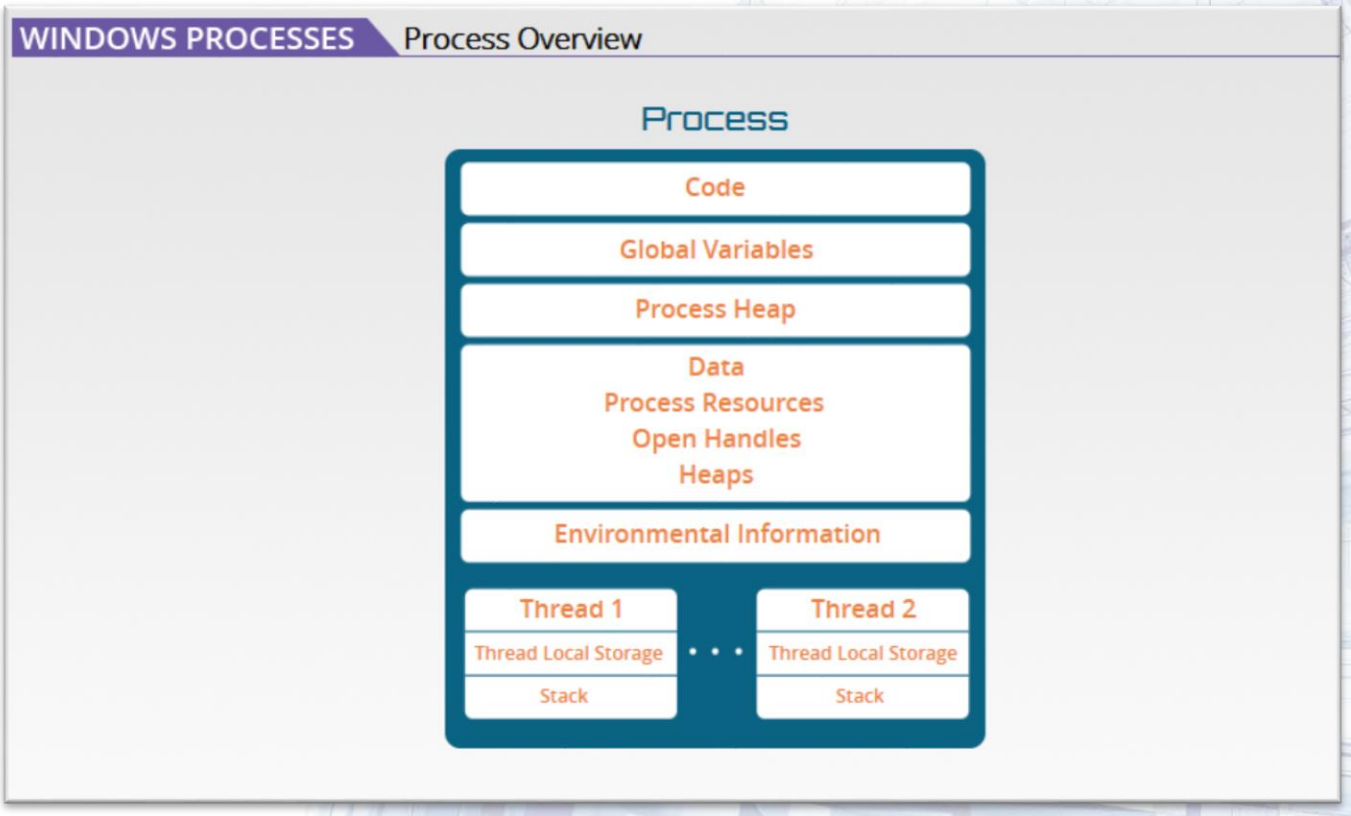
General Guidance

You are not expected to memorize supplemental and reading materials; however, you should be familiar enough with the content to use as a reference in a testing environment.

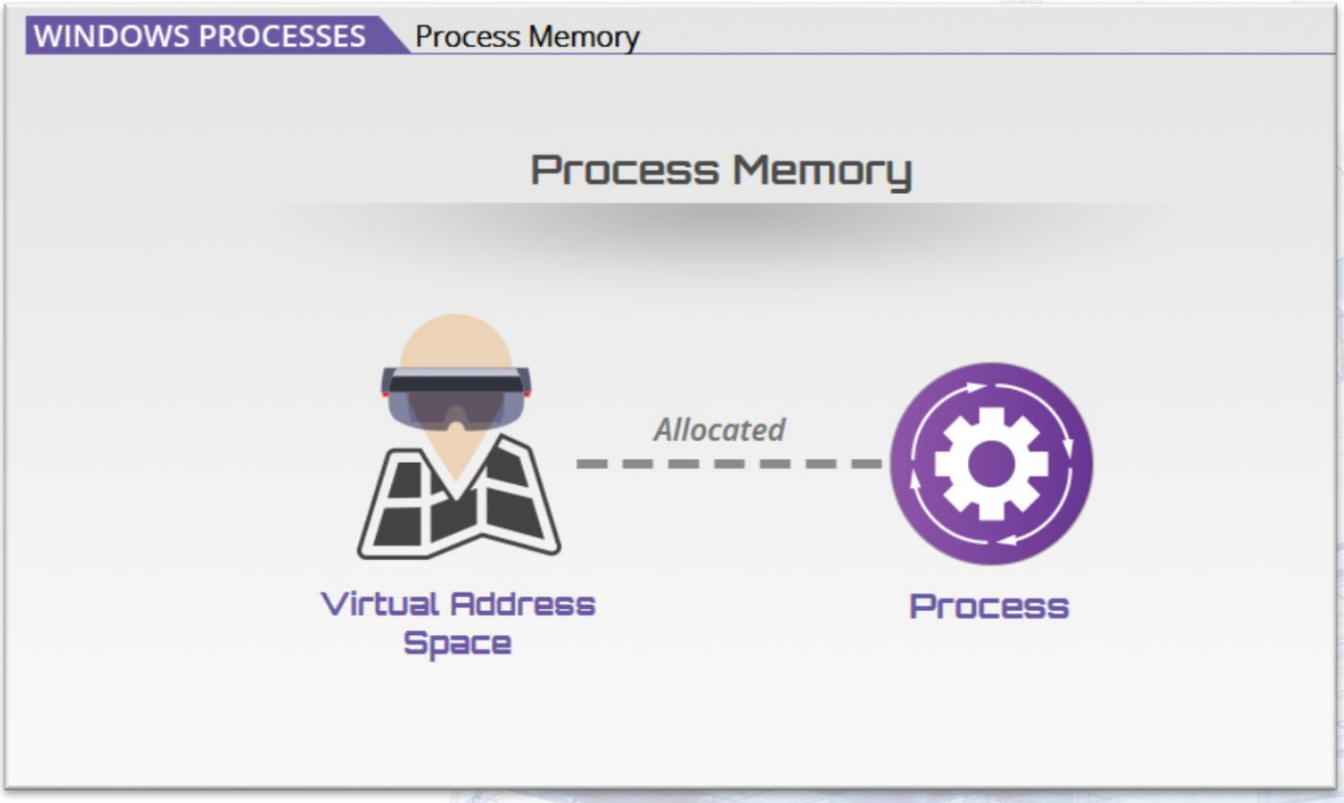
Process

A process is an instance of a computer program that is being executed and is running on a computer. A process is defined as a container for a set of thread resources. Within the Windows operating system, each process has its own assigned virtual address space and is independent to the virtual address space assigned to other processes.



Process Memory

Process memory is the virtual address space allocated to a process at the time the process is created. This virtual address space is the set of virtual addresses the process can use. This address space is private and cannot be accessed by other processes unless it is shared.

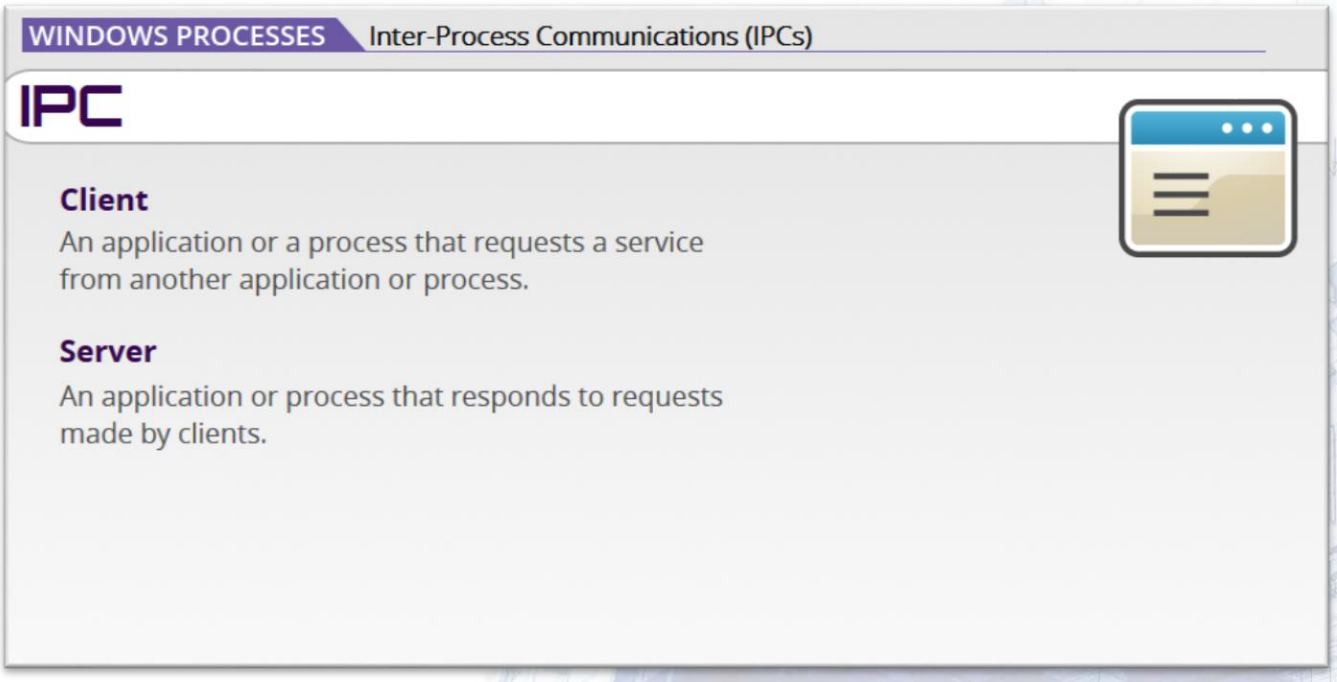


A process' virtual address space consists of the several categories:

Working set	The working set is the amount of memory physically mapped to a process at a given time.
Paged pool	Memory in the paged pool can be transferred to the paging file on the disk, referred to as paged out, when it is not being used.
Non-paged pool	Memory in the non-paged pool cannot be transferred to the paging file on the disk as long as its corresponding objects are allocated.
Pagefile	The pagefile monitors memory usage and how much memory is set aside for the process in the system paging file.

Inter-Process Communications (IPCs)

Inter-Process Communications (IPCs) are Windows operating system mechanisms that allow applications and processes to communicate and exchange data. IPCs are typically categorized as either clients or servers.



IPC mechanisms include, but are not limited to the following:

Pipes Windows implements two types of pipes for two-way communications: anonymous pipes and named pipes.

- **Anonymous pipes** Are used to transfer information between related processes and only support communication in a single direction. In order to exchange data in both directions, you must create two anonymous pipes. Anonymous pipes are only implemented within the local computer. Anonymous pipes provide an efficient way to redirect standard input or output to child processes on the same computer.
- **Named pipes** Are similar to TCP in that they provide reliable, connection-oriented, duplex communication paths to transfer data between unrelated processes or processes on different computers. A named-pipe server process creates the named pipe with a well-known name that can connect to the open end of the named pipe. Once this connection is established, data can be exchanged between the client and server processes. Named pipes provide a simple programming interface for transferring data between two processes, whether they reside on the same computer or over a network.
    - **Mailslots** Are created by a mailslot server and provides an unreliable, connectionless, one way communication path. Mailslot clients send messages to the mailslot server by writing messages to its mailslot. A mailslot is a pseudo-file to which incoming messages are appended and saved until they're read by the mailslot server.
    - **RPC** Remote Procedure Call (RPC) allows applications to call functions remotely (remotely refers to the application, not the computer). The remote function called by the application executes on the local or a remote computer, but external to the calling application
    - **Windows socket** Is a protocol-dependent interface that takes advantage of the communication capabilities of other underlying protocols. It is capable of supporting current and emerging networking capabilities. An application that uses sockets can communicate with other socket implementations on other systems.
    - **Clipboard** When performing copy and paste operations between applications, the clipboard acts a central repository for the data to be shared. Copied or cut data in an application is placed on the clipboard. Once this data is placed on the clipboard, any other application can retrieve it.

Process Enumeration Tools

Command-line tools (CLIs) view and manage processes and process information. Below is a list of CLIs along with an explanation of each.

PsList	PsList is part of the Sysinternals tool suite that provides a listing of the running processes on a system. Its output includes details about each running process and can be used locally or against a remote computer. This tool can run on either a 32- or 64-bit system.
TaskList	TaskList is a Windows native utility built into the command shell. It provides a listing of the system's currently running processes. This tool can run on either a 32- or 64-bit system.
PsKill	Also part of the Sysinternals tool suite, PsKill allows process termination. This tool can be run on either a 32- or 64-bit system.
TaskKill	TaskKill, similar to taskList, is also a Windows native tool built into the command shell. It allows process termination. This tool can run on either a 32- or 64-bit system.
PsSuspend	Also part of the Sysinternals tool suite, PsSuspend provides the capability to suspend a running process or resume a suspended process. This tool can run on either a 32- or 64-bit system.
Handle	This tool is part of the Sysinternals tool suite. When run, handle provides a list of all the open file references on the system for each running process. This tool can run on either a 32- or 64-bit system.
ListDLLs	Another tool that is part of the Sysinternals tool suite. It provides a listing of all DLLs that are loaded by the running processes. This tool can run on either a 32- or 64-bit system.
Pmon	Process Resource Monitor (Pmon) is part of Windows Resource kit. Pmon is a command-line tool that displays several measures of the CPU and memory use of processes running on the system. Pmon only works on 32-bit systems.

Integrity Levels

The component store uses NTFS hard links between itself and other Windows directories to increase the robustness of the Windows platform.

- The different Integrity Levels include:
  - Untrusted
  - Low
  - Medium
  - High
  - System

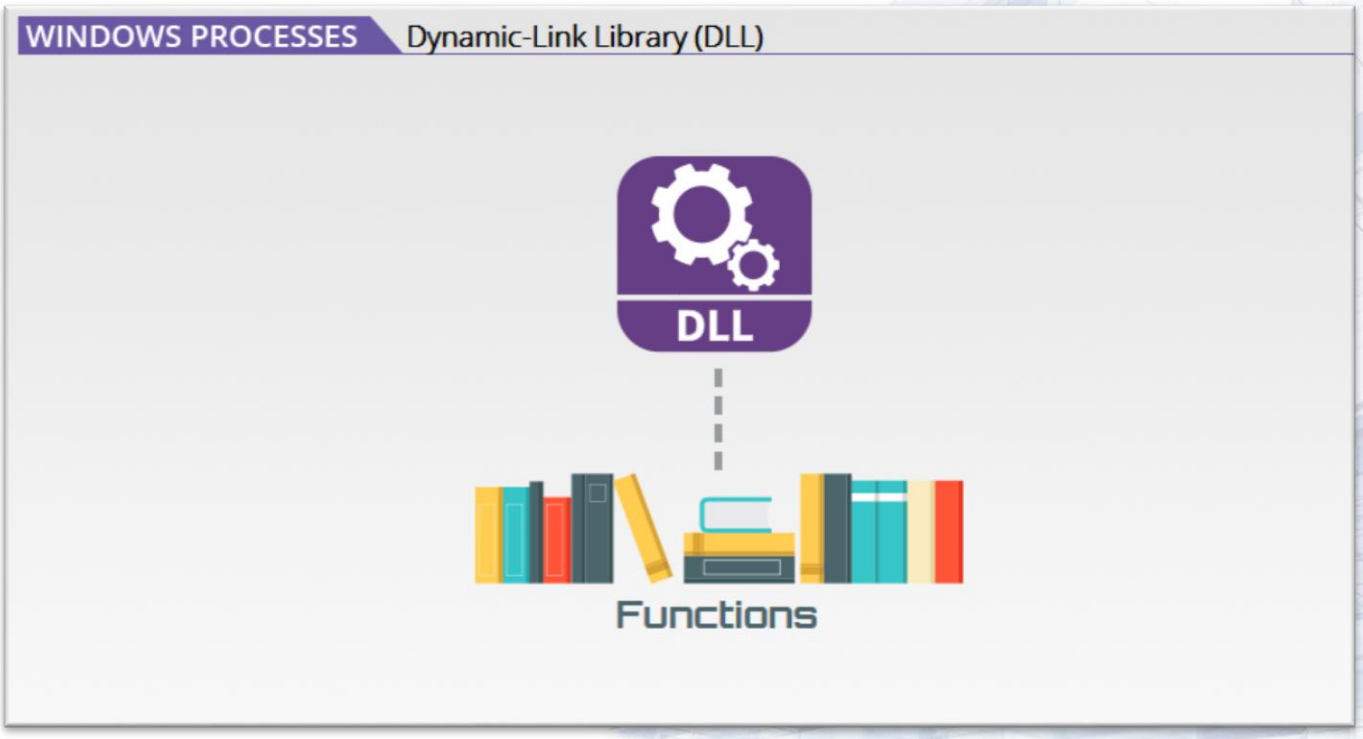
Below is a list of Integrity level features.

Major parts to the mechanism	• Predefined integrity levels and their representation. • Integrity policies that restrict access permissions. • Integrity level assigned to the security access token. • Mandatory label access control entry. • Mandatory labels assigned to objects. • Integrity restrictions within the AccessCheck and kernel-mode SeAccessCheck APIs.
Mandatory access token policies	• No write up - The default policy that is assigned to all access tokens. The policy restricts write access by this subject to any object at a higher integrity level. • New process min - Controls the behavior of assigning the integrity level to child processes. Normally, a child process inherits the integrity level of the parent process when the parent process access token is assigned to the child. With the NEW_PROCESS_MIN policy, the integrity level of the child process will be the minimum integrity level of either the parent access token, or the object integrity level of the executable file for the new process. This policy is set by default in all access tokens.
Mandatory label policies	No write up – the same as the access token policies.

Dynamic Link Library (DLL)

A Dynamic Link Library (DLL) is a library of functions that can be used by an application. DLLs:

- Provide much of the Window operating system's functionality,
- Represents a specific function,
- Are unique, the code contained within a DLL can be used by multiple programs simultaneously, and
- Allow programs to be modularized which makes it easier to apply updates.

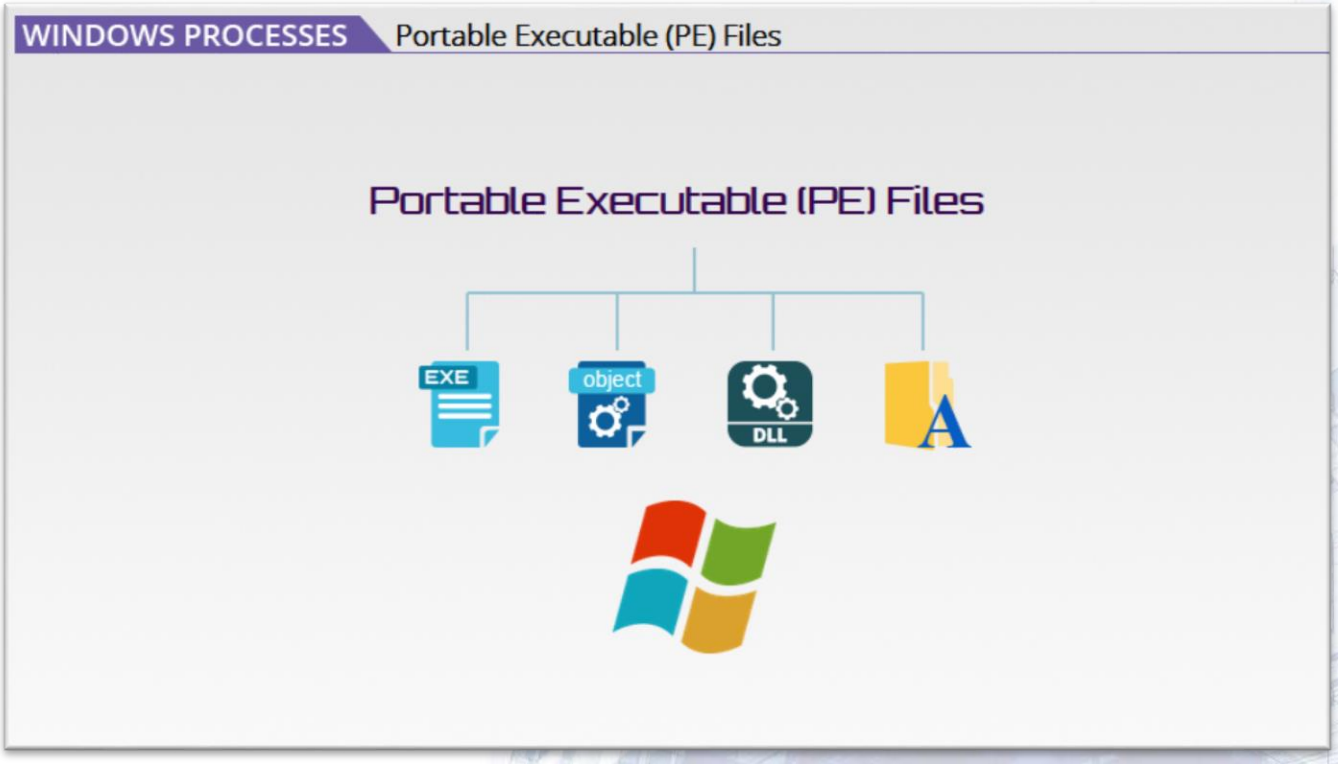


DLLs can have a variety of extensions, the most common of which are:

- .dll- Dynamic Link Library, is a binary file that are shared among running processes. DLLs allow programs to interact with the operating system. Each DLL is written to perform a specific function.
- .ocx – Object Linking and Embedding Control Extension or Active X Controls, is used for interface behaviors that are triggered by users of programs (i.e. scroll bar movements).
- .drv – Device Driver, is associated with legacy device driver files. A device driver is the software that interacts with a specific device or other specific software. Device driver files contain specific information about its associated device/device software interface that enables other programs to communicate with the device.

Portable Executable (PE) Files

Portable Executable (PE) is a file format for executables, object codes, DLLs, and font files that are used in variants of the Windows operating system.



The PE format is a data structure that encapsulates the information necessary for the Windows OS loader to manage the wrapped executable code.

PE File Header

Within the header of a PE file, the following data structures are found: DOS header, PE header, optional header, data directories, and section table. Below is a list of data structures along with information about each one.

DOS Header	The DOS header is the first component in the PE file format and occupies the first 64 bytes of the file. It is there in case the program is run from DOS, to allow DOS to identify the file as a valid executable. When this happens, the DOS stub, stored immediately after the header, is executed. The DOS stub normally prints out the error message, "This program cannot be run in MS-DOS mode." The DOS Header contains a signature and the offset to the PE header.
PE Header	The PE header defines what the rest of the file looks like. It contains essential information used by the loader. You will notice that this main header isn't at the very beginning of the file; instead, it is located a few hundred bytes into the file following the DOS Header. The PE header contains a signature field, processor information and type, the number of sections, the relative offset of the section table, and the characteristics of the file, such as the file type or extension.
Optional Header	The optional header contains the most meaningful information about the executable image, such as the initial stack size, program entry point location, preferred base address, operating system version, section assignment information, etc.
Data Directories	The data directory is the final 128 bytes of the Optional Header. It is basically a data array of 16 IMAGE_DATA_DIRECTORY structures. Each data directory specifies the size and relative virtual memory address relating to an important data structure in the PE file
Section Table	The section table is an array of data structures. Each data structure contains information about one section in the PE file. This includes the section name, virtual size, virtual address size of raw data, pointer to raw data, and characteristics (whether the section contains executable code, initialized or uninitialized data, or whether it can be written to or read from). The sections follow the ordering presented in the section table

Recommended Readings

- Windows Internals Part 1, Sixth edition

Recommended Internet Sites

- **Inter-Process Communications (IPCs)**
  https://web.archive.org/web/20180507165032/https://msdn.microsoft.com/en-us/library/windows/desktop/aa365574(v=vs.85).aspx
- **Data Execution Prevention (DEP)**
  https://web.archive.org/web/20170218063427/https://support.microsoft.com/en-us/help/875352/a-detailed-description-of-the-data-execution-prevention-dep-feature-in-windows-xp-service-pack-2,-windows-xp-tablet-pc-edition-2005,-and-windows-server-2003
- **Address Space Layout Randomization (ASLR)**
  https://web.archive.org/web/20160721115402/http://searchsecurity.techtarget.com/definition/address-space-layout-randomization-ASLR
  (TechTarget)
- **Windows Integrity Mechanism Design** https://web.archive.org/web/20160826033155/https://msdn.microsoft.com/en-us/library/bb625963.aspx
  (Microsoft Developer Network)
- **Portable Executable File Format – A Reverse Engineer View**
  https://web.archive.org/web/20181003154708/http://darkblue.ch/programming/PE_Format.pdf
  (CodeBreakers Magazine, Volume 1 Issue 2 2006)
- **What does Internet Explorer protected mode do?**
  https://web.archive.org/web/20170310131052/https://msdn.microsoft.com/en-us/library/bb250462(v=vs.85).aspx

Please contact the Course Coordinators if you are unable to access any of the Recommended Internet Sites.