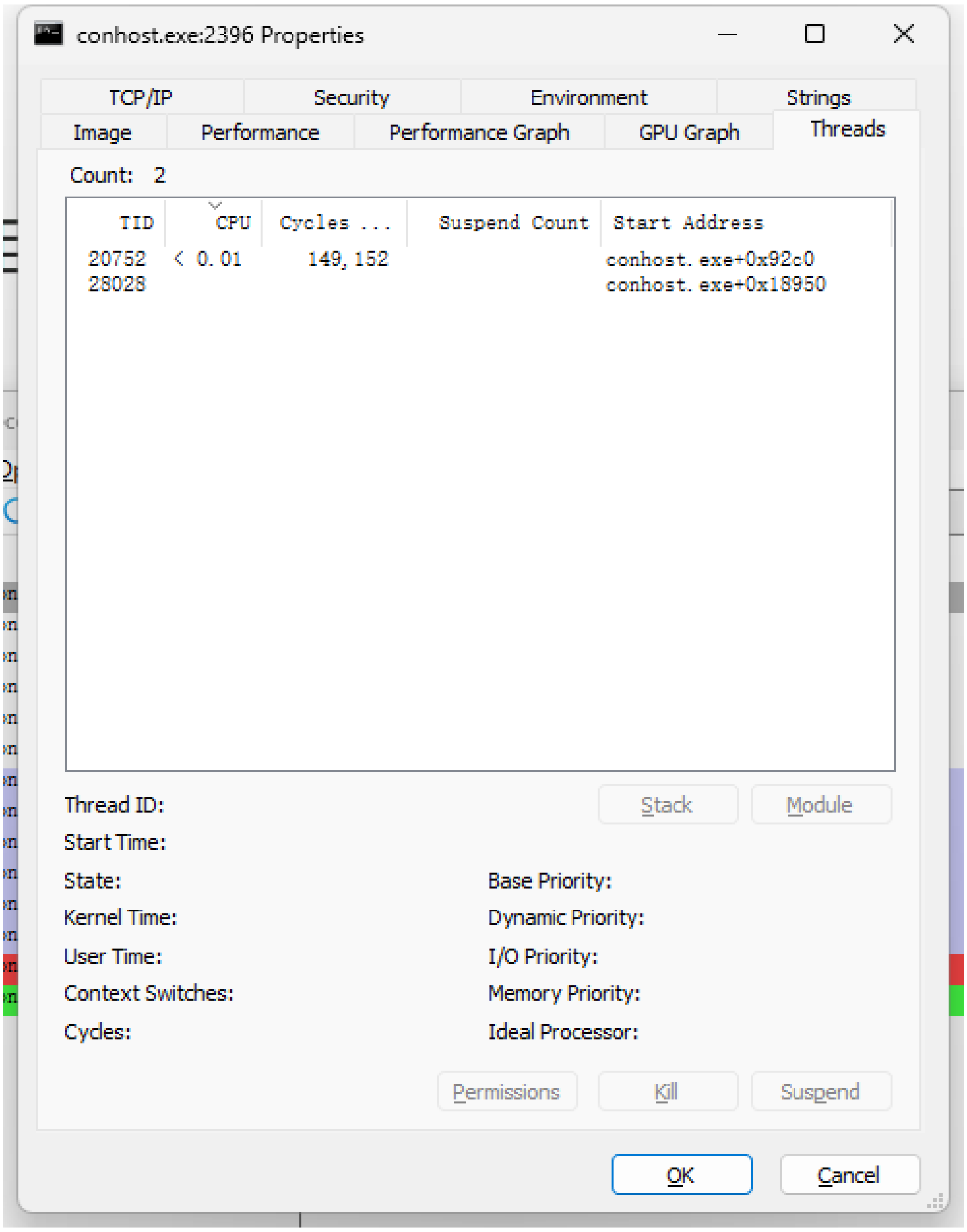
Answer: During the execution of the ping command, new activity can be observed under the cmd.exe process in Process Explorer. Specifically, small fluctuations in CPU usage can be seen, indicating that the ping command is performing network operations. In the thread view, the thread responsible for executing the ping command can be seen in an active state. Additionally, network-related handles being created and used can be observed, which are used for sending ICMP request packets and receiving response packets. The ping command itself does not create a new child process but executes as a command within the cmd.exe process.

Question:

- What happened to the child process conhost.exe when the cmd.exe process was terminated?

Answer: When the cmd.exe process is terminated, its child process conhost.exe (console host process) is also automatically terminated. This is because conhost.exe is a process that provides console window services for cmd.exe, and they have a parent-child relationship. In Windows, when a parent process is terminated, the operating system typically terminates all related child processes. This behavior ensures that no orphaned processes continue to run, preventing resource leaks and system instability.

4.2 Part 2: Exploring Threads and Handles



conhost.exe:2396 Properties

Count: 2

TID	CPU	Cycles ...	Suspend Count	Start Address
20752	< 0.01	149,152		conhost.exe+0x92c0
28028				conhost.exe+0x18950

Thread ID:
Start Time:
State:
Kernel Time:
User Time:
Context Switches:
Cycles:

Base Priority:
Dynamic Priority:
I/O Priority:
Memory Priority:
Ideal Processor:

Figure 5: Process Thread Properties

Question:

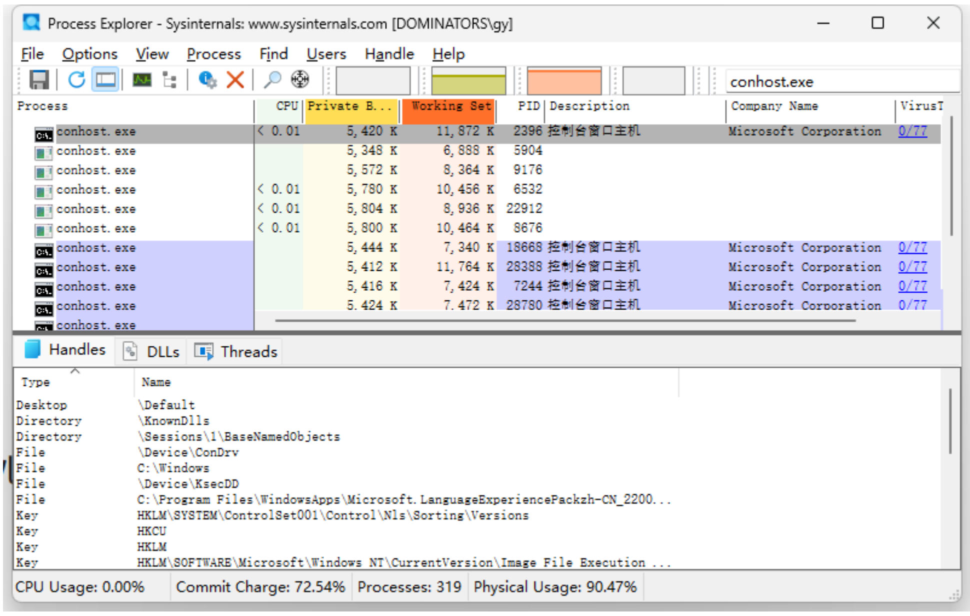
- What types of information are available in the thread properties window?

Answer: In the thread properties window, the following types of information can be seen:

- Thread ID (TID): Unique identifier for each thread
- CPU usage and execution time of the thread
- Thread priority and status (such as running, waiting, suspended, etc.)

- Thread start address and current execution location
- Thread stack information, showing function call hierarchy
- Thread context switch count
- Module (DLL) to which the thread belongs
- Thread creation time

This information is very useful for understanding the execution flow within a process and diagnosing performance issues.



Process	CPU	Private B...	Working Set	PID	Description	Company Name	VirusT
conhost. exe	< 0. 01	5, 420 K	11, 872 K	2396	控制台窗口主机	Microsoft Corporation	0/77
conhost. exe		5, 348 K	6, 888 K	5904			
conhost. exe		5, 572 K	8, 364 K	9176			
conhost. exe	< 0. 01	5, 780 K	10, 456 K	6532			
conhost. exe	< 0. 01	5, 804 K	8, 936 K	22912			
conhost. exe	< 0. 01	5, 800 K	10, 464 K	8676			
conhost. exe		5, 444 K	7, 340 K	18668	控制台窗口主机	Microsoft Corporation	0/77
conhost. exe		5, 412 K	11, 764 K	28388	控制台窗口主机	Microsoft Corporation	0/77
conhost. exe		5, 416 K	7, 424 K	7244	控制台窗口主机	Microsoft Corporation	0/77
conhost. exe		5. 424 K	7. 472 K	28780	控制台窗口主机	Microsoft Corporation	0/77
conhost. exe							

Type	Name
Desktop	\Default
Directory	\KnownDlls
Directory	\Sessions\1\BaseNamedObjects
File	\Device\ConDrv
File	C:\Windows
File	\Device\KsecDD
File	C:\Program Files\WindowsApps\Microsoft.LanguageExperiencePackzh-CN_2200...
Key	HKLM\SYSTEM\ControlSet001\Control\Nls\Sorting\Versions
Key	HKCU
Key	HKLM
Key	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution ...

CPU Usage: 0.00% | Commit Charge: 72.54% | Processes: 319 | Physical Usage: 90.47%

Figure 6: Process Handle Properties

Question:

- What do handles point to?

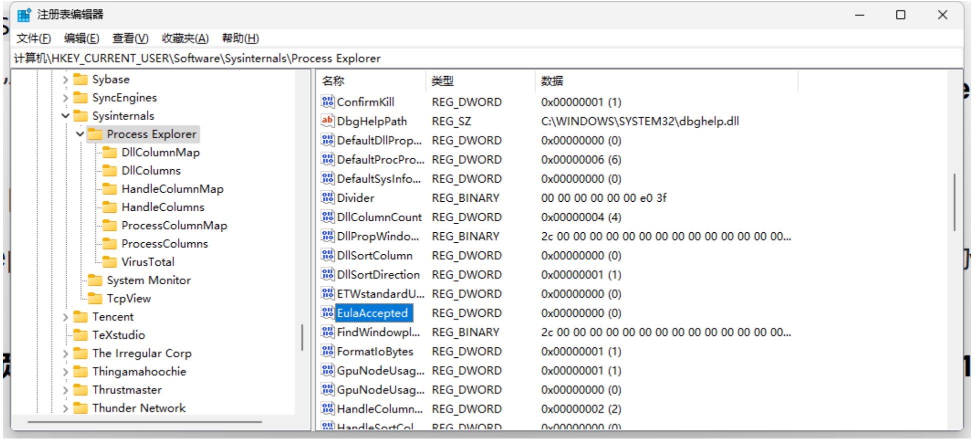
Answer: In the Process Explorer handle view, handles can be seen pointing to various system resources, including:

- Files: Files opened by the process, including executable files, configuration files, data files, etc.
- Registry keys: Windows registry entries accessed by the process
- Directories: File system directories accessed by the process
- Events: Event objects used for inter-process synchronization
- Mutexes: Synchronization objects used to control access to shared resources
- Semaphores: Synchronization objects used to control resource access counts
- Threads: Thread objects created or accessed by the process

- Processes: Other processes referenced by the current process
- Devices: References to hardware devices or virtual devices
- Ports: Communication ports, such as named pipes or network ports

These handles are essentially references to operating system resources, through which processes interact with system resources.

4.3 Part 3: Exploring Windows Registry



计算机\HKEY_CURRENT_USER\Software\Sysinternals\Process Explorer

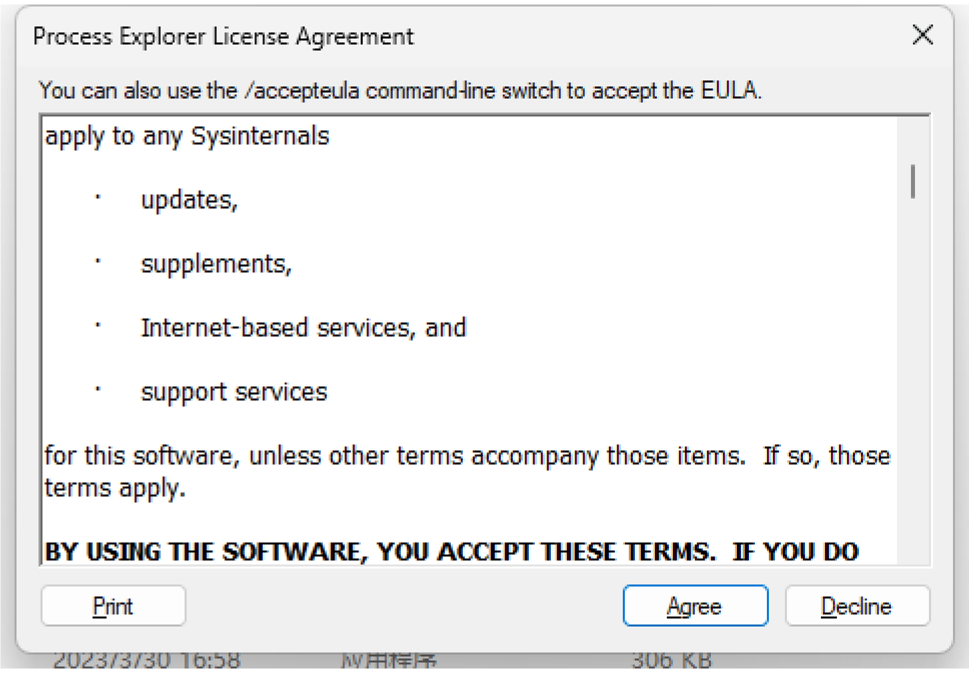
名称	类型	数据
ConfirmKill	REG_DWORD	0x00000001 (1)
DbgHelpPath	REG_SZ	C:\WINDOWS\SYSTEM32\dbghelp.dll
DefaultDllProp...	REG_DWORD	0x00000000 (0)
DefaultProcPro...	REG_DWORD	0x00000006 (6)
DefaultSysInfo...	REG_DWORD	0x00000000 (0)
Divider	REG_BINARY	00 00 00 00 00 00 e0 3f
DllColumnCount	REG_DWORD	0x00000004 (4)
DllPropWindo...	REG_BINARY	2c 00 00 00 00 00 00 00 00 00 00 00 00 00 00...
DllSortColumn	REG_DWORD	0x00000000 (0)
DllSortDirection	REG_DWORD	0x00000001 (1)
ETWstandardU...	REG_DWORD	0x00000000 (0)
EulaAccepted	REG_DWORD	0x00000000 (0)
FindWindowpl...	REG_BINARY	2c 00 00 00 00 00 00 00 00 00 00 00 00 00 00...
FormatIoBytes	REG_DWORD	0x00000001 (1)
GpuNodeUsag...	REG_DWORD	0x00000001 (1)
GpuNodeUsag...	REG_DWORD	0x00000000 (0)
HandleColumn...	REG_DWORD	0x00000002 (2)
HandleSortCol...	REG_DWORD	0x00000000 (0)

Figure 7: EULA Settings in Registry

Question:

- What is the value of the EulaAccepted registry key?

Answer: The value of the EulaAccepted registry key is 0x00000001(1), indicating that the user has accepted the End User License Agreement (EULA) for Process Explorer. This value is stored under the path HKEY_CURRENT_USER
Software
Sysinternals
Process Explorer. When the value is 1, it indicates that the EULA has been accepted; when the value is 0, it indicates that the EULA has not yet been accepted.



Process Explorer License Agreement

You can also use the /accepteula command-line switch to accept the EULA.

apply to any Sysinternals

- updates,
- supplements,
- Internet-based services, and
- support services

for this software, unless other terms accompany those items. If so, those terms apply.

BY USING THE SOFTWARE, YOU ACCEPT THESE TERMS. IF YOU DO

Figure 8: Result After Changing EulaAccepted Value to 0

Question:

- What did you see when you opened Process Explorer after changing the EulaAccepted value from 1 to 0?

Answer: After changing the EulaAccepted value from 1 to 0 and opening Process Explorer, the software displays the End User License Agreement (EULA) dialog box again, requiring the user to re-accept the license terms. This indicates that Process Explorer checks this value in the registry each time it starts to determine whether to display the EULA. If the user accepts the EULA, the value is reset to 1; if the user refuses to accept, the application will not start. This is a mechanism for the software to ensure that users understand and agree to the terms of use.

5 Lab Conclusion

Through this lab, we have successfully used tools from Windows Sysinternals Suite to explore and analyze processes, threads, handles, and the registry in Windows systems. We have learned:

- How to use TCP/UDP Endpoint Viewer and Process Explorer to monitor system processes
- The hierarchical structure of processes and parent-child relationships
- The characteristics and properties of threads as execution units within processes
- How handles serve as references for processes to interact with system resources
- How the Windows registry stores application configuration information

- How to change application behavior by modifying registry values

This knowledge is very important for system administration, security analysis, and software development, helping us better understand the internal working mechanisms of the Windows operating system.