

Lab 1

Windows Basics

ITSC205: Operating Systems Internals

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

Table of Contents

[Lab Outcome(s) 3](#_Toc472025633)

[Reading 3](#_Toc472025634)

[Introduction 3](#_Toc472025635)

[1.0 Set up Windows Virtual Machine 4](#_Toc472025637)

[2.0 Explore Windows Settings and Services 6](#_Toc472025638)

3.0  [Windows Boot Process –BCD -UEFI 7](#_Toc472025640)

[4.0 Explore System Internals Suite 13](#_Toc472025639)

[5.0 Windows Registry 15](#_Toc472025640)

L*abs must be submitted by the due date for full credit. After due date late submissions will be accepted for a period of one week (seven days) and the grade will be reduced by ten percent (10%) per day after due day.* ***Assignments that are submitted more than seven days late will receive a grade of zero (0).***

I certify that the work submitted in this assignment is my own and that it has not been taken in whole or in part from any other source. I understand that the penalty for plagiarism will include a grade of zero (0) for this assignment plus disciplinary action in accordance with SAIT policies.

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**EVALUATION**:

|  |  |  |
| --- | --- | --- |
| Use Windows tools to explore operating system settings and services | 15 |  |
| Windows Boot process – BCD and UEFI | 20 |  |
| Explore System Internals tools | 10 |  |
| Windows Registry | 10 |  |
| TOTAL MARK | 55 |  |

Lab Outcome(s)

* Set up virtual environment for course lab exercise.
* Examine the Windows environment provided by the system user interface and the Sysinternals suite of advanced system utilities.
* Examine Windows boot Process.
* Explore the system components used to boot up a Windows environment.

Reading

* Textbook Chapter 21- Windows 10 –sections: 21.1 History and 21.4.
* Textbook chapter 21 – Windows 10 -sections 21.3.5.10 (Registry) and 21.3.5.11 (Booting)

Introduction

In this lab, we will set up the virtual machine environment that is used as the basis for studying the Windows operating system in detail for this course. We will study some of the tools, provided by the operating system and external, to examine various information regarding the system itself.

Services on Windows are usually non-interactive (does not directly accept user input) and run consistently in the background. Examples of services include event-logging and the firewall. Many antivirus products run as services as well. Malicious code often leverages services for persistence (to survive reboots), to load kernel drivers and to blend in with legitimate components of the system.

1. Prepare Virtual Machine for Labs

In this course, we study operating systems in a safe environment that allows us to experiment without ruining the configuration of the computers we use for everyday work. The safe environment is a *virtual machine*: a program that **acts like** a physically separate computer. Virtual machine (VM) software is installed on and run on a *host* system. A *guest* operating system is installed onto the VM, and the VM runs programs just as if they were running on a separate computer. We will use the Oracle VM VirtualBox virtual machine system.

1. Obtain the files you will need:

* The latest version of “VirtualBox platform package” virtual machine software from <http://www.virtualbox.org/wiki/Downloads>. Select the host suitable for your machine. (E.g. if your machine is running Windows, select “Windows hosts”.)
* A pre-configured “image” of Windows 10 to run in the virtual machine (obtain from instructor).

1. Run the VirtualBox installer you downloaded. Install the program using the defaults provided.
2. Unzip the image file. (The 7-Zip program should be installed on your computer. If it is not, then get it from <http://www.7-zip.org/>.)
3. Configure the Windows 10 virtual machine:
4. Create a new virtual machine (click the “New” button).
5. Name the machine “Win10” or something similar and specify operating system “Microsoft Windows” and version “Windows 10 64-bit”.
6. Select a base memory size of 4096 MB.
7. Configure the Virtual Hard Disk. Select “Use an existing virtual hard drive file” and click the folder icon. Browse to the Windows 10 hard disk image directory you extracted in step 3. Select “Windows10.vdi”.
8. Click “Create”.
9. View the various system settings.
10. Start the Windows 10 virtual machine. Log in to the user account using the password “OSExploits”.
11. The Windows system will take quite a while to configure itself including the automatic installation of needed device driver code and updates to the operating system. The process may require multiple restarts.
12. Restart the Windows virtual machine and configure the VirtualBox virtual display. “VirtualBox Guest Additions” allows the guest OS to resize the display **plus** adds other features used in **later** labs.
    1. Under Windows 10 guest virtual machine window select: Devices 🡪 Insert Guest Additions CD Image; The system should mount the CD with the image
    2. Login into Windows 10 virtual machine and under This PC click twice on the CD image to install the tools. Install using defaults; restart; resize guest window as desired.

**Note:** Rt-Ctrl is the default “Host” key to communicate with VirtualBox on the Host (e.g., Rt-Ctrl-F switches back and forth between full screen and windowed mode).

2.0 Explore Windows Settings and Services \_\_\_\_/15

1. Explore the following Windows tools and complete the information in the following table:
   1. From Windows System you can use : cmd, control panel, task manager and This PC tools to explore Windows operating system settings
   2. Search for System Information or run msinfo32.exe
   3. Command Line. Start cmd and type help to explore Windows commands that can be used to display Windows kernel version, computer name (hostname) and other hardware settings.

|  |  |
| --- | --- |
| Operating system version and build |  |
| System Type (32 bit or 64 bit OS) |  |
| CPU type and speed |  |
| Boot Device Path |  |
| BIOS Mode |  |
| Memory RAM size |  |
| Virtual memory size |  |
| Is Hyper-V Virtualization enable? |  |

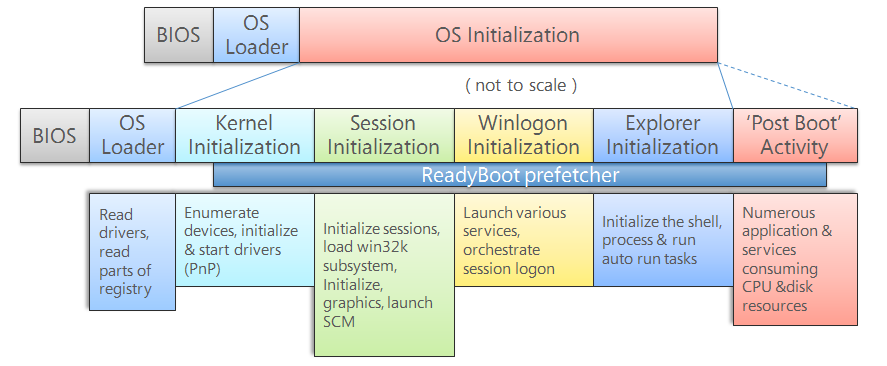
* 1. How many physical CPUs are there in your system?
  2. How many cores are there in your system?
  3. How many Logical processors?

1. Use the Start Menu 🡪 Windows Administrative Tools 🡪 Services utility to view the services installed on the virtual machine.
   1. What service protect users form malware?
   2. What service manages event logs?
   3. What Windows service(s) can be used to manage updates?
   4. Briefly describe all the Windows Defender services running in the system?

3.0 Windows Boot Process \_\_\_/20

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.

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 Boot Process:**

1. BIOS
2. Power-On Self-Test (POST)
3. Load Master Boot Record (MBR) from the device specified in the BIOS setup. MB *is the first sector ( 0 ) of a disk drive and contains boot code and data ( partition table). The partition table contains 4 entries for primary partitions*
4. *MBR boot code searches for partition table for the active partition and loads this partition’s boot sector into memory*
5. *The active partition’s boot sector known as Volume Boot Record (VBR) is the first sector of the partition. The boot code in VBR read’s Partition’s File system and locates Bootloader in Windows called boot manager* ***bootmgr***
6. *BIOS loads Boot manager* ***bootmgr*** *into memory. If the system is configured to boot with Extended Firmware Interface (EFI) instead of BIOS then the boot loader will load* ***bootmgfw.efi***
7. ***bootmgr*** *performs a test to verify the integrity of its own image by loading digital signature catalogue (nt5.cat)*
8. ***bootmgr*** *runs in* ***real mode*** *and reads Boot Configuration Database* ***BCD*** *located in /windows/boot directory and for EFI is under windows/boot/efi*
9. ***bootmgr*** *loads Windows loader* ***winload .exe*** *and**switches to* ***protected mode***
10. ***winload.exe*** loads Windows kernel **ntoskrnl.exe** and will load basic drivers to prepare the system for the kernel to take over. If integrity checking is enabled and do not fail, the DLLs ( Dynamic Link Libraries) imported by ntoskrnl.exe are loaded, digital signatures are verified and initialized.
11. 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.

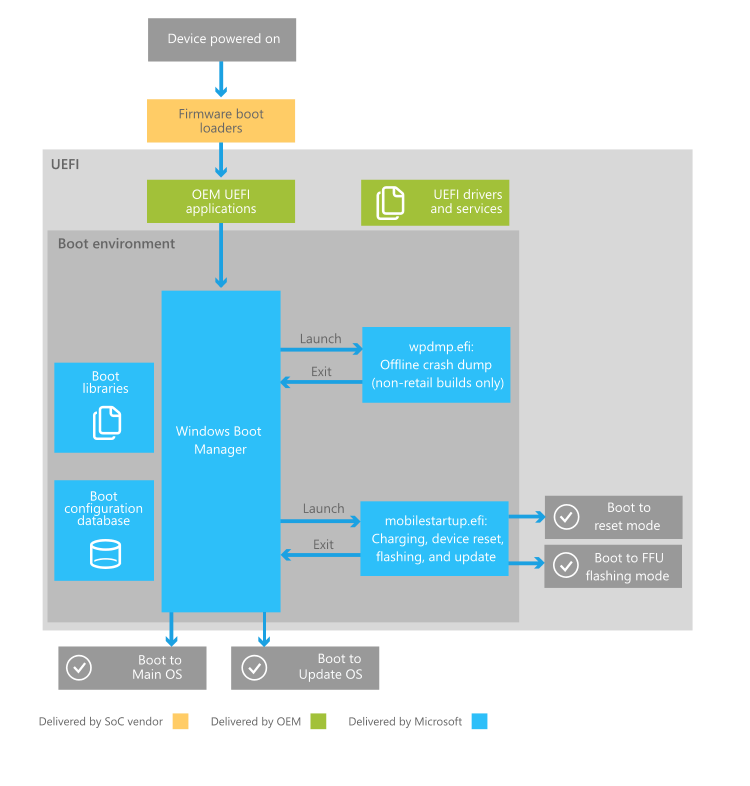
*Note: if the system was hibernating, winresume.exe will be loaded instead of winload.exe.*

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).

1. The **kernel (ntoskrln.exe)** 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.
2. **Smss.exe** is the first process created by the kernel and it performs many tasks such as initialize Win32k.sys system driver and loads **Windows initialization process wininit.exe**
3. **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.
4. 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.
5. 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).**
6. 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!.

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 provided in the following diagram:



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

1. Use the search tool to find the absolute path of windows kernel executable **ntoskrnl.exe** file and write down the purpose of this file, file path and file size.
2. Use notes and handout provided in class to identify the processes that participate in Windows boot process. For this exercise you can use either the host machine or the virtual machine. 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 |  |  |

1. Use system information utility and answer the following question:
   1. Does the system boots using legacy BIOS or firmware UEFI?
2. What are the advantages of using Firmware UEFI instead of legacy BIOS?
3. List two ways to verify if secure boot is enabled or not

**Managing Boot Configuration Database - BCD**

**NOTE: You have to use your Windows Virtual Machine to complete this exercise**

The Windows boot manager **(bootmgr.exe)** read information from the **Boot Configuration Database (BCD).** The ability to configure the boot process is valuable, especially if the boot has been corrupted by a virus or malfunctioning software. In Windows 7, the **boot.ini** file has been replaced with Boot Configuration Data (BCD). Microsoft Windows **bcdedit** is a tool used to manage boot loader entries.

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

1. Start cmd as “Administrator” and use the command **bcdedit** . Analyze the output and differentiate between Windows Boot Manager and Windows Boot Loader.
2. Attach a screen capture that demo the results. Based on the output answer the following questions:
   * What is the device of boot manager?
   * Identify the path of boot manager. Based on this output, is the system booting using BIOS or UEFI?
   * Analyze Windows Boot loader results and identify **nx** . Search <https://docs.microsoft.com/en-us/windows-hardware/drivers/devtest/bcdedit--set> and find out the purpose **nx**.
   * Is nx enable or disable?
   * Identify root directory of the OS?
3. Use **bcdedit /?** to determine the purpose of the command **bcdedit /export**. In what situation would this command be useful?
4. Create a copy of the current **bcd entry** by typing the command :

**bcdedit /copy {current} /d “ New Windows 10 -Safe Mode”**

1. Use bcdedit at command line to verify the new entry. Windows also provides a basic (GUI) tool that can be used to verify and configure boot and startup system settings called System configuration (msconfig)
2. To verify the new entry created run msconfig program. Under Boot Tab select the second entry called “New Windows 10-Safe Mode” and change the boot options to boot with Minimal drivers. Use bcdedit to verify the changes made in the boot.
3. Attach a screen capture that demo the new entry created in the bootloader
4. Reboot the system and boot the system from the second entry “New Windows 10-Safe Mode”
5. Attach a screen capture that demo the new entry created in the bootloader after rebooting the system and screen capture that demo Windows 10 running in Safe Mode
6. To verify the drivers (module) loaded during boot process use the following bcdedit command that will generate a log file called Ntblog.txt in the %SystemRoot% directory.

**bcdedit /set BOOTLOG TRUE**

1. Reboot the system, find the Ntbtlog.txt , analyze the output and identify the Windows kernel and the libraries loaded by the kernel. Write down the name of the executable file and the first 5 libraries (.dll) loaded by the kernel.

4.0 The Sysinternals Suite \_\_\_\_/10

The utilities provided by the Windows operating system is sufficient for everyday use but as an IT security professional, you need more advanced tools to diagnose and troubleshoot your system.

1. Download the **Sysinternals Suite** from [www.sysinternals.com](http://www.sysinternals.com) (which will redirect you to <http://technet.microsoft.com/en-US/sysinternals> since Microsoft’s acquisition of Sysinternals in 2006).
2. Before you extract the tools from the zip file, you can remove the marker that tells Windows to treat the content of the file as coming from the internet and is untrusted. This will get rid of the security warnings when you run any of the tools or display content errors when viewing the help files.

To do this, right-click on the SysinternalsSuite.zip file in folder what you have downloaded it in the previous step. Go to the Properties dialog box, on the General tab near the bottom – select the Unblock checkbox and click OK.

1. Extract the tools from the zipped file to a location where you will be able to locate easily in the future as we will be using many of these tools for all the lab exercises in the course.
2. Open a command prompt and navigate to the directory where you have extracted the sysinternals tools.

*Pro-tip: Type “cmd” in the address bar of file explorer and the command prompt will start in the same directory location.*

1. **Autoruns** is a powerful tool from the Sysinternals Suite. 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.

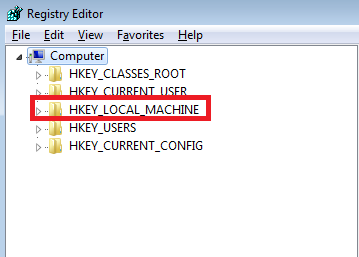
1. Start Autoruns and study the user interface. The 20 tabs may seem intimidating at first but it’s really just a breakdown of the **Everything** tab into specific categories.
2. The Publisher column shows the company name of the file. Anyone can write a program and put the company name as “Microsoft Corporation”! You can verify the entry’s digital signature by right-clicking any entry and select Verify Image.
3. Instead of verifying each entry one by one, do the following:
4. Click on Options from the menu and select Scan Options.
5. Enable Verify code signatures.
6. Rescan.
7. 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)”.
8. Repeat the above steps to check all the entries against VirusTotal.com.
9. Attach respective screen captures to demo results

5.0 Windows Registry \_\_\_\_/10

The Windows registry is a **database** that contains the information required to boot and configure the system. It contains many settings that affect system performance and behavior. 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*.

**NOTE: Use Windows Virtual machine to complete Windows registry activities**

1. The registry may be explored and edited using **regedit command**. Run **regedit** and explore Windows registry. Research, list, and briefly describe each ***hive* key** in the registry.
2. Search the registry **HKEY\_LOCAL\_MACHINE** and find :
   1. Boot Configuration Database **BCD**
   2. System Services
   3. Attach screen captures to demo results



1. 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).

Access this registry value DisableCAD and disable CTRL\_ALT\_DEL by using the respective 0 or 1 value. Restart the computer and attach a screen capture that demo the results

1. Search the registry to find exactly **AutoAdminLogon** value and write down the hive key,subkey and its value data :
   1. Change the value data of AutoAdminLogon from 0 to 1.
   2. In the same place where AutoAdminLogon is located, create a new value with the following settings:
      1. Value name: DefaultPassword
      2. Value Type: String
      3. Value: OperatingSystems
   3. Check the value of DefaultUserName. The value should have the user name of the user you wish to logon automatically.
   4. Restart the machine and create and attach a screen capture to demo the results. What is the effect of changing the value of AutoAdminLogon ?
2. The information in the registry can be used for different purposes. It can be used for troubleshooting the system and also in forensic investigations. Investigators and system administrators can access and analyze useful registry information about attackers’ actions. Example investigators use RecentDocs value stored at: HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs to display the documents opened by the user currently logged into the system. These documents are stored by extension but the files paths and names are stored in binary values.
3. To test the effects of this key, open and use different applications such as notepad , wordpad and games and verify the results at the key:

HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs

1. Attach a screen capture to demo results
2. 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.

1. Attach the screen captures to demo the results .