Resolving Windows Performance Issues without opening a support case

Hands-on Lab Manual

Find Boot Delay Caused From Resource Consumption

Estimated time for completion: **45 minutes**

Introduction

In this exercise you take a boot trace where you will try to identify boot delays that are caused by a resource that is being fully consumed (resource contention)

> Both virtual machines should already be running. If it's not the case, please start them.

1. Take an xbootmgr trace

The Windows Performance Recorder (WPR) demoed in the presentation could also be used, but in this lab we are using xbootmgr because it can be used on XP/Server 2003 and Vista/Server 2008 whereas the Windows Performance Recorder can only be used on Windows 7 and later operating systems.

To start a trace you have to run xbootmgr as an administrator, therefore you will be prompted for elevation executing the command. An alternative that is often used is to open an elevated command prompt and then enter the command.

1. Open the **Win8_Boot1** machine and if necessary login with the following credentials:

   User name: "Win8SlowBoot"

   Password: "Password1"

> It's normal to take a while. This is an issue that many of our customers experience and that we will troubleshoot in this lab.

2. Press **CTRL+X** (or right click the bottom left corner of your screen) and click **Command Prompt (Admin)** to open an elevated command prompt.

If you don't open the command prompt as administrator the trace cannot be completed

3. Say **Yes** to the UAC prompt
4. Change directory with the command **cd c:\tracing**
5. Type **xbootmgr -trace boot -traceflags diageasy** and press enter to start a basic boot trace

Please note that as soon as you press **enter** your machine will reboot.

You should enter the credentials as soon as the machine restarts to make sure that the trace won't contain the delay of entering the credentials.

Important Note: Since the machine will have some resource contention it is possible that while typing the password the display of the characters is lagged. If you need to check whether you are typing the password correctly press the "eye icon"

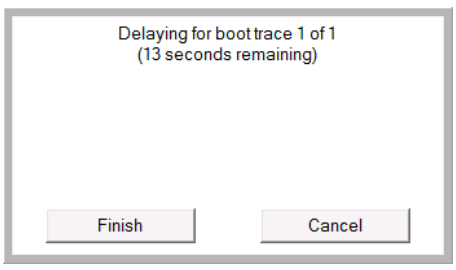
Command Explanation

-trace allows you to choose the trace type. Options are **Boot, Hibernate**, **Standby**, **Shutdown** and **RebootCycle**

-traceflags is used to specify **kernel flags** and/or **kernel groups** to that will be collected. For a full list of kernel flags please use **xperf -providers kf** and for a full list of kernel groups please use **xperf -providers kg** (or just **k** for both)

For more information on those switches and the others available please use **xbootmgr –help**

6. After explorer.exe has started (ie, when you see the Start Screen) xbootmgr will start a countdown of 120 seconds and then stop and merge the trace.



You need to click the desktop tile to see this box that is being presented on the desktop

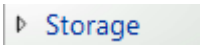
7. When the trace is complete a UAC prompt will pop up to allow xbootmgr to merge the trace. Press **Yes**
8. Rename the trace in **C:\Tracing\boot_diageasy_1.etl** file to **C:\Tracing\Lab_Initial.etl**

2. Looking at Disk Usage

You will open the trace in **Windows Performance Analyzer** and see many different graphs. Let's see how to separate what's relevant from what is not.

9. You can open the trace in one of several ways:
   i. Double click on the ETL file will automatically open it on machines that installed WPT (or if manually added a file association for .etl to wpa.exe)
   ii. Running: **wpa.exe "Path To The ETL File"**
   iii. Running **wpa.exe** and clicking **File -> Open**

   Use one of these methods to open the previously saved **C:\Tracing\Lab_Initial.etl**

10. Next let's have a look at the **Storage** utilization. Does the Storage graph on the left pane indicate that the disk is heavily utilized?
11. To drill down into the disk usage, press the arrow next to the word **Storage** 

12. Drag and drop the **Disk Usage** graph from the left pane to the right pane.
13. Select the highly utilized time period and go to table view by pressing this icon on the top right corner of the graph 
    i. Drag the **Process** column to the first place on the table. It will order disk utilization by process.
    ii. What processes are mainly responsible for this utilization?

    ---

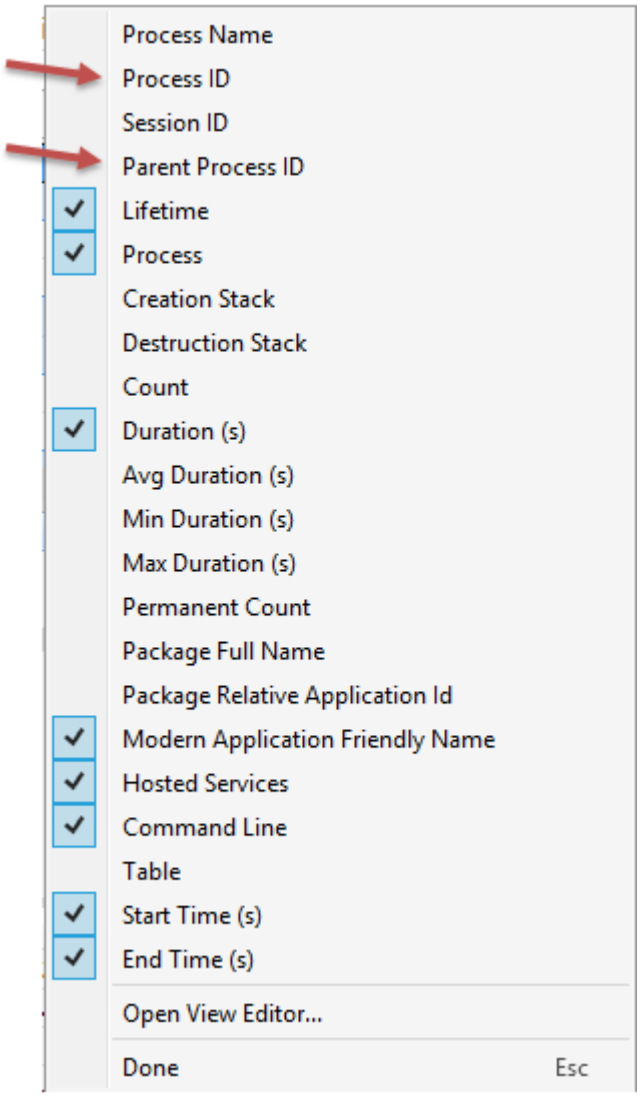
    iii. Now select the next period with some disk utilization after this high utilization period we just analyzed
    iv. What processes are mainly responsible for this utilization?

    ---

    v. Do you think the machine was booting slowly because of these applications?

    ---

14. Select the entire period of high disk utilization
15. Do not close the **Disk Usage** graph
16. On the left pane expand **System Activity** and drag the **Processes** graph to the right pane
17. Open the table view on this graph and locate the process that was responsible for the high disk utilization
18. Right-click one of the column headers and enable Process ID and Parent Process ID *(see screenshot below)*



19. Identify the parent process ID of the process that caused the high disk utilization
20. Locate the parent process by matching the IDs. Do that recursively until you have a clue of how that process that is responsible for a lot of disk utilization was started

Line #	Process ID	Parent Proc...	Lifetime	Process
47	2,968			
48		2,156	Transient	net.exe (2968)
49		52	Transient	xbootmgr.exe (2968)
50	2,976	892	Transient	gpscript.exe (2976)
51	3,012	2,976	Transient	cmd.exe (3012)

i. How was it started?

ii. Let's assume that this script is not necessary at boot time

21. Close the Windows Performance Analyzer

3. Looking at CPU Usage

22. You can open the trace in one of several ways:
    i. Double click on the ETL file will automatically open it on machines that installed WPT (or if manually added a file association for .etl to wpa.exe)
    ii. Running: **wpa.exe "Path To The ETL File"** in a command line
    iii. Opening and clicking **File -> Open**

    Use one of these methods to open the previously saved **C:\Tracing\Lab_Initial.etl**

23. Next let's look at **Computation.** From the graph in the left pane does it seem to be an occasion where the CPU is heavy utilized?
24. Expand **Computation** and drag **CPU Usage (Precise)** to the right pane.

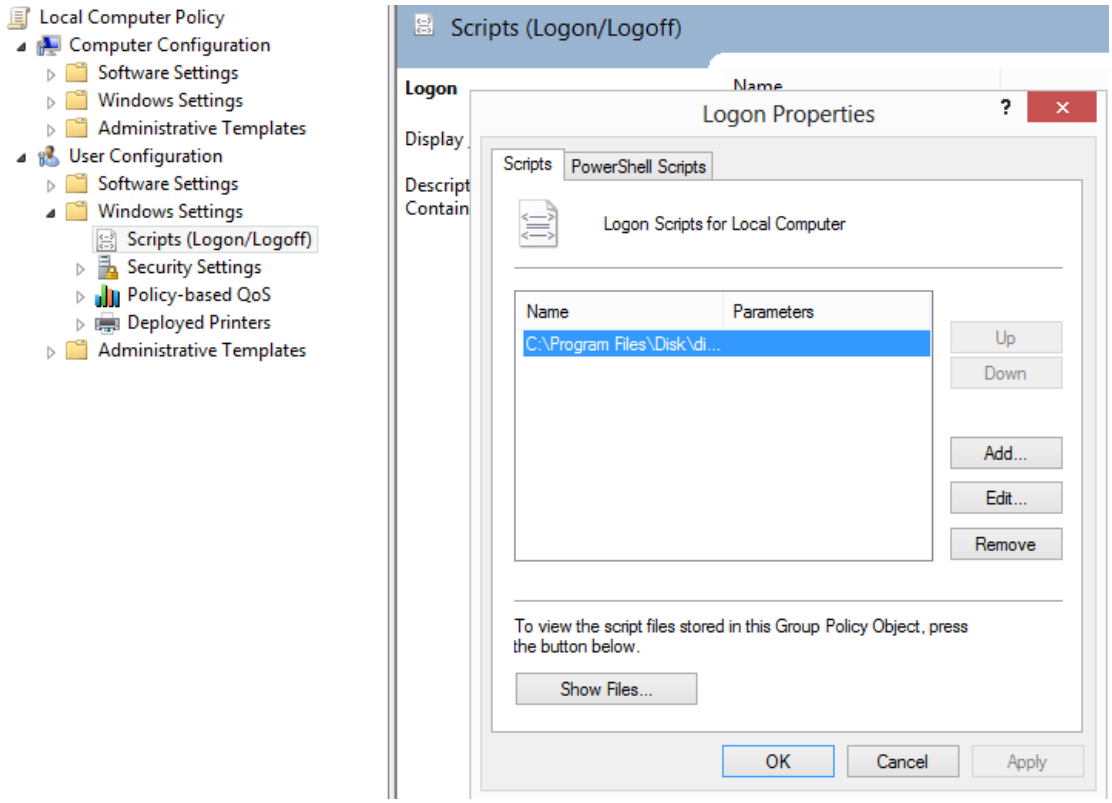
25. Which process is causing this?
    i. Select the fully utilized CPU time frame and change to table view
    ii. Find the highest consuming process
    iii. Search for that process in the **System Activity -> Process** graph to see how it was started and by who (using the **table view**)
        1. Find the parent process starting it (if you are having trouble finding it choose the **Process ID** and **Parent Process ID** columns)
        2. Is it harder to understand how this process was started? In this case the process was started by a Scheduled Task. The Task Scheduler is implemented as a service and that is why you will see Services.exe as the Parent Process.

> Finding which service is starting this executable is possible using the Windows Performance Toolkit but requires more advanced troubleshooting by looking at individual events

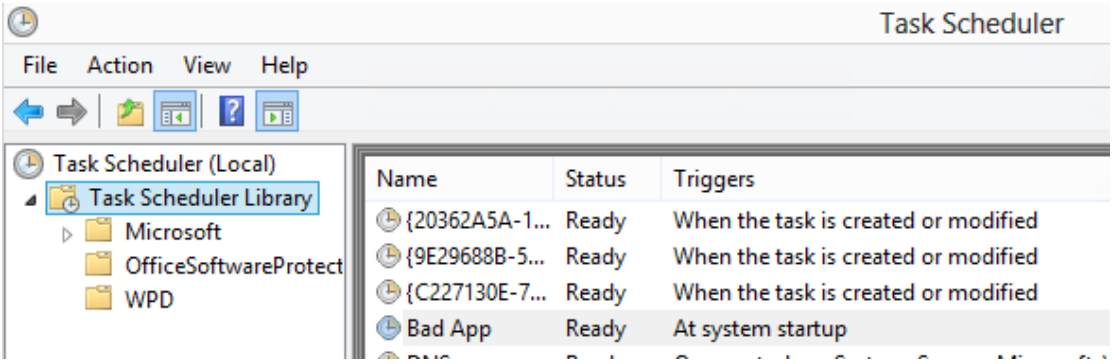
4. Disabling "misbehaving applications"

26. Finally fix the issues you found and reboot the machine.

    i. The group policy script can be disabled through **gpedit.msc**



    ii. The scheduled task can be disabled through **Task Scheduler**



27. Reboot the machine. Did the boot time improve?

28. You can take another boot trace to measure the improvements. Use the command from exercise 1.

29. Rename the trace to **C:\Tracing\Lab_Final.etl** and open with Windows Performance Analyzer

5. Advanced troubleshooting

30. While booting did you see any potentially unnecessary applications starting?
    i. How do you think they were started?

---

    ii. Select the period where they are started in the **Process** graph to see how it was started and by who (using the **table view**) just like you did before and figure out the parent process that started them
    iii. Now go to the **Generic Events** graph (should be the last graph in **System Activity**) and go again to the table view (for the time frame that these processes are starting)
    iv. Make sure that you are grouping (left to Golden bar) by **Task Name** followed by **Provider Name** followed by **Time**
    v. Look for the task name that has 2 things:
        1. The name of the process that started all the applications.
        2. The word “Executing” which leads to the startup of a child process

---