

Lab 1B

Windows and the Boot Process

ITSC205: Operating Systems Internals

NAME: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Table of Contents

[Lab Outcome(s) 3](#_Toc472198838)

[Reading 3](#_Toc472198839)

[Introduction 3](#_Toc472198840)

[1.0 Boot Process 4](#_Toc472198841)

[2.0 Windows Registry 9](#_Toc472198842)

[3.0 Analyzing System Start-up 12](#_Toc472198843)

Lab Outcome(s)

* Examine the boot sequence used by Windows.
* Explore the system components used to boot up a Windows environment.

Reading

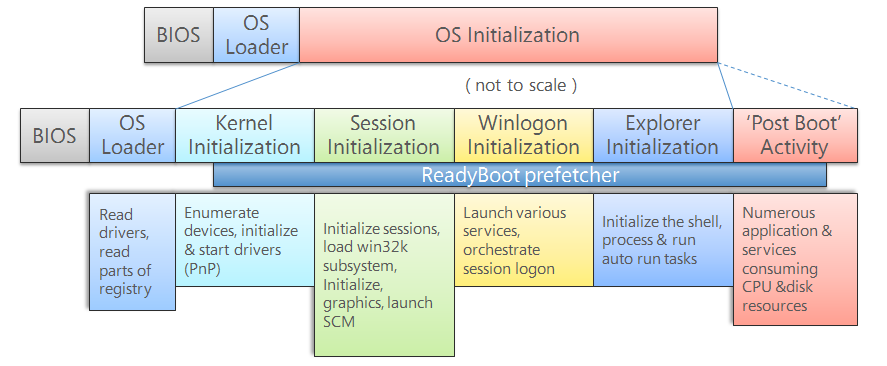
* Textbook chapter 21 – Windows 10 sections 21.2, 21.3.1, 21.3.3.10 and 21.3.3.11.

Introduction

A task as simple as starting up the computer is actually much more complex than it appears. The choice and order of a long series of operations must be configured for the computer to function correctly. The ability to configure and analyze the boot process is valuable, especially if the boot has been corrupted by a virus or malfunctioning software.

1.0 Boot Process \_\_\_\_/18

Many users judge the performance of an operating system by the speed in which it starts up. The time from powering on a machine to the time where the user can start working is a critical benchmark for Microsoft. The Windows boot process consists of several stages as shown below:



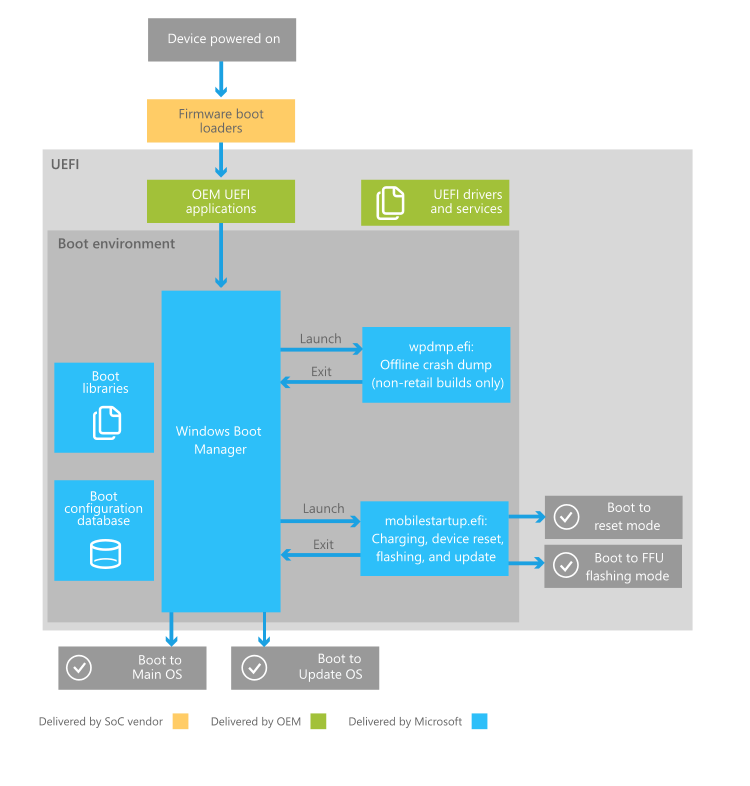
Source: <http://social.technet.microsoft.com/wiki/cfs-filesystemfile.ashx/__key/communityserver-wikis-components-files/00-00-00-00-05/1538.WindowsBootProcess.png>

The initial stage of the boot process is the same for all systems no matter which operating system is installed. This is commonly known as the BIOS initialization stage which includes the power-on self-test (POST) and loading the master boot record (MBR) from the device specified in the BIOS setup.

*Note: MBR is the first sector of a disk drive and the code stored on the MBR is the bootloader program.*

The MBR searches its partition table and reads the partition boot record (also called the volume boot record – the first sector of a partition/volume) and starts the second stage bootloader (OS loader). Windows 10 uses the **Secure Boot** on Unified Extensible Firmware Interface (UEFI) environments (if it’s enabled) to verify the bootloader’s digital signature before it loads the Windows Boot Manager to prevent rootkits from interfering with the boot process.

The boot manager is responsible for setting up the boot environment, executing any pre-boot applications such as the memory test program (memtest.exe) and loading the kernel in any of the various modes it provides in the following diagram:



Source: <https://i-msdn.sec.s-msft.com/en-us/windows/hardware/drivers/bringup/images/oem-boot-flow-detail.png>

The Windows boot manager (bootmgr.exe) read information from the Boot Configuration Database (BCD).

*Note: The BCD replaced the boot.ini file that was used in versions prior to Windows 7.*

1. Research and briefly describe the purpose of the BCD.

1. Open a console window by right-clicking the shortcut on the desktop and select “Run as Administrator”. Execute bcdedit (no options).
2. What is the device and path to the OS?
3. Which directory is the root of the OS?

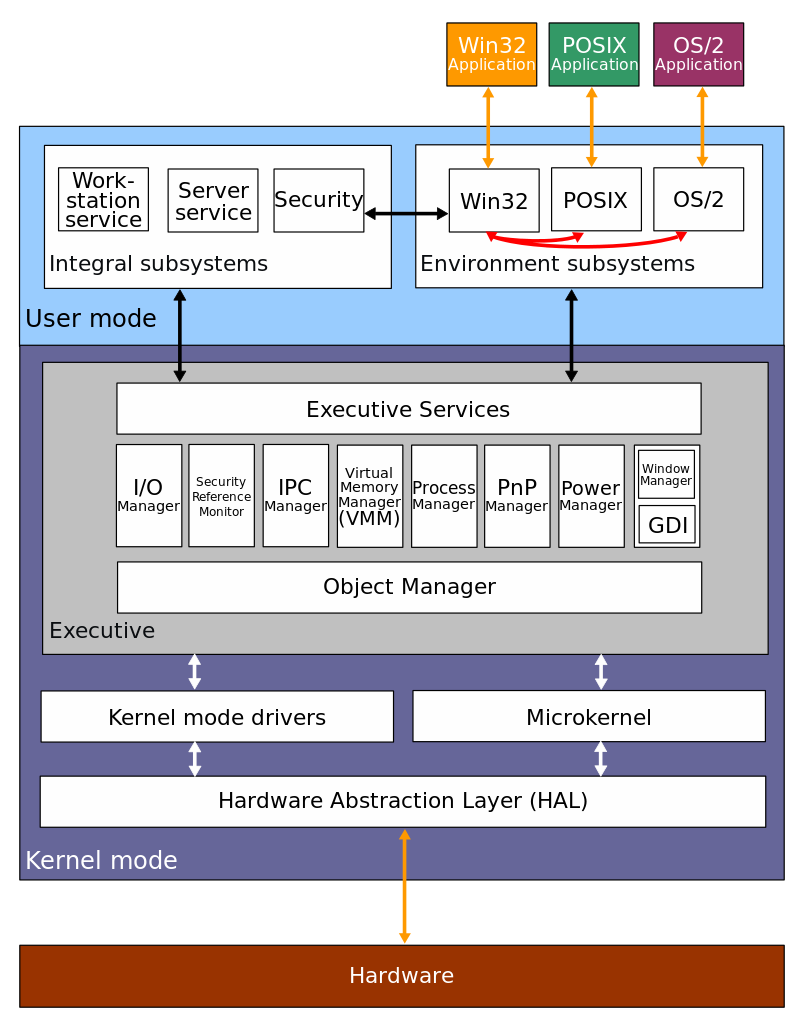
This data is used to boot Windows 10, other operating systems or system versions could be stored in the BCD, read by the bootmgr and the OS selection menu would be displayed.

1. Use bcdedit /? to determine the purpose of the command bcdedit /export. In what situation would this command be useful?

The Windows loader (winload.exe) will load basic drivers to prepare the system for the kernel to take over. The Windows 10 kernel (ntoskrnl.exe) will begin by reading the disk data and loading any additional drivers that are marked as BOOT\_START into memory.

*Note: if the system was hibernating, winresume.exe will be loaded instead of winload.exe.*

During the kernel initialization stage, the system is being prepared for running native applications. This involves initializing the executive subsystem and the hardware abstraction layer (HAL).



Source: <https://commons.wikimedia.org/wiki/File:Windows_2000_architecture.svg>

The kernel than passes control to the session manager (smss.exe). The initial instance (session 0) will initialize the registry, loads other drivers that were not marked BOOT\_START and starts the Windows subsystem processes.

You can see the order in which Windows loads device drivers and starts services using the **LoadOrder** utility from Sysinternals.

Many Sysinternals tools have multiple versions – the command-line version has the character ‘c’ appended to the end of the name for tools that has a GUI version. The numbers ‘64’ indicates the 64-bit version and is launched by the 32-bit counterpart when the tool starts up and detects a 64-bit environment.

1. Run the LoadOrd.exe utility from the Sysinternals Suite folder.

How would you use this output to search for a particular device or service without looking through the list one by one?

The Windows initialization process (wininit.exe) starts the Service Control Manager (SCM), the Local Security Authority (LSASS) and the Local Session Manager (LSM). It also completes the initialization of the registry and performs user-mode initialization.

All other session instances will create its own Client/Server Runtime Subsystem process (csrss.exe) to handle the Graphics Device Interface (GDI) shutdown, and login subsystem process (winlogon.exe) which coordinates logon and user security.

During the winlogon initialization stage, service.exe will load and initialize auto-start drivers and services. The login screen (logonui.exe) will prompt the user for login credentials and if authentication is validated, control will pass to the Windows Program Manager commonly known as the Windows Explorer (explorer.exe).

Windows Explorer will create the Desktop Window Manager (dwm.exe) which initializes the desktop and displays it. Now the system is ready for user input!

1. Open Task Manager and go to the Details tab to complete the following table:

|  |  |  |
| --- | --- | --- |
| **Process Name** | **User Name** | **Description** |
| smss.exe |  |  |
| wininit.exe |  |  |
| csrss.exe |  |  |
| winlogon.exe |  |  |
| explorer.exe |  |  |
| dwm.exe |  |  |

2.0 Windows Registry \_\_\_\_/12

The Windows registry is a large file of settings and related information for installed programs and parts of Windows. Information is kept in groups (like folders) identified by a name called the *key*. An individual unit of information is kept in a *value* that includes its name and its data. A key may also hold *subkeys*. For example, Windows may or may not require CTRL-ALT-DEL (CAD) to reach the login screen at logon time. This is controlled by a value in the registry key:

HKEY\_LOCALMACHINE/SOFTWARE/Microsoft/Windows NT/CurrentVersion/Winlogon

The value is named: DisableCAD. The data can be 0 (false) or 1 (true).

1. Research and briefly describe **three** of the following registry *root* *keys*: HKEY\_CLASSES\_ROOT, HKEY\_CURRENT\_USER, HKEY\_LOCAL\_MACHINE, HKEY\_USERS, HKEY\_CURRENT\_CONFIG.
2. Which root key uses HKLM as an abbreviation?
3. Incorrectly changing registry values can lead to serious problems, including inability to boot. Nonetheless, registry changes are made to obtain a well-tuned system. A system can be protected from serious consequences by having Windows save its most recent software configuration in a *restore point* – a backup file for the software configuration.

Use Start 🡪 Windows System 🡪 Control Panel. Select System and Security 🡪 System 🡪 System Protection to create a restore point.

1. The registry editor utility is used often by IT professionals. It can be made very conveniently available by adding it to the taskbar.

Start 🡪 Search and run REGEDIT. Right-click on the RegEdit program icon on the taskbar and select “Pin to taskbar”. Now it will always be available for a quick start.

1. Study the values in the registry key:

HKEY\_CURRENT\_USER\Control Panel\Desktop

Modify two values that you understand and can verify. What are the values you have modified and what does it do?

1. Research and briefly describe the registry key:

HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Run

Contrast this key with the HKCU\…\RunOnce key:

Modify the relevant key value(s) to automatically start the Notepad program when a user logs in, using **both** of these keys. **Demonstrate and explain** the results to the instructor.

|  |  |
| --- | --- |
| **ASK THE INSTRUCTOR TO SIGN OFF** |  |

Sysinternals includes a useful tool called Registry Usage (RU) that reports the space consumed by the registry key you specify. Beyond this ability to address a bloated registry key, RU can also report the date and time a registry key or its content was last modified when outputted to a comma-separated values (CSV) file.

1. Open a console windows and run the following command:

ru -c hkcu\software\microsoft\windows\currentversion\run | clip

1. Open Notepad and paste the contents of the clipboard into the editor to view the results.

Why would knowing the last modified time of a registry key be useful?

3.0 Analyzing System Start-up \_\_\_/10

Windows provides a basic tool that can be used to configure boot and startup system settings.

1. Start 🡪 Search programs for MSCONFIG. Run that program.
2. Use the program’s help and/or research to describe the differences between Normal, Diagnostic and Selective startup.
3. List the items that are run at startup. In what situation would it be useful to disable one or more startup items?

The system configuration utility provided by operating system only shows a small subset of software that runs automatically during boot. **Autoruns** is a powerful tool from the Sysinternals Suite you had downloaded in the previous lab.

This tool allows you to view every program that has been configured to start automatically and make it easy to disable or remove these programs. It goes beyond system startup and can show startup of any application and extensions. It can even analyze offline systems from a virtual disk image.

*Note: Autoruns is not magic! It exposes as many programs as possible but some very clever malware may not be uncovered (such as kernel-mode rootkits).*

Another advantage of the Sysinternals Suite is its integration with VirusTotal – an Antivirus as a Service (AaaS) which checks the programs against 50+ malware engines to identify suspicious programs.

1. Start Autoruns and study the user interface. The 20 tabs may seem intimidating at first but it’s really just a breakdown of the **Everything** tab into specific categories.
2. The Publisher column shows the company name of the file. Anyone can write a program and put the company name as “Microsoft Corporation”! You can verify the entry’s digital signature by right-clicking any entry and select Verify Image.

Instead of verifying each entry one by one, do the following:

1. Click on Options from the menu and select Scan Options.
2. Enable Verify code signatures.
3. Rescan.

If the file has been signed with a valid code-signing certificate that derived from a root certificate authority that is trusted, the text in the Publisher column changes to “(Verified)”.

1. Repeat the above steps to check all the entries against VirusTotal.com.

*Pro-tip: Crackers check this too! So if you think you’re under a targeted attack, don’t submit upload the suspected file!*

**Demonstrate and explain** the results to the instructor.

|  |  |
| --- | --- |
| **ASK THE INSTRUCTOR TO SIGN OFF** |  |

**EVALUATION**:

|  |  |  |
| --- | --- | --- |
| System boot process | 18 |  |
| Windows registry | 12 |  |
| Analyzing Start up process | 10 |  |
| TOTAL MARK | 40 |  |