Chapter 11 - Key Topics

Description	Element	Page	Summary
Collecting & Analyzing logs from servers	Paragraph	440	Just as you do with endpoints, it is important that you analyze server logs. You can do this by analyzing simple syslog messages or more specific web or file server logs. It does not matter whether the server is a physical device or a virtual machine.
Understanding Windows processes & threads	Paragraph	454	**Process** - a program the system is running.. Each process starts with a single thread known as the primary thread. **Threads** - a process is made up of one or more threads. Threads are the basic units the OS allocates process time to.
Key process & thread concepts	List	456	**Process** - A process is a program that the system is running and is made of one or more threads. **Thread** - A thread is a basic unit that an operating system allocates process time to. **Job** - A job is a group of processes. **Thread Pool** - A thread pool is a group of worker threats that efficiently executes asynchronous callbacks for the application. **Processes** - Processes must have permission to run within Windows. **CreateProcessWithTokenW** - You can use a Windows token to specify the current security context for a process using the CreateProcessWithTokenW function. **Windows Token** - Windows stores data in a token that describes the security context of all processes associated with a particular user role.

Description	Element	Page	Summary
Operating system memory allocation concepts	List	458	**Volatile Memory** - is memory that loses its contents when the computer or hardware storage device loses power. **Non-volatile Memory (NVRAM)** - holds data with or without power. **Static Memory Allocation** - program allocates memory at compile time. **Dynamic Memory Allocation** - program allocates memory at runtime. **Heap** - memory set aside for dynamic allocation. **Stack** - memory set aside as spare space for a thread of execution. **Virtual Address Space** - is a reference to the physical location of an object in memory. A page table translates virtual memory into its corresponding physical addresses. The virtual address space of each process can be smaller or larger than the total physical memory available on the computer.
Windows registration concepts	List	460	**Windows Registry** - is a hierarchical database used to store information necessary to configure the system for one or more users, applications, and hardware devices. **Some** f**unctions of the Registry** - Some functions of the Registry are to load device drivers, run startup programs, set environment variables, and store user settings and operating system parameters. **Hives** - Hives contain values pertaining to the operating system or applications within a key. The five main folders in the Windows Registry are called hives. Three of these hives are reference points inside of another primary hive.

Description	Element	Page	Summary
Windows Management Instrumentation key concepts	List	462	WMI is a scalable system management infrastructure built around a single, consistent, standards-based, extensible, object-oriented interface. WMI is only for Windows systems. WMI comes preinstalled on many Windows systems. For older Windows versions, you may need to download and install it. WMI data must be pulled in with scripting or tools because WMI by itself doesn't show data.
System handle concepts	List	463	A handle is an abstract reference value to a resource. Handles hide the real memory address from the API user while permitting the system to reorganize physical memory in a way that's transparent to the program. A handle not only can identify a value but also associate access rights to that value. A handle leak can occur if a handle is not released after being used.
Understanding system services	List	466	Microsoft Windows services are long-running executable applications that operate in their own Windows session. Services Control Manager enforces the rules and protocols for Windows services. Services are ideal for running things within a user security context, starting applications that should always be run for a specific user, and for long-running functionality that doesn't interfere with other users who are working on the same computer. Windows administrators can manage services using the Services snap-in, Sc.exe, or Windows PowerShell.

Description	Element	Page	Summary
Understanding Windows event logs	List	468	Logs are records of events that happen on a computer. The most common place for Windows logs is the Windows event log. Windows Event Viewer is a common tool used to view Windows event logs. You can generally find the Windows event logs in the C:\Windowsystem3config directory. Event logs typically maintain three event log types: Application, System, and Security log files. Within the log types are generally five event types: Error, Warning, Information, Success Audit, and Failure Audit. A log parser is a versatile tool that provides universal query access to text-based data.
The types of processes that can run in Linux systems	List	469	Child, Init, Orphan, Zombie, Daemon
Understanding Linux processes	List	471	The two methods for starting a process are starting it in the foreground and starting it in the background. The different types of processes in Linux are the child process, init process, orphan process, zombie process, and daemon process. All processes in Linux have a parent, except for the init process, which has a PID of 1. An orphan process results when a parent process is terminated and the child process is permitted to continue on its own. A zombie process is a process that releases its associated memory and resources but remains in the entry table.
Understanding what forks are	List	472	A fork is when a parent creates a child process. The fork command returns a PID. The entire virtual space of the parent is replicated in the child process, including all the memory space.

Aa Description	☰ Element	Page	Summary
Linux file permissions	Paragraph	473	**Read (r)** - Reading, opening, viewing, and copying the file are permitted. **Write (w)** - Writing, changing, deleting, and saving the file are permitted. **Execute (x)** - Executing and invoking the file are permitted. This includes permitting directories to have search access.
Key permissions concepts	List	478	File permissions assign access rights for the owner of the file, members of a group of related users, and everybody else. The **chmod** command modifies file permissions for a file or directory. Read (r) = 4, Write (w) = 2, Execute (x) = 1. A group is the set of permissions for one or more users grouped together. You can modify the group “ownership” of a file using the **chgrp** command. To change the owner of a file, you can use the **chown** command. File permissions in Linux take a top-down approach, meaning denying access for a directory will automatically include all subdirectories and files. Super user privileges provide the highest access level and should be used only for specific reasons, such as performing administrative tasks. All processes, including background daemons, should be limited to only the permissions necessary to successfully accomplish their purpose.
Understanding symlinks	List	480	A symlink is any file that contains a reference to another file or directory. A symlink is just a reference. Removing the symlink file doesn’t impact the file it references. An orphan symlink is a symlink pointing to nothing because the file it references doesn’t exist anymore. A symlink is interpreted at runtime and can exist even if what it points to does not.

Description	Element	Page	Summary
Understanding Linux Daemons	List	481	Daemons are programs that run in the background. From a permissions viewpoint, daemons are typically created by the init process. A daemon's permissions level can vary depending on what is provided to it. Daemons should not always have super user–level access. Daemons are not controlled by the active user; instead, they run unobtrusively in the background, waiting to be activated by a specific event or condition. Not all daemons are started automatically. Children of the init process can be terminated and restarted.
Linux syslog concepts	List	484	The most common form of logging is the general-purpose logging facility called syslog. The default location of logs in Linux is the **/var/log** directory. The facility describes the application or process that submits the log message. A priority is used to indicate the level of importance of the message. **Transaction logs** record all transactions that occur. **Session logs** track changes made on managed hosts during a web-based system manager session. **Alert logs** record errors such as a startup, shutdown, space errors, and so on. **Threat logs** trigger when an action matches one of the security profiles attached to a security rule. **Selectors** monitor for one or more facility and level combinations and, when triggered, perform some action. **Actions** are the result of a selector triggering on a match. **The configuration file /etc/syslog.conf** controls what syslogd does with the log entries it receives. **Newsyslog** attempts to mitigate log management by periodically rotating and compressing log files.

Description	Element	Page	Summary
Apache access log concepts	List	485	**ErrorLog log** - Apache sends diagnostic information and records any errors it encounters to the ErrorLog log. **Access Log** - Apache servers record all incoming requests and all requests to the access log file. The combined log format lists the access, agent, and referrer fields.
The most common types of malicious software	List	486	**Computer virus** - This malicious software infects a host file or system area to perform undesirable actions such as erasing data, stealing information, and corrupting the system's integrity. In numerous cases, these viruses multiply again to form new generations of themselves. **Worm** - This virus replicates itself over the network, infecting numerous vulnerable systems. On most occasions, a worm will execute malicious instructions on a remote system without user interaction. **Mailer & mass-mailer worm** - This type of worm sends itself in an email message. Examples of mass-mailer worms are Loveletter.A@mm and W32/SKA.A@m (a.k.a. the Happy99 worm), which sends a copy of itself every time the user sends a new message. **Logic Bomb** - This type of malicious code is injected into a legitimate application. An attacker can program a logic bomb to delete itself from the disk after it performs the malicious tasks on the system. Examples of these malicious tasks include deleting or corrupting files or databases and executing a specific instruction after certain system conditions are met. **Trojan Horse** - This type of malware executes instructions determined by the nature of the Trojan to delete files, steal data, or compromise the integrity of the underlying operating system. Trojan horses typically use a form of social engineering to fool victims into installing such software on their

Description	Element	Page	Summary
			computers or mobile devices. Trojans can also act as backdoors. **Backdoor** - This piece of malware or configuration change allows attackers to control the victim's system remotely. For example, a backdoor can open a network port on the affected system so that the attacker can connect and control the system. **Exploit** - This malicious program is designed to "exploit," or take advantage of, a single vulnerability or set of vulnerabilities. **Downloader** - This malware downloads and installs other malicious content from the Internet to perform additional exploitation on an affected system. **Spammer** - This system or program sends unsolicited messages via email, instant messaging, newsgroups, or any other kind of computer or mobile device communication. Spammers use the type of malware for which the sole purpose is to send these unsolicited messages, with the primary goal of fooling users into clicking malicious links, replying to emails or messages with sensitive information, or performing different types of scams. The attacker's main objective is to make money. **Key Logger** - This piece of malware captures the user's keystrokes on a compromised computer or mobile device. It collects sensitive information such as passwords, PINs, personally identifiable information (PII), credit card numbers, and more. **Rootkit** - An attacker uses this set of tools to elevate privilege to obtain root-level access to control the affected system completely. **Ransomware** - This type of malware compromises a system and then often demands a ransom from the victim to pay the attacker for the malicious activity to cease or for the malware to be removed from the affected system. The following are examples of

Description	Element	Page	Summary
			ransomware: WannaCry, SamSam, Bad Rabbit, NotPetya
Host-Based Firewalls & Host-Based Intrusion Prevention	Section	488	Host-based firewalls are often referred to as personal firewalls. Personal firewalls and host-based intrusion prevention systems (HIPSs) are software applications that you can install on end-user machines or servers to protect them from external security threats and intrusions. The term personal firewall typically applies to basic software that can control Layer 3 and Layer 4 access to client machines. HIPS provides several features that offer more robust security than a traditional personal firewall, such as host-based intrusion prevention and protection against spyware, viruses, worms, Trojans, and other types of malware.
Application-Level Whitelisting & Blacklisting	Section	490	**Whitelist** - A list of separate things (such as hosts, applications, email addresses, and services) that are authorized to be installed or active on a system in accordance with a predetermined baseline. **Blacklist** - A list of different entities that have been determined to be malicious. **Graylist** - A list of different objects that have not yet been established as not harmful or malicious. Once additional information is obtained, graylist items can be moved onto a whitelist or a blacklist. Decisions can be based on file attributes such as: F**ile path, Filename, File Size**

Description	Element	Page	Summary
System-based Sandboxing	Section	491	**Sandboxing (detonation boxes)** - Sandboxing limits the impact of security vulnerabilities and bugs in code to only run inside the “sandbox.” The goal of sandboxing is to ensure that software bugs and exploits of vulnerabilities cannot affect the rest of the system and cannot install persistent malware in the system. In addition, sandboxing prevents exploits or malware from reading and stealing arbitrary files from the user’s machine.
Untitled			