|  |  |  |  |
| --- | --- | --- | --- |
| **A** |  |  |  |
| **B** |  |  |  |
| **C** |  |  |  |
| **D** |  |  |  |
| **E** |  |  |  |
| **F** |  |  |  |
| **G** |  |  |  |
| **H** |  |  |  |
| **I** |  |  |  |
| **J** |  |  |  |
| **K** |  |  |  |
| **L** |  |  |  |
| **M** |  |  |  |
| **N** |  |  |  |
| **O** |  |  |  |
| **P** |  |  |  |
| **Q** |  |  |  |
| **R** |  |  |  |
| **S** |  |  |  |
| **T** |  |  |  |
| **U** |  |  |  |
| **V** |  |  |  |
| **W** |  |  |  |
| **X** |  |  |  |
| **Y** |  |  |  |
| **Z** |  |  |  |
| **#** |  |  |  |
| **AICPA** | SOC 2 |  |  |
| **ARM platform** | Longer battery life |  |  |
| **File Integrity Checker on a Running Website** | ID changes to the web server config settings |  |  |
| **Firewall - Concurrent TCP Streams** |  |  |  |
| **Hash - Command to get** | sha256sum <filename> |  |  |
| **Lessons Learned Phase - Incident Handling** | Improve the incident response process and identify mistakes |  |  |
| **OWASP** | A1:2017-Injection |  |  |
| **Vulnerability** | a vuln is a weakness in a system that allows an adversary to manifest (code error, etc) |  |  |
| **OneDrive** | Is for the internet-accessible file storage in azure and it is in the same category as Dropbox |  |  |
| **Assets IT** | Servers, desktops, and applications. | 1 | 15 |
| **Know Thy Systems** | we can ONLY secure WHAT we KNOW we OWN (or have), perhaps the most important tenet of security and perhaps, sadly, also the most ignored tenet of security. 1. understanding the architecture of our network (conceptual design, logical design, physical design, understanding where our valuable data is located, understanding communication flow | 1 | 15 |
| **Network Architecture** | Knowing your network architecture is one of the most important principles in security. It is comprised of Conceptual Design, Logical Design and Physical Design. Where my valuable data is located (stored) and communication flow is highly important as well. | 1 | 15 |
| **Non-Traditional IT asset** | Door locks, refrigerators, televisions | 1 | 15 |
| **Up to date network diagram** | When you know what things are connected to your network | 1 | 15 |
| **Conceptual Design components** | A high-level overview of why we need our network. it includes the core components of your network's architecture. These components represent internal and external systems, data flow, and overall system behavior. The conceptual design should consider (a) OS platforms, (b) server services, (c) critical processes, (d) critical core functions, and (e) the critical individuals who leverage these functions, processes, and services to help achieve organizational success. | 1 | 17 |
| **gitClosed box diagraming** | It is describing a system by its functionality and use without requiring any knowledge of how the systems achieves its functionality. | 1 | 18 |
| **Logical Design** | Represents the logical functions in the system, while putting the conceptional design into paper. we are NOT describing, at this point, are the finely detailed aspects of the components themselves. We are not discussing patch levels, hardening configurations, etc. | 1 | 19 |
| **Physical Design** | A physical design has all major components and entities identified within specific physical servers and locations or specific software services, objects, or solutions. This design level represents how the network, and all its components are expected to behave, while still being on paper. The physical design is the last one created before the network design is implemented. | 1 | 20 |
| **Physical Design details** | Include OS versions, patch levels, hardening configurations, risk categorization. | 1 | 20 |
| **Physical Design security** | Physical security can always betray logical security controls. | 1 | 21 |
| **Critical data** | Often lives in critical servers, being accessed by critical applications, an asset. | 1 | 22 |
| **Know Where Your Valuable Data Is Located** | We will often be surprised as to where we might find sensitive data on our internal network, we haven't even taken into full consideration the usage of cloud services and how we might be storing and working with data (intentionally) on systems and networks that are completely external to our organization | 1 | 22 |
| **Knowing your environment - Principals.** | Knowing where your data is located and communication flow | 1 | 22 |
| **Valuable data - Location** | Securing a network requires knowledge of where each piece of valuable data resides as it could end up in the most unusual locations such as desktop of senior leadership, hard drives of systems admin or unintentional like a cloud, Dropbox, OneDrive, etc. | 1 | 22 |
| **Command and Control (C2)** | Leverage compromise, pivot, control | 1 | 24 |
| **Communication flow** | Understanding Who and when accesses data and how much data is accessed. | 1 | 24 |
| **DLP (Data Loss Prevention)** | Is a security control that can attempt to discover sensitive data on our networked systems, monitor for the misuse of that data across the networks and potentially prevent exfiltration of the same | 1 | 24 |
| **Lateral movement** | What an adversary does to gain leverage in our networks. Stage 1 | 1 | 24 |
| **Persistence** | Is what adversaries do to keep ahold on our network and survive a restart. Stage 2 | 1 | 24 |
| **Stage 1 of Compromise** | initial break in, compromise | 1 | 24 |
| **Stage 2 of Compromise - Persistence** | Install additional capability to survive a restart, can be lengthy | 1 | 24 |
| **Understanding Communication Flow** | By understanding how data is supposed to flow across the network, we can build a baseline of expected communication flow. If our data is supposed to be accessed in specific ways, and if we can accurately describe how that data is accessed, we can also determine when that data is accessed in non-normal ways. In other words, knowing 'normal' allows for us to determine 'abnormal' more easily. | 1 | 24 |
| **Critical assets** | Like routers and switches are very important and largely neglected, and even running outdated software due to the impact on security due to downtime. | 1 | 27 |
| **IoT** | Many things are computers, and they are also subject to compromise | 1 | 27 |
| **Key network devices** | Routers and switches | 1 | 27 |
| **Key network devices compromise** | Allows for sniffing and could be catastrophic. Compromise of routers and switches can facilitate communication interception and manipulation. | 1 | 27 |
| **Networks Under Attack** | A compromise of the key network devices (routers and switches) that facilitate the network's communications can allow for an adversary to sniff (capture) the network communication, subsequently inspect the communication in a search for sensitive data (such as authentication credentials), modify the network communication (integrity violation), and perhaps even the bypass of security controls in general | 1 | 27 |
| **Routers** | Are devices that can connect different networks together or facilitate network to networks communication and are the most ignored IT assets. | 1 | 27 |
| **Switches** | Allow all of your devices together to form a network and has software to provide switching functions | 1 | 27 |
| **Threat Enumeration** | Understanding how an adversary operates, the goals they want to achieve, the why of targeting organizations, etc. are parts of understanding that, in general, may come from threat enumeration. Threat enumeration is a part of the overall concept known as "Threat Intelligence". Threat Intelligence strive understanding the TTP: tactics, techniques, procedures. (of adversaries). | 1 | 29 |
| **Threat Intelligence** | Strives to help us understand the TTP: Tactics, Techniques, and Procedures (of adversaries). Who is attacking and why, not all attacks are 'targeted' | 1 | 29 |
| **TTP** | Tactics, Techniques, Procedures (of adversaries). tactics refer to high-level descriptions of the method an adversary might use to formulate an attack; techniques describe, at a more detailed level, a specific instance of the tactic being attempted; and the procedure is the detailed explanation of how the technique works to implement that tactic of a given attack. | 1 | 29 |
| **APT (Advanced persistent threat)** | Defined as highly capable adversaries, acting under the authority of a specific government or military organization. | 1 | 30 |
| **Attackers** | Threat agents | 1 | 30 |
| **Nation State Actors** | Nation state actors would be defined as highly capable adversaries, acting under the authority of a specific government or military organization | 1 | 30 |
| **Opportunistic Attackers** | the attacker is looking for an easy target. Opportunistic attackers might leverage malicious websites, malvertising (malicious advertising), malicious emails and so on, to try and 'catch' an unwitting victim. We must not, however, confuse opportunistic with the concept of less skilled | 1 | 30 |
| **Organized Cybercrime** | From ransomware to traditional extortion (holding data or a system hostage), there is a lot of money to be made by such an adversary. It should be noted that while organized cybercrime fits into a separate category, organized crime will often leverage some of the attacker methodologies of the opportunistic attacker | 1 | 30 |
| **Threat Agents** | Attackers: opportunistic attackers, organized cybercrime, nation state actors & APT (advanced persistent threat). An individual, organization or group that is capable and motivate to carry out an attack of one sort or another | 1 | 30 |
| **Persistence** | Surviving the restart of a computer system | 1 | 31 |
| **ACL (Access Control List)** | A traditional DoS attack might have been launched from one machine, or from one location (from a single IP address). The attack being launched from a single location might allow for defense through the implementation of an ACL (access control list) block at an upstream network provider (like a telecommunications provider) | 1 | 32 |
| **Alanis Morrisette Phenomenon** | Need to secure the critical assets. Router could be one. The more critical the asset, the more unlikely you will be able to secure it. Ironic, don't you think | 1 | 32 |
| **DDoS (Distributed Denial of Service)** | It is similar to DoS but in this case, but the adversary will launch the attack from many different locations, simultaneously | 1 | 32 |
| **DoS (Denial of Service)** | Service of the device is being impacted (denied). At a minimum, it could result in a total impact of device availability. | 1 | 32 |
| **Router attacks** | Most common are Denial of service (DoS) Distributed Denial of Service (DDoS) Packet sniffing, Routing table poisoning. | 1 | 32 |
| **CDN (Content Distribution Network)** | CDN (content distribution network) providers can sometimes provide potential relief from a DDoS attack by leveraging their own massive network infrastructure to assist in the re-routing of a victim organization's traffic, or by providing enough infrastructure to handle both the attack and legitimate traffic at the same time | 1 | 33 |
| **MiTM (Man in the Middle)** | Attacks can be devastating for an organization; communication is still being successfully sent and received, but the victim is unaware the communication is transiting through an untrusted third party. | 1 | 33 |
| **Packet Misrouting** | Integrity based attack. A routers configuration is manipulated such that traffic is no longer routed properly, traffic might be routed to a non-existent network location, traffic might be sent back to a previous router, resulting in a routing loop. | 1 | 33 |
| **Packet Sniffing** | Packet sniffing refers to the capture (and analysis) of the traffic of a network (the network's communication). Confidentiality and integrity can be heavily impacted by the sniffing of network traffic. | 1 | 33 |
| **Routing Table Poisoning** | A routing table poisoning attack results in the modification of a victim router's routing table. Routers use their routing tables to determine the next best routing hop (which best router to next send traffic to), as network traffic moves from router to router. Routers often update their tables by conversing with their routing peers (allowing for routers to dynamically update and re-route traffic as necessary). In a routing table poisoning attack, the adversary convinces a router to update its routing table, resulting in traffic redirection. This is also a machine in the middle attack. | 1 | 33 |
| **CDP (Cisco Discovery Protocol)** | discovery protocols are used to aid devices in discovering other devices that also exist on a network.  CDP is the term used for the discovery protocol utilized in switches developed by Cisco Systems, but the concept exists, regardless of manufacturer (it just might have a different name). | 1 | 34 |
| **CDP Info Disclosure** | The information provided by a network discovery protocol might be crucial to switch operation and perhaps equally crucial to the success of an attacker. Discovery protocols provide a tremendous amount of information. The sniffing of such protocols can yield a substantial amount of valuable information.  One way to try and restrict an attacker's access is to limit the dissemination of the discovery information (limit where on the network the communication is broadcast). This discovery information is important to a switch but isn't important to a desktop. discovery protocol traffic is more than likely being broadcast across the entirety of the network, but it shouldn't be | 1 | 34 |
| **Discovery protocol traffic** | Is being broadcasted across the entire network, and limiting to the management interfaces, we can limit the disclosure of this data to a potential attacker. | 1 | 34 |
| **Hubs** | The predecessor of switches, and it is not as secure as a switch since it will broadcast all information within the network allowing this for sniffing. | 1 | 34 |
| **MAC Downgrading** | This happens when the switch doesn’t have enough memory to record each new MAC address that connects, and it will downgrade itself to a hub rather than shutting down. | 1 | 34 |
| **Mac Flooding** | Is when an attacker floods our MAC ( Media Access Control) address table of a switch with fake MAC addresses. | 1 | 34 |
| **Network interfaces** | Wired or wireless internet | 1 | 34 |
| **Switch Attacks** | Some of the attacks discussed here might be a manipulation of network traffic in a specific manner to convince a switch to do something it shouldn't. CDP Information Disclosure, MAC Flooding, DHCP Manipulation, STP Manipulation, VLAN Hopping | 1 | 34 |
| **Switches** | Leverage the MAC Address to determine which connected computer traffic is destine for. | 1 | 34 |
| **Switches attacks** | Could be identified as a way of manipulating the network traffic in a specific manner to convince a switch to do something it shouldn’t. | 1 | 34 |
| **MAC flooding** | In a MAC (Media Access Control) Flooding attack, the attacker floods the MAC address table of a switch with fake (non-existent) MAC addresses.  The switch will start to enter an error condition, and as a part of said error conditioning, the switch (in order to continue operating as opposed to outright shutting down), might downgrade itself to the concept of a network hub. Hubs (the precursor devices to switches) have a fundamental security issue – a lack of awareness of which MAC address is connected to which physical port of the hub itself, leading to a broadcasting of all traffic to each and every connected computer | 1 | 35 |
| **DHCP (Dynamic Host Configuration Protocol)** | commonly used by computing devices to obtain their network configuration. This network configuration will include items such as the IP address to be used by the computing device, the subnet mask for the network segment, the IP address of the default gateway (default router), and maybe even the IP address of a server that will provide the OS for the computing device | 1 | 36 |
| **DHCP Manipulation** | If the attacker operates quickly enough, the attacker can supply an answer of the DHCP request to the device before the legitimate DHCP server does. In doing so, the attacker has gained control of the network configuration information that will be used by the computing device for its network configuration. The attacker provides network configuration information to deceive the victim computer, perhaps indicating that the attacker's IP address is the IP address for the default router of the network (for example) | 1 | 36 |
| **STP ( Spanning tree protocol)** | STP,  Layer 2 network protocol used to prevent looping within a network topology. | 1 | 36 |
| **STP Manipulation** | STP (Spanning Tree Protocol), One of the responsibilities of STP is to ensure that switch loops do not occur. Just like CDP, STP should be available only to management interfaces (not the network overall). | 1 | 36 |
| **STP Manipulation** | An attacker with sufficient access to a network, might be able to impersonate an STP communication and manipulate to their advantage to get a switch re-configuration to facilitate an MITM attack. | 1 | 36 |
| **VLAN (Virtual Local Area Network)** | Is a concept of network segmentation. | 1 | 36 |
| **VLAN Hopping:** | An attacker will manipulate the characteristics of network packets in such a way that a switch will allow an attacker to hop from one VLAN to another in a way that would otherwise be prohibited | 1 | 36 |
| **Network Segmentation - AWS** | VPC, Security Groups, NACL | 1 | 37 |
| **Network Diagram** | Shows the topology of our network. | 1 | 38 |
| **Physical Topology** | It is defined as how devices are physically connected together, and how communication is sent over the physical connection (electrical) signaling, pulses of light, etc) | 1 | 38 |
| **Star Topology** | type of network topology in which every device in the network is individually connected to a central node, known as the switch or hub. When we connect cables from our desktops, phones, and printers to the switch. | 1 | 38 |
| **Topologies** | Define the inter-relationship of network-based components, and can be defined as physical or logical | 1 | 38 |
| **Logical Topology** | Logical topologies might describe how network communication occurs across the various physical mediums, regardless of medium. A logical topology doesn't describe how signaling works; a logical topology describes how (or what) is communicated across the medium. For most, binary is considered to be the language of computing systems. How those 1s and 0s are re-represented as pulses of light, electrical signaling, etc. would be an example of a physical topology | 1 | 39 |
| **Approaches to Network Design** | Segmentation, Software-Defined Networking (SDN) | 1 | 40 |
| **Least Privilege** | A user needs to have access to only the necessary data to complete their jobs functions and nothing more. | 1 | 40 |
| **Principle of Least Privilege** | What is needed, nothing more | 1 | 40 |
| **Segmentation** | Means separation, assets should not be free to communicated unabated. One of the most important concepts related to network architecture security. The idea of segmentation is that the network is not a single trusted entity and highly sensitive data should be placed in a segmented part of the network restricting access. | 1 | 40 |
| **NotPetya** | Type of malware with devastating consequences. Proof that segmentation could’ve prevented that attack. Costed 10 billion in damages around the world. | 1 | 41 |
| **SND – Software Defined Networking** | Is the concept related to the notion of virtualization. Virtualization is an abstraction. With SDN we are not dependent upon what underlying network hardware is in place since it is becoming a more valuable aspect of cloud computing. | 1 | 41 |
| **VLAN** | A VLAN is an example of a logical segmentation control; a VLAN is a 'virtual' LAN. This logical control allows for the network to be logically separated into virtual network segments. Unless an ACL (access control list) is used, a VLAN provides for easier management – not security | 1 | 42 |
| **VLAN & ACL** | If you add an ACL (Access control list) this will increase security. | 1 | 42 |
| **VLAN Network Security** | Is a Virtual LAN. This logical control allows for the network to be logically separated into virtual segments. Is a very valid form of segmentation. Each VLAN is technically its own network and will have their own IP ranges separating devices allowing it for better management. | 1 | 42 |
| **802.1X** | Limit what and how can connect. a standard that allows for an authentication to occur prior to a user and/or computing device being granted access to a network. The X of 802.1X stands for eXtensible, meaning that the standard can be extended to support different types of authentication methods. | 1 | 43 |
| **802.1X** | Port-Based Network Access Control. Is a standard that allows for authentication to occurs prior to a user and/or computing device being granted access to a network. The “X” represents the eXtensible nature of authentication methods. It could be used on wired and wireless networks. | 1 | 43 |
| **EAPOL (Extensible Authentication Protocol over LAN)** | The passing of the authentication request and the subsequent response to the request is conducted between the supplicant software and the authenticator via the use of EAPOL (Extensible Authentication Protocol over LAN) messages | 1 | 43 |
| **Network Jack** | It is used to connect to the network; however, it could be a mean for attackers to connect to our network. | 1 | 43 |
| **3 tier architecture** | 3 tiers of separation, the tiers represent our network, and the separation is based upon criticality, security, sensitivity. The separation is enforced by firewalls.  Systems in each category serve a similar purpose and have common security requirements. This allows you to group resources within a category by placing them into a common network section. You should locate firewalls:   * Between the Internet and the other networks * Between the semi-public and private network * Between sections of varying trust levels | 1 | 44 |
| **Attack vector** | A pathway or method used by a hacker to illegally access a network or computer in an attempt to exploit system vulnerabilities. | 1 | 44 |
| **Defense in depth** | Defense-in-depth, in this example, could be described by the idea that no single failure of any single part of the environment would result in the total catastrophic failure of the entirety of the environment. Here, the constituent components of the web application are separated, one from another. A compromise of the web server, therefore, would not result in the automatic compromise of the internal data (nor any other part of the segmented environment). | 1 | 44 |
| **Network Design Objectives** | 3 examples | 1 | 44 |
| **Three Tier Architecture** | Presentation Tier, Application Tier, Data Tier: Constituent components of the web app are separated one from another | 1 | 44 |
| **Tiered architecture** | It is the concept separating network segments based on trust while the segments form tiers of separation | 1 | 44 |
| **Web applications – attack vector** | Web applications are hacker’s best attack vector because are difficult to build securely and often provide access to sensitive data found in an internet network segment. Web-apps most time are built insecurely. | 1 | 44 |
| **Public tier (segmentation)** | Resources that reside on the internet are not reliable and cannot be trusted. | 1 | 45 |
| **Segmented tiers** | More defined names are: Public tier, Semi-Public tier (DMZ), Middle tier, and private tier. | 1 | 45 |
| **Semi-Public tier-DMZ (Segmentation)** | The DMZ is a tier that includes Web, email, and DNS servers and they are used for organizational systems that intend to be public facing. The DMZ is a great risk of compromise since they are connected to the internet 24/7 and because of that, better efforts to protect it are made. The DMZ is also considered unreliable but not as much as the public tier. | 1 | 45 |
| **Firewalls** | preventative security devices that will further restrict communication (to only what is necessary to be transmitted) between the various tiers. | 1 | 46 |
| **Middle Tier (Segmentation)** | Also known as application tier, because it is the separation between the DMZ and the private (internal) network tier by using a proxy allowing the communications to be inspected both ways (in and out) before reaching out the webserver. | 1 | 46 |
| **Private Tier**  **(segmentation)** | AKA as the internet network segment and where you find your trusted assets and your internal network and internal systems are located. Also, it represents the last tier in our segmentation example and is of higher trust and lower risks. | 1 | 46 |
| **Proxies** | Proxies separate the two ends of a communication; the web server cannot communicate directly with the back-end database on the internal network segment and vice versa. If the web server desires to send a communication to the back-end database, the request must instead be sent first to the proxy for analysis | 1 | 46 |
| **Three Rules for Tiered Architecture** | * Any system visible from the internet must reside on the DMZ and may NOT contain sensitive data. * Any system with sensitive data must reside on the private (internal) network and must NOT be accessible from the public network (the internet). * The only way a DMZ system can communicate with a private (internal system is via the middle tier. | 1 | 47 |
| **DMZ (segmentation)** | The DMX machine would understand that the traffic received needs to be forward to the internal database server. The DMZ will first send this request through the middle tier (proxy) and the firewall will inspect it. Once its determined safe, it will be forwarded to their destination. | 1 | 48 |
| **Logical topology** | Communication flows from the internet to a machine in the DMZ, through a proxy and then delivered to an internal database server. | 1 | 48 |
| **Physical topology** | Network traffic from the public tier enters our networks through a router. While a router is not commonly used for security but in this instance is used as an initial filtering, once it determines the communication is secure, it will send it to the DMZ for further inspection at the firewall. | 1 | 48 |
| **Tiered Architecture - Final Design** | Picture | 1 | 48 |
| **Network Architecture Understanding: Benefits** | Situational awareness, Prioritization of effort, Reduced cost of effort, Timely detection of attacker presence, Timely detection = timely response = reduction of damage | 1 | 49 |
| **Defensible Network Architecture: Summary** | 1. We can only secure what we know we have. 2. We can only know what criticality we require by understanding our organization's reason(s) for existing. 3. Knowing what makes our organization unique allows us to define criticality. 4. Knowledge truly is power. "Know Thy Systems", "Know Thyself", and "Know Thy Networks" are some of the most important tenets of security. | 1 | 50 |
| **Network Protocol** | It is a set of rules that dictate how computer networks communicate. | 1 | 54 |
| **Communication Protocols (Protocol Stacks) 3 Purposes** | • To standardize the format of a communication  • To specify the order or timing of communication  • To allow all parties to determine the meaning of a communication | 1 | 55 |
| **Protocol Stacks** | To aid in the understanding of how different protocols work together: protocols are often described as part of a protocol stack, where each layer of the stack receives a service from the layer above and provides a service to the layer below. The OSI (Open Systems Interconnection) Protocol Stack is an example of that. See: Three Purposes | 1 | 55 |
| **ISO OSI –**  **Protocol Stack** | The ISO (shorthand name representing the 'International Organization for Standardization') OSI (Open Systems Interconnection) model is perhaps the most prolifically referenced of any protocol stack model. The OSI model divides network communications into seven layers: 7. Application 6. Presentation 5. Session 4. Transport 3. Network 2. Data Link 1. Physical | 1 | 56 |
| **ISO OSI – Layer 6**  **Presentation** | The presentation layer makes sure that the data sent from one side of the connection is received in a format that is useful to the other side. If a sender compresses data prior to transmission, for example, the presentation layer on the receiving end would have to decompress it before it could be used. | 1 | 56 |
| **ISO OSI – Layer 7**  **Application** | The application layer interacts with an application to determine which network services are required. When a program requires access to the network, the application layer manages requests from the program to the other layers below. | 1 | 56 |
| **ISO OSI Protocol stack** | ISO (International Organization for Standardization) OSI (Open Systems Interconnection) model divides network communication into 7 layers. Each layer has a specific job or function. Each layer is only concerned in their own specific function and disregards the layers above and below functions. | 1 | 56 |
| **ISO OSI – Layer 1**  **Physical** | It handles the transmission across the physical media, such as an electrical pulse, modulation of radio waves, connection specifications between interface hardware and network cabling, voltage regulation, etc. | 1 | 57 |
| **ISO OSI – Layer 2**  **Data Link** | It connects the physical part of the network (cables and electrical signals) with the abstract part (packets and data streams). | 1 | 57 |
| **ISO OSI – Layer 3 Network** | IP is the basis for all communication on the Internet. It is so important that it even gets its own layer in the TCP/IP stack. Its primary purpose is to handle the transmission of packets between network endpoints, usually single hosts identified with a unique address. IP includes some features that provide basic measures of fault-tolerance (time to live, checksum), traffic prioritization (type of service), and support for the fragmentation of large packets into multiple smaller packets (ID field, fragment offset) | 1 | 57 |
| **ISO OSI – Layer 4 Transport** | It sequences packets and provides end-to-end data delivery service and ensures the transport of the data is successful. | 1 | 57 |
| **ISO OSI – Layer 5**  **Session** | The session layer handles the establishment and maintenance of connections between systems. It negotiates a connection, sets that connection up, maintains the connection, and makes sure that information exchanged across the connection is in sync on both sides. | 1 | 57 |
| **Layer Independence** | Each layer of the transmitting system is only aware of the requirements that must be fulfilled to communicate with the corresponding layer of the receiving system. Each layer is not aware of the contents of data generated by the other layers of the stack and, in fact, does not need to be aware of the specifics of the other layers to allow communication to succeed. | 1 | 58 |
| **Protocol Stacks - Communication** | Layers do not communicate directly but use headers for each layer. The basic principle of stack-based communication is that data from one layer of the stack on the transmitting computer can be understood only by the corresponding receiving layer of the receiving computer. | 1 | 58 |
| **Stack based communication** | Is when data from one layers of the stack in the transmitting computer can be understood by the corresponding receiving layer of the receiving computer. The layers do not communicate with each other directly, rather they use headers to do so. | 1 | 58 |
| **Decapsulation** | Is the process of moving up a stack with each layer doing work, combined with the reading, and removing of headers. | 1 | 59 |
| **Encapsulation** | The process of moving down a stack, with each layer doing work, leveraging the addition of a header to describe said work. | 1 | 59 |
| **Headers** | Are blocks of data that describe the work that was done at the corresponding layer. They are also referred as “protocol headers”. | 1 | 59 |
| **IP Address Spoofing** | In IP address spoofing, an attacker forges the source IP address found in the IP header (Layer 3 – Network Layer), to make a communication appear to originate from an IP address that it isn't. Considering that many security controls are predicated upon which IP addresses are involved in a communication, the ability to fool a computer into thinking a communication is coming from a specific computer, simply based on the IP address being claimed, is of significant risk. Such an attack is trivial to perform because the Network Layer on either computer does not take into consideration any data from any other layer above or below in the stack. | 1 | 59 |
| **TCP/IP** | The TCP/IP stack uses four layers as opposed to the seven layers used in OSI.   1. Application 2. Transport (TCP) 3. Internet (IP) 4. Network | 1 | 60 |
| **TCP/IP Model** | Has 4 layers and still have to perform the same functions as the OSI model, meaning that each layer needs to perform more work. The network layer comprises both the physical and data link layers of the OSI model, and the application layers encompasses the Application, Presentation, and Session layers. | 1 | 60 |
| **TCP/IP Packet Generation** | Each layer on the sender needs to communicate with the same layer on the receiving computer. However, they cannot directly talk because you must go down the stack, across the network, and back up the stack on the receiving system. This is accomplished by having each layer add a header as you go down the stack on the sender and each layer remove a header on the receiving system as it goes up the stack. | 1 | 62 |
| **IP (Internet Protocol) Basics** | The information provided by IP is used to assist in the determination of the best route (path of transmission) for a communication between two endpoints, on two different networks. IP can define how an IP address should be formed, based on the size of the network; and the facilitation of other desired functionalities such as private IP addresses, broadcasting, and multicasting. Some features of IP that will attempt to provide some resiliency to a communication, such as Time to Live (TTL) and Type of Service. IPv4 and IPv6 | 1 | 64 |
| **IPv4** | Accommodates 4.2 billion unique 32-bit addresses. An evolution of a protocol originally designed and utilized for ARPANET (a network created by the United States' Department of Defense Advanced Research Projects Agency - DARPA). Only considered Availability, not Confidentiality or Integrity | 1 | 65 |
| **IPv6** | IPv6 protocol expands the address size from 32 bits to 128 bits. A 128-bit address allows for approximately 340 undecillion addresses (340,282,366,920,938,463,463,374,607,431,768,211,456). Added IPSec to address Confidentiality and Integrity in addition to Availability. Added QoS features like VoIP and interactive media. | 1 | 65 |
| **IPv4 Header** | Layer 3, contains fields of data that can help describe aspects of data communication | 1 | 68 |
| **IPv4 Header Key field** | All fields of the ip header have importance. The most important are Version, 4 bits; Protocol 8 bits, time-to-live (TTL) 8 bits, Fragmentation 16 bits, and source address and destination address 32 bits. | 1 | 69 |
| **IPv4 Header Key field – version 4 bits** | This field contain a short integer value, corresponding to the internet protocol version used to create the header. A value of 4 decimal indicates IPv4 and value of 6 indicates IPv6. | 1 | 69 |
| **IPV4 8 bits Protocol.** | This field will contain an integer value, denoting the exact type of IP message encapsulated in the packet. | 1 | 70 |
| **IPv4 Header Key field – fragmentation 16 bits** | It is also known but hos 16 bits (3 bits for flags and 13 bits for fragment offset. Sometimes a router encounters a packet that is too large for it to transmit at once. Instead of singling the sender it is too bit, the router fragments. It only happens when the network’s size requires it due to a maximum | 1 | 70 |
| **IPv4 Header Key field – Options** | Options are optional and they happen to be set the length of the IPv4 hear will extend beyond the size of 20 bites, | 1 | 70 |
| **IPv4 Header Key field – protocol 8 bits** | This field will contain and integer value denoting the exact type of IP message encapsulated in the packet. Although the meaning of the possible values are standardized, the values themselves are arbitrary. | 1 | 70 |
| **IPv4 Header Key field – source address and destination address** | Routers and networks need to know who they are speaking to. The source address contains the IP address of the packet’s sender. The destination address list the IP address of the recipient. | 1 | 70 |
| **IPv4 Header Key field – Time to live (TTL) 8 bits** | A packets TTL specifies how many router “hopes” a packet can transit before the packet will expire. If a destination router can be reached in less hops than allowed by the TTL, it will not expire prior to delivery | 1 | 70 |
| **IPV4 Options** | While there are valid uses for IPv4 options, especially for network troubleshooting, they are rarely used by legitimate network traffic today. The presence of IPv4 options in an IPv4 header might indicate a malicious aspect to the communication. | 1 | 70 |
| **Source Address and Destination Address, 32 bits (each) (IPv4)** | The source address contains the IP address of the packet's sender. The destination address lists the IP address of the intended recipient. | 1 | 70 |
| **Time to Live (TTL), 8 bits (IPv4)** | A packet's TTL specifies how many router 'hops' a packet can transit before the packet will expire. | 1 | 70 |
| **ECN (Explicit Cogestion Notification)** | A concept that works with TCP, to help provide overall quality of service to network-based communications. | 1 | 71 |
| **IPv6 header Key field – Traffic class and flow label** | They are 2 fields, and their case can be related. Traffic class is used to provide QoS (quality of service) for network communication Traffic class also indicates the packets priority. Flow label indicates that this packet belongs to a specific sequence of packets between a source and a destination, requiring special handling of the IPv6 routers | 1 | 71 |
| **IPv6 header Key field – Version 4 bits** | Just as the IPv4, this field indicates the IP version in use. | 1 | 71 |
| **IPV6 Version 4** | the version of IP that is in use. 6. | 1 | 71 |
| **Traffic Class (8 bits) and Flow Label (20 bits) (IPv6)** | Similar to Type of Service from v4. Traffic Class provides QoS and considers aspects of ECN | 1 | 71 |
| **IPv6 header Key field -** | It looks a different than IPv4 but has the same purpose as IPv4. It is known for version 4 bits, Traffic class, flow label, payload length, next header, hop limit, source address and destination address. | 1 | 72 |
| **IPv6 header Key field – hop limit** | Is the same concept as time to live field and works in the same manner | 1 | 72 |
| **IPv6 header Key field – Next header** | Contains an integer value denoting the type of IP message encapsulated in the packet. | 1 | 72 |
| **IPv6 header Key field – Payload length** | Specifies the length of a load | 1 | 72 |
| **IPv6 header Key field – source address and destination address** | These fields represent the source and destination addresses of the computer involved in a communication | 1 | 72 |
| **IPV6 Hop Limit, 8 bits** | same as TTL from v4 | 1 | 72 |
| **IPV6 Next Header, 8 bits** | Next Header field will contain an integer value, denoting the exact type of IP message encapsulated in the packet. The values that are assigned to IPv4 embedded protocols (such as TCP, UDP, and ICMP) are forward-compatible with the IPv6 Next Header field. | 1 | 72 |
| **IPV6 Payload Length, 16 bits** | Similar to the Total Length field of an IPv4 header, the Payload Length field specifies the length of the packet (in bytes). | 1 | 72 |
| **IPv6 Features** | * Extended address space: * Route aggregation, improved delegation/management, hierarchy * Auto configuration support * Support for IPv6 over IPv4 (tunneling) * Support for IPv4 over IPv6 (translation) * Authentication of endpoints and cryptographic protection of communication (utilizing concepts of IPsec) * Flexible embedded protocol support | 1 | 73 |
| **Translation** | IPv4 over IPv6 allows for IPv4 addresses to be 'translated' from their current form into IPv6 compatible addresses. | 1 | 73 |
| **Tunneling** | IPv6 over IPv4 allows for IPv6 communication between two IPv6 endpoints without the routing systems in the middle requiring any knowledge of IPv6. A virtual 'tunnel' allows for communication between the two IPv6 endpoints transiting through IPv4 systems along the path (hence the term tunneling). | 1 | 73 |
| **ICMP (internet Control Message Protocol)** | ICMP has 2 purposes, Report errors or troubleshooting such as destination host unreachable and TTL exceeded in transit, to provide network information like pings. | 1 | 76 |
| **ICMP (Internet Message Control Protocol)** | a protocol designed to report error conditions, assist with the performance of network troubleshooting, and to provide other network-related information, such as whether a host is available to receive a communication. It is not meant as a protocol to be used for the transmission of data, per se. Network Layer of the OSI model (the same layer as where IP is described as existing) | 1 | 76 |
| **ICMP versions** | It has 2 versions, ICMP for IPv4 based networks and ICMPv6 for IPv6 based networks | 1 | 76 |
| **Ping** | determine if a host is present and available | 1 | 76 |
| **traceroute** | determine the route a packet will take | 1 | 76 |
| **ICMP Code, 8 bits** | The code field contains an integer that provides additional clarification to the overall 'type' (the condition related to the type as described above). The code provides a better understanding of 'what and why' | 1 | 77 |
| **ICMP Header** | type, code checksum (PRG) | 1 | 77 |
| **ICMP Header – Code** | The code provides a clarification of the header “type” | 1 | 77 |
| **ICMP Header – Type** | It is like a category of conditions that are often related. | 1 | 77 |
| **ICMP Headers** | It is the easiest to understand as it only contains 3 fields such as version, code and the last one is a checksum. | 1 | 77 |
| **ICMP Type, 8 bits** | The type of field contains an integer that identifies which 'type' of ICMP packet is being sent. A type is like a category of conditions that are often related. Type 3, for example, is a category with several conditions related to 'destination unreachable'. | 1 | 77 |
| **ICMP Types and codes** | Types and codes are related to each other. See book for more examples | 1 | 79 |
| **TCP**  **Transmission control protocol** | Works by establishing a session prior to data exchange, session leads to connection- oriented communication, provides a “guarantee of delivery” by providing “receipts of successful delivery” and other protocols leverage from TCP’s benefits (HTTP, HTTPS, SSH, etc). It operates on the transport layer (layer 4 of the OSI model) | 1 | 82 |
| **TCP Benefits** | Flow control, verification of data received, and retransmission of any missed communication are all examples of TCP benefits this is also why is the most utilized of the transport layer protocols. | 1 | 82 |
| **Virtual connection** | Session between hosts that desire to communicate | 1 | 82 |
| **TCP ports** | FTP (file transfer protocol) data (20), FTP(21), SSH(22), telnet (23), SMTP(25), DNS(53), HTTP(80), HTTPs(443). | 1 | 83 |
| **TCP Uses/Benefits** | The first benefit is flow control. The second benefit is found in the comfort level of sending larger amounts of data (overall) as compared to other transport layer protocols (such as UDP). The third benefit is the ease at which application developers can take advantage of the functionality of TCP. | 1 | 83 |
| **Sequence Number** | An integer that will assist in the tracking of data being communicated in either direction. Randomly chosen by each host and then subsequently exchanged on the SYN segments. | 1 | 85 |
| **TCP Connection** | SYN, SYN/ACK, ACK | 1 | 85 |
| **TCP Connection (establishing)** | Is stablished by a 3-way handshake in which ISN(initial sequence numbers) are exchanged. | 1 | 85 |
| **TCP handshake** | It is also known as a 3-way handshake. The client initiates a 3-way handshake by sending a “SYN” requesting a TCP connection to the server. If the server accepts the connection in response of the “SYN” will send a “SYN/ACK” and in turn, the client will send an “ACK” response completing the handshake. In other words, the computer systems are synchronizing their sequence numbers that assist in tracking the data communicated in each direction. By the end of the handshake both computers will have each other’s sequence numbers and after that each sequence number will increase by 1 for each byte of data that is transmitted by each party. | 1 | 85 |
| **TCP Response packet** | The response consists of both an acknowledgment of the client's initial connection request (the ACK flag is set) and a connection request of its own (the SYN flag is set), together in a single packet (a SYN-ACK). | 1 | 85 |
| **ISN (initial sequence number)** | Sent with SYN, ACK adds 1 for each byte of data and sends back | 1 | 86 |
| **TCP ISN (initial sequence number)** | Sent together with “SYN” while the 3-way handshake gets stablished. It is a way for a computer to count how much data is sent from computer to computer. | 1 | 86 |
| **Acknowledgement Number, 32 bits** | The acknowledgment number indicates the value of the next expected sequence number from a party of the communication. | 1 | 87 |
| **TCP Header Fields** |  | 1 | 87 |
| **TCP Headers – Acknowledgement number** | It indicates the value of the next expected sequence number from a party of the communication. 32 bits. | 1 | 87 |
| **TCP Headers – Data offset** | It refers to the length of the header and indicates the number of 32-bit words (4 bytes) that are contained in the TCP header. | 1 | 87 |
| **TCP Headers – sequence number** | Represents either the ISN or the accumulated sequence number which is a totality of data related to the current packet. It is 32 bits. | 1 | 87 |
| **TCP Headers – Source port and destination port** | It is a port number that indicates on a host system where an application is sending and receiving its network communications. It is 16 bits each. | 1 | 87 |
| **TCP Flags (NS, CWR, ECE, URG, ACK, PSH, RST, SYN, FIN), 9 bits** | TCP flags represent a characteristic of the session's status or of the importance of a specific packet. | 1 | 88 |
| **TCP Headers – options** | If TCP options are used, this field is occupied by a minimum of 4 bytes and consumes multiples of 4 bytes. Unneeded space is padded with zeros. Common TCP options include Maximum Segment Size (MSS), Windows scale, Selective ACK (SACK), Timestamp, and No operation (NOOP). | 1 | 88 |
| **TCP Headers – TCP flags** | Represents a characteristic of the session’s status or of the importance of a specific packet such as NS, CWR, ECE, URG, ACK, PSH, SYN, FIN. | 1 | 88 |
| **TCP Headers – Windows size 16 bits** | It represents the value described by the window size that indicates the number of data octets that the sending of this segments is willing to accept (see flow control). | 1 | 88 |
| **TCP Abrupt closure** | Either host can abrupt terminate a session by sending an RST packet and an ACK packet will not be sent due to the fact that whoever sent the RST packet has become unavailable. | 1 | 89 |
| **TCP closing session** | There are 2 methods of closure, Graceful and abrupt. | 1 | 89 |
| **TCP Graceful closure** | Hosts can initiate a graceful closure. Host A sends a “FIN” packet, upon receipt host B sends an “ACK” acknowledging the request and sends a “FIN” terminating then Host A will reply by with “ACK” packet finalizing the session. | 1 | 89 |
| **TCP Teardown** | The two main methods for closure are a graceful closure and an abrupt closure. FIN packets are used in the process of graceful closure; RST packets are used in the process of an abrupt closure. | 1 | 89 |
| **UDP (User Datagram Protocol)** | Layer 4 Transport, UDP is not built on the foundation of session establishment. Faster, very little error checking | 1 | 91 |
| **UDP transmission** | The sender isn’t required to check if the receiving host is available prior to transmission. After the sender transmits the UDP packet the sender forgets the packet was even sent and the receiver doesn’t even acknowledge receipt and there is also no guarantee that the packets arrive in the same order they were sent. | 1 | 91 |
| **UDP Uses** | VoIP, DNS requests and responses. Example UDP ports: DNS (53), BOOTP/DHCP (67 and 68), TFTP (69), NTP (123), SNMP (161 and 162), NFS (2049) | 1 | 93 |
| **UPD ports** | DNS (53), BOOT/DHCP (67, 68), TFTP (69), NTP (123), SNMP (161, 162) NFS (2049). | 1 | 93 |
| **UPD traditional uses** | Applications where small amounts of occasional packet loss may be acceptable or where retransmission doesn’t make sense such as DNS request and responses. Applications where the overhead of TCP could impact performance like VoIP or other multimedia exchange. | 1 | 93 |
| **Network File System (NFS)** | File sharing support for UNIX-based networks. | 1 | 94 |
| **Network Time Protocol (NTP)** | Time synchronization | 1 | 94 |
| **Simple Network Management Protocol (SNMP)** | Monitoring and troubleshooting for server-based devices. | 1 | 94 |
| **Trivial File Transfer Protocol (TFTP)** | Inter-device file transfer without the requirement for authentication. | 1 | 94 |
| **UDP based protocols - Network Time Protocol (NTP)** | Time synchronization | 1 | 94 |
| **UDP based protocols – BOOT/DHCP** | Autoconfiguration of network interfaces and the ability to atomically start an OS from the networks during startup. | 1 | 94 |
| **UDP based protocols – Network File System (NFS)** | File sharing support for UNIX-based networks. | 1 | 94 |
| **UDP based protocols – Simple network management protocol (SNMP)** | Monitoring and troubleshooting for server-based devices. | 1 | 94 |
| **UDP based protocols – Trivial File transfer protocol (TFTP)** | Inter-device file transfer without the requirement for authentication | 1 | 94 |
| **UDP Header** | There are only four fields: Source Port, Destination Port, Length, and Checksum. Further, each field is exactly 2 bytes long, resulting in a mere 8 bytes of UDP overhead per packet. | 1 | 95 |
| **Sniffing** | A sniffer provides the capability to observe, record and analyze traffic as it transits a network. They can also observe network communication on a wired and wireless network, used legitimately by administrations and security professionals, and can asl o be used maliciously to obtain credentials, sensitive data, and audio from VoIP calls. | 1 | 98 |
| **Sniffers** | Are tools that come in different forms based on required functionality, work- flow ability, or application best practice. Here is a list. Tcpdump: initial triage, Wireshark: Detailed and protocol analysis, Snort: intrusion detection system, Kismet: Wireless network sniffer, BetterCAP: Sniffing on switches, via the use of MiTM attacks. | 1 | 99 |
| **Snort** | Is an instance of IDS (Intrusion detection system) it performs a detailed inspection of network traffic to assist in the earlier detection of malicious activity. | 1 | 99 |
| **Wireshark** | Is not a sniffer per say but a network protocol analyzer. It can understand hundreds of different protocols and media types. Its primary purpose is to analyze what’s being communicated. can be used to perform detailed analysis and automatic packet decoding of network packet data in a GUI environment. | 1 | 99 |
| **BetterCAP** | It obtains sniffing capabilities by utilizing stacks, including MiTM. Once MiTM is in place, BetterCAP will sniff the traffic it can observe. | 1 | 100 |
| **Kismet** | Is a sniffer designed for traditional wireless (WLAN) networks. It processes the radio signals produced by WLANs to provide information on what security controls are in place. The data can be combines with geographic mapping tools. | 1 | 100 |
| **Tcpdump Example** | See book | 1 | 101 |
| **Tcpdump** | Is primarily runs on Linux - UNIX OS and it is considered to be “lean and mean” it doesn’t provide a detail interpretation of the data it sees and leaves it to the analyst to interpret. It is a sniffer that displays traffic from your ethernet adapter | 1 | 103 |
| **Abstractions** | Abstractions are a way to separate function from form. Abstractions allow us to focus on the 'what', and not the 'how' a computing function is achieved. Abstractions are not a new concept; they have served our evolving technological world well. Abstractions can allow for the creation of 'virtual hardware' from physical hardware, leading to the first example we discuss – virtual machines (VM) | 1 | 108 |
| **Virtualization** | Abstraction. Virtual machines utilize virtual hardware, which in turn maps back to physical hardware. Software known as a hypervisor interacts with the physical hardware and creates a logical separation of physical to virtual. The hypervisor therefore becomes a layer of abstraction between the physical hardware and the virtual hardware. | 1 | 108 |
| **Hypervisor** | It is like a broker; it is aware of all the hardware you have and allocates that hardware virtually to virtual machines. It also presents simulated hardware to the virtual machine OS. | 1 | 110 |
| **Virtual Machines** | A separation from physical to virtual allows for 'hardware independence' This separation can help to reduce the scope of damage from the compromise of a VM:  Damage can be isolated away from the physical system, away from any other VMs running on the same physical hardware, or both This separation further enables other key functionality such as malware analysis and improved disaster recovery. The physical computer requires an OS. This OS is referred to as the “Host OS”. A virtual machine requires its own OS. This OS is referred as “Guest OS”. The software makes this all possible is known as the hypervisor. | 1 | 110 |
| **Virtualization Security** | It provides incredible security powers; however, it also provides security issues. Benefits are Isolation, disaster Recovery and VDI, Malware and forensic analyses and the risks are the hypervisor, isolation violation, | 1 | 110 |
| **Virtualization**  **Isolation** | Is another word for segmentation. A separation from physical to virtual allows for “hardware independence” This separation can help to reduce the scope of damage from the compromise of a VM since the damage can be isolated away from the physical hardware of both. This separation further enables other key functionality such as a malware analysis and Improved disaster recovery. A compromise of the virtualization doesn’t mean a compromise of the host. The more we abstract and isolate, the more difficult it would be for an adversary to attack us. | 1 | 111 |
| **Virtualization Security** | When focused on capability alone, we tend to forget about potential security weakness  • Strangely, we tend to misconstrue that a security benefit must naturally be secure - that no security vulnerabilities would also be present | 1 | 113 |
| **Isolation** | A compromise of an application is not necessarily a compromise of ALL applications, nor the entirety of the VM | 1 | 114 |
| **Virtualization Security Benefits** | Isolation (114), Disaster Recovery and VDI (115), Malware and Forensic Analysis (117) | 1 | 114 |
| **Disaster Recovery** | The power of a shorter recovery time as applied to an organization is powerful indeed. If any of the organization's VMs ever experience an issue, a simple re-copying of the VM's files might be all that is necessary to recover it. | 1 | 115 |
| **VDI (Virtual Desktop Infrastructure)** | Organizations that have made the move to VDI (Virtual Desktop Infrastructure) are able to take advantage of this concept as applied to all of their client desktops. Once a desktop VM is known to be compromised, it can be destroyed (by deleting its files), with a new replacement desktop VM spun up quickly. | 1 | 115 |
| **Forensic Analysis** | The benefits of virtualization can also apply to forensic analysis. | 1 | 117 |
| **Malware Analysis** | Virtualization isn't a panacea for safety during malware analysis. Improperly configured networking and drivers that allow for copying and pasting in and out of the VM (to and from the host OS), are examples of how malware might be able to escape a virtual machine. | 1 | 117 |
| **Hypervisor** | The hypervisor is software, and software can easily have mistaken (vulnerabilities). Depending on the severity of a vulnerability, a vulnerability could lead to a DoS or even remote code execution. DoS against a hypervisor could have a significant impact on the operation of an organization. | 1 | 118 |
| **Virtualization Risks** | The Hypervisor (118), Isolation Violation (119) | 1 | 118 |
| **Isolation Violation** | One example of an isolation violation is known as a VM escape. A VM escape is best described as software code that is run from inside a guest VM but is executed on the host computer instead. If a hypervisor has a VM escape vulnerability, an adversary might leverage a compromised guest VM and execute, on the host computer, any code of their choosing (more than likely at a high privilege level). | 1 | 119 |
| **Virtual Environment Defense** | Guest OS minimal footprint, disabling of unnecessary services, patching, and baselines. Hypervisor minimal footprint, patches installed as quickly as possible. Maintaining appropriate visibility of VM-to-VM communication | 1 | 120 |
| **Cloud** | When people ask what 'cloud' is, they probably are asking what 'public cloud' is. Abstractions create cloud concepts: applied to our own hardware - private cloud; applied to 3rd party hardware - public cloud. | 1 | 122 |
| **Cloud** | Virtualization is the foundation of cloud. It is used when referring to technological abstraction. | 1 | 122 |
| **Cloud Deployment Types** | Public: Operated by a third party provided. Private: Built of operated by a single company or organization. Hybrid: combination of public and private tools. | 1 | 123 |
| **Cloud Trade off** | Include control and visibility | 1 | 123 |
| **Cloud Public cloud (why?)** | Building and maintain IT infrastructure is expensive, it is computing that can be rented, and providers build the computing infrastructure and spread the cost among its customers (subscribers(. Allowing entities to do more at a lower price point than on their own, and services are often defined (or offered) in terms of “subscription models” or “subscription services” | 1 | 124 |
| **Public Cloud** | Having our own IT infrastructure is a very expensive investment. | 1 | 124 |
| **Cloud Computing benefits** | Cost: trade fixed expense for variable expense.  Speed: resources are a click away.  Scale: Global elasticity when it is needed, Productivity: on-demand resources.  Security: Major cloud providers have a huge business incentive to invest in security.  Security: Major cloud providers have a huge business incentive to invest in security. | 1 | 125 |
| **Cloud Computing Essential Characteristics** | On-demand self-service, Broad network access, Resource pooling, Rapid e Cloud Deployment Types elasticity, Measured service, Multitenancy | 1 | 125 |
| **Cloud Type services** | IaaS, PaaS, SaaS. | 1 | 127 |
| **IaaS (Infrastructure as a Service)** | IaaS: With the Infrastructure-as-a-Service (IaaS) delivery model, the cloud service provider provides access to a console that allows the customer to provision servers on a virtual network according to the customer's needs. In general resources are provisioned as needed from a shared pool and billed based on consumption. | 1 | 127 |
| **PaaS (Platform-as-a-Service)** | PaaS: The Platform-as-a-Service (PaaS) delivery model allows customers to access a "platform" that enables the customer to run custom code or applications. Customers do not need to manage the underlying infrastructure and only have to secure the code running on the platform and ensure the environmental settings on the platform are appropriate. | 1 | 127 |
| **SaaS (Software-as-a-Service)** | SaaS: In the Software-as-a-Service (SaaS) delivery model, the cloud service provider provides access to a web application that is used by several customer organizations. | 1 | 127 |
| **FaaS (Functions as a Service)** | Refers to new, non-traditional architecture as it does not use dedicated containers, event-triggered computing requests, ephemeral environment, servers fully managed by a third party (AWS) and are referred to as Function as a Service (FaaS) | 1 | 128 |
| **Serverless Computing** | * Does not use dedicated containers * Event-triggered computing requests * Ephemeral environment • Servers fully managed by third party (e.g., AWS) * Referred to as Functions as a Service (FaaS) | 1 | 128 |
| **CI/CD (Continuous Integration / Continuous Delivery)** | Software changes to running applications are delivered using automation. This includes the delivery of IaC! | 1 | 130 |
| **IaC (Infrastructure as Code)** | Allows us to create infrastructure from templates.   * DevOps teams rely heavily on IaC * Security teams can introduce controls/checks prior to infrastructure being created * Many different tools available, from a CSP’s native offering to third-party tools: * Terraform, AWS Cloud Formation, Ansible, Chef, etc. | 1 | 130 |
| **Cloud Market Data** | Market share in decreasing order: AWS, Azure, Google Cloud | 1 | 131 |
| **IaaS + PaaS Market Share** | Decreasing order of share: Amazon, Microsoft, Google | 1 | 132 |
| **Could providers:**  **Amazon, Microsoft, Google** | The cloud market space is rapidly growing and the three largest CSPs, Amazon’s AWS has strong offerings in the IaaS and PaaS space, but is now expanding more into serverless computing, artificial intelligence, and IoT. Microsoft offerings are split between Azure (IaaS, PaaS, serverless) and SaaS (Microsoft 365, Microsoft Dynamics). GCP (Google Cloud Platform) is less mature, IaaS and PaaS are still developing, Google workspace is the largest SaaS offering. | 1 | 133 |
| **Cloud security:**  **Point of demarcation** | Is where the cloud provider responsibility ends, and the users begins. Remember, we chose to put our data in the cloud therefore we are responsible.9 | 1 | 137 |
| **Shared Responsibility - Cloud** | Shared responsibility does clearly imply that both the cloud provider and the subscriber have responsibility for security concepts. Second, shared responsibility, while clearly implying that both parties bear some responsibility for security, doesn't articulate how difficult such a shared responsibility naturally is. | 1 | 137 |
| **SLAs (Service Level Agreements)** | Method by which subscribers might be able to contractually negotiate with a provider on certain levels of traditional performance and availability, and shared responsibility. | 1 | 138 |
| **AWS Shared Responsibility Model** | AWS says that it is responsible for security OF the cloud including typical infrastructure services such as compute, storage, database, and networking. On the flip side, AWS says that the customer is responsible for security IN the cloud. This includes ultimate responsibility for data security, appropriate access controls, application, and platform security in addition to configuration settings of the operating system, network, and firewall. | 1 | 139 |
| **Microsoft Azure Shared Responsibility Model** | For on-premises systems, the customer is, of course, responsible for security at all levels of the stack. But the level of customer security responsibility varies for each layer depending on the type of cloud service that is utilized. The customer always has ultimate responsibility for data protection in the top row. | 1 | 140 |
| **Security and Privacy Frameworks** | Spoiler: for the most part, it boils down to the attestations that the cloud service provider makes in its documentation and third-party audit reports. | 1 | 141 |
| **CAIQ (Consensus Assessments Initiative Questionnaire)** | A document intended to be prepared by providers, in order to detail the provider's own security and compliance controls. | 1 | 142 |
| **CCM (Cloud Controls Matrix)** | Maps various cloud security controls to a large list of security and compliance standards. | 1 | 142 |
| **Cloud - CSA Cloud Security Guidance Project** | The CSA Cloud Controls Matrix (CCM) provides fundamental security principles covered by the CSA Security Guidance. The CSA Security Guidance defines critical areas of focus for organizations managing and mitigating cloud risks. | 1 | 142 |
| **Cloud Security Alliance (CSA)** | Organization dedicated to raising security awareness and best practices for secure cloud computer. Publishes the cloud security governance, Risk Management, and compliance (GRC) stack for the community such as Cloud Audit, Cloud Controls Matrix (CCM) and Consensus Assessments Initiative Questionary (CAIQ). | 1 | 142 |
| **Cloud Security Governance, Risk Management and Compliance (GRC)** | Published by the CSA, a toolkit for organizations to assess the security of both public and private clouds against industry standard and compliance requirements. The GRC stack includes the following initiatives: CAIQ, CCM | 1 | 142 |
| **CSA (Cloud Security Alliance)** | The CSA is an organization focused on a mission to help raise awareness of security issues related to cloud computing, and to the development of best practices related to the same. Publishes the Cloud Security Governance, Risk Management and Compliance (GRC) stack for the community. | 1 | 142 |
| **CSA Cloud Security Guidance Project** | The principals covered by the CCM are thoroughly documented by the CSA Security Guidance project. | 1 | 143 |
| **Cloud – CSA Guidance Domain Matrix** | There are 14 CSA guidance domains. See book | 1 | 144 |
| **CSA Guidance Domain Matrix** | 14 CSA Guidance domains in v4.0, released in 2017 | 1 | 144 |
| **Cloud infrastructure leaders** | There are 3, Amazon, Microsoft and google. They all have similar features but for the sake of training, we are just going to focus on AWS | 1 | 145 |
| **Cloud Providers: Biggest** | Amazon, Microsoft, Google | 1 | 145 |
| **AWS Regions and Availability Zones** | A Region is a physical location in the world that is composed of multiple Availability Zones. Each Availability Zone consists of one or more data centers with redundant power, networking, and connectivity. | 1 | 146 |
| **Cloud AWA Regions and Availability Zones** | A region is a physical location in the world that is composed of multiple availability zones and each availability zone consisted of one or more data center with redundant power, networking, and connectivity. | 1 | 146 |
| **VPC -Virtual Private Cloud** | Provides a dedicated network inside an AWS account and logically isolate networks resources from other VPCs. Control networks flow using cloud-based networking (route tables, internet gateways, NAT, and network interfaces. IP address blocks are defined by Classless Inter-Domain Routing (CIDR) . VPCs m ay contain resources in both public and private subnets. Scoped to a single AWS Region. | 1 | 147 |
| **VPC (Virtual Private Cloud)** | Virtual Private Cloud (VPC) provides a dedicated network inside an AWS account | 1 | 147 |
| **IGW (Internet Gateway)** | Internet gateways connect resources or instances within a VPC to the internet and it is responsible for handling traffic from router tables. | 1 | 148 |
| **Cloud – Subnets** | Subnets divide cloud VPCs into smaller virtual network segments. | 1 | 149 |
| **Subnets** | Cloud subnets are similar to traditional network subnets, allowing a VPC to be divided into smaller virtual network segments. | 1 | 149 |
| **Network Address Translation (NAT) Gateway** | Enable instances in private subnets to access the Internet | 1 | 150 |
| **Cloud – Subnet Network Access Control Lists (NACLs)** | VPC and subnet security starts with controlling traffic flow using a network access control list (NACL). The default NACL allows all inbound and outbound traffic and a custom can be associate with a lot of subnets. NACLs are stateless. | 1 | 151 |
| **Elastic Compute Cloud (EC2)** | It provides a virtual computing in the cloud, configurable instance types with different CPU, Memory, storage, etc. Deploy into different regions and availability zones and Access Control using AWS key pairs. It also uses preconfigured templates using Amazon Machine Images and Control Traffic flow using security groups. | 1 | 152 |
| **Amazon Machine Images (AMIs)** | Amazon Marketplace provides AMIs for new EC2 instances and supports several AMIs for customers to consume such as Amazon Linux, ECS Optimized, Windows Server, etc. Customers can create private AMIs stored in their account and can share public AMIs with the community. Vendors publish AMIs on the AWS Marketplace. | 1 | 153 |
| **EC2 Security Groups** | Security groups control the ingress and egress traffic flow to an EC2 instance’s network interfaces. | 1 | 154 |
| **Stateful** | Responses to inbound traffic are automatically allowed. | 1 | 154 |
| **Management subnets** | Secure cloud architecture commends created a separate management VPC as it contains bastion/RPD gateway to admin instances. VPC peering connectors to application servers/resources and can contain CI, CD, CM monitoring resources. | 1 | 155 |
| **Serverless Security Benefits** | For one, the attack surface is much smaller because there are no running servers to attack. Serverless also means that there are no long-running servers that can be scanned or attacked. It's also harder to exfiltrate data as traditional reverse shells are not as effective since serverless architectures can be stricter about limiting connectivity. | 1 | 156 |
| **Serverless Security concerns** | Attack surface is bigger, as anyone can deploy a function, and the cost to deploy a function is zero and it is difficult to track and delete functions. Authentication and access control as we are not sure who is allowed to deploy functions and how to lock down each function. Lastly compliance. | 1 | 157 |
| **Application Security (Serverless)** | Application security is even more important with serverless • If attackers have less infrastructure to attack the focus naturally shifts to the application | 1 | 158 |
| **Code Review**  **(Serverless)** | Functions still rely upon source code implemented by developers. As such, there is always the risk that security vulnerabilities could be introduced through software bugs. | 1 | 158 |
| **Data Sanitization**  **(Serverless)** | Data that is sent to outside sources should always be sanitized to ensure that external sources are not exposed to attack payloads | 1 | 158 |
| **Dependency checking (Serverless)** | These shared libraries need to be scrutinized for security vulnerabilities as closely as the custom code written for a function. | 1 | 158 |
| **Input Validation (Serverless)** | One of the most effective ways to prevent attacks is to perform validation on all inputs processed by any application code, including functions. Any data used in application processing that was not generated by a function should be validated using allowlists or denylists to prevent unwanted data from being processed by the function code. | 1 | 158 |
| **Cloud-Native Security Services** | One distinct advantage of working with the large cloud service providers are the cloud-native security services. This includes capabilities that log every activity taken in the Cloud Management Plane (basically every change made to the cloud environment itself), cloud-native logging and analytics capabilities, as well as Threat Detection services that leverage concepts of artificial intelligence (such as machine learning) and third- party data feeds. | 1 | 159 |
| **Cloud Sandbox** | As you learn and practice your hands-on security skills, don't forget that the cloud is a great place to play and experiment. | 1 | 160 |
| **Windows Defender Firewall:**  **Pros** | * Built-in, free, and enabled by default * Stateful dynamic filtering. * Per-application and per-services rules * IPv4 and IPv6 support. * Both ingress and egress filtering. * Integrated with IPSec driver * Manageable through Group Policy. * Command line management with PowerShell or NTSH.EXE | 1 | 160 |
| **Cloud Security: Some Final Conclusions** | * Public cloud is predicated upon the same foundations of private cloud * defenses covered earlier for virtual environments equally apply to public cloud * Installations with minimal footprints, disabling of unnecessary services, baseline configurations, asset management, segmentation, maintaining of trust levels, etc. * apply to public cloud virtual workload | 1 | 161 |
| **Virtualization and Cloud Essentials Summary** | * Abstractions prove useful in a multitude of ways * Abstractions, applied to IT assets, opened the door to new and powerful computing capabilities, resulting in the concept of private cloud * Abstractions applied to public (shared) infrastructure allow for ANY organization to benefit from such powerful computing capabilities - the concept of public cloud * Cloud is too powerful to ignore. The advantages, HOWEVER, will pale in comparison to the cost of a compromise, if security is not considered | 1 | 162 |
| **Securing Wireless Networks** | An understanding and security applications of: The Pervasiveness of "Wireless" Communications, Traditional Wireless: IEEE 802.11 and its continual evolution, PAN, and NFC, 5G Cellular (Mobile) Communication, IoT (The Internet of Things) | 1 | 165 |
| **Wireless pervasiveness** | What we do NOT focus on is how these conveniences work, why it is automatically available to a human without the human requesting it first (a violation of defense-in-depth and principle of least privilege), and what is being developed to make these conveniences truly convenient: ease of use. | 1 | 168 |
| **Wireless pervasiveness** | If wireless is pervasive and yet unknown at the same time, it can't possibly be secure. You cannot secure what you don't know you have or don't understand. Don't think of wireless relative to the device it's in; think of wireless in how the device connects via wireless | 1 | 169 |
| **IEEE802.11** | Also known as traditional wireless | 1 | 170 |
| **IEEE802.11 – Evolution** | 802.11 refers to Part 11 of the 802 standards, created by IEEE (The institute of Electrical and Electronic Engineers). It was the first type of widely used wireless. The 802 standards give us the methods by which we can create LANs (local area networks) and MANs (Metropolitan Area Networks). Part 11 specifically gives us the methods by which we can create WLANs (Wireless Local Area Networks) | 1 | 171 |
| **IEEE802.11 (802.11)** | IEEE 802.11 is not a single standard, nor a single technology. 802.11 refers to Part 11 of the 802 standards, created by the IEEE (the Institute of Electrical and Electronic Engineers). The 802 standards give us the methods by which we can create LANs (Local Area Networks) and MANs (Metropolitan Area Networks). Part 11 specifically gives us the methods by which we can create WLANs (Wireless Local Area Networks) | 1 | 171 |
| **802.11** | 802.11 is continually evolving; 802.11 represents a collective of the original standards and improvements (known as amendments) 802.11: • Created in 1997 • Ratified in 1999 • Over a dozen amendments to the standard since then • Amendments are eventually incorporated into the full 802.11 standard | 1 | 172 |
| **802.11 Network** | WLAN (Wi-Fi) protective encryption (as there can be varying levels of protection): WEP, WPA1, WPA2, or even perhaps the newer WPA3. | 1 | 173 |
| **802.11 Amendments** | Have focused on making improvements in the areas of performance, range, and reliability. Security has often not been the primary driver of amendments. Each amendment is represented by a letter added like 802.11ac and the letters will represent the amendments. | 1 | 174 |
| **802.11a** | Supported a max of 54MBPs and operates in a 5 GHz. Ratified in 1999 and incorporated to the standard in 2007 | 1 | 175 |
| **802.11b** | Supported a max of 11MBPs and operates in a 2.4 GHz. Ratified in 1999 and incorporated to the standard in 2007 | 1 | 175 |
| **802.11b, 802.11a, and 802.11g** | b - 1999, 2.4GHz, 11Mbps theoretical max. a & g - better range, better bandwidth, theoretical max of 54Mbps | 1 | 175 |
| **802.11g** | Supported a max of 54MBPs and operates in a 2.4 GHz. Ratified in 2003 and incorporated to the standard in 2007 | 1 | 175 |
| **TCP Headers** | Source port and destination port, sequence number, acknowledgement number, data offset, TCP flags, window size, options. | 1 | 87 88 |
| **802.11n** | 2009, higher bandwidth (100Mbps initially), MIMO, device supports is WiFi4 | 1 | 177 |
| **802.11n** | Brought substantial performance improvements and overcome limitations of early amendments. Leveraging new techniques such as MIMO (Multiple Input Multiple Output) and signal reflection and higher bandwidth to up to 100 Mbps. | 1 | 177 |
| **MIMO (Multiple Input Multiple Output)** | Techniques such as MIMO (Multiple Input Multiple Output) and the usage of reflection to "increase" the effective bandwidth allowed for better range and better sustained throughput. | 1 | 177 |
| **802.11ac** | 802.11ac was truly designed to allow for high-speed interaction across the WLAN, supporting large numbers of simultaneous users, with as effective coverage area as possible. 802.11 ac allows for a minimum of 1 Gbps in a multilink scenario, with a minimum of 500 Mbps for a single link scenario. 802.11ac truly facilitates the idea of always on, always connected, and always communicating devices on a network. 2013. Device supports is WiFi5 | 1 | 179 |
| **802.11ax** | Represents the future of WLAN capability. Feb 2021. High Efficiency Amendment. Device supports is WiFi6 | 1 | 181 |
| **Wi-Fi 4** | Device supports 802.11n | 1 | 183 |
| **Wi-Fi 5** | Device supports 802.11ac | 1 | 183 |
| **Wi-Fi 6** | Device supports 802.11ax | 1 | 183 |
| **802.11 Security** | Wireless signals are "in the air". Wireless sniffing allows for people within range of a wireless network to acquire and analyze the signal. 802.11i only amendment for security. | 1 | 185 |
| **802.11i** | Only amendment related to security | 1 | 185 |
| **WEP (Wired Equivalent Privacy)** | WEP was designed to provide confidentiality "equivalent" to what would be granted on a wired network. More simply stated, WEP was designed to protect the confidentiality of data transmitted across the WLAN via the use of encryption. The encryption capabilities leverage the use of a symmetric stream cipher. The particular cipher chosen was RC4 | 1 | 187 |
| **IV (Initialization Vector)** | Combined with a pre-shared key to create the RC4 encryption keys for the actual encryption of the packets of network communication with WEP. Repeating will lead to a potential uncovering of the pre-shared key | 1 | 189 |
| **Stream Cipher** | ex: RC4, fundamental the keying material doesn't repeat | 1 | 189 |
| **WEP Issues Discovered** | A stream cipher, such as RC4, relies on the concept of key material NOT repeating. An IV (Initialization Vector) is combined with the original pre-shared key to create the RC4 encryption keys Initially limited to a 40-bit key length (due to U.S. export restrictions on encryption) The IV was limited to 24 bits, meaning that on a busy network, an IV would eventually be repeated. Research demonstrated that if a sufficient number of packets were acquired from a WEP-protected network, the original pre-shared key could easily be discovered It is generally accepted that a WEP-based network's pre-shared key can be discovered in as little as one minute Because of these weaknesses, WEP was officially deprecated from use as of 2004 and should NOT, under any circumstance, be used today. | 1 | 189 |
| **WEP replacements Wi-Fi Alliance** | The WEP replacements were created in conjunction with work performed by the Wi-Fi Alliance and a membership representing companies, consumers, etc was created. The Wi-fi alliance was created to certified providers. | 1 | 191 |
| **Wi-Fi Alliance** | Founded in 1999. Mission:   * Advancing standardization * Addressing security issues * Ensuring interoperability | 1 | 191 |
| **TKIP (Temporal Key Integrity Protocol)** | A common misunderstanding is that TKIP is a new encryption algorithm. It is not. TKIP is an improvement to how the encryption keys for the underlying RC4 stream cipher are generated. TKIP mixes the IV with the pre-shared key to create unique 128-bit encryption keys for each packet of communication. TKIP integrates the concept of integrity checks, better known as a MIC (Message Integrity Check), and called the Michael MIC | 1 | 194 |
| **WPA1** | WPA1 was released in 2003. RC4 used an underlying encryption algorithm same as WEP to facilitate a software upgrade and TKIP (temporary key integrity protocol) to generate better encryption keys and MIC (Message Integrity Check) to prevent forgery attacks. Deprecated on 2009 due to several vulnerabilities. | 1 | 194 |
| **MIC (Message Integrity Check)** | Integrity Checks help make packet forgery and replay attacks more difficult. | 1 | 195 |
| **Michael MIC** | The specific MIC used for TKIP | 1 | 195 |
| **AES (Advanced Encryption Standard)** | Requires substantially more powerful hardware than was needed to support RC4 for WEP. WPA2 uses keys of 128 bits in length for the underlying AES cipher. | 1 | 196 |
| **WPA2** | Represents the strongest protection for WLAN. Ratified in 2004, and the only amendment with security focus. AES (advanced Encryption Standard) uses 128 key lengths for confidentially protection. It’s been available for 15 years. | 1 | 196 |
| **WPA2** | WPA2 (until recently) represented the strongest protection for a WLAN. Upon official ratification of the 802.11i amendment in 2004, the Wi-Fi Alliance released WPA2. The 802.11i amendment is the only amendment discussed in today's material with a security focus. AES (Advanced Encryption Standard), utilizing key lengths of 128 bits, represents the encryption algorithm used for confidentiality protection. AES support meant a hardware upgrade would be necessary. WPA2 has existed for a long enough period of time for cracks to have begun appearing in its security foundation | 1 | 196 |
| **4-way Handshake** | Let's discuss the fundamentals of where things went wrong with WPA2. WPA2 generates the keys for encryption based on an initial communication exchange between the client device and the access point of the network (as part of the initial setup of the connection). The exchange is better referred to as the 4-way handshake. The 4-way handshake allows for a client device and access point to authenticate each other based on a pre-shared key and then to use that pre-shared key to generate the remaining encryption keys. During this 4-way handshake, the client and access point exchange encrypted data based on their individual knowledge of what the pre-shared key should be. If the other party can decrypt and verify the messages exchanged, knowledge of the pre-shared key is implied. This all occurs without disclosing the pre-shared key itself. | 1 | 197 |
| **KRACK** | In October 2017, Mathy Vanhoef and Frank Piessens released details of major weaknesses discovered in WPA2. The research was given the name KRACK, in light of the fact the most serious attack potential was based on key reinstallation. The vulnerabilities focused on the 4-way handshake used by a client and an access point for the purpose of authentication and encryption key generation. Ambiguity in the wording of the WPA2 specs on how devices should respond to replayed sequences of the 4-way handshake led to some devices reinstalling keys (key re-use). For some vendors, it went beyond key use. With some implementations, the researchers discovered a client device might resort to using a key of all 0's (zeros)! The only way forward is to patch. | 1 | 197 |
| **CES (Consumer Electronics Show)** | In January 2018, at the CES (Consumer Electronics Show) in Las Vegas, the Wi-Fi Alliance announced upcoming enhancements to the WPA specifications: WPA3. The WPA3 specifications were officially released in June 2018 | 1 | 199 |
| **WPA3** | Jan 2018, the Wi-Fi Alliance announced. Full specifications officially released June 2018 Vendors have already pledged support for WPA3; however, ubiquitous access to the capability will take time That being said, we do have (large) examples of vendor support: Samsung's Galaxy S10 smartphone (released March 2019); Microsoft's support in Windows 10 The big surprise: Security will be mandated! All devices that wish to receive certification as being Wi-Fi 6 capable (802.11ax) will need to fully support WPA3 | 1 | 199 |
| **SAE (Simultaneous Authentication of Equals)** | Utilizes the pre-shared key as part of a newer handshake known as Dragonfly. Once a Dragonfly handshake has occurred, the result is a shared secret that is then leveraged as part of the traditional 4-way handshake. The 4-way handshake will result in the session (and other important) keys utilized during the remainder of the communication. The most important takeaways: (a) the weaknesses surrounding the use of a pre-shared key are better mitigated, and (b) Dragonfly, and the subsequent key generation, will result in unique keys PER communicating client. A compromise of one set of keys only compromises the communication of that client, not the entirety of all of the WLAN-based traffic. | 1 | 201 |
| **WPA3 Enhancements** | 192-bit key lengths, Better implementations of pre-shared keys (PSK), Better password guessing protection, Protected Management Frames (PMF), Opportunistic Wireless Encryption (OWE) | 1 | 201 |
| **Dragonfly** | Used by SAE. Once a Dragonfly handshake has occurred, the result is a shared secret that is then leveraged as part of the traditional 4-way handshake. The 4-way handshake will result in the session (and other important) keys utilized during the remainder of the communication. Will result in unique keys per communicating client. Key compromise only compromises communication of that client, not the entire WLAN traffic | 1 | 202 |
| **OWE (Opportunistic Wireless Encryption)** | Designed to provide encryption on open, insecure networks. If both a client device and access point support OWE, they will attempt a Diffie-Hellman Key Exchange (specifically Elliptic-Curve Diffie-Hellman: ECDH). this will result in a shared secret of the client and the access point. From that point forward, the shared secret will be used in the appropriate handshakes to generate the remaining (including session) keys. | 1 | 202 |
| **Protected Management Frames (PMF)** | The greatest danger could come from an adversary impersonating an access point and transmitting spoofs of management frames. The example we will use is the spoofing of disassociate frames. If an access point determines, for whatever reason, that it needs its clients to disconnect, it can command them to do so via the use of a disassociate frame. If an adversary can successfully impersonate an access point and send these disassociates to all connected clients, the connected clients will obey. The adversary might continuously transmit these frames such that every time the clients reconnect, they are told to disconnect once again. The impact could be a sustained DoS, resulting in a completely unusable network. Protected Management Frames are designed to try and prevent these types of attack scenarios. PMF are required and mandated for WPA3 | 1 | 202 |
| **Diffie-Hellman Key Exchange** | Used in OWE | 1 | 203 |
| **ECDH (Elliptic-Curve Diffie-Hellman)** | Specific key exchange in OWE. This will result in a shared secret of the client and the access point. | 1 | 203 |
| **Downgrade Attack** | A downgrade attack occurs when an adversary convinces technology to use older protocols or software. Can exist wherever there is backwards compatibility. | 1 | 204 |
| **Dragonblood** | Dragonblood, the name given to a research paper written by Eyal Ronen and Mathy Vanhoef (yes, the same author of the KRACK research paper) describes a set of vulnerabilities discovered in early implementations of WPA3. The authors worked with the Wi-Fi Alliance and impacted vendors on a coordinated disclosure approach, just as had been done with the KRACK vulnerabilities. The potential attacks (there were five, in total, discussed in the paper) fall broadly into the categories of downgrade attacks, password cracking, and Denial of Service (DoS) | 1 | 204 |
| **WPA3 Attacks** | Dragonblood research paper released in 2019 found 5 potential attacks in the broad categories of downgrade attacks, password cracking, and Denial of Service (DoS) | 1 | 204 |
| **Backward Compatibility** | Dual support for newer security and older more problematic security, opens door for downgrade attack. | 1 | 205 |
| **Transitional Mode** | In transitional mode, an access point can support both newer WPA3-capable client devices and ones that only support WPA2. In transitional mode, the access point will advertise to potential client devices during the association process that both WPA3 and WPA2 are supported. The client device will choose the option best suited to itself. WPA3 client devices will choose WPA3, and WPA2 client devices will choose WPA2. | 1 | 205 |
| **Side-Channel Attack** | An adversary leverages peripheral knowledge of a system in order to infer something important about the system. The one we will discuss is the one that leads to a potential discovery of the pre-shared key, via off-line password cracking. | 1 | 206 |
| **WPA3 Side-Channel Attack Protection** | Can be leveraged by adversary to cause access point to crash causing DoS | 1 | 206 |
| **Rogue Access Points (Rogue AP)** | Rogue AP initially referred to an unauthorized access point connected to the internal wired network Rogue AP today is sometimes used in place of a different term—a masquerading AP (evil twin AP). An evil twin AP has the same "name" as a legit AP. Mitigation: 802.1X, Certificates for mutual authentication | 1 | 208 |
| **802.1X** | 802.1X is a network access control method that attempts to restrict access to a network. Rogue AP mitigation | 1 | 209 |
| **802.11 Denial of Service (DoS)** | Jamming, leveraging timing-based attack protection on itself. Mitigation: quickly detect, locate, respond, and react, or fall back to a wired network. | 1 | 210 |
| **DoS** | Wireless communications are simply radio signals being sent through the air all around us and can only be successful if the communication devices can “hear” each other. bSignals that are present that we didn’t create can be referred as noise and noise can be created to “Jam” a signal. Best mitigation is early detection. | 1 | 210 |
| **SNR (Signal to Noise Ratio)** | SNR is a way to describe and measure how well devices are able to "hear" each other. | 1 | 210 |
| **Jamming** | intentionally generated noise? Noise that was so pervasive and interfering that our devices could no longer detect their own network's signals? A DoS would occur. Most common type of DoS envisioned in discussions of DoS. | 1 | 211 |
| **PAN Wireless** | PAN has a different focus: A "Personal" Network. "Personal" has a specific meaning: In the physical proximity of a human being. Personal ≠ Private. PAN isn't just one technology. The PAN concepts we will discuss: Bluetooth, NFC (Near Field Communication), Zigbee, RFID (Radio Frequency Identification) | 1 | 213 |
| **Bluetooth – Wireless PAN** | First cable replacement applications: Mice, keyboard, printers. Bluetooth's popularity, though, came from a different application: Hands-free communication. Bluetooth devices today might offer a variety of services; Bluetooth profiles describe which services are being offered. Due to variety of potential applications for Bluetooth, Bluetooth hardware and software have evolved to send Bluetooth signals further, making attacks from afar easier. Largest concerns stem from ubiquity of Bluetooth and the fact that Bluetooth is almost always on by default for any device that has it. | 1 | 215 |
| **Bluetooth SIG (Special Interest Group)** | Bluetooth is now based on standards provided by the Bluetooth SIG (Special Interest Group). The Bluetooth SIG does for Bluetooth what the Wi-Fi Alliance does for WLAN; the SIG promotes standardization, interoperability, ease-of-use, product certification, etc. | 1 | 215 |
| **A2DP (Advanced Audio Distribution Profile)** | Bluetooth profile for music streaming | 1 | 216 |
| **Hands-Free Profile (HFP)** | Bluetooth profile for hands-free phone calls | 1 | 216 |
| **OPP (Object Push Profile)** | Bluetooth profile for contact and address book synchronization. | 1 | 216 |
| **BlueBorne** | September 2017, Armis, an Enterprise IoT security company, detailed a series of Bluetooth vulnerabilities which are a result of issues in vendor implementations and include remote code execution and information disclosure. | 1 | 217 |
| **Bluetooth attack** | “BlueBorne” vulnerabilities include remote code execution and information disclosure that can result in exposure of encryption keys. The number of impacted devices were in the billions including Android, Windows, smartphones, refrigerators, Etc. | 1 | 217 |
| **Bluetooth Vulnerabilities** | BlueBorne | 1 | 217 |
| **Bluetooth Protections** | Endeavor to use only the latest versions of Bluetooth. New Bluetooth support might mean software AND hardware upgrades. Most recent version of Bluetooth is 5.3, as of July 13, 2021. Disable unnecessary Bluetooth profiles. Don't leave Bluetooth persistently on. Even better, if you don't need it at all (think of servers), then don't have Bluetooth capability installed. Always install all available patches. Central management systems, like Active Directory, might be able to help manage Bluetooth profiles, Bluetooth enablement, patching, etc. | 1 | 219 |
| **SoC (System on Chip)** | Many computing devices utilize SoC (System on Chip) to provide functionality; SoCs are chips that are usually soldered to the motherboard inside your device. These SoC chips integrate a specific function (or functions), such as Bluetooth. Using the latest version of Bluetooth might, therefore, translate into a replacement of that SoC. May need an entirely new device. | 1 | 219 |
| **dotdot** | A framework to help vendors and consumers integrate various Zigbee devices and technologies together. | 1 | 221 |
| **Zigbee** | Designed as a low-cost, low-power alternative to other PAN technologies. Can be run on battery; potential battery life might be measured in years. Best defined as an "automation" technology. Not a single standard but rather a collection of standards and related technologies. Security is included in the standards; vendor implementation is optional. Zigbee powered devices include television set top boxes, light bulbs, utility meters, medical devices, HVAC, and more. There are hundreds of millions of Zigbee devices in use worldwide. | 1 | 221 |
| **Mesh Networking** | The low-powered nature of Zigbee means limited range. A limited range of signal would appear, at initial glance, to be a major obstacle in the creation of automation—especially if we are discussing Zigbee in terms of a connected smart home or smart building. Zigbee itself addresses this issue via the incorporation of mesh networking. In a Zigbee mesh, Zigbee devices relay messages among each other, from one device to the next, until the destination device receives its communication | 1 | 222 |
| **NFC (Near Field Communications)** | “Near Field Communications” design to operate in close proximity of 1 to 2 inches. Used in contactless debit cards, transit passes. Security is determined by the vendor since this technology is widely used in many different markets. | 1 | 223 |
| **RFID (Radio Frequency Identification)** | “Radio Frequency Identification” linked to tags and can be attached to an object for identification, almost like a serial number. | 1 | 225 |
| **5G - Wireless** | Is the next evolution for mobile communications, and has improvements such as low latency, high bandwidth and multiclient support. Smartphones will start using it first but won’t be available everywhere in the world. | 1 | 228 |
| **LTE (Long Term Evolution)** | 4G | 1 | 228 |
| **3GPP (3rd Generation Partnership Project)** | Provide guidance and propose standards for implementation of 5G | 1 | 229 |
| **IMT-2020** | IMT-2020 is the name given by the ITU to the overall framework that will enable 5G communications. | 1 | 229 |
| **ITU (International Telecommunications Union)** | The most accepted authority on 5G standards is the ITU (International Telecommunication Union), a United Nations organization | 1 | 229 |
| **NR 5G (New Radio 5G)** | The 3GPP's proposal for 5G | 1 | 229 |
| **IoT - Wireless** | Concept created to represent of non-traditional computing devices on networks such a light bulb, refrigerators, thermostats, toothbrushes. These devices are computers with RAM, storage, software code and OS and requires the same protections given to desktops and servers. IoT can leverage Bluetooth, Zigbee, Wi-Fi, 5G, etc. | 1 | 231 |
| **Timing Attack** | Adversary leverages peripheral knowledge of a system in order to infer something important about the system. The computer should take the exact same amount of time in all three situations to make its determination. The variances in the timing lead to a timing attack. | 1 | 260 |
| **Defense in depth** | Protections need to be layered: a principle called Defense-in-Depth. We explain some principles that will serve you well in protecting your organization. The following are the key areas that will be covered:  Ø Defense-in-Depth Overview  • Risk = Threats x Vulnerabilities  • CIA Triad ( Confidentiality, Integrity, and Availability)  Ø Strategies:  • Strategies for Defense-in-Depth  • Core Strategies | 2 | 3 |
| **Defense in depth** | Defense-in-Depth is a comprehensive, integrated approach in which multiple solutions are tied together to accomplish a goal. There is no single security solution that will make an organization secure because any single measure could be bypassed (and miss an attack all together) or compromised. When protecting any entity, multiple measures that complement each other must be put in place across a variety of control options. | 2 | 5 |
| **Defense in depth – Layers** | * perimeter security, * network security, * host security, * application security * Data security. | 2 | 5 |
| **Defense in depth – Network Security** | Aims to protect the internal network. “Network Segmentation” and “Network Access Control” are key security controls. | 2 | 5 |
| **Defense in depth – Perimeter security** | Focuses on security controls at the borders of a network such as firewalls, proxy servers, and anti DDoS appliances. This is referred at the protections between you and the internet. | 2 | 5 |
| **Host Security** | All about hardening systems and securing the setup of endpoints and servers. This can be done by securely configuring the system, implementing an application control solution, scanning the system for vulnerabilities, and installing an anti-malware agent. A good reference for a secure system configuration would be the CIS benchmarks. | 2 | 5 |
| **Network Security** | Aims at protecting the internal network. Network segmentation and network access control are key security controls to be implemented. | 2 | 5 |
| **Application Security** | Creating a secure application by the means of implementing authentication mechanisms, writing overflow protections, performing input validation, and much more. | 2 | 6 |
| **Data Security** | Encompasses techniques such as encryption and data classification in order to handle data properly according to its sensitivity and stay in control of the content. | 2 | 6 |
| **Defense in depth –**  **Data Security** | Techniques such as encryption and data classification in order to handle data properly according to its sensitivity. | 2 | 6 |
| **Defense in depth –**  **Host security** | It is all about hardening systems and securing the setup and endpoints of servers by securely configuring the system, implementing an application control solution, scanning the system for vulnerabilities, and installing anti-malware agent. | 2 | 6 |
| **Defense in depth – Application Security** | Refers to creating a secure application by the means of implementing authentication mechanisms, writing overflow protections, performing input validation, etc. | 2 | 6 |
| **Risk** | Is the level of exposure to certain danger and is composed of a vulnerability threat level. Risks, threats, and vulnerabilities are highly interrelated. Risk (due to a threat) = Threat x Vulnerability (to that threat) | 2 | 6 |
| **Threat** | All possible dangers that would result in any type of damage. | 2 | 7 |
| **Vulnerabilities** | Weaknesses in a system that can be exploited. | 2 | 7 |
| **CIA Triad** | Confidentiality, Integrity, Availability | 2 | 9 |
| **Risk Key Focus – CIA: examples** | * **Confidentiality:** customers will expect that the privacy of their credit card numbers, or other information shared during a transaction to be ensured. * **Integrity:** Customer will expect product availability to be accurate, to quantities and quality be what they agreed to, and anything that they download be authentic and complete. * **Availability:** Customers will expect to be able to place orders when is convenient for them and the employer will want the revenue stream to not be disturbed. | 2 | 9 |
| **Risk Balance: Security vs. Functionality** | Application updates bring more functionality over time; however, more functionality typically means a drastic drop in security. Key challenge: Finding the right balance! | 2 | 12 |
| **Anti-DDoS Appliances** | High-throughput devices that inspect all incoming traffic and will filter out any excessive network packets. These devices are typically installed at the border of the network. | 2 | 14 |
| **Filtering** | The foundation of Defense-in-Depth. Network filtering technologies:   * Firewalls * Anti-DDoS * Proxy servers * Mail relays Host-based filtering technologies: * Anti-malware software * Application control | 2 | 14 |
| **Firewalls** | Todays are stateful meaning they keep track of the state of connections. Latest are next-generation which not only inspect state but also contents of packets and drop those that do not conform to the security policy of the environment. | 2 | 14 |
| **Proxy Server** | Proxy servers are intermediate systems that handle requests to resources on behalf of other systems. They are specialized in inspecting a specific protocol and/or content type. Examples include mail relays, web application firewalls, and API gateways | 2 | 14 |
| **Anti-malware software** | They inspect every file of a system and look for malicious code patterns called virus signatures. Evolving into EDR tools, which also look for sequences of activity. | 2 | 15 |
| **Application Control** | Another way of applying a filter on a system. Such solutions will kick in every time a new process is started and will only allow approved applications to run. | 2 | 15 |
| **EDR (Endpoint Detection and Response) tools** | Look for malicious sequences of activity. | 2 | 15 |
| **Defense in depth – Approaches** | 4 Approaches:   1. Uniform protection 2. Protected enclaves 3. Information centric 4. Vector oriented | 2 | 16 |
| **Uniform Protection Defense-in-Depth** | This is the most common approach to DiD. In this setup, all parts of the organization receive equal protection and are treated the same. | 2 | 17 |
| **(N)ACLs (Network Access Control Lists)** | (N)ACLs on firewalls, routers, and switches specify which traffic flows are allowed or blocked between the different segments on your network. These access control lists can generally be configured to control both inbound and outbound traffic. | 2 | 18 |
| **Private VLAN** | Also known as port isolation, is a technique in computer networking where a VLAN contains switch ports that are restricted such that they can only communicate with a given "uplink". | 2 | 18 |
| **Protected Enclaves** | Assets that require additional protection are segmented from the rest of the internal organization, and access to it is restricted. This can be done by using VLANs or Network Firewalls or (N)ACLs. | 2 | 18 |
| **VLAN** | VLANs are virtualized LANs, meaning logical segments within the same physical network. | 2 | 18 |
| **information-centric** | Any organization uses multiple layers to access confidential information and they need to build successive layers of protection on top of your other layers. It is key that proper data classification standard is in place, security controls for each confidentiality level are defined. | 2 | 19 |
| **Vector-Oriented Defense-in-Depth** | Involves identifying the various vectors that are employed by threats and providing security mechanisms to shut down those vectors -- for example, disabling USB drives, blocking or scanning attachments, verifying addresses at email server. | 2 | 20 |
| **Vector-Oriented threat analysis** | A threat requires a vector to cross the vulnerability. Preventing the threat from using that vector will stop its capability to exploit it. Vectors such as malicious USB Drives, attachments containing malware, spoofed emails. | 2 | 20 |
| **System Security Layer** | Most of the DiD principles are focused on the network layer which focused on filtering. At a system level we can implement configuration hardening on assets such as network components, operating systems, software and cloud environments to protect this layer. | 2 | 21 |
| **Vulnerability Criticality** | Ratings applied to identified vulnerabilities are likely subjective and have different meanings to different organizations. Without providing any context around those rating they have little meaning. It is often the case where asset value and impact are multiplies by the likelihood of compromise. Rating examples: Low, medium, high, severe, grave. | 2 | 22 |
| **Configuration Management** | Is the discipline of establishing a known baseline condition such a document compliant with applicable standards and then managing that condition. Revie and change control are critical as they provide a way to detect when a change occurs to that baseline. | 2 | 23 |
| **Heartbleed: Cerebration over computation** | With a common vulnerability scoring system (CVSS) score of 5.0/10, it is often overlooked during triage and placed in the “some-da” stack of info-sec to do list. However, it could have calamitous impacts or organizations, but still a 5.0. Vulnerability rating systems may or may not even approximate rea-world risk. | 2 | 23 |
| **Configuration management –**  **Fixing it** | The main strategy for fixing an infected system is to rebuild that system from scratch because you would never know if there were anything that was left behind if we don’t have a baseline. | 2 | 25 |
| **Fixing the Problem** | The main strategy for fixing an infected system is to rebuild the system from scratch | 2 | 25 |
| **Defense in depth –**  **Not limited to on-premises** | Did is not only applicable to your own-premise environment, thing larger cloud environments, you will find several security controls that can be implemented on different layers. | 2 | 26 |
| **Defense in depth – in the Cloud** | Firewalls, Web Application Firewall (WAF), IPS/IDS, E-mail protection, VPN, IAM, Centralized logging, Container Security | 2 | 27 |
| **Firewalls (DiD - Cloud)** | In the cloud, we have (Network) Security Groups, which is a set of access control rules that can be wrapped around a VM or subnet. These rules then inspect the ingress and egress traffic to determine whether to allow or deny a packet | 2 | 27 |
| **Web Application Firewall (WAF) (DiD - Cloud)** | Will inspect the traffic up to the application Layer (7) | 2 | 27 |
| **E-mail protection (DiD - Cloud)** | Using a cloud email server also means that these tasks can be unified in a single solution, such as Microsoft Defender for Office 365 or Exchange Online Protection (EOP), offering an all-in-one solution to automatically analyze all incoming and outgoing emails for threats and allowing each email account user to manage their own email quarantine. | 2 | 28 |
| **IAM (Identity and Access Management) (DiD - Cloud)** | The process that deals with assigning a digital identity to the users of a system and making sure the right people have access to the right system resources at the right time. | 2 | 28 |
| **Identity Lifecycle** | Account Creation, Authenticator Issuance, Changes in responsibility/access rights, Account Retirement | 2 | 28 |
| **IPS/IDS (DiD - Cloud)** | While IDS and IPS are, in essence, very similar, an IDS will prioritize availability of the system over stopping an attack immediately, and, thus, instead of preventing it, will simply alert the administrator of a potentially malicious event, rather than blocking the traffic altogether. An IPS would do the opposite and block the traffic that was detected as malicious immediately, even at the risk of blocking genuine traffic in the event of a false positive. | 2 | 28 |
| **VPN (DiD - Cloud)** | When making use of Cloud Hosting, a user may want to connect to services running in the cloud as if they were on the local network. To make this possible, Cloud providers offer VPN solutions that combine extremely high availability and ease-of-use, by taking the burden of the VPN server setup and configuration out of the organization's hands. | 2 | 28 |
| **Centralized logging (DiD - Cloud)** | When using an on-premises infrastructure, specific logging tools need to be deployed to keep track of all system events on hosts within the network, as well as to analyze the network traffic itself. In the cloud, this process is streamlined because of the unified ecosystem offered by the cloud provider. | 2 | 29 |
| **Container Security (DiD - Cloud)** | Cloud Services recognize the popularity of containerization and need of organizations to run containerized applications. As such, Azure, AWS and GCP all offer a fully featured containerized environment that provisions the containers automatically and secures their configuration and environments by default. | 2 | 29 |
| **On Premise IAM** | Identity And Access Management Authentication includes multiple components. Context based authentication is enforcer via multiple technologies such as multifactor authentication and device compliancy checks. | 2 | 30 |
| **Azure IAM** | Azure makes use of a ‘IF-This-THEN-That' mechanism that allows an administrator to tell the system what to do in certain conditions. This system of access control is called Ruleset-Based Access Control | 2 | 31 |
| **Conditional Access** | IF-This-THEN-That | 2 | 31 |
| **Ruleset-Based Access Control** | IF-This-THEN-That' mechanism that allows an admin to tell the system what to do in certain conditions. | 2 | 31 |
| **AWS AIM** | Similar to the other platforms, AWS allows for multi-factor authentication and “Identity Federation”, meaning a user can either be logged in with a designated AWS account or with an account on another platform (Google account, Microsoft account, etc) That has been linked to AWS. After the user has been authenticated, they can then access the resources on the system that they have been authorized to. | 2 | 32 |
| **AWS Policies** | Policies in AWS are composed of different elements: Version, ID, The SID, Effect, Principal, Action, Resource, Condition | 2 | 32 |
| **GCP IAM** | In the Google Cloud IAM paradigm, access is controlled by grouping permissions into Roles, which are then, in turn, granted to one or more (authenticated) Members. This IAM system is an example of a Role-Based Access Control mechanism (RBAC). | 2 | 34 |
| **On-premises Network Segmentation** | Network segmentation is typically done by splitting up networks into different zones (VLANS) and firewalls to control and allow access to specific resources within your network. | 2 | 35 |
| **Azure Network Segmentation** | For IaaS Azure uses virtual networks band network security groups to do stateful packet inspection. A network security group is reusable object within azure that can be assigned to an azure resource. | 2 | 36 |
| **Azure VNet (Virtual Network)** | A VNet is nothing more than the virtual representation of what a regular network layout would look like. It allows the administrator to connect resources such as Azure Virtual Machines securely to each other, the internet and on- premises networks. | 2 | 36 |
| **Network Security Groups (NSG)** | Azure allows to define Network Security Groups (NSG). A security group can be seen as a "virtual firewall" in which the user can put security rules that allow or deny inbound or outbound traffic to and from several types of resources. | 2 | 36 |
| **AWS Network Segmentation** | Amazon AWS uses VPC, security groups and Networks access control list to enable segmentation. A Security group acts as a virtual firewall that controls the traffic for one or more instances. | 2 | 37 |
| **VPC (Virtual Private Cloud)** | AWS's service offering does not differ strongly from Azure's. Those who look to design their virtual network structure are able to do so in the form of Virtual Private Clouds, or "VPC" for short, just like a VNet would define a network layout in Azure. | 2 | 37 |
| **GCP (Google cloud Platform) Network Segmentation** | Google Cloud Platform uses Virtual Private Cloud and VPC Firewall Rules that filter both ingress and egress traffic on a per-instance basis to allow network segmentation. VPC Firewall Rules:   * Allow all TCP traffic from Internet to VM1 * Default denies for VM2 (no ingress rule) * Allow VM4 to communicate with VM3 (client tag) * Allow VM4 to communicate with VM1 (no egress rule) | 2 | 38 |
| **Defense in depth – shortfalls** | Internal threats already bypassed a few of the outer security controls. The rise of unmanaged devices within a network such as mobile phones and tables from a new challenge to secure environments. | 2 | 39 |
| **Zero-Trust** | Zero-trust is based on two key factors: authentication and encryption. Every time someone wants to access data or perform a certain operation, regardless of whether they are internal or external, this person will have to authenticate and be authorized before they can continue. | 2 | 39 |
| **Zero-Trust:**  **Basic principles.** | Internal and external threats will always be present.  Every user, device or network connection must be proven  log and inspect all network traffic. | 2 | 40 |
| **Zero-Trust: Securing Your Traffic** | All traffic must be: Authenticated to prove the user, device, or network connection is legitimate. Encrypted to protect the confidentiality of the traffic while cryptographically proving its source and destination. | 2 | 41 |
| **Zero-Trust: Variable trust** | Variable trust enables you to dynamically change access for good or bad based on conditions. Some conditions that can be used to dynamically change access are Type of user access, Geo Location, Device compliancy and type of application. | 2 | 42 |
| **Zero-Trust: Variable trust examples** | An employee wants to access an internal application. The access is granted when the variable trust score is 30 or above. | 2 | 43 |
| **Zero-Trust: Log Inspection** | Logging and Inspection is required for zero-trust to work effectively. Authentication Logs, Event Logs, Network Logs, Data Access Logs, Email and Messaging Logs | 2 | 44 |
| **Identity & Access Management** | Identity Management: Digital Identities, Enrollment, & Identity Proofing. Access Control: Managing, Controlling, & Monitoring Access | 2 | 46 |
| **Accountability** | The process of identifying who did what on the system, as well as when they did it. | 2 | 48 |
| **Authentication** | A process in which a subject proves they are in the possession of one or more valid authenticators associated with an identity. Can rely on 3 types of methods:   * Something you know * Something you have. * Something you are | 2 | 48 |
| **Authorization** | The process of determining what a subject is allowed to do or access after authentication. | 2 | 48 |
| **Digital identity** | Is the online persona of a subject. A single subject can have multiple digital identities for different services. | 2 | 48 |
| **IAAA** | Identity, Authentication, Authorization, Accountability | 2 | 48 |
| **Social Engineering - Types** | Computer-based: Pop-up windows, Mail attachments. Human-based: Urgency, Third-person authorization | 2 | 48 |
| **Identity Access Management IAM** | Is the organizational process for identifying, authenticating, and authorizing individuals or groups of people to have access to applications, systems, or networks by associating user rights and restrictions with established identities. | 2 | 50 |
| **Enrollment** | Process of Enrollment: An individual can become a subscriber to a system by going through the process of enrollment. The outcome of enrollment is typically the assignation of a user ID and credentials. To this end, the applicant will undergo identity proofing by the system owner (or delegate). | 2 | 51 |
| **Identity Proofing** | The process in which an applicant proves they are who they claim to be. After successfully undergoing the identity proofing, the applicant is provided with one or more credentials they can use at a later time to authenticate to the system. 3 steps:   1. Resolution: Evidence of identity and required attributes of the applicant are collected. 2. Validation: Authenticity of the provided evidence is checked. 3. .Verification: The contents of the provided evidence are checked. | 2 | 52 |
| **Identity Assurance Levels (AIL) 1:**  **Self-asserted** | Any information on the applicant’s attributes is deemed self-asserted and, thus, cannot be confirmed | 2 | 53 |
| **Identity Assurance Levels (AIL) 2:**  **Evidence based** | Applies to situations where the system owner is somewhat assured the applicant is associated with the real-world identity they claim to have. Additionally, there is the possibility for the identity of the applicant to be verified by a third party,. The credential service providers who will assert their attributes. | 2 | 53 |
| **Identity Assurance Levels (IAL)** | The NIST 800-63 standard, which deals with identity management, defines three assurance levels, aptly named "Identity Assurance Levels," as follows: IAL 1 - Self-asserted. IAL 2 - Evidence-based. IAL 3 - Physical check. | 2 | 53 |
| **Identity Assurance Levels (AIL) 3**  **Evidence based** | Applies to situations where the system owner has performed a direct verification of the identity of the individual, the applicant needs to be physically present and has to provide substantial evidence to prove their identity. | 2 | 54 |
| **Authentication** | A process in which a subject proves they are in the possession of one or more valid authenticators associated with a subscribed identity. During authentication, the subject is referred to as the claimant. The validity of the authenticator is checked by the verifier.   1. The verifier is presented with the authenticator(s) by the claimant. 2. The verifier checks the correctness of the validity of the provided authenticator(s). 3. The verifier asserts the identity of the claimant to the relying party. | 2 | 55 |
| **Authenticator Assurance Levels (AAL)** | Depending on the sensitivity of the resource being accessed and the context of the access requested, the user can be asked to provide a different level of user authentication. NSIT outlines three different authenticator assurance levers with varying degrees of proofing requirements such as Single factor at least, any two factors and strong crypto, and selected 2 factors and strong crypto | 2 | 56 |
| **Authenticator Assurance Levels (AAL1) Single Factor at least** | Requires either single-factor or multifactor authentication using a wide range of available authentication technologies. | 2 | 56 |
| **Authenticator Assurance Levels (AAL2) Any two factors & strong crypto** | Proof of possession and control of two different authentication factors is required through secure authentication protocol and approved cryptographic techniques are required. | 2 | 56 |
| **Authenticator Assurance Levels (AAL3) Selected 2 factors & strong Crypto** | Based on proof of possession of a key through a cryptographic protocol. It requires a hardware-based authenticator and an authenticator that provides verifier impersonation resistance. | 2 | 56 |
| **Single Sign-On (SSO)** | The SSO technology allows a user to log on once with one set of credentials and use those credentials to access different resources. 2 types of authentication protocols that support Single Sign on capabilities: SAML and Oauth. | 2 | 57 |
| **SAML 2.0** | “Security Assertion Markup Language” Is an XML based protocol for exchanging identities. It uses tokens containing information about the identities to enable SSO on different environments. | 2 | 59 |
| **Oauth 2.0** | Is an open standard for access delegation, it is used to grant websites or applications access to their information on other websites but without exchanging passwords. In general Oauth 2.0 provides secure delegated access and supports SSO. | 2 | 60 |
| **Access control** | Is the broader concept of controlling the access to resources, as well as managing this access. Additionally, constant monitoring is needed in the process to ensure access is revoked when it is no longer needed. The 3 most important:   * controlling access * managing access * monitoring access. | 2 | 61 |
| **Controlling Access** | Least Privilege, Need to Know, Separation of Duties, Rotation of Duties | 2 | 62 |
| **Least Privilege** | Give someone the least amount of access they need to do their job | 2 | 62 |
| **Need to Know** | Only grant someone access to resources when they need it— and revoke it when it is no longer required | 2 | 62 |
| **Separation of Duties** | Break critical tasks across multiple people to limit your points of exposure | 2 | 62 |
| **Rotation of Duties** | Change jobs on a regular basis to prevent anyone from being able to get comfortable in a position and, therefore, being able to cover their tracks and to minimize the chance of collusion | 2 | 63 |
| **Access control techniques** | There are many types but the most important are: Discretionary access control (DAC), Mandatory access control (MAC), Role-Based Access control (RBAC) and Lattice-Based Access Control (LBAC) | 2 | 64  65 |
| **Discretionary Access Control (DAC)** | A type of access control that controls access to certain resources via a credential given to the authorized party. This party is able to transfer this credential and, thus, authorization to another party at their own discretion. | 2 | 64 |
| **Mandatory Access Control (MAC)** | A type of access control that controls access to all resources via system-enforced credentials that are nontransferable by the authorized party. MAC requires all users of the system to be assigned a clearance, as well as all data to be assigned a classification level. | 2 | 64 |
| **Lattice-Based Access Control (LBAC)** | A type of mandatory access control that defines access restrictions on the interactions between subjects and objects. A subject is only allowed to access an object if the security level of said subject is greater than or equal to that of the object. | 2 | 65 |
| **Role-Based Access Control (RBAC)** | A type of discretionary or mandatory access control that assigns users to roles or groups based on their organizational function(s). Each group is associated with an authorization to access certain resources. | 2 | 65 |
| **Access Management** | User accounts, data, and their relationships must be actively maintained, perhaps by an entire team of employees. This process, called “access management,” consists of four tasks: account administration, maintenance, monitoring, and revocation. | 2 | 66 |
| **Account Administration** | A set of processes and controls. The account administrator—who can be the system owner or the member of a dedicated team—enrolls a user, assigns authenticators, and assigns authorizations. | 2 | 66 |
| **Maintenance** | The process of reviewing account data and spot-checking for inconsistencies or errors. | 2 | 66 |
| **Monitoring** | order to enforce accountability of accesses to information, authentications and authorizations should be monitored. | 2 | 66 |
| **Managing and Monitoring access** | It is a process called “Access Management” and consists of 4 tasks. Account administration, maintenance, monitoring and revocation.  Account administration is a setoff processes and controls. Maintenance is the process of reviewing account data and spot-checking for inconsistences or errors. Monitoring is enforcing accountability of accesses to information, authentications and authorizations and Revocation is when account management staff and systems administrators revoke privileges when they are no longer needed, especially if users have been fired. | 2 | 67 |
| **Revocation** | Administrators should promptly revoke privileges when they are no longer needed, especially for users who have been fired. | 2 | 67 |
| **Privileged Access** | The access to a computer system with elevated access rights, such as root or administrator access, or access to service accounts. | 2 | 68 |
| **Password Vaults: Privileged Access Management Tools** | To help organizations manage their privileged access credentials, they can make use of Privileged Access Management (PAM) tools. | 2 | 69 |
| **Privileged Access Management (PAM) Most used tools:** | 1. CyberArk 2. HashiCorp Vault 3. Azure Active Directory | 2 | 69 |
| **Privileged Access Management (PAM) Tools** | * PAM tools usually provide one or more of the following features: * Provide transparency to the user * Policy enforcement point * Generate strong shared secrets * Securely store credentials * Rotate credentials * Monitor and log privileged access * Generate reports. | 2 | 69 |
| **Password Compromise** | Password compromise remains one of the most sought-after and effective access methods used by penetration testers   * This includes password cracking, spraying, reuse, and others * This should not still be the case... Turn on Multi-Factor Authentication! (MFA) | 2 | 70 |
| **Azure privileged identity management** | To implement the least privilege principle within Azure AD, privileged identity management can be used to assign on-demand, time limited, administrative privileges when approved by a workflow. | 2 | 72 |
| **Privileged Identity Management (PIM) - Azure** | To implement the least privilege principle within Azure AD, Privileged Identity Management can be used to assign on-demand, time- limited, administrative privileges when approved by a workflow. | 2 | 72 |
| **3-Tiered privileged access management.** | It is basically separating the administration of, for example, end-user workstations and domain controllers. For example, tier 0 users have access to active directory, critical and secret servers (crown Jewels) and not tier 1 tools such as exchange servers, intranet servers, and tier 2 tools such as workstation, printers, mobile devices and vice versa. | 2 | 73 |
| **Tiered AD Administration** | The tiered administration model is basically separating the administration of, for example, end-user workstations and domain controllers. The tiered model is composed of three levels and only includes admin accounts:   * Tier 0 has direct control of enterprise identities in the environment. This tier includes accounts, groups, and other assets that are used to manage the domain controllers, AD forest, domains, and all the assets in it. * Tier 1 is able to control the enterprise servers and applications. This includes server operating systems, cloud services, and enterprise applications. The administrators have control of a large number of assets. * Tier 2 controls the user workstations and devices. The administrators in this tier are often part of the helpdesk or other support group | 2 | 73 |
| **Tiered Privileged Access Management** | Mitigate Lateral Movement | 2 | 73 |
| **Tiered Privileged Access Management (2)** | this slide, an example of the devices in each tier is given. | 2 | 75 |
| **Authentication Types (1)** | * Something you know: password, pin * Something you have: Look-Up Secrets, Out-of-Band Devices, One-Time Password (OTP) Tokens, Cryptographic Devices * Something you are: biometrics | 2 | 78 |
| **Look-Up Secrets** | "A look-up secret authenticator is a physical or electronic record that stores a set of secrets shared between the claimant and the CSP." An example of a look-up secret would be a cardboard card that has a table of authentication codes on them. | 2 | 79 |
| **One-Time Password (OTP) Tokens** | NIST distinguishes two OTP token categories: "A single-factor OTP device generates OTPs. This category includes hardware devices and software- based OTP generators installed on devices such as mobile phones. These devices have an embedded secret that is used as the seed for generation of OTPs and does not require activation through a second factor. The OTP is displayed on the device and manually input for transmission to the verifier, thereby proving possession and control of the device" An example of a single-factor OTP token is an RSA SecurID keychain token. "A multi-factor OTP device generates OTPs for use in authentication after activation through an additional authentication factor. This includes hardware devices and software-based OTP generators installed on devices such as mobile phones." An example of a multi-factor OTP token is a multi-factor authentication app on your mobile phone. | 2 | 79 |
| **Out-of-Band Devices** | "An out-of-band authenticator is a physical device that is uniquely addressable and can communicate securely with the verifier over a distinct communications channel, referred to as the secondary channel.“ An example of an out-of-band device is a mobile phone receiving a text message with a secret code in it. | 2 | 79 |
| **Something you are** | Fingerprints, retina scans, voice recognition, facial recognition | 2 | 79 |
| **Something you have** | Look-up secrets, out-of-band devices, OTP Tokens, Cryptographic devices | 2 | 79 |
| **Something you know** | A memorized secret authenticator or a password, or a pin, it is a secret value intended to be used by the user. | 2 | 79 |
| **Cryptographic Devices** | Just like with OTP tokens, NIST identifies two different categories for cryptographic devices: "A single-factor cryptographic device is a hardware device that performs cryptographic operations using protected cryptographic key(s) and provides the authenticator output via direct connection to the user endpoint. The device uses embedded symmetric or asymmetric cryptographic keys and does not require activation through a second factor of authentication. Authentication is accomplished by proving possession of the device via the authentication protocol." "A multi-factor software cryptographic authenticator is a cryptographic key stored on disk or some other "soft" media that requires activation through a second factor of authentication. Authentication is accomplished by proving possession and control of the key. The authenticator output is highly dependent on the specific cryptographic protocol, but it is generally some type of signed message." An example of a cryptographic authentication device is a YubiKey. | 2 | 80 |
| **Hashing** | Typically, you want to store a password in a unique and irreversible format, which is referred to as password hashing. Hashing is a one-way transformation of a password (different from encryption which is two- way) that turns it into a unique string called a password hash. One-way means it's practically impossible to turn that string into the original readable password. | 2 | 81 |
| **Password Storing** | This is typically how a user is authenticated without knowing the clear text password. Based on the users' input, the hash is calculated from the clear text password and compared with the hash stored on the system. | 2 | 81 |
| **Difficulty Factor** | This is a value that makes it intentionally more difficult to calculate the resulting hash to slow down any brute force guessing attacks. | 2 | 82 |
| **Input Transformation** | Can be used to obtain keys of a specifically required format. | 2 | 82 |
| **Irreversible Hashing Function** | The beauty of hashing the password is that after hashing, the value is completely unrecognizable, and the original input value cannot be retrieved even when the output value is known. | 2 | 82 |
| **Key Derivation functions** | Irreversible hashing function, input transformation, salt (and pepper( values, difficulty factor. | 2 | 82 |
| **Key Derivation Functions (KDF)** | Main characteristics: Irreversible Hashing Function, Input Transformation, Salt (and Pepper) Values, Difficulty Factor | 2 | 82 |
| **Key Stretching** | Input transformation can be used to obtain keys of a specifically required format. This allows us to create longer, more difficult to brute force keys from potentially weak, shorter passwords. We call this technique key stretching. | 2 | 82 |
| **Salt (and Pepper) Values** | The salt value is a randomly generated string of characters that is added to each password before hashing it and is stored next to the hashed value. Even more secure is a pepper, which is, in essence, a salt that is kept secret and stored in a secure location. | 2 | 82 |
| **Confusion** | The algorithm renders an output very different from the input | 2 | 84 |
| **Diffusion** | Means that it generates widely different outputs for closely matching inputs. | 2 | 84 |
| **Hashing Algorithm** | It is recommended to use an algorithm that has been available for a long time and well tested. Currently good choices are PBKDF2, SHA256, SHA512 bcrypt, scrypt, and the newer Argon2, which won the Password Hashing Competition in 2015. | 2 | 84 |
| **Password Hash Strength** | 4 factors:  1. Quality of Key Derivation Function  2. Password & Derived Key Length  3. Character Set Support  4. Difficulty Factor | 2 | 84 |
| **Difficulty Factor** | The number of passwords tries per second an attacker is able to launch against a hash depends on the number of CPU cycles the hashing function uses to compute the hash of a password. The more tries per second, the sooner all possible combinations can be tested. | 2 | 85 |
| **Key derivation function: PBKDF2** | DK = PBKDF2(PRF, Password, Salt, c, dkLen)  PRF: a pseudorandom function to 2 parameters with output length hLen such as HMAC-SHA3. C: cost factor or number of iterations, and dkLen: derived key length. | 2 | 86 |
| **PBKDF2** | The function generates a Derived Key (DK) based on five inputs:   1. A Pseudo-Random Function (PRF) that is used for hashing. The chosen hash function to be used to form the output of the KDF. A sample PRF that is approved by NIST is HMAC-SHA3, but any secure one-way function will do. 2. The password from which the derived key should be generated. 3. The salt, a random sequence of bits. As outlined by NIST, the salt should be chosen randomly and has to be at least 32 bits in length. 4. A cost factor (c) that defines the amount of iterations for each calculation. NIST specifies the iteration count should be as high as the verification server would allow. Typically, at least 10,000 iterations should be used, preferably more. 5. The desired length of the derived key in bits. | 2 | 86 |
| **MD5** | When downloading software from internet can verify the integrity of the DL | 2 | 87 |
| **MD5** | MD 5 is a cryptographic hash algorithm which produces a hash value in hexadecimal format. This algorithm has serious weaknesses as it known to have hash collisions. A collision happens when two unique plaintext hash to the same hash value. | 2 | 87 |
| **SHA** | Secure Hash Algorithm . There are several iterations: SHA, SHA1, SHA2 and SHA3. Passwords saved in online databases are typically stored as SHA hash. | 2 | 88 |
| **LM (Lan Manager)** | LM or Lan Manager hash was introduced by Microsoft in 1980 and has multiple weaknesses. The LM Hash is typically stored in the system in the SAM/NTDS database. Two severe weaknesses are: a maximum password length of 14 characters and passwords are converted to uppercase. Example Hash: 299BD128C1101FD6 | 2 | 89 |
| **LM hash** | LM or Lan Manager hash was introduced by Microsoft in 1980 and has multiple weaknesses. The LM hash is typically stored in the system in the SAM/NTDS database. 2 severe weaknesses are a maximum password length, or 14 characters and passwords converted to uppercase. | 2 | 89 |
| **NTLM (NTHash)** | The successor of LM. NTLM is challenge and response authentication mechanism which uses messages to authenticate a certain client. You also have NTMLv2 which is stronger than NTMLv1. Example Hash:eed224b4784bb040aab50b8856fe9f02 | 2 | 89 |
| **Rainbow Tables** | A list of known hashes | 2 | 90 |
| **Nonce** | A nonce is a random number of 16 bytes. Using a nonce allows the two parties to perform authentication without having to send the password (cleartext or encrypted) over the network. | 2 | 91 |
| **NTLMv2** | When Kerberos is not available, windows will fall back to NTLMv2. Authentication is a 2-party: the client and the server and it takes 3 steps, Negotiate, challenge and response. Within a windows domain, it is vulnerable to relay attacks. | 2 | 91 |
| **Collection #1 Breach** | Jan 2019. Leaked on 'MEGA'. Largest dump of user credentials known so far. | 2 | 92 |
| **LinkedIn Breach** | June 5, 2012, +-6.5 million passwords | 2 | 92 |
| **Password dumps** | Password cracking requires the attacker to possess a list of hashed passwords. Often, these passwords are obtained from the internet in so-called password dumps. | 2 | 92 |
| **Password Cracking** | Is the process of trying to guess or determine plaintext passwords, given only hashed passwords. The process does not actually break the encryption, it mimics the actions that would take place if a user tried passwords until guessing the right one. The cracking operation is usually performed offline against a recovered password file. | 2 | 93 |
| **Password Cracking method** | Using a password hashing tools like Mimikatz or hashcat, determine the encryption algorithm used, create a list of possible passwords, encrypt each password in the list, and determine whether there is a match with the collected hashes. | 2 | 94 |
| **Brute force attack** | The most powerful cracking method. Enumerate all possible combinations, slowest attack, 100% success rate given enough time. | 2 | 96 |
| **Dictionary attack** | Enumerate all entries from a dictionary or wordlist, fastest attack, only effective against weak passwords. | 2 | 96 |
| **GPU Acceleration** | Hardware like Graphics Processing Units (GPU's) can be effectively used to gain a significant speed improvement. | 2 | 96 |
| **Hybrid attack** | Extends dictionary attack with numerals and symbols, combines effectivity of brute force with speed of dictionary attack | 2 | 96 |
| **Password Cracking: Specialized Attacks** | 4 general methods: The Dictionary Attack, The Hybrid Attack, The Brute Force Attack, The Pre-Computation Attack | 2 | 96 |
| **Pre-Computation Attack** | Aka Rainbow Table Attack. computing hashes of possible passwords and storing the results in a database or table, the CPU time can be invested at times when there is plenty of processing power available. By the time the real cracking needs to be done, matching hashes with passwords is only a matter of searching through the pre-computed tables, which requires more memory, but takes only a fraction of the time needed to brute force a hash. The files containing the pre-computed password hash values are called "rainbow tables". | 2 | 97 |
| **GPU Acceleration – Speeding cracking process** | Password cracking is highly suitable for applying data parallelism. This means the process of calculating hashes can be split up over large amounts of processing units (CPU cores). Highly specialized password cracking can be done using tools that make use of the larger number of cores presents on hardware like GPUs, with OpenCL or CUDA APIs. | 2 | 98 |
| **Password Cracking Expedition** | Suitable for applying data parallelism/GPU Acceleration, Rainbow Tables | 2 | 98 |
| **Rainbow Tables** | Rainbow tables is the name given to the files that are produced by pre-computing password hash values and storing the data in an optimized manner to reduce the amount of disk space needed. After pre-computing the hash values, the cracking tool has to search only the tables for a given hash value to determine the corresponding plaintext password. | 2 | 98 |
| **Rainbow tables –** | Enables the attacker to trade CPU or GPU cycles for memory, As password cracking comes with a high computational cost, it can prove beneficial to calculate hashes in advance. The use of pre-computation attack can immensely speed up the cracking process. | 2 | 98 |
| **Speeding cracking process: Password Policy** | **Enforce a Strong Password Policy:**  • Force length > 8 characters  • Check for recognizable words or number sequences  • Block after x failed attempts  • Force change in case of (suspected) breach  **DO NOT:**  • Truncate passwords  • Allow password hints  • Force specific composition rules  • Force periodic password changes  • Save passwords in clear text | 2 | 99 |
| **Strong password: DO** | Force length to >8 characters, check for recognizable words of numbers of sequences, block after x failed attempts, force change in case of (suspected) breach. | 2 | 99 |
| **Strong password: DO NOT** | Truncate passwords, allow password hints, force specific composition rules, force periodic password changes, save passwords in clear text | 2 | 99 |
| **Fighting pre-computational attacks** | Can be fought by preventing hashes from being exact representations of passwords or “salting and Peppering” | 2 | 101 |
| **Peppering** | Additional KDF ensures hashes cannot be reproduced, peppers are unique per application and secret. | 2 | 101 |
| **Salting** | Additional input for KDF ensures output is different for different salts. Salts are unique per password, but not secret | 2 | 101 |
| **Salting example** | Please see book | 2 | 102 |
| **Password Cracking Tools** | Hashcat, Mimikatz | 2 | 103 |
| **Hashcat** | Hashcat is a password cracking tool that takes advantage of the potential hardware acceleration provided by parallelizing the workload over multiple cores. 4 core attack modes: Dictionary Attack - This attack is also known as a wordlist attack. The attack is very simple: Go through a given list of words from a text file and try each one as a password candidate. Combinator Attack - The combinator attack builds on the dictionary attack by concatenating two strings from the provided wordlist to each other. Brute-force Attack & Mask Attack - The brute-force attack tries each possible combination from a given Keyspace. It is the most straightforward attack but takes the longest, as it will test every possible combination. Mask - It adds the option to provide a mask on which password guesses will be based. This effectively reduces the possible Keyspace to a smaller one, which makes the cracking much faster. Hybrid Attack - Hybrid Attack A hybrid attack is a variant of the combinator attack. Instead of concatenating two strings from a given wordlist, the attack will combine the entries from the wordlist on one side and add the result of a mask attack on the other side. | 2 | 104 |
| **Mask** | A mask is a string that defines the Keyspace in which hashcat should look for the passwords. | 2 | 105 |
| **crypto** | Mimikatz module: allows you interact with the cryptographic system library. | 2 | 106 |
| **Kerberos** | Mimikatz module: exploits the Microsoft Kerberos API to create a golden ticket. | 2 | 106 |
| **lsadump** | Mimikatz module: can be used to dump system data from the windows local security authority. | 2 | 106 |
| **Mimikatz** | A feature-rich post-exploitation tool that can be used to extract plaintext passwords, PINs, hashes, and Kerberos tickets from system memory on Windows machines, as well as exploit several vulnerabilities and apply multiple credential-gathering techniques. Main feature modules: 1. sekurlsa 2. Kerberos 3. crypto 4. Lsadump. It can be used as a standalone tool, has a PowerShell version, and is also integrated in Metasploit and meterpreter modules | 2 | 106 |
| **sekurlsa** | Mimikatz Module: this module extract credentials from the Lsass (Local security authority system service) memory. | 2 | 106 |
| **Multi-factor authentication** | Is a method of authentication in which access is only granted after being presented with more than one authenticator. Often MFA combines a shared secret (something you know) with an OTP token (something you have) or Biometrics (something you are). | 2 | 108 |
| **Adaptive authentication** | In an adaptive authentication mechanism, the party that is requesting access will need to provide one or more authenticators, depending on the context of their request and the sensitivity of the resource they are attempting to access. | 2 | 109 |
| **CIS Controls** | "This consensus document of 20 crucial controls is designed to begin the process of establishing that prioritized baseline of information security measures and controls. The consensus effort that has produced this document has identified 20 specific technical security controls that are viewed as effective in blocking currently known high-priority attacks, as well as those attack types expected in the near future. Fifteen of these controls can be monitored, at least in part, automatically and continuously. The consensus effort has also identified a second set of five controls that are essential but that do not appear to be able to be monitored continuously or automatically with current technology and practices. Each of the 20 control areas includes multiple individual sub-controls, each specifying actions an organization can take to help improve its defenses" | 2 | 114 |
| **NIST Framework** | To reduce the risk of cyberattacks on election systems, the National Institute of Standards and Technology (NIST) has released draft guidelines that provide a road map to help local election officials prepare for and respond to cyber threats that could affect elections. This framework was published in 2014 but it still contains relevant information to protect your infrastructure from cyber-attacks. This framework is a risk-based approach to cyber security and contains the framework core, implementation tiers, and framework profiles. | 2 | 114 |
| **Security Frameworks** | Provide guidance for organizations such as “CIS Center for Internet Security”, “NIST National institute of Standards and Technology”, and “MITRE ATT&ACK" on prioritizing their most critical risks. | 2 | 114 |
| **MITRE Framework** | "MITRE ATT&CK™ is a globally-accessible knowledge base of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community." | 2 | 115 |
| **CSC (Critical Security Controls)** | We need priorities and someone to take a stand and provide the industry with a set of real priorities for defense | 2 | 116 |
| **CIS Guiding Principles** | Defenses should be automated where possible and periodically or continuously measured using automated measurement techniques where feasible. To address current attacks occurring on a frequent basis against numerous organizations, a variety of specific technical activities should be undertaken to produce a more consistent defense. | 2 | 117 |
| **CIS: Principles** | * Defenses should be automated where possible and periodically or continuously measured using automated measurement techniques where feasible. * To address current attacks occurring on a frequent basis against numerous organizations, a variety of specific technical activities should be undertaken to produce a more consistent defense. * Root cause problems must be fixed in order to ensure the prevention or timely detection of attacks. * Measures should be established that facilitate common ground for measuring the effectiveness of security measures, providing a common language to communicate about risk. | 2 | 117 |
| **CIS Guiding Principles (2)** | Root cause problems must be fixed in order to ensure the prevention or timely detection of attacks. Measures should be established that facilitate common ground for measuring the effectiveness of security measures, providing a common language to communicate about risk. | 2 | 118 |
| **CIS Controls summary** | There are 5 critical rules: Offense informs defense, prioritization, measurements and metrics, continuous diagnostics and mitigation, and automation. | 2 | 119 |
| **CIS Controls** | The controls are split into three groups:   1. Basic (Controls 1–6) 2. Foundational (Controls 7–16) 3. Organizational (Controls 17–20)   Top 5 things to do:   1. Application Control 2. Patching Apps 3. Patching OS 4. Reduction of Admin Privilege 5. Secure Baseline Configuration | 2 | 120 |
| **CIS Application Control** | The measures for this control focus on discovering new software applications on systems that are unauthorized. | 2 | 122 |
| **CIS Control #2 (example)** | Business goal of this control: Only authorized software should be installed on the organization's computer systems | 2 | 122 |
| **CIS Control #2 Example** | “Inventory and Control of Software Assets”. The goal of this control is that only authorized software should be installed on the organization’s computers systems. | 2 | 122 |
| **CIS control #2 Core Evaluation Test** | Install a benign software application on 10 unauthorized devices on various portions of the organization’s network unannounced to see how long it takes for the software to be detected. | 2 | 123 |
| **CIS control #2 Measures** | 1. How may unauthorizes software applications are presently located on business systems within the organization (By business unit)? 2. How long, on average does it take to remove unauthorized applications from business systems within the organization (by business unit)? 3. What is the percentage of the organization’s business systems that are not running application control software that blocks unauthorized software application software applications (by business unit)? 4. How many software applications have been recent blocked from executing by the organization’s application control software (by business unit)? 5. How long does it take to detect new software installed on systems in the organization (time in minutes, by business unit)? 6. How long does it take to remove unauthorized software from one of the organization’s systems (time in minutes, by business unit)? | 2 | 124 |
| **NIST Cyber Security Framework** | The NIST framework is a risk-based approach to managing cybersecurity risk and includes 3 parts:   * The framework core: consists of 5 concurrent and continuous functions. “Identify, Protect, Detect, Respond, Recover” * Framework implementation tiers: contextualize risk for a specific organization and how these risks can be managed. “T1 Partial, T2 Risk informed, T3 Repeatable, T4 Adaptive” * Framework profiles: the profile is typically used a target and provides guidelines or us being used as a roadmap to further strengthen your security controls. | 2 | 125 |
| **NIST Framework Core: Detect** | Develop and implement appropriate monitoring capabilities, this means looking at certain events and alerting policies. | 2 | 126 |
| **NIST Framework Core: Identify** | Develop an organizational understanding to manage cybersecurity risk to systems, people, assets, data and capabilities. | 2 | 126 |
| **NIST Framework Core: Protect** | Develop and implement appropriate safeguards to ensure delivery of critical services, an example is to use perimeter filtering. | 2 | 126 |
| **NIST Framework Core: Recover** | Develop and implement appropriate activities to plan for resilience and restore capabilities | 2 | 126 |
| **NIST Framework Core: Respond** | Develop and implement appropriate activities to take action regarding a detected security incident. | 2 | 126 |
| **NIST Security Incident Lifecycle** | 5 functions: Identify, Protect, Detect, Respond, Recover. | 2 | 126 |
| **NIST Implementation Tiers** | When you are looking at your own organization, they help determine the extent to which cybersecurity risk management is informed by business needs and is integrated into the organization. 4 tiers:  **Tier 1: Partial**  **Tier 2: Risk Informed**  **Tier 3: Repeatable**  **Tier 4: Adaptive** | 2 | 128 |
| **NIST Implementation tiers: T1 Partial** | Certain controls are partially implemented, processes are not formalized and there is limited awareness of cyber security risks. This is the lowest tier within the NIST Cybersecurity Framework. | 2 | 128 |
| **NIST Implementation tiers: T2 Risk informed** | Risk management practices are formalized and approved by management. It has not been published organization wide and cyber risk awareness is limited to the organization level. | 2 | 128 |
| **NIST Implementation tiers: Tier 3 Repeatable** | Risk management practices are formally approved and expressed as policy. There is an organization-wide approach to manage cybersecurity risk. | 2 | 128 |
| **NIST Implementation tiers: Tier 4 Adaptive** | Adapts cybersecurity practices based on previous and current cybersecurity activities, including lessons learned and predictive indicators. this includes continues improvement. | 2 | 128 |
| **NIST Framework Profiles** | Within the framework profile, you align the functions, categories, and subcategories based on your business needs. Your profile is based on risk appetite and risk tolerance and, depending on your security profile, additional security controls can be implemented. | 2 | 130 |
| **MITRE ATT$CK** | Is a globally-accessible Knowledge base of adversary tactics and techniques based on real-world observations. The ATT$CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community. | 2 | 131 |
| **MITRE ATT$CK Tactics** | Are used to describe high-levels attack steps used by an adversary. | 2 | 131 |
| **MITRE ATT$CK Techniques** | How a certain tactic is executed is described by a variety of techniques. | 2 | 131 |
| **MITRE ATT&CK Matrix** | Tactics, Techniques, Sub-techniques | 2 | 132 |
| **MITRE ATT$CK: Matrix** | It is pretty extensive and is divided in sub-techniques, tactics, and techniques. Please see book for more info | 2 | 132 |
| **MITRE ATT$CK: Technique details** | It gives information about the techniques being used by actual adversaries. The technique valid account is categorized as initial access and explains how attackers are using compromised credentials. In addition, ATT$CK also describes the data sources required to detect these types of techniques. | 2 | 133 |
| **MITRE ATT$CK: Mapping to known adversaries** | Knowing which adversaries are known to abuse the technique (threat intelligence), you can map this technique toward a known campaign and look for specific details within your environment. | 2 | 134 |
| **Security Frameworks - Other Frameworks** | While mappings to other standards are rarely ever a one-to-one match, clearly, the ideas expressed in these guides often represent common ideas. Defenses should focus on addressing the most common and damaging attack activities occurring today and those anticipated in the near future. Enterprise environments must ensure consistent controls across an enterprise to effectively negate attacks. | 2 | 135 |
| **Data Leakage** | In most cases, data leakage is categorized as a security incident that result in data being access in a security incident that results in data being accessed in an unauthorized way. One of the typical examples is exposing data on publicly available sites by using external sharing platforms | 2 | 141 |
| **Data Leakage –**  **Data Breach** | In most cases, data leakage is categorized as a security incident that results in data being accessed in an unauthorized way. One of the typical examples is exposing data on publicly available sites by using external sharing platforms. | 2 | 141 |
| **Data Leakage –**  **Data Loss** | Data loss occurs when data is being corrupted, deleted or made unreadable in any way | 2 | 141 |
| **Data Leakage –**  **Cost** | Value of lost data, cost of continuing without, cost of recreation, reputational damage. | 2 | 142 |
| **Data Leakage –**  **Prevention** | Backups, Data Redundancy. | 2 | 142 |
| **Data Leakage –**  **Recovery** | Data Recovery labs, dependent on storage mediums. | 2 | 142 |
| **Data Leakage –**  **Types** | Procedural, (un)intentional, failure, disaster, crime. | 2 | 142 |
| **Ransomware** | Ransomware is a variety of malware that, upon infecting a system, encrypts all the data on the hard drive it can find. Subsequently, the user of the system is prompted with a screen explaining what has happened and a ransom note requiring them to pay a certain amount of money, often in the form of a cryptocurrency like Bitcoin to get their data back. | 2 | 143 |
| **Data Loss Prevention strategies –**  **Access Control** | Need-to-have, Prevents Accidental deletion | 2 | 144 |
| **Data Loss Prevention strategies –**  **Strategies** | * Redundancy * Backups * Access Control. | 2 | 144 |
| **Data Loss Prevention strategies –**  **Backups** | On-site, Off-site, Automated. | 2 | 144 |
| **Data Loss Prevention strategies –**  **Redundancy** | In-house, Cloud infrastructure. | 2 | 144 |
| **copy-on-write** | Some filesystems built specifically for scalability and robustness, such as btrfs or zfs, implement a technique called copy-on-write, which avoids in-place changes to the data altogether by writing changes into new blocks, adapting the metadata to point to the new data and subsequently delete the old data. | 2 | 145 |
| **Error-Correction Code (ECC) memory** | Adds some extra bits to each data unit, enabling the system to detect if a bit in the data has been changed accidentally and to correct the issue. It is often used in computer systems that do not tolerate memory corruption, such as financial computing. | 2 | 145 |
| **Journaling** | Redundant data can also be built into the filesystem of the computer. An example of this is journaling filesystems that keep a log of the changes they made and intend to make to the system. In the event of a crash during a write or delete operation, the system can simply reconstruct a valid state by repeating the steps as kept in the journal logbook. | 2 | 145 |
| **Parity Bits** | A bit added to a piece of binary code to indicate if the total number of 1-bits is even or odd. This way, if a bit has changed from a 1 to a 0, or vice versa, the system can detect it based on the parity bit and correct the issue. | 2 | 145 |
| **RAID configuration** | "Redundant Array of Inexpensive Disks" (former definition) or "Redundant Array of Independent Disks" (recent definition). It is a way of combining multiple physical hard drives into one or more logical drives on the system, allowing data to be replicated across disks, speed up read/write performance by splitting data across drives (this technique is called "striping"), and add parity bits to the data for error detection. | 2 | 145 |
| **Redundancy** | A first step in preventing the loss of data is building redundancy into the system. In-House: -ECC memory -RAID configurations -Data redundant file systems (btrfs, zfs). Cloud: -availability -integrity -geo-replication. | 2 | 145 |
| **Redundancy: in House** | Error-Correction Code (ECC Memory) adds some extra bits to each data unit, enabling the system to detect if a bit in the data has been c hanged accidentally and to correct the issue. RAID configurations (Redundant Array of Independent Disks) it is a way of combining multiple physical hard drives into one or more logical drives on the system. Data redundant file systems (btrfs, zfs) or journaling filesystems that keep a log of the changes made and intend to make to the system. In the event of a crash during a write or delete operation, the system can simply reconstruct a valid state by repeating the steps as kept in the journal logbook. | 2 | 145 |
| **Striping** | Splitting data across drives | 2 | 145 |
| **Amazon DynamoDB** | The key-value store developed by Amazon for use in the backend of their own systems, as well as a service for Amazon Web Services customers. | 2 | 146 |
| **CAP Theorem** | CAP stands for "Consistency, Availability, Partition Tolerance." The theorem states that a distributed system can have, at most, two of the three properties; therefore, a choice has to be made. | 2 | 146 |
| **Cloud Redundancy** | Cloud storage solutions often offer replication between so-called "storage nodes." These are abstractions of physical storage devices in the data center on which the data is stored. When one node fails, the system will automatically retrieve the data from another node. | 2 | 146 |
| **Consistent Hashing** | Assigns each storage node a position in a ring of nodes. When a new key-value pair is added to the storage, its key is first hashed. The resulting hash corresponds to a position in the ring of nodes on which the data will get stored | 2 | 146 |
| **Distributed Systems** | Often, cloud systems enable the replication of data between multiple servers, data centers, or even geographical regions (called "geo-replication"). This means that cloud systems are highly distributed systems. | 2 | 146 |
| **Geo-Replication** | Replication between regions ensures data can be recovered even in the event of a regional disaster (e.g., an earthquake). | 2 | 146 |
| **Redundancy: cloud** | Solutions often offer replication between so-called “storage nodes”. There are abstractions of physical storage devices in the data center on which the data is stored when one node fails it will retrieve the data from another node. Availability, integrity, Geo-Replication | 2 | 146 |
| **Redundancy: Cloud – Data Availability** | Administrators can set a system parameter that forces replication on at least N nodes. This ensures there are always N copies of the data. | 2 | 146 |
| **Redundancy: Cloud – Data integrity** | When retrieving data, replication nodes can come to a consensus on the value of the data to ensure it wasn’t corrupted on a single node | 2 | 146 |
| **Redundancy: Cloud – Geo-Replication** | Replication between regions ensures data can be recovered even in the event of a regional disaster like an earthquake. | 2 | 146 |
| **Auto-failover groups** | Enable the administrator to define groups of nodes to automatically take over from the primary group in case of a large-scale error event. Often, this secondary group is located in another region to provide resilience against regional disasters, but this does not always have to be the case. | 2 | 147 |
| **Azure SQL Database** | Availability and integrity are configured by defining replication and failover groups. | 2 | 147 |
| **Preference List** | Subsequently, based on a configured parameter N, the data will be replicated to the next N–1 node in the ring. These nodes, responsible for storing that particular key, are called the preference list. | 2 | 147 |
| **Backup Methods** | * Full System Imaging * Incremental Backups * Differential Backups * Continuous Backups | 2 | 148 |
| **Continuous Backups** | Stores data automatically to a backup medium each time a change is made to the data. | 2 | 148 |
| **Differential Backups** | Only stores the data that has changed between the present and the last full system image. | 2 | 148 |
| **Full System Imaging** | The entire system is copied to a single image and stored on the backup medium. | 2 | 148 |
| **Incremental Backups** | Only the changed data since a previous point in time is changed. | 2 | 148 |
| **Backups: Common Pitfalls** | Keep backups on separate networks, Cloud storage also requires backups, Use different technologies, Frequently verify backup systems and backup data | 2 | 149 |
| **Data Recovery** | A procedure in which corrupted or unreadable data is recovered from a faulty storage medium.  1. Repair Storage Medium  2. Image Data to Another Medium  3. Logical Data Recovery  4. Repair Damaged Data | 2 | 150 |
| **Data Leakage** | It is when we lost control of our data | 2 | 151 |
| **Data Leakage –**  **Cost** | Business losses, reputational damage, fines. | 2 | 151 |
| **Data Leakage –**  **Insider threats** | Insider threat program, user access monitoring, third-party risk management | 2 | 151 |
| **Data Leakage –**  **Prevention** | Secure data storage, intrusion detection & prevention, Data exfiltration | 2 | 151 |
| **Data Leakage –**  **Types** | * Cost * Prevention * Insider Threats | 2 | 151 |
| **Data Leakage –**  **Why is it important?** | Business Losses, Reputational Damage, Fines | 2 | 152 |
| **Data Leakage –**  **Business losses** | Leaked trade secrets or sensitive data can lead to large business losses | 2 | 152 |
| **Data Leakage –**  **Fines** | Strict regulations apply to data security. Data breaches may lead to high fines | 2 | 152 |
| **Data Leakage –**  **Reputational Damage** | Data breaches involving sensitive (user) data can lead to reputational damage | 2 | 152 |
| **CCPA** | “The California Consumer Privacy Act” is a recently enacted law from 2018 that grants rights concerning privacy to Californian consumers and outlines requirements for organizations on how to process consumer data. | 2 | 153 |
| **GDPR** | Regulation (EU) 2016/619, also known as the “General Data Protection Regulation”, is a European law enacted in 2016 that outlines data protection measures that need to be taken when an organization collects and processes personal data. | 2 | 153 |
| **Breaches - Expensive** | British Airways, Equifax | 2 | 155 |
| **Data Leakage –**  **Prevention Strategies** | Secure Data Storage, Intrusion Detection, Exfiltration Detection | 2 | 156 |
| **DLP Storing Sensitive Data** | Digital: Encrypt sensitive data should always be stored with strong encryption, delete sensitive data when it is no longer needed. Physical: having a clean desk policy, shredding data. | 2 | 157 |
| **Secure Archival** | Sometimes, sensitive data is only used sporadically but cannot be deleted, as it may be needed in the future. In this case, the data needs to be securely archived. Secure archival means the files should be stored on a medium only privileged people within the organization can access, and it's always encrypted when not in use. Ideally, the archiving storage medium is not connected to any production networks or is even fully air-gapped from other systems. | 2 | 157 |
| **Secure Data Storage** | This means that when dealing with digital data, strong encryption should be used for storage. Equally as important is the secure deletion of sensitive data when it is no longer needed. Secure archival should be used for data that is only used sporadically. | 2 | 157 |
| **DLP Policies** | A policy defines where the tool must monitor for sensitive data, under which conditions it should intervene, and which actions need to be taken automatically, by the tool, when these conditions are fulfilled. | 2 | 158 |
| **Data Classification Labels** | Each piece of data must be assigned such a label to designate how sensitive it is and what the appropriate security requirements for the data are. | 2 | 159 |
| **DLP Policies: Data Classification Labels** | To properly function, DLP tools allow the definition of Data Classification Labels. Each piece of data must be assigned such label to designate how sensitive it is an what the appropriate security requirements for the data are. The owner of the data should assign the label to the data. If needed a DLP tool can often make suggestions to the data owner on which label to use depending on the contents of the data and whether any sensitive information has been found. | 2 | 159 |
| **Data At-Rest** | * Enable automatic encryption of sensitive data * Detect wrongly classified documents and suggest a higher (or lower) classification level where needed * Detect sensitive data stored in the wrong location and automatically alert security staff, move or delete the data if needed. | 2 | 160 |
| **Insider Threats** | According to the 2016 US State of Cybercrime Survey, insider incidents make up 27% of electronic crimes. | 2 | 160 |
| **Data In-Transit** | • Detect transmissions with sensitive data in real-time and take appropriate action  Potential actions include:  • Escalate to manager for approval  • Block transmission  • Alert security staff  • Remove sensitive data before transmission | 2 | 161 |
| **DLP Policies: Data In-Transit** | Detect transmission with sensitive data in real-time and take appropriate action such as escalate to a manager for approval, block transmission, alert security staff, remove sensitive data before transmission. | 2 | 161 |
| **Data Loss Prevention strategies –**  **Tools** | 1. DigitalGuardian 2. Forcepoint 3. Office 365 Data Loss Prevention | 2 | 162 |
| **DLP Tools** | DigitalGuardian, Forcepoint, Office 365 DLP | 2 | 162 |
| **Digital intrusion** | A way for advanced adversaries to steal your online data. Data that is kept offline is not accessible to a digital intruder. Most data have to be online, but archived data, can be stored on storage media that is not directly online thus limiting the impact of a ransomware attack. | 2 | 163 |
| **Host IDS** | Encompasses logging activity on the hosts themselves (configuration changes, unusual behavior or activity, executable files being run, and so on), combined with the active detection of malware signatures on the system using anti-malware products. | 2 | 164 |
| **Intrusion Detection and Prevention** | Intrusion detection can help identify adversaries that have breached the system and enable the organization to take action before any data can be exfiltrated. | 2 | 164 |
| **Intrusion Detection and Prevention** | We have 2 types of Intrusion Detection Systems (IDSs): Network IDS – Online: is done by real-time traffic analysis, and applies basic detection rules, and Offline: stored traffic analysis, allows for more thorough analysis. Network IDS-Hosts: Logs, and Malware Detection | 2 | 164 |
| **Network IDS** | is done by monitoring traffic on the organization's network by passively looking at the packets that are passing through. Online & offline network IDS. | 2 | 164 |
| **Data Exfiltration (Data Theft)** | 1. Search for interesting files 2. Collect and prep interesting files 3. Exfiltrate interesting files | 2 | 165 |
| **Data Exfiltration - Defense** | Prevent, Detect | 2 | 167 |
| **Data exfiltration: Detection** | A system-wide search generates a lot of activity on the system being searched. Monitoring for searches through filesystems in not trivial, through there will be several false positives such as antivirus scanners, search indexers, backup programs, etc. Access to network shares can however be monitored (Event ID 5140) – “A network share object was accessed”). Look for repeated audit failures from one source. | 2 | 167 |
| **Data Exfiltration: Prevention** | Ensure the organization knows what data it possesses and that it is correctly classified. Limit user access only to data they should be allowed to access (“need to know”). Next to limiting user access to data, also consider what type of data you store where.. this includes network segmentation, but also even considering storing some data offline. | 2 | 167 |
| **Insider Threats** | Insiders in the organizations are in a position to easily access sensitive assets and can form a significant threat to your organization’s data. | 2 | 169 |
| **Insider Threats: Humans** | In order to mitigate insider threats, your organizations need an “Insider Threat Program” and it begins with raising awareness about its existence within the organization. This serves as a deterrent for insiders to leak information. Research have shown that positive incentives can be used to encourage employees to act in the interest of the organization. Individuals who pose an elevated risk, should be placed under increased monitoring. | 2 | 170 |
| **User Activity Monitoring** | Tools enable the organization to keep track of internal end-user activity when using the organization's infrastructure. Many regulations (HIPAA, SOX, GDPR, etc.) require organizations to implement some degree of UAM for auditing purposes. | 2 | 171 |
| **Digital watermarking** | Can be used to embed markers in data to track the data source and ownership. | 2 | 172 |
| **Document Watermarks** | Ensure that the source of a document can always be traced. They can be visible or invisible, depending on the use case. | 2 | 172 |
| **Printer Dots** | Tiny, yellow dots embedded on each printed page. They are almost invisible to the naked eye but can be used to identify the source of a printed page. | 2 | 172 |
| **Secure centralization** | Secure centralization is the protection of logs in transit with encryption. Some examples include stunnel (SSL encryption), OpenSSH (with SSH), and IPsec. VPNs offer this capability and can be used if log encryption is needed. However, many logging devices (such as network devices) might not support encryption "end to end", and, thus, this security measure becomes difficult. If possible, choose a log agent or service that supports TLS encryption. | 2 | 174 |
| **Mobile Device Security** | Android vs iOS, Security features of each | 2 | 177 |
| **Android vs. iOS** | Most important: Android - open, iOS - closed | 2 | 179 |
| **Android Security** | Previously viewed as more focused-on functionality than security. Since Android Oreo significantly focused on security. More built-in security and trying to be similar to iOS. Still open operating system but more embedded security features. Now focused on multiple layers of security, right out of the box. | 2 | 180 |
| **Android Security Features** | Worldwide Active Scanning through Google.   1. Application security 2. Active scanning 3. Android Pay 4. Virtual Sandbox 5. Device Manager 6. Built-in encryption | 2 | 181 |
| **Android Security features: Active Scanning** | When applications run, they are monitored to look for suspicious activity. | 2 | 181 |
| **Android Security features: Android Pay** | Secure payment options do not expose credit card data. | 2 | 181 |
| **Android Security features: Application Security** | Applications are rigorously tested for security before there are available for download | 2 | 181 |
| **Android Security features: Built-in Encryption** | Cryptographic functionality is built in to protect data at rest and in transit. | 2 | 181 |
| **Android Security features: Device manager** | Allows for locating and securely wiping remote devices. | 2 | 181 |
| **Android Security features: Virtual Sandbox** | Critical Data is sandboxed or isolated from applications to minimized exposure. | 2 | 181 |
| **Android - What you need to know** | Open Handset Alliance, led by Google. Manufacturers manipulate the OS, add or remove content and control software features. Rapid hardware development. Wide disparity of hardware and software features. | 2 | 182 |
| **Android important characteristics** | Open Handset Alliance, led by Google. Manufacturers manipulate the OS add or remove content and control software features. Rapid hardware development. Wide disparity of hardware and software features. | 2 | 182 |
| **Android Fragmentation** | Multiple competing hardware manufacturers whose sole profit is hardware sales. Manufacturers are free to customize Android source: incompatibilities. Multi-chain update cycle is complex: issuing updates. Reduces the platform's effective value; Common complaint from Android users. | 2 | 183 |
| **Android Security Fix Process** | it is helpful to understand the process adopted by the Open Handset Alliance (OHA), which vendors and MOs must go through to get a security update to an end user. 3 steps. | 2 | 184 |
| **Android security process** | First a flaw is disclosed to the OHA, publicly or privately. When a fix is made available, notice is returned to the OHA. Second, each vendor must test and incorporate the software update into the Android OS. After the update is prepared, it is shared with the MOs. After the MO completes testing and integrations of the fix, it can be made available to Android users as an OTA update or as a download on a website. | 2 | 184 |
| **Open Handset Alliance (OHA)** | Android Security Fix Process, 3 steps. | 2 | 184 |
| **Apple iOS Security** | * iOS is designed with security at its core. * Closed model unless someone jailbreaks the phone. * Security is integrated into the entire architecture. * Most security is transparent to the user. * Key security (that is, encryption) cannot be disabled of turned off. | 2 | 185 |
| **Apple iOS security features: Application Security** | Creates a platform to enable third party applications to run in a secure environment. | 2 | 186 |
| **Apple iOS security features: Encryption and Data protection** | Protect critical data id the device is stolen, or unauthorized access is attempted protecting data at rest. | 2 | 186 |
| **Apple iOS security features: Network Security** | Complements data protections and provides security for data in transit. | 2 | 186 |
| **Apple iOS security features: System Security** | Both hardware and software contain security features that support the architecture and applications that run on the system. | 2 | 186 |
| **Apple iOS security features: Apple Pay** | Enables you to make secure payments | 2 | 187 |
| **Apple iOS security features: Device control** | Enables management of device to include secure wipe. | 2 | 187 |
| **Apple iOS security features: internet services** | Provides cloud-based services for backup and capability for communications. | 2 | 187 |
| **Apple iOS security features: Privacy controls** | Enables device information to be shared with external services, such as with location services. | 2 | 187 |
| **Apple important characteristics** | * Massively popular platform, common for enterprise-owned and user-owned deployment. * Most restrictive of the 2 major platforms. * Apple end-to-end ownership model of hardware and software. * Apple forbids mobile operators (MO) software. * Minor software differentiation between iPhone, iPod, and iPad. Constant hardware evolution and improved performance. * Hardware capabilities frequently dictate software feature capabilities. | 2 | 188 |
| **Mobile Problems and Opportunities** | * Resisting mobile devices contributes to organizational risk and exposure. * Users leverage mobile phones and tablets with or without enterprise support. * Mobile devices introduce new problems and opportunities for organizations * End users see mobile devices as sophisticated, cutting- edge, desirable technology * From a security perspective, mobile devices lack the security functions we expect in modern devices: * Commonly lacking functionality needed for secure use * Immature or hampered enterprise controls * Organizations have sound motivators to leverage mobile devices for many industries | 2 | 188 |
| **Mobile devices threats: TH 1** | Loss of controls and visibility with BYOD (Bring your own device) | 2 | 190 |
| **Mobile devices threats: TH 2** | Always-on devices through multiple interfaces such as Bluetooth leaving the door opened for potential attacks. | 2 | 190 |
| **Mobile devices threats: TH 3** | Device patching and extended vulnerability periods. | 2 | 190 |
| **Mobile devices threats: TH 4** | Device theft and loss. | 2 | 190 |
| **Mobile Threats** | 1. Loss of control and visibility with BYOD  2. Always-on devices through multiple interfaces  3. Device patching and extended vulnerability periods (Android)  4. Device theft and loss | 2 | 190 |
| **Mobile device stolen threat: Loss impact.** | An attacker can do many things to a stolen device such as access device resources, extract data, synchronize device, jailbreak/unlock/root, access stored authentications credentials, backdoor device. | 2 | 191 |
| **Stolen Device Threat** | * Access device resources * Extract data * Synchronize device * Jailbreak/unlock/root * Access stored authentication credentials * Backdoor device | 2 | 191 |
| **Mobile Device Management (MDM)** | A system that companies use to control, secure, and enforce policies on the mobile devices of their employees. Security-wise, its main goal is to protect company data and intellectual property against theft. MDM solutions usually require software on both the device and the backend infrastructure | 2 | 192 |
| **Mobile device stolen threat: :Loss device reporting** | Time sensitive actions, greater risk if not timely reported, encourage quick reporting. | 2 | 192 |
| **Mobile device stolen threat: Mobile device management** | Protect company data, enforce policies, locate devices remotely, updates and installs., diagnosis and troubleshooting, secure communications. | 2 | 192 |
| **Unlocking, Rooting and Jailbreaking.** | Overcoming restrictions and bypassing intended controls as the process of creating unrestricted devices. | 2 | 193 |
| **Unlocking, Rooting, and Jailbreaking** | "unrestricted devices" | 2 | 193 |
| **Mobile Malware** | Mobile device malware is a growing threat for devices:   * New opportunities for exploiting users * New opportunities for attacker financial gain Mobile malware is a small fraction of the overall malware threat * Growing at an alarming rate Platform exposure varies significantly | 2 | 194 |
| **Mobile user credential theft.** | Mobile phones are increasingly relied upon for two-factor authentication via SMS primarily for banking applications and related financial activities. If your credentials are stolen, by an attacker, your whole online persona could be compromised as well. | 2 | 195 |
| **The Zitmo (Zeus In The Mobile)** | The Zitmo (Zeus In The Mobile) malware is an evolution of the ZeuS Windows-based malware, targeting BlackBerry, Android, Windows Mobile, and Symbian phone users. Zitmo-infected devices enable an attacker to control SMS and phone functionality, blocking inbound and outbound calls, and silently intercepting SMS messages. Combined with a ZeuS-infected system, Zitmo enables an attacker to bypass the mTAN authorization function by intercepting and redirecting the mTAN value to an attacker. | 2 | 195 |
| **User Credential Theft** | Mobile phones are increasingly relied on for two-factor authentication via SMS. Primarily for banking applications and related financial activities. Zitmo variant of the ZeuS trojan controls SMS and phone functionality, blocking select calls and intercepting SMS messages. Works with the PC variant of the ZeuS trojan for effective banking control bypass. | 2 | 195 |
| **ZeuS Trojan** | Malware that targets blackberry, android, windows mobile and Symbian phone users. It enables an attacker to gain control of SMS from infected mobile devices to intercept authentication pin numbers. | 2 | 195 |
| **Mobile malware delivery methods** | Official app store repositories, typically short lived. Third party app stores repositories, primarily Android device or jailbroken iPhone. Malicious websites for direct download installation. Direct victim targeting through email, SMS, and MMS. | 2 | 196 |
| **Android Malware** | Highly targeted among four major mobile device vendors. Platform accommodates silent SMS delivery, untrusted applications, third-party, applications stores. Premium rate/short code SMS for quick financial gains. Easy for attackers to repackage legitimate applications with malware. Significant market share. Platform fragmentation creates extended lifetime for exploit applicability. | 2 | 197 |
| **iOS Malware** | * Platform security prevents unauthorized executables from running * Small number of early malware samples targeted jailbroken devices * No option to automatically send SMS * Handful of questionable applications retrieving sensitive data that were not rejected • OpenFeint, Path, Twitter, and Facebook retrieval and storage of contacts * Storm8, mogoRoad phone number retrieval * Phishing due to the consistency of the iOS UI * Users are often nagged for their Apple ID password * Exploited by websites pretending to be a native dialog | 2 | 198 |
| **Buffer overflow: Normal stack memory allocation** | This is when the attacker tries to determine how many characters are allowed in memory or hold text. | 2 | 102 103 |
| **Vulnerability Assessments** | Attackers use many different tools and techniques to identify vulnerable systems. These same tools and techniques must be used by security teams to remediate any findings prior to a compromise. Vulnerability assessments include ways to help expose misconfigured systems, vulnerable features, missing patches, incorrect permissions, and many other areas of concern. | 3 | 3 |
| **Perimeter Security** | Focuses on security controls at the borders of the network, such as firewalls, proxy servers, and anti-DDoS appliances. Typically, this is referred to as a mitigation layer between your organization and the internet or other untrusted networks. | 3 | 5 |
| **Vulnerability assessment at scale** | Modern enterprises have enormous numbers of systems, it can be a challenge to know where they are. New vulnerabilities are discovered every single day. Vulnerabilities have resulted in many recent large-scale breaches. To defend themselves at scale, enterprise organizations must have a consistent, automate vulnerability management program. | 3 | 6 |
| **Penetration Testing** | Actualizes risk to demonstrate the business implications | 3 | 8 |
| **Security Audit: Definition** | Asses the adequacy of controls and evaluate compliance. | 3 | 8 |
| **Vulnerability** | Flaw or weakness in a system that can be exploited | 3 | 8 |
| **Vulnerability Assessment** | Description and analysis of vulnerabilities in a system. When performed properly, the most value-added security service in which an organization can invest is vulnerability assessment. Must be able to expose risk with a similar degree of confidence to pen testing, across a larger environment, without exploitation, and while somehow limiting the incidence of false positives. | 3 | 8 |
| **Vulnerability Management** | Ongoing, repeatable processes for identifying, remediating, or accepting risk | 3 | 8 |
| **Vulnerability:**  **Definition** | Flaw or weakness in a system that can be exploited | 3 | 8 |
| **Vulnerability Assessors** | It is not penetration testing. Many consider vulnerability assessment to be a lesser from of penetration testing. Many think the vulnerability assessors are the people who run the scanners, but they are not. | 3 | 9 |
| **Penetration Testing** | Their goal is to report exploitable vulnerabilities under controlled circumstances. Actualizes risk to demonstrate the business implications. | 3 | 10 |
| **Vulnerability Assessments** | Increased precision to produce the same. Qualitative and quantitative assessment metrics. Scoring systems and triage at scale. To achieve this, VA MUST be more than running a scan, without breaking in. | 3 | 10 |
| **The Vulnerability Assessment Framework (VAF)** | Introduce the vulnerability assessment methodology, seven phases:   1. Engagement Planning 2. Threat Modeling 3. Discovery 4. Scanning 5. Validation 6. Remediation 7. Reporting   Validation, remediation, and reporting can be an iterative loop. Reporting then loops back to engagement planning. | 3 | 11 |
| **Vulnerability Assessments: Modules** | The work flow we follow through the methodology process reflects the modularity:   * Output from module > database import * Database queries > inputs to next module * Reporting module > ticketing * Tickets > vulnerability management and mitigation * Close the loop back to the test team process * Retest where necessary | 3 | 12 |
| **Engagement planning: VA Step 1** | This is a crucial step, often skipped or not done correctly. It consists of planning such as rules of engagement scoping, process, procedures and checklists, access and visibility, logistics, and resource management. | 3 | 14 |
| **Vulnerability assessments steps**  **VAF** | Step 1 Engagement planning,  Step 2 Intelligence and Threat Modeling  Step 3 Resource management  Step 4 Scanning  Step 5 Validation  Step 6 Remediation  Step 7 Reporting. See slides on the book. | 3 | 14 |
| **Intelligence and Threat Modeling:**  **VA Step 2** | Gathering intelligence, identify prioritize most likely threats to the organization. Most likely targets/critical assets/functions. How likely are the mitigation controls able to detect/respond. Level of sophistication and resourcing required. | 3 | 15 |
| **Threat Model** | Helps identify threats and prioritize the defenses most likely to succeed against the most probable attacks. The threat model guides the entire defensive program for an organization. | 3 | 15 |
| **Discovery: VA Step 3** | This phase validates documentation and network diagrams  Input: Domain names, list of host names, subnets, CIDR ranges.  Goal: which subnets are routable, which hosts are accessible, visibility available.  How: Network is mapped using various active and passive methods.  Result: identify individual live hosts. | 3 | 16 |
| **Scanning: VA Step 4** | Major considerations are scanner placement and impediments to testing, such as firewalls and load balancers. Various strategies and techniques are used to scan endpoints and network elements across the enterprise. | 3 | 17 |
| **Validation: VA Step 5** | Assign a confidence value and validate potential vulnerabilities. Some may require exploitations, pillaging, or pivoting to ascertain impact. Meet compliance and audit requirements. Major issues we face when performing enterprise assessments include collaboration tools and data management. | 3 | 18 |
| **Remediation: VA Step 6** | Changes must be prioritized based on risk assessment and cost-benefit analysis that has leadership approval. Remediation loops back into enterprise functions such as change management, configuration management, architectural changes, ticketing system, framework to tie into other systems. To assign risk and priority ratings to vulnerabilities | 3 | 19 |
| **Reporting: VA Step 7** | Assign risk and priority ratings to confirmed vulnerabilities. Determine the most appropriate options for remediation/mitigation of the issues identified as being a priority by performing analysis, write finding reports, executive summary. | 3 | 20 |
| **OWASP Risk Rating Methodology** | Risk = Likelihood \* Impact | 3 | 22 |
| **Common Vulnerability Scoring System (CVSS)** | x / 10.0 A universal language to convey vulnerability severity and help determine urgency and priority of response. | 3 | 23 |
| **Heartbleed** | Heartbleed broke the internet CVSS Score: 5.0/10 | 3 | 23 |
| **Common Vulnerability Scoring System (CVSS)** | Universal language to convey vulnerability severity and help determine urgency and priority of response. It serves as the standard by which other systems are judged. | 3 | 24 |
| **Common Vulnerability Scoring System (CVSSv2)** | Lacked considerations for targets of opportunity, active and successful exploitation in the wild, consideration on easy of exploitation. | 3 | 24 |
| **Common Vulnerability Scoring System (CVSSv3)** | Environmental considerations are modified to include base modifiers, impact metrics, and impact subscore modifiers. | 3 | 24 |
| **CVSS Version 3 NVD Calculator** | CVSSv3 calculations are broken into three categories: Base Score Metrics, Temporal Score Metrics, and Environmental Score Metrics. | 3 | 25 |
| **CVSS Version 3 NVD Calculator** | The national vulnerability database contains a tailored calculator for CVSSv2 and CVSSv3. CVSSv3 calculations are broken into three categories: Base score metrics, temporal score metrics, and environmental score metrics. The National vulnerability database (NVD) calculator provides a five-rank vulnerability categorization. | 3 | 25 |
| **CVSS Scores - Calculating** | The calculation for CVSS scoring is broken into three segments: Base Score Metrics, Temporal Score Metrics, and Environmental Score Metrics. | 3 | 26 |
| **CVSS Scores calculating** | The calculation for CVSS scoring is broken into 3 segments: Base score metrics (Exploitability metrics, impact metrics), Temporal score metrics (exploitability, report confidence, remediation level) Environmental score metrics (Base modifiers, impact metrics, impact subscore modifiers). These components are further broken down to perform the actual calculation. | 3 | 26 |
| **Auto-Generated Risk Ratings: Dangers** | Risk ratings, such as Microsoft’s DREAD model, use automated data sources and calculation I order to generate a quantitative risk rating. automated risk ratings fail to account for data singularity. Accurate risk calculation requires customized usage per application, in spite of sophisticated sounding verbiage, computers continue to lag behind in this discipline. | 3 | 27 |
| **DREAD Model** | Risk ratings such as the Damage | Reproducibility | Exploitability | Affected users | Discoverability (DREAD) from Microsoft use automated data sources and calculation in order to generate a quantitative risk rating. Fail to account for data singularity. Requires customized usage per application. | 3 | 27 |
| **Customized Risk Calculation** | Customized risk analysis is vital component to any holistic assessment strategy. When developing personalized ratings, 2 disciplines should be considered: Quantitative Risk Assessment – factors in the financial impact and other metrics, and Qualitative Risk assessment – Factors in the likelihood, difficulty in exploitations, and other factors. | 3 | 28 |
| **Qualitative Risk Assessment** | Factors in the likelihood, difficulty in exploitation, and other factors. | 3 | 28 |
| **Quantitative Risk Assessment** | Factors in the financial impact and other metrics. | 3 | 28 |
| **MS17-010** | Zero day. The NSA developed an exploit that took advantage of a vulnerability in a Microsoft protocol known as SMB. It is the language spoken by their operating systems to do pretty much everything, including file and print sharing and logging into Active Directory (AD). It was not intended to be released to the public. | 3 | 29 |
| **Meltdown** | Meltdown can allow a lower-privileged process to read data from a higher-privileged process. There are patches for some operating systems; however, they have been problematic. | 3 | 30 |
| **Spectre** | Spectre can allow one process to read data from other processes at the same level of privilege. It can be exploited using JavaScript in a browser. | 3 | 30 |
| **Cisco System's Dynamic Trunking Protocol (DTP)** | A feature that allows devices connected to a switchport to determine if they are to be a trunk port or standard managed port. If an attacker determines that this feature is enabled, they can use tools to negotiate their connection as a trunk port. This gives the attacker access to all VLAN's on the switch, bypassing security controls. | 3 | 31 |
| **CVSS scores: Not all vulnerabilities have scores** | Most “vulnerabilities” triggered by attackers do not have CVSS scores. Attackers use systems the way that they were designed, but with nefarious intentions. | 3 | 31 |
| **Penetration Testing** | The role of penetration testing is well-understood by the majority of organizations and gave birth to newer testing techniques such as Red Teaming, adversary emulation, and Purple Teaming. | 3 | 33 |
| **Pen testing: customer** | System owners, operations, engineering, and application stakeholders. | 3 | 36 |
| **Pen testing: Effort** | ~10% tools based and ~90%. Manual testing | 3 | 36 |
| **Pen testing: Frequency** | ~once per year. | 3 | 36 |
| **Pen testing: goal** | Report all exploitable vulnerabilities under controlled circumstances. | 3 | 36 |
| **Penetration Testing** | Involvers modeling the techniques used by real-world computer attackers to find vulnerabilities; to exploit those flaws under controlled circumstances; in a professional, safe manner according to a carefully designed scope and rules of engagement; to determine business risk and potential impact, all with the goal of helping the organization improve security practices. | 3 | 36 |
| **Red Team** | Emulates tactics, techniques, and procedures (TTPs) of real adversaries to improve the people, processes, and technology in their target environment. “The practice of looking at a problem or situation from the perspective adversary” | 3 | 37 |
| **Red Team: Customer** | Blue teams | 3 | 37 |
| **Red Team: effort** | Manual; some Red Team Automation tools. | 3 | 37 |
| **Red Team: Frequency** | Intelligence-led (New exploit, tool, or TTP) | 3 | 37 |
| **Red Team: Goal** | Trans and measure blue teams’ detection and response policies, procedures, and technologies are effective. | 3 | 37 |
| **Adversary emulation (Red Team)** | A type of Red Team exercise where the red team emulates how an adversary operates, following the same tactics, techniques, and procedures (TTPs), with a specific objective similar to those of realistic threats or adversaries. | 3 | 38 |
| **Adversary Emulation: Customer** | Entire organization | 3 | 38 |
| **Adversary Emulation: Effort** | Manual; more setup than a limited scope penetration test. | 3 | 38 |
| **Adversary Emulation: Frequency** | Twice a year or yearly. | 3 | 38 |
| **Adversary Emulation: Goal** | Emulate an end-to-end attack against a target organization. Obtain a holistic view of the organization’s preparedness for a real, sophisticated attack. | 3 | 38 |
| **Purple Team** | Is a function or process, not an individual team. Where red and blue work together to improve the overall security of the organization. Red Team does not focus on stealth as they normally would. | 3 | 39 |
| **Purple Team Effort** | Manual | 3 | 39 |
| **Purple Team Goal** | Red team emulates adversary TTOs while blue teams watch and improve detection and response policies, procedures, and technologies in real time. | 3 | 39 |
| **Purple Team: Customer** | Red Team and Blue Team. | 3 | 39 |
| **Purple Team: Frequency** | Intelligence led (new exploit, tool, or TTP) | 3 | 39 |
| **Pen Testing: Why performing it.** | Penetration testing allows for the validation of vulnerability findings. Vulnerability scanners do not typically perform validation. A penetrations test attempts to exploit a given vulnerability to demonstrate the true impact. It shows the effectiveness of security controls. Identify gaps in network and system architecture. Build confidence in potential customers. | 3 | 40 |
| **Rules of Engagement** | Covers how the test is to be conducted: - Start dates and end dates of testing - Time of day testing is permitted - Contact information for testers, system owners, and business owners - What data is and is not allowed to be viewed—Think PII data and regulatory requirements - How to respond if an exploit attempt has been blocked or new rules were added to block the tester's IP addresses - How results should be submitted and in what format | 3 | 41 |
| **Scoping** | The defined scope instructs the penetration testing team as to what systems and services they can and cannot target. Too narrow of a scope reduces the value to the customer. The pen test team can help explain as to why additional targets should be in score. Sometimes the customer intentionally limits the scope. | 3 | 42 |
| **Scoping document** | It should include items before performing a pen testing such as:  What systems, networks, domains, and applications are in play?  What if new unidentified systems are discovered?  Which ones should explicitly be avoided?  What if the system resides at a vendor location or in the cloud? | 3 | 42 |
| **Penetration Testing: External** | An external penetration test focuses on the perspective of an attacker from outside the organization.   * Targets externally-facing systems such as DNS, Mail, and Web servers * May include phishing attacks as another means to gain access. * Relies heavily on Open-Source intelligence (OSINT) and scanning. It is often seen as a "real-world" campaign since the testers are not given internal access to start. But is this correct? "Zero-Knowlegde Box" | 3 | 44 |
| **Penetration Testing: Internal** | Post-exploitation capability. An internal penetration test focuses on what an attacker could do if starting from within the organization's network:   * This can be a malicious insider with privileged access, such as an employee, contractor, or vendor * It could also be an attacker who has successfully gained a foothold inside the organization - We can assume a breached state. | 3 | 45 |
| **Penetration Testing: Web Application** | Focuses specifically on web applications and databases:   * These systems are the ones most often exposed to the Internet and provide access to sensitive data * If poorly coded they can be exploited to gain unauthorized access. * The OWASP Top 10 Web Application Security Risks: <https://owasp.org/www-project-top-ten/> * Zed Attack Proxy: https://owasp.org/www-project-zap/ | 3 | 46 |
| **Penetration Testing: Social Engineering** | The clever manipulation of the natural human tendency to trust. Use of influence and persuasion to deceive people for the purpose of obtaining sensitive target information or for the victim to perform an action. | 3 | 47 |
| **Social Engineering types: Computer based** | Pop-up windows. Mail attachments | 3 | 48 |
| **Social Engineering types: Human-based** | Urgency. Third-person authorization | 3 | 48 |
| **Mobile Device Penetration Testing** | The attack surface of mobile devices certainly includes similar attacks as those related to web clients and apps. Mobile devices are used for much more than just the web and require a unique focus. | 3 | 49 |
| **Penetration Testing - Additional Types** | Mobile Device Testing, Product Security Testing, Physical Penetration Testing | 3 | 49 |
| **Penetration Testing: Product Security** | This type of testing could include internet of things (IoT) devices, network appliances, such as VOIP, phones, printers, switches, and routers, often involves reverse engineering, fuzzing, and debugging. | 3 | 49 |
| **Physical Penetration Testing** | This type of testing factors in devices such as a badge reader, elevators, security gates, biometric devices, safes, and many others. | 3 | 49 |
| **Penetrations testing phases** | 1. OSINT 2. Scanning and enumeration 3. Vulnerability identification 4. Exploitation 5. Post exploitation | 3 | 52 |
| **OSINT: Open-Source Intelligence (OSINT) Gathering** | Reconnaissance.  Mostly self-defined!   * It is the practice of collecting data via open- source locations such as the public internet, presentations, printed media, and more * Reconnaissance is a passive effort allowing the analyst to go mostly undetected. Most of us use search engines to search for data; however, much of the internet is not indexed by search engines * Social media quickly comes to mind! * Other interesting areas include the dark web * Shodan and Maltego are examples of tools that to help | 3 | 53 |
| **Scanning and enumeration** | Tools such as Nmap are commonly used to identify systems, services, OS versions, and more quickly and effectively. We can enumerate discovered systems to learn information such as what services are tied to listening ports. It is not uncommon for administrators to bind services to non-standard ports, such as running an HTTPS services over TCP port 4433. | 3 | 54 |
| **Vulnerability Identification** | * Vulnerability identification phase helps us to select the best targets and services to use in the exploitation phase * If we identify that a service is running, we can check to see if we have a corresponding exploit available * With vulnerability assessments, any validation would be performed manually. * Tools such as Nessus and OpenVAS excel at this phase. | 3 | 55 |
| **Exploitation** | * Is where we take all the knowledge gained thus far and attempt to gain access. * Access gained could include a system, network, or other resource. * We want to choose the technique and exploit most likely to be successful and go undetected. * Exploitation may require that we account for evading protections such as a antivirus, antimalware, and other controls. | 3 | 56 |
| **Ghostwriting** | AV-evasion is a very common technique used by penetration testers and attackers. One technique, known as ghostwriting, changes the way malware, a payload, or other code looks at a very low level in order to evade signature detection. It is performed by opening up compiled code and inserting benign processor instructions or moving things around to change its signature. | 3 | 56 |
| **Data Exfiltration** | Depending on the scope of the pen test, we may be able to move data off a system and onto the tester's system. This could help us to demonstrate the impact; however, regulations and other policy may forbid us from performing such actions. | 3 | 57 |
| **Maintaining access** | If is scope, there are tools which we can use to help us maintain access on a system. This will help to ensure that if a system becomes temporarily unavailable, we can regain access. | 3 | 57 |
| **Pivoting and Lateral Movement** | These are techniques performed where we use our access inside the organization to expand influence and gain further access. Assuming we are on a system on the trusted side of a network, new opportunities should be exposed. We may be able to make it look like traffic is coming from a system inside the network using a compromised host as opposed to the true external system. | 3 | 57 |
| **Post exploitation** | Includes activities performed once an initial foothold is gained on a system or network. Initial access is great, but a pen tester must demonstrate the true impact. This includes many actions such as pivoting and lateral movement, privilege escalation, data exfiltration, maintaining access. Tools include Metasploit framework, Empire, Covenant, etc. | 3 | 57 |
| **Privilege Escalation** | Our access to a system may be limited. We may be in a sandbox or running as a low-privileged user. Escalation efforts are aimed at promoting our access level on a system to Administrator, System, Root, or other. | 3 | 57 |
| **Appendix** | This is where verbose scan results, code snippets, images, and other relevant data should be placed. | 3 | 58 |
| **Conclusion** | This should simply be a summary of the entire document, similar to the executive summary. | 3 | 58 |
| **Executive Summary** | This should be no longer than a page or two. It should include a summary of the overall project, a high-level overall risk statement, and then some of the top findings in non-technical terms. Assume that anyone in the organization may read the summary. | 3 | 58 |
| **Findings and Risk Assessment** | This is the section where you list out the findings in order of most critical to least critical. A qualitative and quantitative assessment can be performed on each finding including impact, likelihood, and other factors. | 3 | 58 |
| **Introduction** | Include the Rules of Engagement details, scope of testing, tester information, most significant findings, and consider including information such as the tools or services used. It isn't a bad idea, especially if it is an internal pen test team testing their own organization, to demonstrate that the money spent on security tools and processes are put to use. | 3 | 58 |
| **Methodology** | This should include details on what was performed related to each phase of the overall process. If there aren't many significant findings, this is the section where you can demonstrate how thorough the testing was and provide positive comments on the security posture. | 3 | 58 |
| **Recommendations** | For any findings, recommendations can be made to improve the security of the organization. What types of things would have prevented the testers from being successful? | 3 | 58 |
| **Reporting** | The pen test report is what you leave behind to your customer and it should include. Executive summary, introduction, methodology, findings, and Risk assessment, Recommendations, Conclusions, Appendix. | 3 | 58 |
| **Penetration Testing Tools** | Example Tools of the Trade:  NMAP, Metasploit, Meterpreter, C2 Frameworks: • Empire • Cobalt Strike • Covenant • C2 Matrix | 3 | 60 |
| **Network Mapping** | Network mapping is simply the process of enumerating all hosts that respond on a given network. | 3 | 61 |
| **Nmap** | A popular freeware network mapping tool. It supports a large number of scanning techniques, including port scanning (TCP and UDP), SYN, FIN, ACK, and ICMP ping sweeps. It also offers advanced features, such as remote OS detection, stealth scanning, and other functionality. Nmap is a free, aware winning network scanner. Supports a large number of scanning techniques. Numerous other features supported. Remote operating System detection. Application detection.  Common back door is to open a port. Port scan scans for open ports on remote host. Scan0-65535 twice, once for TCP, once for UDP, Various tools available. Scanport and NMAP. | 3 | 61 |
| **Port Scanning** | Port scanning goes a step further and tells you which ports on each machine have listener processes bound to them. | 3 | 61 |
| **Nmap - Port Scanning** | Although we still can't determine the application that is running on the remote system with 100 percent accuracy, tools like Nmap have included functionality to decipher the responses returned after connecting to a port to identify the software that is listening, sometimes including version and patch level information. | 3 | 62 |
| **Application Fingerprint** | Someone running Nmap to identify applications listening on ports can submit the application fingerprint data and the name and version of the application at the Nmap website to improve the accuracy of the scanner. Over time, as more and more people submit application fingerprints, the Nmap database grows and becomes capable of identifying a variety of different applications listening on a given port. | 3 | 63 |
| **Class-C Address** | For example, 192.168.1.\* would scan the entire 192.168.1.0/24 network (the so-called "Class C" address). | 3 | 63 |
| **Filtered (Nmap)** | Not open or closed. This is a kind of in-between state, which means that it might be listening, but Nmap can't tell for sure. | 3 | 63 |
| **Unknown (Nmap)** | This means that the port is not a well-known port, so Nmap has no idea what is running on it. | 3 | 63 |
| **-randomize\_hosts (Nmap CL option)** | Command line option. This causes Nmap to skip around the IP address space you gave it and scan the hosts in fairly random order. | 3 | 64 |
| **Operating System Identification** | Looks for subtle differences in target responses. Develops a fingerprint. Compares the fingerprint against a pre-built database of operating system fingerprints. | 3 | 65 |
| **Signatures (Nmap)** | The logic for figuring out which OS responds in which way to which probes isn't hardcoded into Nmap, either. It comes with a database of hundreds of "signatures" of different operating systems and networking equipment and relies on its user community to help keep it up-to-date by submitting new signatures as new operating systems become available. | 3 | 65 |
| **Exploitation Frameworks** | These are tools that, if a vulnerability exists, will exploit the system, and provide the person running the tool access to the system. | 3 | 66 |
| **Metasploit** | Exploitation framework. If a module for vulnerability exists, it will attempt to exploit the target and provide access. Useful for organizing and managing exploits. Complementary to vulnerability scanning to reduce false positives. | 3 | 66 |
| **Meterpreter** | Meterpreter is a custom Metasploit shell, originally created by Matt Miller (Skape) which offers an enormous amount of power and functionality to the user. When using Metasploit, the Meterpreter payload can be specified. The payload is what executes if the exploit being used to compromise a system is successful. In the case of Meterpreter, successful execution results in a Transport Layer Security (TLS) connection between the victim and the attacker. A custom menu is made available, and the user can import additional functionality. Actions which can be performed include a wide range of post-exploitation capabilities such as sniffing from the victim's network interfaces, screen capture, interacting with the webcam and microphone, impersonating other users, spawning new processes, and countless others. | 3 | 67 |
| **C2 Frameworks and Implants** | Empire: Written in Python and focuses on PowerShell is a pure PowerShell post-exploitation agent build on cryptological-secure communications and a flexible architecture. Original project no longer supported, but a fork was created.  Cobalt Strike: a commercial adversary emulation product from SpectreOps focusing on covert communication and post-exploitation.  Covenant: a C2 Framework that leverages the attack surface of .NET2. an alternative option to Empire as PowerShell is heavily monitored.  Sliver: A DNS-based implant framework from BishopFox. | 3 | 68 |
| **7T** | Matrix of a command-and-control Frameworks. Google doc of most C2 frameworks. Documents various capabilities of each framework. There is no right or wrong, better, or worse framework. Find the ideal C2 for your current objective. Wizard-like UI to select the appropriate one. | 3 | 69 |
| **C2 Matrix** | The C2 Matrix is a working project to document all public command and control frameworks in a single place. | 3 | 69 |
| **Multifactor authentication** | Microsoft study: 99.9% less likely to be compromised if you used MFA. As October 2019 only 8.2% of Azure AD accounts were using MFA. Solutions such as FIDO2 and YubiKeys can greatly increase MFA security. | 3 | 70 |
| **Password compromise** | A Password compromise remains one of the most sought-after and effective access methods used by penetration testers, this includes password cracking, spraying, reuse and others. | 3 | 70 |
| **Password Reuse and Stuffing** | Many of us use the same username and password combinations for multiple resources. This attach is popular thanks to the large dumping of credentials associated with data breaches. Attackers take the dumped credentials and try them over a large number of websites. | 3 | 71 |
| **Password Spraying** | Is a simple concept where commonly used passwords are tried against many user accounts. Helps prevent against account lockout. By the time a second password is tried, many sites will have reset the account lockout security threshold. Typically targets web and cloud sites. | 3 | 72 |
| **Poisoning Tool** | Responder, a popular tool among penetration testers, is used in attempting to capture credentials. It is a "poisoning" tool that responds to Link-Local Multicast Name Resolution (LLMNR), multicast DNS (mDNS), and NetBIOS Name Service (NBT-NS) service requests coming from systems where DNS is unavailable or fails to successfully resolve a name. | 3 | 73 |
| **Responder** | Responder, a popular tool among penetration testers, is used in attempting to capture credentials. It is a "poisoning" tool that responds to Link-Local Multicast Name Resolution (LLMNR), multicast DNS (mDNS), and NetBIOS Name Service (NBT-NS) service requests coming from systems where DNS is unavailable or fails to successfully resolve a name. | 3 | 73 |
| **Attacks and Malicious Software** | High-Profile Breaches, Ransomware, Malware, Attack Techniques | 3 | 78 |
| **Marriot Data Breach: Summary** | Marriott discovered breached September 8,2018. Reported the breach November 30, 2018. 9.1 million payment cards | 24 million passports. Breach was active from 2014. | 3 | 81 |
| **Marriott Data Breach: Impact** | Reservation database, credit card numbers, passport numbers, mailing addresses, name, phone, and emails. | 3 | 83 |
| **Marriott Data Breach: Failure** | Step taken postmortem: Free web watcher enrollment, website set up for information, huge sum spent on additional security technology, thousands of man-hours still going into finding the root. | 3 | 84 |
| **Marriott Data Breach: Fallout** | Marriott implemented a website for customers who thought they might have been impacted through a 3rd party, and it is most likely that it had been the company in charge of the post compromise, and it wasn’t on corporate domain. Customers thought it was phishing. | 3 | 85 |
| **wtmp** | Key Log Files | 3 | 85 |
| **Equifax Data Breach** | The Equifax breach was another massive hit to confidentiality of consumer information. In total 143 million people affected Please see book. | 3 | 86 |
| **WannaCry Ransomware** | Confidentiality, integrity, and availability attack. Spread across over 150 countries. Cripples over 200,000 devices. Much like other high-profile breaches, a few basic security practices could have reduced or prevent these attacks. This is not a zero-day, it was a patchable vulnerability. | 3 | 87 |
| **WannaCry Ransomware: The attack** | Crippled hospitals, banks, and other industries around the world. Encrypts data and demands ransom. Utilized tools initially believed to have been developed by a nation state. | 3 | 88 |
| **WannaCry Ransomware: Breach impact** | One of the first public examples of ransomware. The impact of these attacks was widespread. Roughly 230,000 systems breached. Over 150 countries. Gave way to several follow-on attacks. | 3 | 89 |
| **WannaCry Ransomware - Root Cause** | The Attack:   * Began May 12, 2017, in Asia * Ransomware came as a worm, not a virus * Largely targeted Microsoft systems * Patch released two months before initial attacks | 3 | 90 |
| **WannaCry Ransomware: Tools** | * There were multiple layers to the attack that relied on each other. * Utilized code to scan for vulnerable systems. * Eternal blue: SMBv1 exploit. DoublePulsar: Backdoor Trojan. | 3 | 91 |
| **WannaCry Ransomware: Gaining access.** | It used 3 primary commands:   * Ping: Diagnostic tool used to test system reachability * Kill: system command used to terminate running processes. * Exec: Which can be used to load and execute malware on the victim system | 3 | 92 |
| **Breach (3 discussed) Attack Summaries** | Behind every attack lies one or more of the following commonalities:   * The system was visible from the internet * Unchecked scanning and enumeration * An unpatched vulnerability was exploited • The system had weak authentication | 3 | 93 |
| **WannaCry Ransomware: Attack summary** | * The system was visible from the internet. * Uncheck scanning and enumeration. * An unpatched vulnerability was exploited. * The system has weak authentication. | 3 | 93 |
| **WannaCry Ransomware: Current attacks** | WannaCry is mostly patched, but other ransomware has followed in its footsteps. Ransomware remains one of the biggest threats. Methods are changing. Exploiting mostly know vulnerabilities. Spreading due to lack of segregation. | 3 | 94 |
| **Ransomware as a Service (RaaS)** | Cyber criminals can create a customized version of various types of ransomwares for profit. Satan ransomware has the ability to make your own customized version, in the example, 30% of the profits go to the developer or the service. | 3 | 95 |
| **Ransomware as a Service (RaaS) (2)** | On this slide are screen captures from the author of "Bitcoin Stealer" and "Bitcoin Blackmailer." The JigSaw ransomware came from this source. The author allows you to create your own customized version of "Bitcoin Blackmailer" for profit, providing instructions on how to compile the code, create new Bitcoin payment addresses, and more. It is unlikely that the link provided will remain active forever, but the screenshots should get the point across. | 3 | 96 |
| **Ransomware as a service: examples** | Bitcoin stealer and bitcoin blackmailer. Jigsaw ransomware can be customized at this site, and it allows you to create your own customized version of bitcoin blackmailer for profit, providing instructions on how to compile the code, create new Bitcoin payment addressed and more. | 3 | 96 |
| **Common Attack Techniques** |  | 3 | 97 |
| **Application/Protocol Fuzzing** | Application or protocol fuzzing are testing techniques where malformed data is sent to a network protocol or to a program opening a supported file format. By intentionally including the wrong input, the application can be checked to see how it handles the requests. | 3 | 98 |
| **Input Attacks** | Applications receive client data in many forms. Treat all user-supplied input as potential attack points. Examples:   * OS command injection * Buffer overflows * SQL injection | 3 | 98 |
| **Trust Boundaries** | One reason applications tend to have vulnerabilities is because they accept and process input from the user in a variety of different entry points. We often call these entry points "Trust Boundaries." There are many ways an attacker can cause issues with an application by sending invalid or malicious data to these entry points or across these trust boundaries. | 3 | 98 |
| **Command Injection** | Attacker sends OS commands as form or other input. Relies on developer using input to build calls back to the OS. Some web applications use operating system level commands to perform certain functions. For example, a mailbox application might make a call to the operating system to create a new folder for a user's attachments and name the folder to match the username supplied by the user. If the input is not properly validated, the user could have typed ksmith; rm -rf / as his user ID. When the create folder command was run within the operating system, the rm -rf / command would be run as a separate command meant to delete the entire filesystem. Attacker sends OS commands as form or other input. Relies on developer using input to build calls back to the OS. EX: jsmith; cat /etc/passwd | 3 | 99 |
| **Command Injection - Defense** | * Avoid making system calls from within your application, especially to the system() function * Input processed based on user input at biggest risk * Where possible, use built-in application functions instead * Strip OS commands and characters from input * Even better: Define valid characters for input used in this way; delete all others from input | 3 | 100 |
| **Buffer Overflow** | In poorly coded applications, no validation is performed to ensure the data being copied is not larger than the allocated memory  When the buffer is overrun, important variables such as the return pointer can be hijacked | 3 | 101 |
| **Stack** | A segment in the memory of a process used during function calls. | 3 | 102 |
| **Return Pointer** | This is a very important variable as it is used to return control back to the caller of this function. To use an analogy, when you rent a car, you agree to return it back to the company from which the car was rented. If your car is stolen, then it will not be returned and likely be taken to some other destination. We do not want someone to hijack control of our process and as such, we must protect the return pointer from being overwritten. | 3 | 103 |
| **strcpy()** | The strcpy() (string copy) function is a deprecated function used to copy text data from one location to another. The problem with strcpy() is that it does not offer the option to check the length of the user input. | 3 | 103 |
| **Buffer overflow: Basic stack overflow I.** | This is when we have overwritten the memory because the developer didn’t put a limit on the characters in the input and goes to the return pointer, and we will get a segmentation fault. | 3 | 104 |
| **Segmentation Fault** | This is the error message reported if an access violation occurs within the process, such as that of a return pointer pointing to an invalid location in memory. | 3 | 104 |
| **Buffer overflow: Basic stack overflow II** | 1. Using a debugger, the attacker gets the static memory address of the buffer 2. . They place their shellcode into the buffer and overwrite the return pointer with the address of the buffer 3. . When the process goes to return control to main(), control is instead passed to the attacker's shellcode | 3 | 105 |
| **Buffer Overflow - Defenses** | * Run the latest versions of all your installed software. * Update and patch your software. * Update and patch your languages/runtime environment/server add-ons. * Run a vulnerability scanner against your applications. * Utilize endpoint protection suites offering exploit mitigations. * Validate and sanitize user input. | 3 | 106 |
| **SQL Injection:** | SQL Injection is yet another vulnerability taking advantage of insufficient input validation. The technique, if successful, can be used to execute arbitrary SQL commands to which the web server application database account is authorized. Ex: http://www.example.com/login.php?passwd=' or userID='admin';--' | 3 | 107 |
| **SQL Injection - Defenses** | **Validate user input:**   * Only allow a specific set of valid characters, i.e., an allowlist   **Have length limits on input:**   * Many SQL attacks depend on long strings   **More tiers:**   * Add an application layer between the web server and the database * Utilize stored procedures instead of SQL queries * Database access: Web account should not have rights to add/drop/modify tables or stored procedures * Do not display SQL errors to web users * Monitor SQL error messages | 3 | 109 |
| **Allowlist** | The preferred way of validating input is to define an allowlist (to allow only known good values). | 3 | 110 |
| **Denylisting** | Another method to validate input is to check for or sanitize known bad input. This is also known as denylisting. With denylisting, you try to guess all the bad input that a user might try to send into the application and delete or replace the bad input before it is processed by the application. | 3 | 110 |
| **Input Attack - Defenses Summary** | "Always validate and sanitize user input." | 3 | 110 |
| **Input attack defenses summary** | * Only allow specific valid characters from user input to avoid characters that have special meaning in scripting languages. * Be suspicious of all input, including HTTP headers, and cookie data. * Validate on the server, not the client. * Use an up-to-data well-validates third party library of input and check for encoded characters * checking routings to use throughout the application. | 3 | 110 |
| **Sanitization** | Sanitization while denylisting can be done by deleting the unsafe characters or by changing them to safe characters. | 3 | 110 |
| **Validation** | Validation can be done using scripts on the client side, but this really should be used only to make the web application more usable. Client-side validation provides no additional security because a malicious user can modify or bypass any scripting or validation done on the client; all data must be validated on the server as well. The preferred way of validating input is to define an allowlist (to allow only known good values). Another method to validate input is to check for or sanitize known bad input. This is also known as denylisting. | 3 | 110 |
| **Malware and Analysis** | Viruses, Worms, Trojans, Rootkits; Fully Automated, Static Properties, Interactive Behavior, Manual Code Reversing | 3 | 111 |
| **Virus** | * A virus is a type of malware, historically destructive in nature, that typically requires user interaction to infect a system * They require a host file, such as an executable or Microsoft Office macro * Once executed, they often copy themselves onto file shares or removable media, email copies of themselves, etc., in an effort to spread * Their goal is often destructive, such as deleting files; however, they can perform whatever actions they were programmed to do by their author | 3 | 112 |
| **Worm** | A worm is historically a self-replication piece of code that typically carries one or more exploits and payloads. They often scan systems for vulnerable serves, launch an attack, and execute their payloads. Their goal is often to infect as many systems as possible and set up C2 (Command and control). | 3 | 112 |
| **Virus vs. Worms** | Virus typically requires user interaction to infect whereas worms do not. They also require a host file, such an executable or document whereas worms’ scans for listening ports associate with vulnerable services. With viruses the goal is data destruction whereas with worms if exploitations are successful, it executes a payload and copies itself onto the infected system. Viruses spread via file shares, removable media, email clients whereas worms’ goals are to set up command and control network and viruses may intentionally or inadvertently impact systems and network performances whereas worms inadvertently impact systems and networks | 3 | 113 |
| **Trojans** | A Trojan horse is a program that often performs the desired action of the victim, as well as a malicious action:   * If we have the source code, we could add our malicious code to the beginning of the program, or at any other point within it, compile the code, and publish it * When the victim executes the program, they will get their desired result of using the tool; however, our code will also execute | 3 | 114 |
| **Trojans** | A trojan horse a program that often performs the desired action of the victim, as well as a malicious action. If we wanted to create a trojan horse on a netcat and if we had the source code, we could add our malicious code to the beginning (hide it) or any other point within it, compile the code, and publish it. When the victim executes the program, they will get their desired result of using the tool, however, our code will also execute. | 3 | 114 |
| **Rootkits** | Rootkits are a form of malware that, from a high level, typically have the goals of subverting userland and kernel detective controls and providing ongoing access to the compromised system by an attacker. | 3 | 115 |
| **Malware Analysis Additional Stages** | Fully automated analysis, Static Properties Analysis, Interactive Behavior Analysis, Manual Code Reversing. | 3 | 116 |
| **Malware Analysis Stages** | Traditionally, malware analysis has consisted of 2 phases: Behavioral Analysis and code analysis. | 3 | 116 |
| **Fully Automated Malware Analysis** | Automation of malware analysis is a good way to perform quick analysis at a relatively low cost when compared to a human analyst. However, they can provide false positives, it is normal for programs to contain strings that also reside in malware, critical function calls by legitime programs will result in alerts, VirtuaProtect, VirtualAlloc, WritePRocessMemory, HeapAlloc, etc. | 3 | 117 |
| **File utility** | The "file" utility uses signatures and examines metadata to determine the file type of our selected file. | 3 | 118 |
| **Static Properties Analysis** | Ex: file utility. Performing a static properties analysis allows us to learn about the intentions of the malware without actual execution. This analysis must still be performed carefully in a controlled environment to prevent unintended copying, execution, or other undesired results. By examining static properties, we can learn things such as any ASCII or Unicode strings, calculate hashes of the specimen to submit to online resources, and learn about potential packing or obfuscation, potential IOCs, and other relevant information. | 3 | 118 |
| **Interactive Behavior Analysis** | By allowing the malware to run in a controlled environment, we can monitor its behavior as well as manually interact   * Registry and filesystem access * Process behavior and interactions * Network activity | 3 | 119 |
| **Decompilers** | convert assembly language or intermediate language back to pseudocode | 3 | 120 |
| **Disassemblers** | convert machine code back to assembly language | 3 | 120 |
| **Manual Code Reversing** | Manual code reversing and analysis is considered to be the most advanced form of malware analysis.  Requires the use of special tools such as disassemblers and decompilers:   * Disassemblers convert machine code back to assembly language * Decompilers convert assembly language or intermediate language back to pseudocode | 3 | 120 |
| **Marriot & WannaCry: Lessons learned** | Attackers follow a strategy of reconnaissance, enumeration, and penetration. It is important to follow security core concepts such as a patching and limiting network visibility. There are many different methods of attack, including buffer overflows, and SQL injection, etc. Malware remains a real threat. | 3 | 121 |
| **Web Application Security** | Understand how web applications work, Learn best practices for creating secure web applications, Identify and fix vulnerabilities in web applications | 3 | 126 |
| **Web Application Basics** |  | 3 | 127 |
| **HTTP (Hypertext Transfer Protocol)** | Transaction-oriented. The format of a transaction is simple. There are two parts: The client's request and the server's response. | 3 | 128 |
| **Port 443** | HTTPS connects to this port and all secure transfers are done here. | 3 | 128 |
| **Port 80** | TCP connects to this port on server/HTTP. End to end unprotected. | 3 | 128 |
| **Web communications Basics** | Stateless communications. Retrieving information: GET, HEAD. Sending information: POST, PUT. The web can be considered a transport mechanism for the information it contains. It boils down to the protocol browsers and servers use to communicate: the Hypertext Transfer Protocol (HTTP). The HTTP is transaction-oriented. Clients make requests, and servers send responses. It is stateless since once a request is responded to; it will forget all about it. | 3 | 128 |
| **Body** | Some requests include a third piece: The body. This is used only in the case where the client is going to send some data to the server, such as when a user POSTs some form data or uploads a file via the HTTP PUT method. | 3 | 129 |
| **Header stanza** | The next piece of the client request is the header stanza. Headers immediately follow the request and can convey almost any piece of information that the client wants the server to know. As mentioned, the client usually includes a Host: header just to let the server know to which website it's trying to connect. | 3 | 129 |
| **Host: header** | The client is also required to send a Host: header to specify at which domain the request is aimed. This allows a single web server on a single IP address to process requests for multiple domains. | 3 | 129 |
| **PUT** | Clients use PUT when they need to upload files to a web server, such as when publishing new webpages or sending attachments with a web-based email service. | 3 | 129 |
| **Request** | The client starts the conversation with the request. The request is a one-line string that includes the method (for example, the type of request), the resource being requested, and the HTTP version the client expects to use. In the current version of the protocol, the client is also required to send a Host: header to specify at which domain the request is aimed. | 3 | 129 |
| **Response** | Responses consist of a status line followed by some header lines and, usually, the body which contains the requested resource. | 3 | 129 |
| **Cookies** | Stores data from a browser session on the client and are sent to the server. Are often to keep state. Can be: Persistent (Text, file/database) and non-persistent: Session/ in-memory. Cookies can be blocked in you wanted. A Cookie is a named piece of data created by a web server and stored at the web browser. Both the name and the contents are chosen by the application and can be almost anything the programmer wants. | 3 | 130 |
| **Cookies: Non-Persistent.** | Session cookies, stored in memory, do not survive a reboot, stored for a short period of time, could require additional authentication, since user ingo is not remembered. | 3 | 132 |
| **Cookies: Persistent** | Stored on a hard drive, survive a reboot, typically stored for a long period of time, used to track user activity, create privacy concerns. | 3 | 132 |
| **Cookies: Session** | The other type of cookie is the session cookie, sometimes called a "non-persistent cookie". As the name implies, session cookies are good only during the current browser session. They are usually stored only in memory and, when the browser exits, these cookies are lost forever. As you might guess, session cookies are good for applications that track their own state, especially if users might be accessing them from a shared computer (in a public library, for example).  After ending a session, simply closing the browser destroys them and renders them inaccessible to the next user of that computer. | 3 | 132 |
| **Proxy** | In memory, cookies can still be edited by either a user or machine in the middle by using an application that sits between the client computer and the web server, commonly called a proxy. An example of a web proxy is OWASP ZAP (http://www.zaproxy.org/) | 3 | 132 |
| **SSL/TLS** | Secure Sockets Layer (SSL, also called TLS, or Transport Layer Security) comes in. SSL is a protocol that provides an encrypted tunnel between two SSL-aware applications. It's the de facto standard for secured communication, and virtually all web browsers and HTTP servers support SSL, at least as an option. HTTP traffic over SSL uses port 443 by default, although this is subject to change by the server administrator. Performs 3 important functions: 1. Encryption 2. Server identity verification 3. Data integrity | 3 | 133 |
| **SSL/TLS: How it works** | Client connects to server, servicer indicates SSL configuration, client and server exchange crypto keys, secure session begins, it is not a guarantee of security. | 3 | 133 |
| **SSL/TLS: What is it** | Protocol for encrypting network traffic. Operates on port 443, provides encryption, server identity verification, and data integrity. | 3 | 133 |
| **Broken access control** | Access control enforces policies such that users cannot act outside their intended permissions | 3 | 136 |
| **Cryptographic failures** | Leads to sensitive data exposure or system compromise. | 3 | 136 |
| **Injection** | Happens when a user-suppled data is not validated, filtered or sanitized by the application. | 3 | 136 |
| **Insecure design** | Represents different weaknesses, expressed as missing or ineffective control design. | 3 | 136 |
| **OWASP top 10 critical issues** | 1. Broken access control 2. Cryptographic failures 3. Injection 4. Insecure design 5. Security misconfiguration 6. Vulnerable and outdated components 7. Identification and authentication failures 8. Software and data integrity failures 9. Security logging and monitoring failures 10. Server-side request forgery (SSRF) | 3 | 136 |
| **Security Misconfiguration** | "This is commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information." | 3 | 136 |
| **Vulnerable and Outdated Components** | "Components, such as libraries, frameworks, and other software modules run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts." | 3 | 136 |
| **Web App Security Issues** | Broken Access Control, Cryptographic Failures, Injection, Insecure Design, Security Misconfiguration, Vulnerable and Outdated Components, Identification and Authentication Failures, Software and Data Integrity Failures, Security Logging and Monitoring Failures, Server-Side Request Forgery (SSRF) | 3 | 136 |
| **Identification and Authentication Failures** | Application functions related to authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities temporarily or permanently." | 3 | 137 |
| **Security logging and monitoring failures** | Allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data. | 3 | 137 |
| **Server-Side Request Forgery (SSRF)** | "SSRF flaws occur whenever a web application is fetching a remote resource without validating the user-supplied URL. It allows an attacker to coerce the application to send a crafted request to an unexpected destination, even when protected by a firewall, VPN, or another type of network access control list (ACL)." | 3 | 137 |
| **Software and Data Integrity Failures** | "Software and data integrity failures relate to code and infrastructure that does not protect against integrity violations." | 3 | 137 |
| **Developing Secure Web Applications** | * Developer training on vulnerabilities and secure coding * Peer reviews to identify errors or bad practices * Formal and thorough testing using expected and unexpected input * Configuration management and version control * Separate development, testing, and production environments; separation of duties between developers and production administrators | 3 | 138 |
| **Load Testing** | This helps to demonstrate that the architecture and resources provided are sufficient for the web application's needs. | 3 | 138 |
| **Peer Review** | A peer review is the manual examination of source code by a group of the author's peers. Peer reviews are effective for detecting defects or other errors and can be especially effective for finding security issues if reviewers are trained in common mistakes that lead to vulnerabilities. | 3 | 138 |
| **Software testing** | Software testing is the process of executing a program or system with the intent of finding errors. With web-facing applications, it is important to also test how the application will respond to unexpected or invalid input. | 3 | 138 |
| **Application Performance Testing** | In performance testing, careful attention should be paid to the error messages and other abnormal behaviors of the system under an excessive load to ensure they don't disclose sensitive information about the system or indicate the creation of other vulnerabilities. | 3 | 139 |
| **Configuration Management** | Important components include:   * Separate, distinct workspaces or environments for different developers and different releases of the same product * A version control system that tracks changes to the code, allows developers to check in/check out components, and ensures code changes do not overlap * Formal processes for use of the versioning systems and development environments | 3 | 139 |
| **Secure Coding - Basics** | * Initialize all variables before use * Validate all user input before use * Don't make your application require admin permissions on the server or database * Handle errors and don't display errors to end users * Employ least privilege/limit access * Don't store secrets in your code * Use tested, reliable libraries or modules for common functions (authentication, encryption, session tracking) * Watch for vulnerability notifications in any utilized open-source libraries | 3 | 140 |
| **Web application vulnerabilities** | In order for an adversary to compromise a web application, they only have to find one vulnerability. Most common are Authentication, access control, session tracking. | 3 | 142 |
| **Basic Authentication** | 1. The client sends a standard HTTP request to load a page: GET /documents/JulyReport.html HTTP/1.1 2. The server responds with HTTP 401 Authorization Required: HTTP/1.1 401 Authorization RequiredDate: Sun, 09 Dec 2016 19:35:01 GMTWWW-Authenticate: Basic realm="Users" ... 3. At this point, the browser displays a pop-up dialog prompting for a user ID and password. 4. After the user enters the password, the client sends the same request, but this time, the entered user ID and password are included in the headers. The user ID and password are base64 encoded. This encoding is easily reversible; base64 encoding is not encryption and provides no protection: GET /documents/JulyReport.html HTTP/1.1 Authorization: Basic n3786sd9maGY5OWm8dcT= 5. After the web server receives this request, it decodes the base64 encoded user ID and password and tests whether it is correct for a valid user on the system. | 3 | 143 |
| **Digest Authentication** | The process for digest authentication is similar to basic authentication; however, instead of simple base64 encoding, it uses a one-way cryptographic MD5 hash to create a hashed password that is sent within the HTTP headers. | 3 | 143 |
| **Form-based authentication** | Using HTML form fields to request the user's authentication credentials. It is common to use the <INPUT TYPE="PASSWORD"> tag for the password input field. This field type obscures the typed characters with asterisks as they are displayed on the screen, and the contents of password fields are not cached or auto-filled when you reload or navigate between screens. The user ID and password are sent cleartext along with any other form data, so a separate mechanism to create a secure channel, such as SSL, is required for secure form-based authentication. | 3 | 143 |
| **HTTP Authentication** | With HTTP authentication, the user's authentication credentials are sent within the HTTP headers. | 3 | 143 |
| **Web Application authentication** | HTTP authentication: Credentials sent in HTTP header, basic mode: credentials sent in cleartext (base64 encoded) and Digest mode: Sends MD5 hash of password. Form based authentication: credentials entered and sent as HTML form data. Authentication attacks: password guessing, brute forcing, or bypassing authentication mechanism. Multifactor authentication: Relies upon more than just the user’s password. | 3 | 143 |
| **Authentication attacks** | * Password guessing, * brute forcing * bypassing authentication mechanisms | 3 | 144 |
| **Certificate-Based Authentication** | Provides a second factor by relying on something you have, namely a client certificate. | 3 | 144 |
| **Client Certificate** | A client certificate is a digital file with a cryptographic signature that is provided to the user either by the website owner or a trusted third party. Client certificates work very well to validate a user's identity, but they have gained minimal popularity because the distribution, setup, and management of digital client certificates are difficult for both website operators and end users. | 3 | 144 |
| **Footprint** | Because of the expense and complexity of these multifactor authentication schemes, many web applications are beginning to rely on the "footprint" of the user's device as a second factor for authentication. Attributes the application can look for to confirm the footprint of the device can include cookies left by the web application during a prior visit, software or other signatures installed to the hard drive, the client IP address, and system and browser configuration. Footprint schemes are easier to implement than other methods, but they are also much easier to fool or break. | 3 | 144 |
| **Hardware token authenticators** | Hardware token authenticators work well because they are easy to use, but they are expensive relative to other mechanisms. | 3 | 144 |
| **Multifactor Authentication** | In response to these weaknesses in password-based web authentication schemes, multifactor authentication is gaining popularity for stronger web-based authentication. Multifactor authentication is the use of more than one "factor" to verify a user's identity. Certificate-based web authentication, Token-based authentication (Hardware token authenticators, Software token authenticators), "Footprint" | 3 | 144 |
| **One-time password** |  | 3 | 144 |
| **Token-based Authentication** | Token-based authentication schemes also rely on something you have: a token. The token is a small device that produces a one-time password for each authentication attempt. Generally, this works by initializing and running the same algorithm on both the web server system and the token, so both systems generate the same one-time password based either on time or sequence of prior passwords. Ex's: Authy, Google Authenticator, Duo Mobile, and Microsoft Authenticator | 3 | 144 |
| **Access Control**  **Web users** | Typical users follow the path you anticipated through the site. Keep users out of parts of the server you don’t intend them to be in. default pages, sample sites, code library pages and configurations files, disable directory browsing, URL directory traversal. | 3 | 146 |
| **URL Directory Traversal Attacks** | Are kind of a combination of flawed access controls and an input attack. With directory traversal, the user exploits vulnerabilities on the web server to access restricted directories, execute commands, and view data outside of the directories meant to be published by the web server. | 3 | 146 |
| **Session IDs** | The most popular technique for tracking a user through multiple web requests is the use of session IDs. When the user requests the first webpage in their session, the server creates a unique identifier, usually a random number or string, and sends it back to the client along with his web request. This session ID is often stored as a hidden form element, part of the URL query string, or in a cookie. These methods cause the browser to send that same session ID back to the web server on all subsequent web requests. | 3 | 147 |
| **Cookie Manipulation** | The third method of setting state is to use a cookie. Cookies are usually the preferred method of saving state because you have a little more control over them. You choose whether they are session or persistent cookies, and you can set the secure parameter to indicate whether or not they're allowed to be sent over non-SSL encrypted channels. Cookies, however, remain relatively easy for an end user to manipulate using tools such as the Paros or Burp web proxies. | 3 | 148 |
| **Hidden fields** | Hidden form elements: The User session information is passed in the HTML itself but not displayed  Cookies : The user session information is stored by the browser as a cookie .  No matter what mechanism we use (URL, Hidden fields, or Cookies) the session state is the prime target of the attack. Session attacks can be very simple as convincing an application you logged in as another user. | 3 | 148 |
| **Hidden Fields** | The user session information is passed in the HTML itself, but not displayed:  • <INPUT TYPE="HIDDEN" NAME="Session" VALUE="22343"> • Can save source to the local drive and alter the session ID • Can modify session ID, on the fly, using a proxy (developer mode?) | 3 | 148 |
| **Session Information - Hacking** | URL Session Tracking: The session ID is passed with the URL | 3 | 148 |
| **URL Session Tracking** | • https://www.bank.com/acctbal.asp?sid=34112323 • Edit the session ID in the URL, enter another user's SID | 3 | 148 |
| **Session Attacks - Protection** | * Ensure session IDs are random and sufficiently long * Use an established session toolkit, don't home-grow your own * Use a tool to test the predictability of session IDs * Digitally sign or hash session IDs to confirm validity. * Store and pass only session IDs between the browser and server; store other session information in a database keyed by session ID. * If session information is sent to a client or stored in a cookie, encrypt it. Provide a new session ID immediately upon user authentication. * Have session IDs expire on logout or periodically timeout. | 3 | 149 |
| **File Integrity Checker** | A file integrity checker monitors the filesystem based on a number of preset rules and generates alerts when files are added, modified, or deleted out of compliance with those rules. | 3 | 150 |
| **Web application Monitoring** | Monitoring web content and file integrity. Understand that SIEM correlation is critical for timely detection of attacks. Check availability of web application components. Track performance and look for trends and anomalies. Examine web server log files regularly. Provide more scrutiny to website areas that publish user provided content. Verify that backend databases are properly protected and secured. | 3 | 150 |
| **Next-generation WAF** | Newer iterations of WAFs, sometimes referred to as "next-generation" WAFs, are deployed at the application level with integration with the web server or a web proxy, usually via a plugin. | 3 | 152 |
| **Traditional WAF** | There are two ways that WAFs are typically deployed. Traditional WAFs are deployed at the network level with traffic flowing from either a switch span port or inline via a tap. | 3 | 152 |
| **Web Application Firewall** | Provides ability to monitor and block HTTP/HTTPS traffic. Deployment options: Traditional WAFs deployed at the network level, receives traffic via switch span port or inline via a tap. Next generation WAFs deployed at the application level: integrated with the web server or a web proxy.  Many WAFs deployed in monitor mode only. | 3 | 152 |
| **Monolithic Architecture and Security Controls** | * Monolithic application architecture is one where all functionality required to perform duties exists on one tier. * Traffic from the client browser enters the load balancer over HTTPS on port 443 and traffic on other TCP ports is blocked. Security teams might also include a WAF, which can be used to perform rate limiting, block automated scanners, deny known bad IP addresses, and block common web application attack signatures. * The web application tier is responsible for performing all authentication, authorization, access control, entitlement, validation, and encoding logic for each request. Remember that whole OWASP Top Ten problem? * The database tier is responsible for data encryption, requiring client systems to properly authenticate to the server, and ensuring data is transmitted securely over TLS. | 3 | 153 |
| **Command Query Responsibility Segregation (CQRS)** | Other microservice architectures might be broken down even further by Command Query Responsibility Segregation (CQRS). Martin Fowler describes this pattern on his blog as one that has separate read, write, and delete APIs. The idea is that this pattern can isolate sensitive operations (e.g., delete) and reduce the risk of an unauthorized party invoking the API. In his blog, Martin also mentions that CQRS adds complexity, which can actually weaken security controls. | 3 | 154 |
| **Microservice Architecture** | It is when a web application is divided into a smaller independent service. Very different from monolith in a sense that it could be web application could be broken down further by command queries. | 3 | 154 |
| **Microservice Architecture attack surface** | The attack surface for a microservice architecture is huge, as every single component of a web application architecture could become vulnerable. | 3 | 155 |
| **Security Operations and Log Management** | * Logging Overview * Setting Up and Configuring Logging * Key Logging Activity | 3 | 160 |
| **Alert** | message usually sent to notify an operator | 3 | 162 |
| **Alerting** | Sending a signal to an operator (the signal alert is treated as a type of log as well) | 3 | 162 |
| **Auditing** | Sometimes used to name act of reviewing log records for the purpose of auditing | 3 | 162 |
| **Debug traces** | Special logs specific to application debugging or tracing | 3 | 162 |
| **Device/Log source** | A source of security-relevant logs | 3 | 162 |
| **Log File** | Collection of the above log records | 3 | 162 |
| **Logging** | The act of creating a log | 3 | 162 |
| **Logging terms and definitions** | * Message: indication that an even has transpired * Log file: collection of the above log records * Alert: message usually sent to notify an operator * Device/log source: a source of security-relevant logs * Logging: the act of creating a log * auditing reviewing log records * monitoring: real-time or near-real * alerting: sending a signal to an operator * debug traces: special logs specific to application debugging or tracing. | 3 | 162 |
| **Message** | Some system indication that an event has transpired | 3 | 162 |
| **Monitoring** | Real-time or near-real | 3 | 162 |
| **Logging data** | Every machine has the capability to write logs, more and more data to collect, each has a different format, logs can be overwhelming if they are not properly managed. | 3 | 163 |
| **Log Server – Operate** | 1. Forward logs to that server (syslog or agent) 2. Configure log retention 3. (Optional) parse and augment logs. | 3 | 164 |
| **Log Server – Prepare** | 1. Build a Linux server 2. Deploy it on the network 3. Allow inbound log ports: TCP 22 -SSH management, UDP 514 – least secure option, TCP 514 for TCP support, other ports such as log agents. | 3 | 164 |
| **Log collection Architecture** | Logs are either pushed to or pulled into a log relay or even collector. SIEM may be referenced as a log relay, log aggregator or a central collector. Using WEF is optimal as it keeps the full log format while other solutions may convert to windows logs to syslog. Once logs are received by the log collection service the logs are filtered, parsed, enriched, and sorted in a back-end storage service. | 3 | 165 |
| **Common Event Format (CEF)** | The top log is from a Palo Alto firewall and is in the Common Event Format (CEF) log format. It uses key-value pairs to identify fields. For example, src=1.1.1.1 has a key of src and a value of 1.1.1.1. The field name would be the key name, or src. | 3 | 166 |
| **CSV (Common Separated Values)** | CSV used to stand for Comma Separated Values but is more accurately referred to as Common Separated Values as the separator does not have to be a comma. | 3 | 166 |
| **JSON log** | The next log is a Suricata intrusion detection system (IDS) log. It is a JSON log. JSON is a log format that is like key-value pairs but supports identifying field types and supports fields that are subfields of other fields. For example, src\_port 51321 does not have double quotes around it. The lack of quotes signifies this is an integer. JSON is another log format that is widely being adopted. | 3 | 166 |
| **Key-value log format** | Key-value log formats do not require regex parsing as log solutions can simply read keys and assign values. As such, key-value formats are becoming standard in network equipment such as firewalls. | 3 | 166 |
| **Log Event Extended Format (LEEF)** | Other key-value log formats exist, such as the Log Event Extended Format (LEEF). Key-value log formats do not require regex parsing as log solutions can simply read keys and assign values. As such, key-value formats are becoming standard in network equipment such as firewalls. | 3 | 166 |
| **Log standards (Lack of accepted)**  **Palo Alto** | Palo Alto firewall is in the Common event format (CEF) log forma. It uses key-value pairs to identify fields. The field name would be the key name, do not require regex parsing as log solutions can simple read keys and assign values. As such, key-value formats are becoming standard in networks equipment such as firewalls. | 3 | 166 |
| **Log standards (Lack of accepted) Fedora (Linux)** | Linus default is syslog. Unfortunately, syslog is not a true format. It does include basic information like timestamp, host, and message but the message can be in any format. A log even may end up being written in 10 different formats even though it occurred on the same system for the same user. Syslog is slowly being replaced by key-value log formats or other methods such as CSV | 3 | 166 |
| **Log standards (Lack of accepted) Suricata** | Suricata intrusion detection system (IDS) it is a JSON log format that is like key-value pairs but supports identifying field types and supports fields that are subfields of other fields. Does not have double quotes around it and it means it is an integer. | 3 | 166 |
| **Log standards (Lack of accepted) Windows** | Windows logs are stored in a proprietary binary format, inside the binary log. It uses XML schema to associate fields and values. Windows log agents will read the XML schema and convert the logs into Jason, key-value pairs, or syslog before sending to a log management system or SIEM. Log agents that convert to syslog requires regex parsing in order to know what part of the log belongs to a given field. | 3 | 166 |
| **Output-driven** | Log only what you need to know | 3 | 166 |
| **Recommended logs** | Drop high-volume logs you do not need | 3 | 166 |
| **Syslog** | Linux systems default to using syslog as a log format. Unfortunately, syslog is not a true log format. It does include basic information like timestamp, host, and message, but the message can be in any format. The result is application developers each producing their own log formats. A login event may end up being written in ten different formats even though it occurred on the same system for the same user. Syslog is the de facto log standard, although, it is slowly being replaced by key-value log formats or other methods such as Common Separated Values (CSV). | 3 | 166 |
| **Windows logs** | Windows logs are stored in a proprietary binary format. Inside the binary log, Windows uses an XML schema to associate fields and values. Because the logs are binary, they are fast but require specialized agents to read the logs and to send them. Windows log agents will read the XML schema and convert the logs into JSON, key-value pairs, or syslog before sending to a log management system or SIEM. Log agents that convert to syslog requires regex parsing in order to know what part of the log belongs to a given field. | 3 | 166 |
| **Logs – how to manage logs** | Progressing in this order considers both the need to analyze certain log types and typical available expertise.   1. Firewalls, network gear 2. Other network security gear 3. Servers (UNIX and Windows) 4. Other server appl.(web, mail) 5. Databases 6. Applications (big challenge) 7. Desktop & workstations | 3 | 167 |
| **Logs – Key points** | Logs are hey organizational records, logs are produces by a huge number of components , logs often look different due to limited log standards, there is a certain logical sequence for tackling logs. | 3 | 168 |
| **Hybrid (recommended) log filtering** | Drop high-volume logs you do not need first. | 3 | 169 |
| **Input driven log filtering** | Logs and keeps everything | 3 | 169 |
| **Log filtering** | Logs come in large quantities with lots of information, and some data is irrelevant, is just noise of confusing to analysts, and some data is outside scope of compliance.  There are 3 methods: input driven, output driven and hybrid (recommended) | 3 | 169 |
| **Output-driven log filtering** | Log only what you know you need. | 3 | 169 |
| **Authentication and Authorization Reports** | See book | 3 | 170 |
| **Change Reports** | See book | 3 | 170 |
| **Log reports – best** | Top 7 are:   1. Authentication and authorization reports 2. Change reports 3. Network activity reports 4. Resources access reports 5. Failures and critical reports 6. Analytic reports, mostly using “Never Before seen” (NBS) | 3 | 170 |
| **Failures and Critical Errors Logs** |  | 3 | 171 |
| **Malware Activity Reports** |  | 3 | 171 |
| **Network Activity Reports** |  | 3 | 171 |
| **Resource Access Reports** |  | 3 | 171 |
| **Analytic Reports, Mostly Using "Never Before Seen" (NBS)** |  | 3 | 172 |
| **Never Before Seen (NBS)** |  | 3 | 172 |
| **Logging - Setting Up and Configuring** | What is needed to begin collecting logs, the various tools that can assist in the storage and reviewing of logs, and the importance of an organizational workflow for log management and analysis. | 3 | 173 |
| **Log Analysis Tools** | SEC, OSSEC, OSSIM, (Swatch, logwatch, logsentry), Elastic Stack: Elasticsearch, Logstash, Kibana | 3 | 174 |
| **Log collection tools** | Log collection tools are numerous and solve the problem of getting logs from where they are produced to where we need them. Examples are syslog-ng (general purposes logging + reliable delivery + a lot of convenience features), NXLog, Winlogbeat, Fluentd, and more. | 3 | 174 |
| **Log storage** | Log storage is not a challenge when we are talking about megabytes or even gigabytes, but multiple terabytes and then petabytes present a non-trivial challenge. One can always store logs in flat files (such as all the standard files in /var/log/syslog). Currently, major logging systems prefer non-relational storage such as Elasticsearch or Hadoop. There are multiple open-source and commercial log storage systems available to choose. It is not recommended to roll your own storage solution. | 3 | 174 |
| **OSSEC** | OSSEC is a leading open-source tool for real-time log analysis; it is being actively developed and contains a large list of rules, usable out of the box | 3 | 174 |
| **Pre-processing** | Pre-processing is a task of converting logs from one format to another. For example, NXlog helps convert multi- line logs to a single line format and can also convert between CSV, JSON, and other log formats. | 3 | 174 |
| **SEC** | SEC for rule-based correlation and analysis of logs in near-real-time; flexible but sometimes hard to use. | 3 | 174 |
| **Elastic Stack** | Elastic Stack is a large-scale enterprise solution for scalability, security, and log handling. It includes Elasticsearch for storage, Logstash for data ingestion, filtering, and enrichment, and Kibana for visualizations. It also supports multiple lightweight agents. It comes in multiple flavors including an open-source, free basic license, or commercial subscription. | 3 | 175 |
| **OSSIM** | OSSIM is a project to analyze and correlate logs; use them if your goals go beyond real-time alerting on logs. | 3 | 175 |
| **Swatch, logwatch, logsentry** | Swatch, logwatch, and logsentry are examples of literally hundreds of scripts that can look for specific strings in logs and then send an email or other alert. Many people choose to write their own instead of using others, which leads to a proliferation of such tools. | 3 | 175 |
| **Logs - How to Start Using the Tools** | 1. Collect logs Tools: Syslog-ng, standard syslog, log agents, and more 2. Store logs Tools: MySQL, Elasticsearch, and others 3. Search logs Tools: grep, Splunk, LogParser, Kibana, and more 4. Correlate and alert Tools: OSSEC, OSSIM, sec, nbs, elastalert, and others | 3 | 176 |
| **Log monitoring Strategy** | See book for diagram | 3 | 177 |
| **Log Monitoring Program: Phased Approach to Setting Up** | * it is important to understand what information is needed to drive the setting up and the configuration of the log monitoring program * Log types to collect * Networks/systems to collect logs from * Add more users of log data, expand scope * Plan retention * Plan response procedures * Potential problem areas and how to avoid them | 3 | 178 |
| **Log management deployment (Challenges to organization-wide)** | 1. Organization political boundaries 2. Data crossing network and state boundaries 3. Access to remote locations with data sources 4. Customer applications 5. Data retention and potential legal liability with storage of logs 6. Compliance and regulations. | 3 | 180 |
| **Log Aggregation and SIEM** | One of the best ways to begin creating a baseline on your network and systems is to aggregate all the logs on your network and correlate them for analysis of the behaviors on the network. Centralizing your logs in your network is a first step to begin aggregation for correlation and analysis. After centralizing the logs, you should copy them so that the copies of the original logs are normalized and correlated. . When your logs are normalized, you can then begin to correlate across multiple systems and platforms to gain an understanding of what is happening across the environment throughout the day. This helps you quickly discern whether there are anomalies. The solution used to do this aggregation and correlation is called a SIEM, which stands for Security Information and Event Monitoring. Using UTC is best if spanning multiple time zones. | 3 | 181 |
| **SIEM (Security Information and Event Monitoring)** | When your logs are normalized, you can then begin to correlate across multiple systems and platforms to gain an understanding of what is happening across the environment throughout the day. This helps you quickly discern whether there are anomalies. The solution used to do this aggregation and correlation is called a SIEM, which stands for Security Information and Event Monitoring. | 3 | 181 |
| **SIEMs: Overview** | Security information and Event monitoring. Can aggregate all the logs in your environment if you want them to be managed on a central log server. Then they will parser, normalizes, and correlate them for you or enamel you to correlate the logs. Their purpose is to assist you in dinging current or emerging threats in the monitored environment, whether there are security-related or resource-related, such as disk filling up. | 3 | 181 |
| **Sigma** | High-level generic language for analytics. MISP compatible: share and store aligned with threat intel. Decouples rule logic from SIEM vendor and field names. Eliminates SIEM tribal knowledge. | 3 | 183 |
| **Sigma Format** | Sigma rules are written in YAML format. The Sigma rule in this slide is an example showing that each rule has a title, status, and logsource, as well as information about the rule, such as a description and tags. The format allows for ease of readability and customization. | 3 | 184 |
| **Sigma - Conversion of Signatures to Alert Queries** | To make Sigma rules work with a SIEM, they must be converted using sigmac. The process looks as follows:  Write rule in Sigma Format  Create a file mapping generic field names to field names within a given SIEM platform  Use sigmac to convert from Sigma format to a specific SIEM product rule format | 3 | 185 |
| **Sigma - Sharing Rules** | Because Sigma rules are a generic format, they are perfect for sharing. Projects such as MISP make this a simple task. Now, organizations can create custom rules and share them openly with anyone, or strategically with specific partners or industries. | 3 | 186 |
| **Sigma orchestration** | 1. Pull shared rule from MISP 2. Convert Sigma rule with sigmac to a SIEM specific format. 3. Test rule outside alert engine by seeing if it would match on existing data. 4. If matches are found, move to a location for manual assessment and open a ticket. | 3 | 187 |
| **Key Logging Activity** | Examples of detections and the specific actions to be taken as they relate to specified periods of time. | 3 | 188 |
| **Logging - Real-time tasks** | Outbreaks of malware, intrusion evidence that is reliable, significant abuse of the internal network, service loss to assts deemed critical, regulated data theft. | 3 | 189 |
| **Logging – Daily tasks** | Configuration changes that aren’t authorized, service disruption, evidence of intrusions, login failures deemed suspicious, malware activity: minor to medium and summary of activities | 3 | 190 |
| **Logging – weekly tasks** | Review of all log trends, routine create and removal of accounts, device and network changes, and summary of less critical probes and attacks. | 3 | 192 |
| **Logging – monthly tasks** | Review long-term network and system log trends, summary of minor policy violations, various resource usage reports, security technology performance measurement. | 3 | 193 |
| **Logging – quarterly tasks** | Create audit reports (for quarterly audit), review longer-term trends across log data, review infrastructure changes that affect log monitoring and review log management system performance. | 3 | 194 |
| **Logging – annual tasks** | Review log policy, verify log retention and archival, review longest-term trends across log data, use logs to analyze and prepare for next year’s security budget, review possible new regulations that affect logging. | 3 | 195 |
| **Logging - How Logs Help** | * Logs help to figure out the who, where, when, how, what, and more But... * Who: the system or person in question? * Where: is it legitimate? * When: is the time zone correct? * How: what was the vector? * What: what exactly happened and what was logged? | 3 | 196 |
| **Logging – how they help** | * Logs help to figure out the who, where, when, how, what, and more. * Who; the system or person in question: * Where: is it legitimate? * When: is the time zone correct? * How: what was the vector? * What: what exactly happened and what was logged? | 3 | 196 |
| **Digital Forensics** | It is considered the application of science to the identification, collection, examination, and analysis of data while preserving the integrity of the information and maintaining a strict chain of customer for the data. | 3 | 201 |
| **Digital Forensics –**  **What Is Digital Forensics?** | The National Institute of Standards and Technology, or NIST, defines digital forensics (DF) as "the application of science to the identification, collection, examination, and analysis of data while preserving the integrity of the information and maintaining a strict chain of custody for the data." | 3 | 201 |
| **Digital Forensics: who uses it?** | 1. Infosec teams: detect and respond to breaches, in house organizational support, external third parties. 2. DoD/intelligence community: bad guy elimination, DoD media exploitation, profile and expose attackers. 3. Law enforcement: bad guy elimination, support to national interest that may not see court. Work with prosecutors where appropriate. 4. Legal support: prosecute or defend the bad guy, civil/criminal cases, expert witness, often more eDiscovery than forensic. | 3 | 202 |
| **Digital Forensics –**  **The Investigative Process** | Remember, forensics is a science. One of the most effective ways to conduct a thorough forensic investigation is to go back to grade school—to the scientific method! | 3 | 203 |
| **Follow the evidence** | 1. User activity: web browsing, downloading files, connecting USB drives, visiting malicious websites. 2. Creates evidence: browsing history, disk artifacts, event logs, shortcut files, network artifacts. 3. Recovered by digital forensics: data carving/ recovery, packet carving, artifact parsing, timelining , malware analysis. | 3 | 204 |
| **Chain of Custody (CoC)** | The CoC maintains the history of a piece of evidence, including the first person who acquired it, all the way to the person who has it currently. | 3 | 205 |
| **Digital Forensics –**  **Remaining Forensically Sound** | Detailed Documentation:  • Chain of Custody  Start/End Times  Commands Typed.  Forensically Sound Methods:  Write-blockers  Read-only files  Protected storage environments | 3 | 205 |
| **Write-blockers** | Examiners will often use write-blockers to interact with digital evidence such as hard drives, USB drives, or files stored on an external drive. These tools allow for the access of digital evidence but avoid writing any additional data to the disk. Write-blockers actually have hardware limitations that prevent an analyst from impacting data. | 3 | 206 |
| **Digital Forensics Artifacts: Examples** | * File Download (Purple): How files were introduced to a system * Program Execution (Red): Details about which programs ran on the system * File/Folder Opening (Green) Which filesystem objects were viewed or opened * Deleted File or File Knowledge (Black): Details regarding what was searched for, or file metadata items that remain even if the file is completely deleted from disk * Physical Location (Yellow): Geographic data points * External Device/USB Usage (Blue): All activity regarding the use of removal of devices, typically performed by way of USB. * Account Usage (Gray): Local or remote login/logout data and other session authentication details * Browser Usage (Orange): Artifacts that can be traced to a web browser | 3 | 207 |
| **$FILENAME** | timestamps (on poster) | 3 | 208 |
| **$STDINFO** | Timestamps | 3 | 208 |
| **Digital Forensics Artifacts: Filesystem Timestamps** | There are two groups of four timestamps for each file and directory on the NTFS filesystem. One group, the "$STDINFO" timestamps, is shown here. (The other group, the "$FILENAME" set, will not be discussed here but is detailed on the poster itself.) The four timestamps and their alpha designators are:   * Modified Time (M): The last time the file contents were modified * Access Time (A): The last time the file was accessed by a process * Metadata Change Time (C): The last time the file's metadata (permissions, filename, and so on) was changed * Creation Time (B): The time the file was created Together, these are referred to as the "MACB" (mac-bee) times. | 3 | 208 |
| **MACB times** | The four timestamps and their alpha designators are:  • Modified Time (M): The last time the file contents were modified  • Access Time (A): The last time the file was accessed by a process  • Metadata Change Time (C): The last time the file's metadata (permissions, filename, and so on) was changed  • Creation Time (B): The time the file was created Together, these are referred to as the "MACB" (mac-bee) times | 3 | 208 |
| **Digital Forensics; battle against time** | Operating systems don’t store artifacts forever, some may be cleared due to size limitations or maximum system quantity, forensic examiners have tricks they can use to peer “back in time” within a system. | 3 | 209 |
| **Order of Volatility** | Operating systems don't store artifacts forever! Some may be cleared due to size limitations or maximum system quantity. Forensic examiners have tricks they can use to peer "back in time" within a system. RAM memory (shortest) ---> Hard drive (longest) | 3 | 209 |
| **DFIR Subdisciplines** | DIFR has multiple subdisciplines that may provide different artifacts and types of evidence but work together to define a security incident. Ex: Endpoint, Network, threat intelligence and reverse engineering. | 3 | 210 |
| **Endpoint (DFIR)** | This type of forensics and analysis focuses solely on artifacts originating from a computer system, or the endpoint. Endpoint artifacts fall into two categories: disk-based and memory. We will examine memory next but know that forensic analysts can pull a wealth of artifacts from disk images. | 3 | 210 |
| **Endpoint (Memory) (DFIR)** | Memory forensics is quickly becoming its own primary investigative technique as analysts find more and more value in system memory. Many forensic collections these days include disk and memory artifacts, especially as attackers find more and more ways to hide from investigators. | 3 | 210 |
| **Event** | An "event" is any observable occurrence in a system and/or network  Examples of events include:   * The system boot sequence * A system crash * Packet flooding within a network * All incidents are composed of events, but not all events are incidents | 3 | 210 |
| **Network (DFIR)** | Network forensics focuses on artifacts generated by network traffic. This may include data such as NetFlow (IP traffic data) or full-packet capture (a carbon copy of network traffic), both of which can be used to determine IP traffic flows and potentially inspect network traffic content. Collection of network data has often been overlooked by many organizations but is quickly growing in popularity as many security teams realize the need for networking data to understand the scope of an incident. | 3 | 210 |
| **Threat Intelligence (DFIR)** | Threat intelligence (TI) is a useful subdiscipline for incident response teams to help guide investigations and understand the type of attacker that may be behind the incident. Threat intelligence teams can compile reports and indicators on attacker tools, tactics, and procedures, also known as "TTPs." These reports can help guide the team toward what the next target may be or how the attacker may try to remove data from the network. | 3 | 210 |
| **Reverse Engineering (DFIR)** | Reverse engineering is an advanced skillset that allows team members to gain a deep understanding of malware, malicious code, and other attacker files/tools. Reverse engineers will typically serve as support for an incident response team, or reverse engineering may be a skill shared among senior IR team members. | 3 | 211 |
| **Digital Forensics tools** | Open-source tools: the sleuth kit/ autopsy, custom Perl/PowerShell/python scripts.  Standalone virtual machines: SANS SIFT workstation, REMnux.  Proprietary/closed source tools: EnCase, X-Ways. | 3 | 212 |
| **Digital Forensics tools: SIFT** | Is a free virtual machine full of all sorts of digital forensics and incident response tools, as well as commonly used SANS references and cheat sheets. | 3 | 213 |
| **Incident handling fundamentals** | Is an action plan for dealing with intrusions, cybertheft, denial of service, malicious code, fire, floods, and other security-related events. Incidents can be intentional or unintentional. Incident response plans help to know what to do when and incident occurs, sooner or later and incident is going to occur. Incident handling plans are similar to having car insurance. Planning is everything. | 3 | 216 |
| **Damage** | "damage" means "impairment to the integrity or availability of data, a program, a system, or information". | 3 | 217 |
| **Incident** | • An "incident" is an adverse event in an information system and/or network or the threat of the occurrence of such an event  • Incident implies harm or the attempt to do harm  • Incident handler reduces or minimizes harm  • Depending on circumstances, a single event could or could not be an incident. All incidents are composed of event but not all events are incidents. Ex: the system’s boot sequence, a system crash, packet flooding within a network | 3 | 217 |
| **Incident handling process – 6 steps** | Often called PICERL has 6 steps:  Preparation, identification, containment, eradication, recovery, and lesson learned. | 3 | 218 |
| **PICERL** | Preparation, Identification, Containment, Eradication, Recovery, Lessons Learned | 3 | 218 |
| **Incident response process – common problems** | Poor security hygiene, lack of visibility and threat intelligence, little scoping (if any), leads to incomplete containment, not fixing the vulnerabilities, and failure to apply lessons learned. | 3 | 219 |
| **PICERL - Common Problems** | A deeper problem lies in a linear (static) approach to incident response   * Poor security hygiene * Lack of visibility and threat intelligence * Little scoping (if any) * Leads to incomplete containment * Not fixing the vulnerabilities * Failure to apply lessons learned | 3 | 219 |
| **Incident response dynamic approach** | There is no “recipe” for incident response, one of the reasons is that incidents are non-linear, and they don’t happen as a single set of events. There are multiple events happened, in the environment when an incident occurs. | 3 | 220 |
| **Incident Response –**  **preparation** | Know thy organization: what is important to the business? What policies and procedures are in place?  Internal visibility is key for responding to incidents: this will often form the bulk of evidence during an investigation, network, and host visibility are both important, who reviews loges? How often? Identify existing resources and gaps can be an eye-opening exercise.  Have plans for rebuilding systems: often the most cost-effective way to eradicate rootkits. | 3 | 221 |
| **Preparation** | The preparation step is the first and most critical step of the incident handling process. The tasks associated with this step must be performed in advance—before the incident has occurred. This is the reason why it is often overlooked—or even skipped. It is recommended that you spend enough time preparing all the elements that are required during an incident, with the goal of increasing the efficiency and success of your incident handling efforts. | 3 | 221 |
| **Identification, Detection, Verification, & Triage** | Many possible sources for detecting an incident:   * Network devices (e.g., firewall, IDS) * Host devices (e.g., system logs, application logs * Admins and users noticing something isn't right * Threat intelligence feeds • Third-party notification. The first decision is always verification: * Do you have an incident, or is it a false positive? * Sometimes easy, such as a website defacement * Sometimes difficult, requiring a full forensic examination. It's also a good idea to triage: * What type of incident is it? * Affects how you respond | 3 | 223 |
| **Containment** | Goal is to stop the threat actor from continuing to operate in the environment:   * Lateral movement and persisting in the environment are not uncommon * Requires proper scoping Examples: * Isolating a system * Patching systems * Removing attacker processes, accounts, etc. * Applying filters to routers and/or firewalls * Changing entries in DNS May help meet other goals such as eradication Containment may occur in multiple phases: * business decision (e.g., resources, focus) * Evidence collection | 3 | 225 |
| **Incident Response –**  **Containment** | Goal is to stop the threat actor from continuing to operate in the environment, lateral movement and persisting in the environment are not uncommon, requires proper scoping, may help meet other goals such as eradication. Containment may occur in multiple phases such as business decisions, evidence collection. Examples: isolation a system, patching systems, removing attacker’s processes, accounts, etc. Applying filters to routers and/or firewalls, changing entries in DNS. | 3 | 225 |
| **Backdoor** | Attackers often try to establish additional ways of ensuring remote access to the compromised system, so they have control of it even if the vulnerability exploited originally is fixed. Such backup access methods are known as "backdoors" and are implemented using several methods. Some of the most common ones include a process of listening on a specific port and offering shells access (without requiring authentication), creating a new user account with high privileges, and scheduling jobs that periodically run programs that open new paths to access the system. As an incident handler, you need to not only fix the vulnerability used during the initial system compromise but also identify and remove every additional backdoor left by the attacker. | 3 | 226 |
| **Eradication** | * "Undoing" the threat actor actions * Containment is about stopping control * Eradication is about removing Examples * Restoring systems from trusted backups * Removing attacker processes, accounts * Performing a vulnerability assessment May also help meet goals such as recovery * Dealing with fraudulent transactions * codified source code | 3 | 226 |
| **Incident Response –**  **Eradication** | Undoing the threat actor actions, by restoring systems from trusted backups, removing attacker processes, accounts, performing a vulnerability assessment. May also help meet goals such as recovery by dealing with fraudulent transactions, modified source code. | 3 | 226 |
| **Incident Response –**  **Recovery** | Business needs to get “back up and running”, if the compromised systems were not used, much easier. Rebuilding a system is often the most cost-effective, however is not always possible or may consider shorter-term containment options as a band-aid to buy time. Try to bring systems online during off-hours, because it is easier to monitor but it is a business decision, and you may be overruled. | 3 | 227 |
| **Flat Network** | A "flat network" could have contributed to the ease in which an attacker was able to move around. This is when the internal design of the organization is such where there are no protected enclaves and controls preventing inside users from reaching sensitive systems such as those in a data center. Fixing this contributing factor may require a complete re-architecture of the network. | 3 | 228 |
| **Incident Response –**  **Remediation** | Try to identify and fix the root cause, ex: compromise due to a weak password, containment is changing the password, why was the password allowed? Lack of enforcement, weak policy? It can be difficult if. Multiple factors were involved. | 3 | 228 |
| **Incident Response –**  **Lesson learned** | Develop a final report, this is usually not the only documentation you develop. Right after an incident there is a lot of “momentum” and the impact fades over time. Schedule a follow-up review to see what has been implemented. Has the organization been compromised again? Original compromise? Timeline varies but 30,60, and 90-day increments are not uncommon. | 3 | 229 |
| **Incident Handling - Key Mistakes** | Some key mistakes that are common in many organizations are listed here:   * Failure to report an incident or ask for help * Incomplete or nonexistent notes * Mishandling or destroying evidence * Failure to create working backups * Failure to contain or eradicate the incident * Failure to prevent reinfection * Failure to apply lessons learned | 3 | 230 |
| **Incident Handling - Taking Notes** | Critical. Beyond the aforementioned benefits, note-taking allows others to see the incident through the eyes of the handler. It is important that the steps captured, and the outcomes reached be based on facts, staying clear of any bias or unprofessional qualities. | 3 | 231 |
| **Incident Handling Steps** | 1. Preparation: It is essential to plan for the eventuality of an incident. Remember, an incident will happen; it's simply a matter of time. 2. Identification: This is the ability to distinguish between an event and incident. Staying current on potential vulnerabilities and exploits is a critical step in being able to identify an incident on your system. 3. Containment: You must isolate the incident to prevent it from spreading or causing more damage to the organization. This might also involve gathering information to be used as evidence or making the decision to pull a system from the production network. 4. Eradication: Eliminating the source or cause of the incident is an integral part of the incident handling process. 5. Recovery: This is the restoration of service and turning over the affected system back to the system owner or the administrator. The incident handler should take all precautions necessary to ensure the system is fully recovered before returning it back to the production network. 6. Remediation: Validate that the root cause has been identified and closely observe the environment. 7. Lessons Learned: Conducting a follow-up meeting after the incident is critical to understanding what happened and why, and to ensure that the proper steps were taken to prevent similar incidents from occurring | 3 | 232 |
| **Threat hunting – incident response.** | Threat hunting is using knowledge of attacker activities and methodologies to search for historical, previously unknown events within an organization. Results from threat hunting also typically turn into additional activities for the digital forensic team, as they will likely turn up events or systems to be investigated. | 3 | 233 |
| **Remediation** | Closely monitor any compromised systems since a rootkit or backdoor suggest intent to return, monitor network sensors, host, and application logs, look for incident-specific indicators. | 3 | 288 |
| **Cryptography** | Cryptosystem fundamentals. General types of cryptosystems   * Symmetric * Asymmetric * Hashing. * Steganography. | 4 | 4 |
| **Algorithm** | Is defined as a series of instructions (often mathematical) that can be processed, in a specific order by a computer system | 4 | 6 |
| **Cryptanalysis** | Is often seen as the science that studies the methods used by cryptographic systems in an effort to try and defeat them. | 4 | 6 |
| **Cryptography** | Is often referred to as a science that is concerned with the methods of how to hide the meaning of a communication. Hidden writing | 4 | 6 |
| **Cryptology** | science concerned with data communication and storage in secure and usually secret form. It encompasses both cryptography and cryptanalysis. | 4 | 6 |
| **Ciphers** | It is also common to see encryption algorithms referred to as ciphers. Ciphers are a concept of describing the fact that a message has been written in a secretive way and, as such, are not directly (but are somewhat analogous) to algorithms. | 4 | 7 |
| **Ciphertext** | Message in its encrypted form | 4 | 8 |
| **Cryptography: components** | Cryptography, decryption, encryption, plaintext, ciphertext. | 4 | 8 |
| **Decryption** | Process of transforming an encrypted message into its original form. | 4 | 8 |
| **Encryption** | Coding a message so that its meaning is concealed. Describes the transformation of plaintext into ciphertext. | 4 | 8 |
| **Plaintext** | Message in its original form | 4 | 8 |
| **Cryptosystems** | A cryptosystem is the collection of all possible inputs and all possible outputs, in addition to the algorithm and keys. But don't forget about the humans. | 4 | 9 |
| **Cryptographic keys** | Cryptographic keys are simply values used to initialize a particular algorithm. The important aspect of keys in regard to cryptosystems is that only the key, not the algorithm, needs to be protected. This means that algorithms might be widely distributed, and their internal workings publicly documented. Only the key must be protected from thievery by communicating entities. | 4 | 10 |
| **Keys – Cryptographic** | Keys permit the existence of unrestricted algorithms and might be any one of a large number of values. The strength of a cryptosystem rest with the strength of its keys. Key space matters. | 4 | 10 |
| **Keyspace** | Is a critical concept concerning cryptographic keys, the larger the Keyspace, the less likely an attacker is to discover a give key through brute force. | 4 | 10 |
| **Key protection** | The longer the length of the key, the more possible combinations an adversary must work through (brute force) they length won’t protect us if the key isn’t protected ( ad adversary discovers the key or forcers its disclosure). Keys must be in some way accessible toa computing system in order to be used for encryption and decryption, but is a computer can access the key, the adversary could too. Keys must be protected, yet usable at the same time. | 4 | 12 |
| **Ciphers – types** | See book for the diagram. | 4 | 13 |
| **Cryptography - Types** |  | 4 | 13 |
| **Enterprise Crypto: The Big Picture** | Symmetric stream ciphers are fast, and asymmetric factoring algorithms are slow. Diffie-Hellman is great for secure key exchanges, but not necessarily optimal for encryption. This means that cryptography in the enterprise is a multifaceted endeavor. Different types of cryptography are used for different types of situations and, often, cryptographic systems are employed in concert. | 4 | 13 |
| **Cryptography - 3 Core Components** | There are three core components that we must address in implementing cryptography. First, we must make sure the information is protected at rest. Second, we must make sure the information is protected at transit. Finally, we must make sure the keys are properly protected and managed. If any one of these three areas is neglected, the effectiveness of our crypto deployment is degraded, and we are leaving the door open for the adversary. | 4 | 14 |
| **Cryptography - Challenges** | Communication in the presence of adversaries ... Confidentiality – Integrity – Authentication – Non-repudiation | 4 | 14 |
| **Cryptography: implementing** | We must make sure the information is protected at rest.  We must make sure the information is protected at transit  We must make sure the keys are properly protected and managed | 4 | 14 |
| **Cryptography: Goals** | 1. Confidentiality: Symmetric 2. Authentication : Asymmetric (knowledge of private keys) 3. Integrity: Hash 4. Non-repudiation: Digital Signature (Asymmetric + Hash) | 4 | 15 |
| **Hybrid**  **(Encryption)** | The main goal of encryption is to garble text so that a third party cannot understand it. Two basic methods of encrypting or garbling text are substitution and permutation. A third approach is actually a hybrid, which is a mixture of both | 4 | 16 |
| **Permutation** | The main goal of encryption is to garble text so that a third party cannot understand it. Two basic methods of encrypting or garbling text are substitution and permutation. A third approach is actually a hybrid, which is a mixture of both | 4 | 16 |
| **Substitution** | The main goal of encryption is to garble text so that a third party cannot understand it. Two basic methods of encrypting or garbling text are substitution and permutation. A third approach is actually a hybrid, which is a mixture of both | 4 | 16 |
| **Symmetric encryption techniques** | The goal is to scramble the original message so that its meaning is concealed. These techniques are used by the symmetric key systems   * Basic techniques: substitution, XOR, Rotation. * Arbitrary substitution: permutation, hybrid. | 4 | 16 |
| **XOR** | Substitution. The Boolean exclusive OR (XOR) function is one of the fundamental operations used in cryptography. The output of an XOR is TRUE if exactly one of the inputs is TRUE; otherwise, the output is FALSE | 4 | 17 |
| **Rotation Substitution** | It "rotates" the alphabet by X characters, where X is the key. Easy to remember; for example: Plaintext: A B C D E ..... Ciphertext: D E F G H ..... So, "CAT" becomes "PNG". It uses a one-to-one substitution of characters. Caesar Cipher was ROT-3. Usenet uses ROT-13 (symmetric). | 4 | 18 |
| **Arbitrary Substitution** | It uses one-to-one substitution of arbitrary characters. Give one character mapping you cannot determine the key as with rotation substitution. You can see repeated use of more than one character in the ciphertext word, and the ciphertext is the same length as the cleartext. | 4 | 20 |
| **Character Frequency Analysis** | Monolithic ciphers or one-to-one substitution ciphers can often be broken with frequency analysis | 4 | 21 |
| **Frequency Analysis** | Monolithic ciphers or one-to-one substitution ciphers can often be broken with frequency analysis | 4 | 21 |
| **Permutation** | Substitution. Permutation, also called transposition, shuffles the order in which characters (or bytes) appear rather than substituting one for another. The letters in the ciphertext are the same as the plaintext; they are just in a different order. | 4 | 22 |
| **Cryptosystems: types**  **Number of keys** | There are 3 types:  Symmetric: Secret key, single or 1-key encryption  Asymmetric: public key, Dual or 2-key encryption  Hash: one way transformation | 4 | 24 |
| **Pre-shared Key (PSK)** | Requires a secure channel? | 4 | 25 |
| **Symmetric Key Cryptosystems** | Secret key encryption   * Fast, single key for encryption and decryption * Requires secure key distribution channel (scalability) * No technical non-repudiation * Requires a secure channel: * Pre-shared key * Asymmetric encryption * Diffie-Hellman Key exchange. | 4 | 25 |
| **Asymmetric cryptography: secure channel.** | For asymmetric cryptography, you can used either public or private and it comes down to the size of the message. If you are going to encrypt small messages then it would be ok to use a private key. However, if you are going to be encrypting large messages, it would be best to use a public key. | 4 | 26 |
| **Public Key Technology** | Asymmetric protection to exchange key, then communicate using symmetric protection. Due to speed differences in encryption. | 4 | 26 |
| **Diffie-Hellman Key exchange secure channel** | Two-key Crypto, asymmetric algorithm was developed, and it is used for key exchange. this method provides a mechanism whereby 2 people cane exchange a key on a network with someone observing all their communications. Essentially, it allows 2 parties to exchange a secret Key in the presence of an adversary over a nonsecure network | 4 | 27 |
| **Asymmetric Key Cryptosystems** | * Public key encryption * Slow * Public/private key pair * Public keys widely distributed within digital certificates * Used as a secure channel for symmetric key exchange * Technical non-repudiation via digital signatures. | 4 | 29 |
| **Hash Functions** | **Primary use:**  Message integrity!  **No key used during encryption:**  • Irreversible one-way transformation • Plaintext (and length of plaintext) is not recoverable from the hash  • **Examples:** MD2, MD4, MD5, RIPEMD-160, SHA-1, SHA-2, and SHA-3  • Some algorithms have issues with predictable collisions  • Also called "message digests" or "one-way encryption" | 4 | 30 |
| **Hash collisions** | Where 2 different inputs are hashed and produce the same output. In order to reduce the risk of a collision occurring, prove hashing algorithms will ensure that since the hashes produce a fixed length output, the number of possible inputs to a hash algorithm could be larger than the number of outputs. If the number of possible inputs exceeds the number of possible outputs, a collision is a given. Similar inputs will result in drastically different outputs (regardless of how similar the inputs are). There is no way to predict when a collision will occur. | 4 | 32 |
| **Integrity (hashing)** | When someone is concerned only with the message’s integrity or with third party eavesdropping, they input the message into a hashing algorithm of her choice, and it will output a hash of the message. then both the message and the hash (fingerprint) is transmitted and if the receiver confirms the hash, it means that the message has not changed in transit. | 4 | 33 |
| **Digital Signatures** | Digital signatures use public key cryptography to "sign" documents. The signatures are non-repudiable. They "sign" a document by encrypting a hash with a private key. | 4 | 34 |
| **Digital Signature - Example** | To digitally sign the document (message), Alice employs her private key to encrypt the hash. Upon receiving the message and Alice's digital signature, Bob employs Alice's public key to decrypt the signature. By decrypting the signature, Bob retrieves the hash of the message as originally generated by Alice. Bob then generates a hash (fingerprint) of the message he received and compares it to the received hash (the one he decrypted from the 'signature'). | 4 | 36 |
| **Steganography (Stego)** | Conceals the fact that you are sending additional (usually secret or sensitive) information. Data hiding (steganography means "covered writing"). Dates to Ancient Greece; modern awareness is relatively new. Can hide in a variety of formats:   * Images (bmp, png, gif, jpg) * Word documents * Text files * Machine-generated images (fractals) | 4 | 39 |
| **Crypto versus Stego** | (crypto) provides confidentiality but not secrecy. It is fairly easy to detect that someone is sending an encrypted message; it is just difficult for someone to read it. With stego, you might not even know someone is sending a message; the true intent is hidden. | 4 | 40 |
| **Crypto Vs. Stego** | Crypto provides confidentiality but no secrecy. It is fairly easy to detect that someone is sending an encrypted message, it is difficult for someone to read it. With stego, you might not even know someone is sending a message, the try intent us hidden. Note, stenography doesn’t guarantee safety. | 4 | 40 |
| **Inference Attack** | An encrypted conversation can also raise suspicions. If two parties suspected of a crime had suddenly started trading extensive encrypted messages the week before the crime occurred, even though we might not know what they were saying, it would definitely raise some flags and concerns. This is known as an inference attack. You do not know all of the facts, but you can infer something is happening. | 4 | 40 |
| **Cryptography - Detecting** | Non-readable, Histograms | 4 | 41 |
| **Histograms** | "Normal" text - non-uniform. Encrypted text - very flat | 4 | 42 |
| **Steganography - How it Works** | There are two general components of standard steganography. The first is the carrier or host file. This is the medium used to hold the hidden data. The second component of stego is the 'how' of the message being hidden within the host file. | 4 | 43 |
| **Stenography components** | 2: the carrier and the host file.  The carrier can be any type of file such as images: BMP, gif, and jpeg. Word documents, sound files, movies: mpeg, test documents, machine-generated images: fractals, and HTML files. The second component of stego is the “HOW” the message being hidden with the host file. | 4 | 43 |
| **Stenography: how it works** | Stego requires a host (to carry the data) and the hidden message. Host usually a file can be generated on the fly or use existing data. Message can be hidden in certain parts of existing file or can cause a new file to be generated. | 4 | 43 |
| **Stenography methods** | Injection, substitution, file generation. | 4 | 44 |
| **Injection Stego** | We place the information into "holes", or unused areas of the file. | 4 | 45 |
| **Substitution Stego** | Most popular. The goal with this technique is that only insignificant data should be overwritten to prevent degradation. one of the most common forms is the least significant bits (LSB) in the color table of a graphic. | 4 | 46 |
| **File Generation Stego** | This is the only form of stego where a carrier isn't needed beforehand. | 4 | 47 |
| **Generate a new file (stego)** | The hidden data can also be to generate a new file, no host file is needed, and for example the input text can be used to generate fractals or “human-like” text. | 4 | 47 |
| **Steganography - Detecting** | There is no known universal method to detect steganography.  Detection typically works on a tool-by-tool basis  Common way with images is by analyzing the use of the least significant bit  Tools to detect Stego   * StegExpose * StegSecret | 4 | 48 |
| **Cryptography - Concepts in (1)** | What if: We can find a mathematical "problem" that exhibits characteristics of one-way functions (with trapdoors)? Or, as mathematicians would prefer to say, a problem that is "impossible" to solve in polynomial time? | 4 | 56 |
| **Cryptography: Concepts** | 4 main goals are confidentiality, integrity of data, authentication, non-repudiation. | 4 | 56 |
| **Trapdoor** | One-way mathematical functions. Such functions can have trapdoor properties that make them well-suited for public key cryptography, in which the trapdoor allows a message to be decrypted using a different key than the one used to encrypt the message. If the public key were used to encrypt the message, the trapdoor, in this case, is the corresponding private key. | 4 | 56 |
| **Big-O Notation** | Big-O notation is used to give a general idea of how many operations a problem takes relative to the input size n. The big-O function isn't usually specifically defined; it is mostly used as a notational shorthand to indicate a problem's complexity. | 4 | 57 |
| **Computational complexity** | Deals with time and space requirements for the execution of algorithms. | 4 | 57 |
| **Cryptography - Concepts (2)** | Computational Complexity deals with time and space requirements for the execution of algorithms. Problems can be classified as tractable or intractable. | 4 | 57 |
| **Intractable** | Hard problems. Can't be solved in polynomial time. Examples:   * Exponential or super-polynomial problems * Factoring large integers into primes (RSA) * Solving the discrete logarithm problem (ElGamal) * Computing elliptic curves in a finite field (ECC) | 4 | 57 |
| **Polynomial Time** | Relatively easy problems (symmetric encryption) can be solved in polynomial time—that is, the relationship between the input size and the number of operations required to solve the problem is constant, linear, quadratic, cubic, and so on. | 4 | 57 |
| **Problems** | Can be classified as tractable or intractable. | 4 | 57 |
| **Problems: intractable** | Hard problems cannot be solved in polynomial time. EX: exponential or super-polynomial problems, factoring large integers into primes (RSA), solving the discrete logarithm problems (ElGamal), computing elliptic curves in a finite field (ECC). | 4 | 57 |
| **Problems: tractable** | Easy problems can be solved in polynomial time for certain inputs.  Ex: constant problems, linear problems, quadratic problems, cubic problems | 4 | 57 |
| **Intractable Problem - RSA** | Difficulty of factoring large integer into its two prime factors | 4 | 59 |
| **Intractable problems: Factoring integers** | A difficulty of factoring large integers into its 2 prime factors. A hard problem, years of intense public scrutiny suggest intractability. no mathematical proof so far. based on difficulty of factory a large integer into its prime factors. 1000 times slowed than DES. Considered secure, de-facto standard. Patent expired in 2000. | 4 | 59 |
| **RSA** | Difficulty of factoring large integer into its two prime factors. | 4 | 59 |
| **ElGamal** | Cipher built upon the intractability of the discrete logarithm problem over finite fields. | 4 | 60 |
| **Intractable Problem - ElGamal** | Difficulty of solving the discrete logarithm problem—for finite fields | 4 | 60 |
| **Intractable problem: Discrete logarithm for finite fields** | The difficult of solving a discrete logarithm problem – for finite fields. A hard problem. Years of intense public scrutiny suggest intractability. No mathematical proof so far, the discrete logarithm problem is a difficult as the problem of factoring a large integer into its prime factors. Ex: ElGamal encryption and signature schemes. Diffie-Hellman key agreement scheme. Schnorr signature scheme, NIST’s digital signature algorithm (DSA) | 4 | 60 |
| **Elliptic Curve Cryptosystems (ECC)** | Difficulty of solving the discrete logarithm problem—as applied to elliptic curves | 4 | 61 |
| **Intractable problem: Discrete logarithm as applied to elliptic curves** | A hard problem. Years of intense public scrutiny suggest intractability. No mathematical proof so far, in general, elliptic curve cryptosystems (ECC) offer higher speed, lower power consumption and tighter code. Ex: elliptic curve ElGamal encryption and signature schemes. Elliptic curve diffi-hellman key agreement scheme. Elliptic curve schnorr signature scheme. Elliptic curve NIST’s digital signature algorithm (DSA). | 4 | 61 |
| **DES (Symmetric)** | Data Encryption Standard: Released March 17, 1975. Rather fast encryption algorithm. Widely used, a de facto standard. Symmetric key, 64-bit block cipher. 56-bit key size: Small 2^56 Keyspace. Today, DES is not considered secure | 4 | 63 |
| **DES Weaknesses** | Because of DES's relatively small Keyspace, brute force attacks are feasible. DES is no longer considered secure because of its key size, not because the algorithm has been broken. | 4 | 64 |
| **DES - Is Not a Group** | Whether or not an algorithm is a group is an important statistical consideration. If it is a group, then applying the algorithm multiple times is a waste of time. In 1992, it was proven that DES is not a group, in fact, so encrypting multiple times with DES is not equivalent to encrypting once. That's good news; it means that encrypting more than once with DES can increase the security of the ciphertext. | 4 | 65 |
| **DES is not a group** | This means that multiple DES encryptions are not related, making it stronger than a single encryption. Multiple DES encryptions will increase security. | 4 | 65 |
| **DES encryption X2** | Turns out that encrypting twice with DES does not significantly increase the effective key size. If a cryptanalyst is able to obtain both a cleartext (plaintext) message (M) and its corresponding ciphertext (c), they can perform a meet-in-the-middle attack. Brute force attacks are feasible on 2DES. Key length is only 57. One more than DES. | 4 | 66 |
| **DES Encryptions x 2 (2DES): Meet-in-the-Middle Attack** | Brute force all combos of key 1 and key 2 | 4 | 66 |
| **Meet-in-the-Middle Attack** | Brute force all combos of key 1 and key 2 | 4 | 66 |
| **DES encryption X3** | Usage: still widely used today, even though the key length is 169 bits and not considered secure in some environments. Prefer triple DES over DES: des should not be used under any circumstances.  Vulnerabilities: cracking triple DES means examining all possible pairs of crypto-variables. So far, there have been no public reports claiming to have cracked triple DES. | 4 | 67 |
| **DES Encryptions x 3 (3DES):Triple DES** | Usage:   * Still widely utilized today, even though the key length is 168 bits and not considered secure in some environments * Prefer Triple DES over DES: DES should not be used under any circumstances   Vulnerabilities:   * Cracking Triple DES means examining all possible pairs of crypto-variables. * So far, there have been no public reports claiming to have cracked Triple DES. | 4 | 67 |
| **Triple DES** | Usage:   * Still widely utilized today, even though the key length is 168 bits and not considered secure in some environments * Prefer Triple DES over DES: DES should not be used under any circumstances   Vulnerabilities:   * Cracking Triple DES means examining all possible pairs of crypto-variables.   So far, there have been no public reports claiming to have cracked Triple DES. | 4 | 67 |
| **AES (Symmetric) - An Overview** | The DES Replacement. AES has supplanted the inadequate 56-bit DES. AES has three initial key sizes: 128-bit, 192-bit, and 256-bit. Rijndael. | 4 | 68 |
| **AES: advances Encryption standard** | A new encryption algorithm designed to be effective well into the 21st century. | 4 | 68 |
| **AES Algorithm** | Employs 4 basic transformations:  AddRoundKey: XOR Round key with state.  SubBytes: Substitute bytes in state S to form state S’ on a byte-for-byte using S-box.  ShiftRows: left circular shift of rows 1-3 in state s by 1,2, and 3 bytes, respectively.  MixColumns: apply mathematical transformation to each column in State s to form State s’. | 4 | 70 |
| **AES** | Usage: The AES Algorithm is widely used today. You will find it in wireless protection mechanism, whole disk encryption implementations, and VPNs, to name only a few.  Vulnerabilities: No major vulnerabilities reported. Only feasible attack is brute forcing the keys. | 4 | 72 |
| **RSA (Asymmetric)** | The RSA algorithm has been widely implemented all over the world in all kinds of cryptography-enabled applications. It can be used to support both encryption and digital signature schemes. It is often used to facilitate Secure Sockets Layer (SSL) and Transport Layer Security (TLS). Although there have been a large number of claims to having cracked the RSA algorithm, they have all turned out to be false. | 4 | 73 |
| **ECC Elliptic Curve cryptosystem (asymmetric)** | Usage: high security even at relatively small key lengths (that is, a higher strength per bit); high-speed implementations; low processing power requirements; and low storage requirements. This makes ECC a particularly attractive cryptographic option for use in resource-constrained computing environments, such as mobile telephones, information appliances, and smart cards  Vulnerabilities: cracking ECC generally means compromising poor implementation, or weak points on the elliptical curve. So far, there have been not public reports claiming to have compromised the ECC algorithm itself. | 4 | 74 |
| **Key length: comparing** | Typically, within a given crypto algorithm, stronger ciphers result as the key size increases,. When comparing key sizes across different algorithms, then the rule changes. | 4 | 75 |
| **Hashing** | Produces a unique digital fingerprint of input data to verify integrity. | 4 | 76 |
| **SHA-1/SHA-2/SHA-3 (Hashing)** | SHA-1 (Deprecated) Output is 160-bit SHA-2 SHA2-256 Output is 256-bit SHA2-512 Output is 512-bit SHA-3 SHA3-256 Output is 256-bit SHA3-512 Output is 512-bit | 4 | 76 |
| **SHA-1/SHA-2/SHA-3 (Hashing)** | Hashing algorithms provide for the goal of integrity via the generation of the hash (fingerprint). The algorithm does not, however, provide any intrinsic means of protecting the fingerprint from subsequent, intentional modification. | 4 | 76 |
| **Analytic Attack** | Uses algorithms and mathematics to determine the key or reduce the key space to be searched. | 4 | 78 |
| **Cryptanalytic Attacks** | Many of the attacks can be categorically organized: Analytic, Statistical, Differential, Linear, and Differential Linear | 4 | 78 |
| **Differential Attack** | Analyzes resultant differences as related plaintexts are encrypted using a cryptographic key. Differential cryptanalysis is mostly applicable to block and stream ciphers (symmetric), but also has practical applications against hashing cryptographic systems. | 4 | 78 |
| **Differential Linear Attack** | Combines the methods of linear and differential analysis together. | 4 | 78 |
| **Linear Attack** | Linear analysis focuses on pairs of plaintexts and ciphertext, as well as potential weaknesses in the keys. Linear analysis is mostly applicable to block and stream ciphers (symmetric). | 4 | 78 |
| **Statistical Attack** | Uses statistical characteristics of language or statistical weaknesses against keys. | 4 | 78 |
| **Birthday attack** | When 23 people are put together in a room, the odds are greater than 50% that 2 or more people share a birthday. Hash collisions are related to that probability. Pairs of messages might have the same hash signature. Of course, there are many more possibilities for hash signatures than birthdays, but the same logic applies. It an attacker can find any 2 messages that generated the same hash value, that is, a collision, they could substitute one message for the other at will. | 4 | 79 |
| **Cryptography - Practical Applications** | Data in transit:  Virtual private networks (VPNs) Data at rest:  Data encryption  Full disk encryption  GNU Privacy Guard (GPG) Key management:  • Public key infrastructure (PKI)  Digital certificates  Certificate authorities (CA) | 4 | 83 |
| **Private networks: confidentiality in transit** | Pros: dedicated lines and equipment are not shared by others.  Cons: Dedicated lines are expensive, grow more so with distance, and are underutilized. | 4 | 86 |
| **VPN** | Cheaper, more flexible alternative to private circuits connection that assures confidentiality in transit over a public, untrusted network. | 4 | 88 |
| **VPN – Confidentiality in transit.** | Data is encrypted at one end of the VPN from cleartext into ciphertext. Ciphertext is transmitted over an untrusted network. Data is decrypted at the other end of the VPN from ciphertext back into the original cleartext. | 4 | 88 |
| **VPN Advantage: Flexibility** | A VPN "tunnel" over the Internet can be set up rapidly; a private network can take weeks (or months) | 4 | 89 |
| **VPN: Breakdown** | Pros: Connect easily and cheaply to the internet. Create 2 internet connection points. Encrypt traffic over the internet, a lot more bandwidth, VPNs are ideal for file transfers, email and more. If time is not critical, VPNs can save a lot of money. It is cheaper.  Cons: VPNs not ideal for financial, medical, and other real-time operations. Don’t get dedicated bandwidth across the internet. Only 2-secpnd delay compared to private network. Can be critical for some operations. | 4 | 90 |
| **VPN – Remote access Types** | Client-to-site VPN (transport), Example: Laptop connection to remote access server at HQ. Site-to-site (tunnel) Example: Sales office connection to HQ office location. Client-to-client VPN: protected link between 2 specific computers. It is a low-cost solution that allows secure sharing of files. | 4 | 91 |
| **IPsec Overview** | Can prevent replay attacks.  Issued by the IETF as an open standard (RFC 2401), thus promoting multi-vendor interoperability Can enable encrypted communication between users and devices Transparently fits into network infrastructure Can be used on networks on a small or very large scale Commonly implemented: Most VPN devices and clients are IPsec-compliant | 4 | 93 |
| **Replay Attack** | Attackers use replay attacks by copying a message as it goes across the network, then r-transmitting the copy to the destination. Even if the attacker cannot read the encrypted message, he can cause undesired results. For example, if the message was a request to transfer $1,000, the replay might be able to cause an additional transfer, making the total transferred $2,000. IPsec includes specific mechanisms to detect and prevent replay. Replay attacks are often used to capture encrypted authentication sessions and replay them later to log on to a given system. | 4 | 93 |
| **Integrity Check Value (ICV)** | In the ICV computation, AH includes every field that does not change during its trip from source to destination. This includes the source address, destination address, length, and the data. | 4 | 94 |
| **IPsec Headers** | Authentication Header (AH), Encapsulated Security Payload (ESP) | 4 | 94 |
| **IPsec Headers: types** | * Authentication header (AH) * Data integrity: No modification of data in transit. * Origin authentication: identifies where data originated. * No confidentiality. * Anti-replay capabilities * Encapsulating Security Payload. (ESP) * Data integrity: No modification of data in transit. * Origin authentication: identifies where data is originated. * Confidentiality: data can be encrypted. | 4 | 94 |
| **Source IP Field** | Used to tell the recipient who sent the message. | 4 | 94 |
| **Encapsulated Security Payload (ESP)** | ESP is the companion protocol to AH. Like AH, it offers message integrity, anti-replay, and authentication features, but it also offers confidentiality by providing the capability to encrypt the contents of the message. ESP authentication includes only the information in the ESP message, so the source and destination of the packet do not enter into the calculation. It does not matter whether the payload of the ESP message is encrypted or not. The calculation is the same. | 4 | 95 |
| **Keyed Hash** | However, what about the authentication? The hash, in the example of AH IPsec, is referred to as the keyed hash. | 4 | 95 |
| **NAT (Network Address Translation)** | With a NAT, your address is essentially translated (by an edge device such as your firewall or router) into the public IP address assigned by your Internet Service Provider (ISP). This happens transparently, behind the scenes. | 4 | 96 |
| **Private IP address** | Private IP addresses are designed to be used by devices that require network connectivity but aren't to be directly available from the internet. | 4 | 96 |
| **IPsec Modes** | Transport mode: Between two hosts,  Tunnel mode: Applies a Tunnel to IP Traffic | 4 | 97 |
| **Transport Mode** | Places IPsec header after original IP header. | 4 | 97 |
| **Tunnel Mode** | Works by accepting an entire IP packet, which is then packaged in an IPsec packet. This new IPsec packet is not addressed to the destination of the packet it is carrying. Instead, its destination address is the address of the gateway system at the other side of the tunnel. When the destination gateway receives a tunnel packet, it un-packages it to get out the original packet. This packet is then routed onward to the host listed in its destination field.  The original IP header and data are encrypted; a new IPsec header is prepended and a new IP header is created for routing to the destination. | 4 | 97 |
| **Internet Key Exchange (IKE)** | Internet Key Exchange (IKE) is an IPsec protocol that negotiates the connection details of a session and then documents them as SAs. IKE is a hybrid protocol composed of a key management framework and a key exchange protocol. | 4 | 98 |
| **Security Associations (SAs)** | the security services (called "transforms") that a particular IPsec connection is using. Unidirectional. | 4 | 98 |
| **SSL VPNs**  **SSL VPN Important note** | 99 Less operational problems than IPsec, cryptographically equivalent, but from an application perspective, perhaps not quite as secure All you need is a browser for client side. Portal VPNs works with almost any browser. SSL tunnel VPNs might require browsers where active content is allowed Web browsers and servers have vulnerabilities: application vulnerabilities, authentication, attack surface of the browser and the VPN portal are examples of risk that need to be considered.  The SSL pf SSL VPN doesn’t Imply that the insecure SSL (Secure Sockets layer) protocol is being used. An SSL-based VPN should only use more secure forms of the data in transit to protect concept such as through the use of the TLS (transport later security) protocol, which was designed to replace SSL. It is very common to heal older terminology used such as SSL to describe a modern cryptographic implementation, but it doesn’t mean that the implementation is actually using a weakened or insecure capability( or at least it shouldn’t) | 4 | 99 |
| **SSL-based Tunnel** | SSL-based VPN | 4 | 100 |
| **VPN Portal** | SSL-based VPN | 4 | 100 |
| **Client-to-site VPN security** | It suffers from the trusted client problem. Controlling the security of a corporate device used. By a remote user is one thing, controlling the security of a personal device used by a remote user is a completely different situation. | 4 | 101 |
| **VPN - Security Implications** | Encryption not only stops an adversary from reading your network traffic, but it also stops you from reading your own network traffic | 4 | 101 |
| **Data At Rest** |  | 4 | 102 |
| **Full Disk Encryption** | Files are encrypted in memory (RAM) prior to being stored on the hard drive. Files are read into memory (RAM, and decrypted there, prior to use. If the system is turned off and is stolen, data is in a protected (Encrypted) state, and is therefore inaccessible. If a computer is powered on, a user is logged in, and the user is subsequently compromised, the adversary will have the same access to the encrypted data as that user does. Encryption is a perfect example of how we need to truly understand which risks are, or are not, mitigated with the application of a security control. | 4 | 103 |
| **GPG: Critical part of key generation** | Use strong passphrase principles:  Length and complexity.  Easy to remember, difficult to guess | 4 | 104 |
| **GPG: Crypto Made Easy (To Use)** | GNU Privacy guard (GPG). It is a command line tool. Brings privacy to public communication medium via providing encryption for personal use. Provides individual file/folder level encryption. Provides secure email abilities (digitally signed email, encrypted email) | 4 | 104 |
| **GPG: Generate a public/private key pair** | * Choose an appropriate asymmetric algorithm. * SSL VPN Important note * Choose an appropriate passphrase to secure the private in storage | 4 | 104 |
| **GPG - Establishing a Key and Choosing a Passphrase** | In order to try and provide some additional protection for the private key, prior to the storing of the key on the hard drive, GPG will first encrypt it (by using a symmetric key based upon your passphrase). This will give the private key protection of data at rest. | 4 | 106 |
| **GPG - Using** | Can be as easy as clicking a button in a graphical interface and working through the resultant dialog boxes. | 4 | 107 |
| **Public Key Infrastructure (PKI)** | Is a system of functionality for the creation, maintenance, and revocation of certificates: PKIs consist of a hierarchy of trusted authorities, creating a chain of trust. 109 Certificates contain a public key; certificates bind an identity to a public key PKI | 4 | 109 |
| **Certificates** | Certificates are meant to be the digital equivalent of physical, real-world, identification documents (such as driver's licenses and passports). | 4 | 110 |
| **Chain of Trust** |  | 4 | 110 |
| **Trusted Authority** |  | 4 | 110 |
| **Certificate Authority (CA)** |  | 4 | 112 |
| **Certificates: The Easy Picture** | Now that the authority has verified Bob's identify and ownership of the public key, the authority can move forward with certificate creation and issuance. The next thing the authority does is take a hash of the public key (for integrity purposes). Then, the authority will encrypt the hash using their private key to create a digital signature of Bob's public key. Bob's public key and the authority's digital signature of the same are then placed into a digital certificate. | 4 | 112 |
| **Registration Authorities (RA)** |  | 4 | 112 |
| **Certificate - Life Cycle** | * Certificate registration * Certificate creation * Certificate distribution * Certificate revocation * Certificate expiration | 4 | 114 |
| **Certificate Signing Request (CSR)** | Is data structure which contains information on the requestor and a copy of the public key. | 4 | 114 |
| **Distinguished Name (DN)** |  | 4 | 114 |
| **Private CA root and subordinate Key protection** | The entire infrastructure becomes useless if these private keys are not carefully protected. If a Cas private keys are compromised, any ( an all) certificates that have been created by that CA can no longer be trusted. | 4 | 115 |
| **Private client-side keys protection** | New client-side private keys are imported onto a key store (often locally on a computing device), users can protect their private keys with passwords( read passphrase). There passwords can be used simply to keep someone else from exporting (or stealing) their private key. Or the private key can be stored in such a way that a password is required before they can even be accessed and used. | 4 | 115 |
| **Private server-side keys protection** | Server-side private keys such as those associated with SSL, must also be protected adequately to preserve the integrity of the messages between the client and the server. | 4 | 115 |
| **Public Keys protection** | The public key can be used to determine the authenticity of messages, can serve as part of non-repudiation scheme, and can be used to encrypt messages, which only the owner of the private key can decrypt. So they must be stored where everyone can get to them. | 4 | 115 |
| **Certificate Revocation List (CRL)** | A Certificate Revocation List (CRL) is a list of revoked certificates. Unfortunately, CRLs have significant issues:   * A client must download the entire CRL each time it is updated * CRL downloads can be network-intensive; a CRL might be cached by a client to counter-act * Cached CRLs result in a time-gap during which a client might unknowingly rely upon a revoked certificate | 4 | 117 |
| **Online Certificate Status Protocol (OCSP)** | The long-term replacement to traditional CRLs:  • OCSP operates in a client/server model; clients can request the validity status of a certificate by providing the certificate's serial number to an OCSP server  The OCSP server will respond with a validity result; OCSP allows for a one-to-one lookup  Requesting a validity result indicates to a CA that a client is accessing resources from the entity to which the certificate belongs (privacy concern)  OCSP stapling can help to mitigate such privacy concerns | 4 | 118 |
| **Stapling** | Allows for a web server to reach out to the OCSP server as opposed to the web browser do it. | 4 | 119 |
| **Certificate Extensions** | Extensions can limit how a certificate may be used (authentication versus signing of email, for example) | 4 | 120 |
| **Certificates: Some Additional Information** | Important items found in a certificate:   * X.509 version number * Unique serial number * Identity information of the certificate's owner * Validity period * Owner's public key and related algorithm info * Signature by issuing CA | 4 | 120 |
| **X.509 Standard .** |  | 4 | 120 |
| **HTTPS (SSL/TLS)** | Secure Web Traffic. Cryptographic protocols can provide security and data integrity over TCP/IP networks. Two such protocols, SSL, and TLS, encrypt the segments of network connections at the Transport Layer. | 4 | 122 |
| **Secure Socket Layer (SSL)** | Uses cryptography to provide message privacy, message integrity, and client and server authentication. It was designed to operate on TCP port 443. Do not use. | 4 | 122 |
| **Transport Layer Security (TLS)** | Long term replacement for SSL. TLS does take on additional considerations as to how secure sessions are formed in order to provide additional security beyond simply ensuring data is protected in transit. | 4 | 122 |
| **HTTPS - SSL Handshake steps** | 1. Client web request 2. Server responds 3. Client validates certificated and crypto 4. Client generates ad encrypts the session key and sends the session key to the server 5. Optional client certificate exchange 6. Server decrypts the session key 7. Key exchange finishes 8. Encrypted messages are exchanged. | 4 | 124 |
| **SSL Handshake** | See HTTPS -SSL Handshake | 4 | 124 |
| **PKI - Problems** | * Competing/incomplete standards * Certification of Cas * Important issue, but easy to overlook * Do it yourself or outsource * Extensive planning requirement * User education and/or perception | 4 | 126 |
| **PKI - Benefits** | Can be used for more than secure web traffic and secure email, Partial or whole disk encryption   * Code and driver signing * General user authentication * IPsec and VPN authentication * Wireless authentication * Network Access Control/Protection (NAC/NAP) And much more | 4 | 128 |
| **Network Security Devices** | Firewalls - Prevention (in line), NIDS - Detection (out of line), NIPS - Prevention (in line) | 4 | 133 |
| **Network Security Devices** | * Deployed on the network to provide security by monitoring network traffic. * Very scalable based on their placement on the network. * Focused on prevention and timely detection * Net technologies are being developed on a regulate basis * Many categories of network security devices include: Firewalls (prevention, NIDS (Detection) and NIPS (prevention. | 4 | 134 |
| **Network Security Devices - Layers** | See book for picture. | 4 | 135 |
| **Firewalls - Why** | Preventive technology:   * A router with a filtering ruleset * Reduces risks: * Protects systems from attempted vulnerability exploitation * Increases privacy: * Makes information gathering more difficult. * Enforces an organization's security policies | 4 | 137 |
| **Firewalls – Implementing** | * Dedicated network appliances * Hardware or software inserted into network device, such as a router (that is primarily performing other duties) * Software running on a general-purpose computer. | 4 | 137 |
| **Firewalls – Network locations** | Between the public internet and an organization’s private internal network.  Between a PCs network interface card (NIC) and the rest of the PC | 4 | 137 |
| **Firewall Benefits** | * Protect internal/external systems from attacks. * Filter communications based on content. * Perform network address translation (NAT) * Encrypt communications for VPN (IPsec) * Logging to aid in intrusion detection and forensics. * Can be layered to provide defense-in-depth. * Valuable aid in intrusion detection. | 4 | 138 |
| **Firewall leaks** | When “raindrops” get through the perimeter umbrella | 4 | 138 |
| **Firewall Shortcomings** | * Attacks at the application layer might sneak through. * Encrypted traffic, VPN extranet connections might bypass firewalls. * Organizations might let down their guard in other security areas (passwords, patches, encryption) * Management sees firewall as a silver bullet. | 4 | 138 |
| **Firewall Default Allow** | Default allow: More permissive | 4 | 139 |
| **Firewall Default Deny** | Default deny: More restrictive | 4 | 139 |
| **Firewall Default Rules** | Firewalls are designed with someone called a “default rule”. If a packet doesn’t match any of the other specific rules, the default rule would be implemented. 2 rules: Default Deny, and Default allow. | 4 | 139 |
| **Firewall Egress Filtering** | Traffic flowing out | 4 | 140 |
| **Firewall Filtering** | Ingress/Egress | 4 | 140 |
| **Firewall Ingress Filtering** | Traffic flowing in. Ex: Drop all incoming traffic headed to a specific subnet only allow TCP port 80 and 443 to systems in a web DMZ. | 4 | 140 |
| **Anti-Spoofing** | Anti-spoofing is a control to prevent network addressing defined behind an interface from sourcing from another interface. Without this control, an attacker may attempt to change their source IP address to make it look like it's coming from a trusted address in an effort to circumvent controls. | 4 | 141 |
| **Firewall Anti-spoofing** | Anti-spoofing is a control to prevent network addressing defined behind an interface from sourcing from another interface. Network addressing defined behind an interface should never source from another interface. | 4 | 141 |
| **Firewall - Types** | Stateless Packet Filter, Stateful, Proxy (NGFW or next generation firewalls) | 4 | 142 |
| **ACK Flag Set** |  | 4 | 143 |
| **ACK Scan** |  | 4 | 143 |
| **ACK Scan - No Open Ports, DMZ accessible** | The firewall dropped the packets because there was no ESTABLISHED state for the connection | 4 | 143 |
| **Stateless Packet Filter** | * Packet filters are "low-end" firewalls: * Minimal security * Very fast * Can easily be bypassed by attackers: * They examine a packet by itself with no context * They have to make assumptions, which are not always true * Data content passes through unchecked. Packet filtering firewalls rely on TCP flags: * Based on flag determine state of connection * If ACK flag is set, existing connection * Assumes step 1 and 2 of 3-way handshake was already initiated by an internal host but cannot verify * Attacker can send ACK packets to bypass firewall | 4 | 143 |
| **Stateful Firewalls** | * Stateful firewall maintain state of traffic flows by creating a session and tracking the state of the connection * Table of source address and port paired with the destination address and port. * Tracks the progress of the connection via flags. * Each packet is compares to the previous packets to put it in proper context before making a decision on whether to deny or allow the packet. | 4 | 144 |
| **Stateful Firewalls**  **Flags values** | * SYN\_SENT: a SYN packet has been sent from host A to host B, the first step in the 3-way handshake. * SYN\_RECV: A SYN ACK packet has been received from host B, the second step in the 3-way handshake * ESTABLISHED: The 3rd step in the 3-way handshake has been completed and the connection has been established. * FIN\_WAIT1: One host has issued a FIN packet indicating the connection should be gracefully closed * LAST\_ACK: The other host has acknowledged the request to gracefully close * FIN\_WAIT2: The other host has issued a FIN packet in response to the request. Both sides finished communicating. * CLOSED: No connection between hosts. | 4 | 144 |
| **Stateful Firewalls Error packets** | ICMP error packets are another issue for stateful firewalls. If an internal client connects over UDP to an external server that is not listening on the destination port, it is appropriate to issue an ICMP port unreachable error message in response. | 4 | 145 |
| **Proxy or Application Gateway** | The proxy firewall must maintain a complete TCP connection state and sequencing through two connections:  Session user (the source) to the proxy  Proxy to the destination server | 4 | 146 |
| **Network Intrusion Detection Systems (NIDS)** | See also IDS | 4 | 147 |
| **IDS Intrusion Detection System** | •IDPS report attacks against monitored systems/networks.  Alarm system  Mature technology  Requires monitoring, alerting, and reaction | 4 | 148 |
| **IDS what is NOT:** | Not a replacement for firewalls, strong policies, system hardening, timely patching, and other defense- in-depth techniques  Not a low-maintenance tool  Not an inexpensive tool  Not a silver bullet | 4 | 149 |
| **Events of Interest (EOI)** | IDS alerts are generated from events of interest (EOI) | 4 | 150 |
| **IDS - Alerts** | * True positive (ATTACK - alert) * False positive (Investigate alarm) * True negative (Normal, no alert) * False negative (BREACH - no alert) | 4 | 150 |
| **Network Tap** | Is designed to be placed into the network medium itself, perhaps yielding a higher fidelity to the collected packets, and/or avoid performance issues by spanning at a switch. | 4 | 152 |
| **NIDS - Overview** | * Deployed as a passive sniffer/sensor at network aggregation points * Sniffs and/or captures traffic * Detects events of interest on the network * Uses signature, anomaly, or application/protocol analysis | 4 | 152 |
| **Signature Analysis (Attack) - How it works** | * Performs pattern matching * Rules indicate criteria in packets that represent events of interest * Rules are applied to packets as they are received by the IDS. * Alerts are created when matches are found | 4 | 153 |
| **Signature Analysis - Rules** | A flexible rules language is valuable in an IDS, allowing an organization to augment the rules that ship with its IDS:   * Protocol, address, and port information * Payload contents * String matching * Traffic flow analysis * Flags in protocol headers * Any fields in the packet. | 4 | 154 |
| **Signature Analysis – Data characteristics** | * Protocol information * Address information * Port information * Payload contents * String matching * Traffic flow analysis * Flags in protocol headers. | 4 | 154 |
| **Anomaly Analysis - How it works** | Baseline of network must be performed  Enables an understanding of what "normal" is. Flags anomalous conditions in traffic on the network  Unexpected conditions are identified as suspicious. Can catch zero-day exploits. | 4 | 156 |
| **Application/Protocol Analysis - How it works?** | Where traditional anomaly analysis uses a set of conditions that the IDS looks for to flag an even, protocol analysis works by carefully examining the entirety of protocols and how they operate.  Protocol activity that is not known as normal is flagged  Difficult to implement  Few protocol implementations are "standard" | 4 | 157 |
| **Application/Protocol Analysis – Hurdles** | * Standards definitions * Implementations nuances * Changes to a protocol. | 4 | 157 |
| **Packet Inspection** | Shallow packet inspection:   * Fast, but provides little fidelity * Examines header information, limited payload data.   Deep packet inspection:   * Slow * Requires stateful tracking of data * Inspects all fields, including variable-length fields * In practice, both are used together. | 4 | 159 |
| **Data Normalization** | • Attackers try to denormalize traffic to evade detection  Numerous opportunities are available to do this  IDSs normalize data for understood protocols  They give the analyst a consistent basis for traffic analysis and rule generation | 4 | 160 |
| **NIDS - Advantages** | * Scalability * Provides insight into traffic on the network * Helps detect problems with network operations * Can help organizations react swiftly to incidents * Provides auditing for other security measures * Provides additional flexibility in securing information assets * Can be more aggressive than preventive measures in detecting attacks | 4 | 161 |
| **Honeytoken** | Involves labeling information with keywords that are unique, such as a project name or numerical project identifier. Should the NIDS identify the honeytoken content, the analyst can use the source address information to identify the person who sent the document without authorization. | 4 | 162 |
| **NIDS - Challenges** | Encryption.   * Deployment challenges including topology and access limitations * Analyzing encrypted traffic * Quantity versus quality of signatures * Performance limitations with extensive analysis techniques * • Very costly for proper management | 4 | 163 |
| **Snort as a NIDS** | * Open-source tool, low cost (Free software, inexpensive hardware) * Suitable for monitoring multiple sites/sensors. * Efficient detention system * Low effort for reporting. | 4 | 165 |
| **Snort - Rule Flexibility** | * One of the most significant advantages of Snort is the rule language * Administrators can create custom rules to detect any type of pattern match * Rules for new worms, exploits, or vulnerabilities are quick to be published by the user community * Rules can be developed to support honeytokens or any custom requirements | 4 | 166 |
| **Snort - Writing Rules** | There are a few options to keep in mind when writing rules:   * Pass: Ignore the packets and take no action. * Log: Allows you to log the traffic to a specific location, but otherwise take no action * Alert: Send alerts to a central syslog server or write the file to a separate alert file. This can alert to signs of intrusion and log the event, but not necessarily take action. | 4 | 167 |
| **Snort Rules - Simple Example** | Rule looks like the following:  alert tcp any ‐> 192.168.1.0/24 80 (msg: "Inbound HTTP Traffic"; sid: 2012033;)  Output looks like the following:  [\*\*] [1:0:0] Inbound HTTP Traffic [\*\*]  09/02-13:03:22. 734392 192.168.1.104:1460 -> 192.168.1.103:80  TCP TTL:128 TOS: 0x0 ID:28581 IpLen:20 DgmLen:48 DF  \*\*\*\*\*\*S\* Seq: 0x2550D716 ack: 0x0 Win 0x4000 TcpLen: 28  TCP Options (4) => MSS: 1640 NOP NOP SackOK | 4 | 168 |
| **Snort Rules - Advanced** | Rule Looks like the following and gets into application layer data:  alert tcp $HOME\_NET any ‐> $SERVER\_NET 22 (msg:"EXPLOIT ssh CRC32 overflow NOOP"; flow:to\_server,established; content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; reference:bugtraq,2347; reference:cve,2001‐0144; reference:cve,2001‐0572; classtype:shellcode‐detect; sid:1326; rev:6;)  Output looks like the following:  [\*\*] [1:0:0] EXPLOIT ssh CRC32 overflow NOOP [\*\*]  09/02-13:18:30 .550445 192.168.1.108:58362 -> 192.168.2.5:22  TCP TTL:128 TOS:0x0 ID:29951 IpLen: 20 DgmLen:466 DF  \*\*\*AP\*\*\* Seq: 0x32D8E9C1 Ack: 0xB427699E Win 0x4470 TcpLen: 20 | 4 | 169 |
| **NIDS - Key Points** | * Train operation staff in IDS analysis techniques; this is a high- end skill * If you can afford a security management console, keep it populated with a passive sniffer; this can help you manage your false positive problem * Consider a Security Information and Event Management (SIEM) solution * Be prepared with incident response and supportive policies * Perform ROI calculation: In-sourced or outsourced IDS management | 4 | 170 |
| **SIEM** | A security information event and management (SIEM) station allow organizations to combine data from multiple discreet sources, including IDS alerts, syslog messages, Windows event log data, and more. | 4 | 170 |
| **Active vulnerability analysis** | uses traditional vulnerability assessment tools to routinely scan systems to document their vulnerable services. | 4 | 171 |
| **NIDS - Developments** | * Reduction of false positive reporting through target OS identification. * Integrated vulnerability assessment for threat profiling/alert prioritization. * NIDS integration in networking devices. * IDS for wireless networks | 4 | 171 |
| **Passive Fingerprinting** | A technique that monitors network traffic and characterizes the nature of traffic to identify the host operating system. | 4 | 171 |
| **Hardware Buffering** | Lets the switch hardware buffer packets for later delivery to the IDS when the processor is too high. | 4 | 172 |
| **IDS Blade** | Can simply insert into the switch a custom hardware that inserts into the chassis and monitors traffic as it traverses the switch, despite the port of interface that the traffic arrives on. | 4 | 172 |
| **Passive vulnerability analysis** | Uses a new method to identify vulnerable services on a host by passively monitoring traffic. The IDS attempts to identify the characteristics of an application that are exhibited only when the application is susceptible to a specific vulnerability | 4 | 172 |
| **Rogue Wireless LAN activity** | “Rogue” activity in unauthorized locations. | 4 | 172 |
| **WIDS (Wireless IDS tools)** | Are deployed to co-exists with wireless LAN deployments or instead of wireless LAN deployments if an organization wants to identify “rogue” wireless LAN activity in unauthorized locations. | 4 | 172 |
| **NIPS** | See also IPS | 4 | 173 |
| **Intrusion Prevention System (IPS)** | * IPS stops attacks on systems and networks from being effective * IPS can be network-based (NIPS) or host-based (HIPS) * Technology is rapidly maturing | 4 | 174 |
| **IPS – Is NOT** | * Not a replacement for firewalls, IDS, strong policies, system hardening, timely patching, and other defense-in- depth techniques * Not a low-maintenance tool * Not an inexpensive tool * Not a silver bullet. * An active IDS stops an attack in progress, but a true IPS stops an attack automatically. * In addition, an IDS is deployed passively, whereas an IPS is traditionally deployed inline. | 4 | 175 |
| **NIPS – How they work** | * NIPS are typically deployed at the perimeter in front and/or behind a firewall. * Deploying a NIPS between the firewall and ISP router ensures that the firewall and DMZ servers are protected. * Behind-the-firewall NIPS deployments protect the internal network from remote access VPN users and can also assist in locating infected internal hosts. | 4 | 176 |
| **NIPS - Detail** | * Deployed inline at network aggregation points * Uses custom Application-Specific Integrated Circuits (ASICs) to support high-speed analysis with complex inspection * Uses data normalization and reassembly techniques on aggregate traffic * Hierarchical rule classification schemes are used to classify and identify traffic * Because of risk for false positives, NIPS cannot identify as many attacks as NIDS | 4 | 177 |
| **Zero Power High Availability (ZPHA) device** | Others offer a Zero Power High Availability (ZPHA) device that will reroute network traffic should the IPS device lose power. This option is extremely beneficial for organizations that simply cannot afford downtime. | 4 | 177 |
| **Custom Application-Specific Integrated Circuit (ASIC) technology** | Is typically used for these types of devices such as NIPS. Is designed to perform a specific job and not intended for general purpose use. This allows them to perform better but staying focused and limited to specific instructions. | 4 | 178 |
| **Multiresolution Filtering** | A technique that uses a rule classification scheme to quickly sort through traffic in order to rapidly identify malicious events. | 4 | 178 |
| **NIPS - Challenges** | * Organizations (and NIPS vendors can’t afford false positives * A nips false positive drops legitimate traffic * Throughput of NIPS device must be able to keep up with traffic demands. * A NIDS would miss traffic: NIPS stops legitimate traffic * NIPS tend to have a less-extensive rule base * More false negatives that IDS tools. | 4 | 179 |
| **NIPS - Developments in** | * Improved throughput and response times * Near-real-time analysis and forwarding * Automated analysis/signature updates * Might be a good or a bad thing * Environmental anomaly analysis * Protocol "scrubbing", rate limiting, and policy enforcement * Passive analysis | 4 | 180 |
| **Passive Analysis** | • Correlates OS and vulnerability information with identified attacks  • Supports "network learning" mode to identify network architecture, structure | 4 | 182 |
| **Network Security Devices: How to deploy them** | See book for diagram and examples. | 4 | 183 |
| **Endpoint Security - Objectives** | * Endpoint Security Overview. * Endpoint Security Solutions. * Host-based Intrusion Detection System (HIDS) Overview. * Host-based Intrusion Prevention System (HIPS) Overview. | 4 | 188 |
| **Endpoint Security - Core Components** | **Ultimate goal is to control the damage and reduce the impact of an attack by reducing the attack surface by:**  Limit attack surface:   * Patching * Updating services * Turning off services * Controlling access.   Foundation for effective security:   * Asset inventory * Configuration management * Change control | 4 | 190 |
| **Endpoint security Pivot Points** | When a system is compromised, it is used a pivot point to break into other systems. | 4 | 190 |
| **Endpoint Security - Enhancing** | * Many operating systems have built-in security features. * Goals for enhancing OS security: * Better visibility * Reducing the attack surface * Controlling the Damage * Early detection. | 4 | 191 |
| **Baselines** | * Baselines are determined through the use of several commands and logging their output * Create a picture of how a system normally looks and behaves * Compare past logs to present logs for changes | 4 | 192 |
| **Baseline - Establishing** | To establish a baseline, you must create an image of what is normal activity and the behaviors on the systems as well as the network you are monitoring.  **Considered normal:**   * The type of network traffic * The amount of network traffic * The types of logs generated * The number of logs generated * The resource utilization of the systems * Access times to systems and length of access * The current state and configuration of a server. | 4 | 193 |
| **Anomalies - Detecting** |  | 4 | 194 |
| **Detecting Anomalies** | * Using tools to sift through the background noise enables you to find anomalies. * Visualizing all the chances can make it much easier to find outliers of system and network activity in your environment. | 4 | 194 |
| **Anomalies - Looking for** | Not all outliers are volume-based. Often, the outliers are simply those that deviate slightly from the norm but do so in two or more areas at a time, which is where behavior-based anomaly detection comes in. | 4 | 195 |
| **Endpoint Security Solutions** | Antivirus Software, Endpoint Firewalls, File Integrity Checking, Log Monitoring, Application Control, Host Intrusion Detection Systems (HIDS), HIPS, Application Behavior Monitoring | 4 | 196 |
| **Antivirus Software** | While there are many variances, typical AV software scans the files on a computer looking for indicators of malicious software that might be present on a system. Email attachments and files downloaded from the web are often scanned to make sure they are not infected with a virus. | 4 | 197 |
| **Antivirus Software: Types** | AV, or antivirus, software has become a generic term to refer to a wide range of software products, Anti-adware, Antivirus, Anti-malware, Antispyware, Anti-Trojan. | 4 | 198 |
| **Endpoint Firewalls** | * Firewalls are critical to manage and filter traffic * Network firewalls provide a boundary defense. * Host-based firewalls are needed to complement network firewalls * Provides additional protection from a defense-in-depth perspective. * Allows for protection of travelling laptops that directly connect to the internet from untrusted locations. | 4 | 199 |
| **Host-based Firewall** | Can be customized to protect a specific system, they are sometimes hard to manage in a large organization. | 4 | 199 |
| **Endpoint Firewalls - Types** | Packet Filter (Stateful):   * Permit only authorized services * Block unauthorized IP addresses. * Application Control and Operating System Control * Endpoint security suites focus on desktop lockdown, which includes personal firewalls | 4 | 200 |
| **Personal Firewalls** | These usually exists on a PC instead of being a separate appliance. | 4 | 200 |
| **File Integrity Checking** | * The analyst defines a list of critical files that should be monitors for change: * HIDS software calculates a hash for each file * Hash is regenerated frequently: * if a change is made to the file, the hash will change. * Upon next scan, the change is detected, and alert is generated and sent to the analyst. | 4 | 201 |
| **One Way Hash** | Mathematical function that produces a hash-value result when applied to a monitored file. Used to identify unauthorized changes to files. | 4 | 201 |
| **File Integrity Checking - Steps** | 1. Define a list of files to check 2. Generate a cryptographic hash for each file 3. Store the hashes in a secure location 4. Confirm cryptographic hashes cannot be modified 5. t set intervals, rerun cryptographic hashes on the specified files 6. Compare the new hashes against the original 7. Alert on any files where the hashes no longer match 8. Optional: Alert on new files within a certain directory | 4 | 202 |
| **Exclusive Analysis** | Like inclusive analysis, the analyst generates a file of keywords and phrases for use by the HIDS. Unlike inclusive analysis, the HIDS uses the keyword and phrase list to exclude log entries. The log entries that do not match the exclusive keyword list are raised as alerts to the administrators. | 4 | 203 |
| **Inclusive Analysis** | This measure of log monitoring uses a list of keywords or phrases that defines the events of interest for the analyst. The keyword list might contain words from operating system log files, such as "login failed" or "unauthorized access", or more specific events, such as a web server exploit, that would show up in web server log files. | 4 | 203 |
| **Log Monitoring - How it works** | * Uses inclusive or exclusive analysis: Inclusive analysis uses a list of keywords to watch for * When a match is found, an alert is raised Exclusive analysis uses a list of events that can be ignored * When an event is identified that does not match the ignore list, an alert is raised | 4 | 203 |
| **Active Mode** | Blocking | 4 | 205 |
| **Application Control**  **Endpoint protection)** | * Utilizes an approved list of applications: * Only those applications can run * All applications and associated files are cryptographically verified with hashes * Applications can only run if they are approved, and all cryptographic hashes are valid * Can run in passive (alerting) or active (blocking) modes * Relies heavily on environments that have robust configuration management and change control processes | 4 | 205 |
| **Application control Passive Mode** | Alerting | 4 | 205 |
| **Host Intrusion Detection Systems (HIDS) - Overview** | Host-based intrusion detection works on a single host, identifying events of interest that are configured by the administrator.   * Provides much of the functionality of a NIDS, distributes to each host. * Can be more granular than NIDS, analyzing activity on the host. * Uses signature and anomaly analysis with unauthorized change monitoring, log monitoring and network monitoring. * local processing/alerting may be done, but data generally is sent to a central location for parsing. | 4 | 207 |
| **HIDS - Network Monitoring** | * Monitors network traffic to the host * Typically listens on all interfaces * Ethernet, wireless, VPN * Uses signature analysis to identify events of interest * Much like a distributed NIDS * Very powerful for not only monitoring inbound traffic but also outbound traffic * Monitoring outbound traffic can be used to detect pivoting, internal reconnaissance, lateral movement, and C2 | 4 | 208 |
| **HIDS - Advantages** | Can provide additional information the NIDS can't see:  Notably pre/post-encrypted data and more monitoring/analysis capacity  Provides detailed insight into the network, not just at the perimeter  Identifies inside attacks against systems  Has more details about the host that can increase accuracy and reduce false positives  Last line of defense because if the HIDS alerts, it means all other security devices failed to detect the activity | 4 | 209 |
| **HIDS - Challenges** | * Managing updates to signatures and HIDS software can be complex. * Each sensor has tunnel vision * HIDS requires a centralized console to identify trends and wide-scale events * Full HIDS monitoring can be more costly than NIDS. * HIDS requires resources on monitored hosts and can impact system performance (more cost). | 4 | 210 |
| **HIPS –**  **Advantages** | The added benefit for HIPS, of course, is the capability to stop attacks from being successful. | 4 | 210 |
| **HIDS - Developments in and Recommendations** | * Monitoring change at the application level * Protecting your website with HIDS * Appliance platform support * File integrity monitoring for networking devices. * Moving from HIDS being a self-contained solution with a sensor and monitoring station to HIDS becoming another data feed for the SIEM. * HIDS solutions are morphing into HIPS * Logical progression for mixed functionality coming together | 4 | 212 |
| **HIPS –**  **Detail/Specs** | * Can stop common attack techniques, known and unknown. * Traps system calls that are marked as dangerous * In-depth protection requires understanding of how applications function * Uses a combination of file integrity monitoring, network monitoring, and application behavior monitoring. * Can monitor and correlate activity and when it gets above a certain threshold, block the attack. | 4 | 214 |
| **System Call Interception** | Similar to what antivirus vendors have been doing for years. HIPS software inserts its own processes between applications accessing resources on the host and the actual OS resources, denying, or allowing requests depending on if they were determined benign or malign. | 4 | 214 |
| **HIPS –**  **Advantages** | * Includes many of the same advantages of HIDS but allows for more timely prevention * Anomaly analysis techniques can stop unknown attacks. * Can be used to buy more time in the patch management race * Provides a better defense of systems with an expanding network perimeter * Allows for protection for traveling laptops. | 4 | 215 |
| **HIPS –**  **Challenges** | * False positives are a problem, but less so on a distributes scale. * Includes implementation and maintenance challenges * Supports a limited suite of applications (little support for protecting custom applications) * Not a replacement for system patching or antivirus defenses. * Requires more system resources for in-depth anomaly analysis. | 4 | 216 |
| **Application Behavior Monitoring** | * Vendor identifies intended behavior in applications * HIPS software monitors how the application interacts with the host * Can detect unintended behavior * Initially monitors activity and when it is deemed malicious, can automatically block | 4 | 217 |
| **HIPS –**  **Recommendations** | * Maintain a requirements document and testing procedure for HIPS software selection * Develop a centrally managed policy for controlling HIPS client rules and updates * Don’t blindly install updates to software without testing * Don’t rely on the vendor to test for you * Don’t rely solely on HIPS to protect systems * Used them to buy more time for testing patches before a company-wide rollout. | 4 | 218 |
| **Advanced Application Shielding** |  | 4 | 219 |
| **HIPS –**  **Developments in** | * Proven protection against zero-day attacks * Vendors have some real-life scenarios to defend against. * Dynamic rule creating for custom applications based on observed behavior. * A new target for attackers * Application shielding * Like all security software, must be updated and properly maintained. | 4 | 219 |
| **Client Operating System**  **💠** | Client operating systems are intended for devices such as tablets, laptops, PC workstations, and gaming consoles. Users directly interact with these devices by touch, voice, gesture, keyboard, and mouse. Users often personally own these hardware devices. Client operating systems are designed for ease of use, graphical applications, and backward compatibility. Examples include Windows 7 and Windows 11. | 5 | 5 |
| **Embedding Operating Systems**  **💠** | Embedded operating systems are intended for devices such as Point of Sale terminals, automobile dashboards, electronic signs, industrial control equipment, robotics, sonogram machines, handheld laser scanners, welding machines, and all the myriad "Internet of Things" devices. Although the client and server operating systems are general purpose and can be easily repurposed later, the embedded operating systems are usually customized by equipment manufacturers to suit just their hardware. Examples include Windows Embedded 8.1 and Windows 10 IoT Core. | 5 | 5 |
| **Windows Operating System**  **💠** | Server operating systems are intended for devices such as rack-mounted computers that often have RAID storage, 128 GB or more of memory, and multiple network interfaces. The servers can also be virtual machines (VMs). Users normally interact with these physical or virtual machines over the network, not directly by mouse or touch input. Server operating systems are designed for web servers, email gateways, VPN gateways, file and printer sharing, VM hosting, domain controllers, DNS, DHCP, and so on. Examples include Windows Server 2016 and Windows Server 2022. | 5 | 5 |
| **Windows - Operating Systems**  **💠** | Three classes of operating systems  Client  Server  Embedded  Note that these are not completely different products under the hood, as they usually share core binaries for their kernel’s libraries and protocol stacks as Microsoft wants to standardize on a single, modular base OS platform for everything going forward. | 5 | 5 |
| **Client editions**  **(Client Operating System)**  **💠** | Editions that include words such as Starter or Home are intended for personal use, have fewer features, cannot be joined to an Active Directory (AD) domain, and cost less. Editions that include words such as Business, professional, or enterprise are intended for business uses, have more features, can be joined to a domain, and cost more. The ultimate editions are aimed at enthusiasts, have more features than the Home editions, can be joined to a domain, and often cost the most. Recent operating versions do not have an Ultimate edition. | 5 | 6 |
| **Client Operating Systems**  **💠** | Microsoft often experiments with a nee kernel or graphical interface by releasing it as a new OS version, and then fixes and polishes that experiment for the next OS release, which is usually more popular.  Windows 7 • Windows 8 • Windows 10 • Windows 10X • Windows 11 • Windows 10/11 in "S Mode".   * Personal Editions:   Starter, Home, Ultimate   * Work Editions:   Business, Pro, Enterprise   * Platforms:   AMD or Intel • ARM | 5 | 6 |
| **Full Mode Vs S Mode**  **(Client Operating System)**  **💠** | Mode “S” is a special locked down configuration of Windows 10 or windows 11. Users cannot install any applications except those in the Microsoft Store, and the only browser allowed is Edge. PowerShell can’t be run, and it can only join an on-premises Active Directory domain inside the LAN. If an admiration turns off “S” mode, it can’t be enabled again. | 5 | 7 |
| **Licensing**  **(Client Operating System)**  **💠** | The started, Home, Business, Professional, and Ultimate editions are Usually purchased and licensed one at a time as either a retail license or as an OWM license. The Enterprise edition is purchased and licensed as part of an organization wide agreement between your company and Microsoft | 5 | 7 |
| **Platforms (x86, x64, ARM)**  **(Client Operating System)💠** | Microsoft wants to move away from the ancient 32-bit (x86) platform and support only 64-bit (x64). For the ARM platform, there are special editions of windows 10 and later, they don’t have the same features and cannot run all the of same applications as windows for x86/x64. | 5 | 7 |
| **Windows 10X**  **(Client Operating System)**  **💠** | * Released to compete with the iPad Pro tablets and Google Chrome OS devices. * Multi-year beta test of techniques that became standard. * Has a simplified graphical interphase similar to Chrome OS * Separation of OS and application containers, much faster updating, deeper integration with Azure and Microsoft 364, and very strict code signing requirements. Every application runs in a container for isolation, including legacy x86/x64 application | 5 | 7 |
| **Server Editions**  **(Server Operating Systems)**  **💠** | * There are 3 primary editions of Windows Server: DataCenter, Enterprise, and Standard. * The different editions have different scalability and fault tolerance capabilities, such as for clustering and Network Load Balancing (NLB). | 5 | 8 |
| **Server Operating Systems**  **💠** | * Windows Server 2000, Windows Server 2003, Windows Server 2008, Windows Server 2012, Windows Server 2016, Windows Server 2019, Windows Server 2022. * Not intended for desktop, laptop, or tablet use. * It us normally installed in virtual machines running on computers that open have multiple CPUs. Multiple storage device, multiple network interface cards, a lot of memory, and possible no monitor. * Intended for providing network services, such as DNS or HTTP, and not for running graphical applications with a directly attached monitor, keyboard, and mouse. | 5 | 8 |
| **Licensing**  **(Server Operating Systems)💠** | * Is even more complex than client devices licensing. | 5 | 9 |
| **Platforms**  **(Server Operating Systems)💠** | * Windows server 2008 was the last server OS to be available in 32-bit (x86). * Server 2008 R2 and later is only available as 64-bit. | 5 | 9 |
| **R2**  **(Server Operating Systems)**  **💠** | * Microsoft released a new server every 4 years such as windows server 2012 and referred as “First Release” (or “R1”, “Release to Manufacturing, or “RTM) version of that OS. * 2 years after the first release, came the second release (“R2”) of that same OS. * New significant functionalities come out of any “R2”. * Best to consider it a whole new OS, as Microsoft has mostly dropped the whole “R1/R2” naming. | 5 | 9 |
| **Server Core and Server Nano**  **(Server Operating Systems)**  **💠** | * Do not refer to Windows Server editions * Refer to installation options for windows server * When choosing Core or Nano during the Windows Server installation, you are choosing to remove the graphical desktop from the OS either mostly (with Core) or completely (with Nano). * Mostly managed with PowerShell or using graphical tools over the networks from a management workstation. | 5 | 9 |
| **Windows Server Roles** | * Domain Controller * Web Server (IIS) * Hyper-V Virtualization (VMs and Containers) * Remote Desktop Services * DirectAccess and VPN * File and Print Services * DHCP Server * DNS Server * Network Policy Server (RADIUS) | 5 | 10 |
| **Windows Embedded** | * Windows for the Internet of Things (IoT). Intended for industry-specific hardware appliances, such as ICS/SCADA equipment, retail point-of-sale, MRI scanners, robotics, digital signs, drones, 3D printers, and more. * Supports the ARM platform, as well as x86 and x64 CPUs. * Runs on Raspberry Pi, MinnowBoard, and Arduino devices * “Windows” IoT” is just another name for Windows Embedded. | 5 | 11 |
| **Windows for the Internet of Things (IoT)** | "Windows IoT" is just another name for Windows Embedded | 5 | 11 |
| **Windows Workgroups**  **Characteristics.** | * No domain controllers. * Standalone computers only. * Local user accounts only. * Local groups cannot have users from other machines. * Users are typically local administrators of their own machines. * A "workgroup admin" simply has a separate administrative account on every machine. Workgroups tend to be small, usually less than 50 devices. * Permissions can be assigned to local users and groups only. | 5 | 14 |
| **Workgroups** | * 2 or more Windows computers that share information in the absence of any domain controllers are called a workgroup. * Even if they were thousands of user accounts on each of them, they still form only a workgroup, not a domain. | 5 | 14 |
| **Windows Workgroups - Benefits** | * Simple to understand * Each computer manages and protects itself * Lower initial deployment cost * Users are typically administrators of their own machines, allowing personal creative expression and joy | 5 | 16 |
| **Windows Workgroups - Drawbacks** | * Users are insane, as they seem to enjoy destroying their machines. * Workgroup = Chaos + Anarchy * Difficult to manage large numbers * No centralized policy control or auditing * No single sign-on without great effort * No consistent permissions across machines | 5 | 17 |
| **Computer Management Tool** | To manage local users and groups. Search for it on the Start screen or just run CompMgmt.msc to launch | 5 | 18 |
| **Script User Account (Local Account Management)** | If you want to script user account management, there are PowerShell commands and the old NET.EXE program too. In PowerShell 5.1 or later, run "Get-Help LocalUser" or "net.exe help user" to see the available options. | 5 | 18 |
| **Security ID Numbers (SID)** | Windows only cares about SID numbers when enforcing permissions and privileges, not names. SID numbers are unique, except for the SID numbers of some well-known users and groups, which are shorter and standardized across all boxes. For example:   * S-1-1-0 = Everyone group * S-1-5-11 = Authenticated Users group * S-1-5-32-544 = Local Administrators group | 5 | 19 |
| **Security Access Token (SAT)** | all of your SID numbers go onto your "ID card", so to speak, when you log on. This ID card is how the computer identifies your programs and regulates your activities | 5 | 20 |
| **SID number –**  **Finding it** | You can see the SID number of your user account in PowerShell or CMD.EXE by running: whoami.exe /all /fo list | 5 | 20 |
| **Security Access Token (SAT)** | * Your SAT is attached to every process you start. * Contains your user SID and all your group SIDs. * Contains a list of all your privileges. * Windows uses your SAT to check your permissions and privileges before allowing attempted actions. * In PowerShell, run: whoami.exe /all /fo list | 5 | 21 |
| **Domain Controller** | * A central shared database of SID numbers that all standalone computers agree to use for single sign-on and authorization * A secure authentication protocol for distributing SID numbers from this database to computers making SATs | 5 | 22 |
| **Workgroup –**  **Making a perfect workgroup.** | * Standalone computers do not trust each other’s SAT, that is why workgroups do not scale. * A central shared database of SID numbers that all standalone computers agree to use for single sign-on and authorization * A secure authentication protocol for distributing SID numbers from this database to computers making SATs. * We need a domain controller. | 5 | 22 |
| **Access Control List (ACL)** | List of permissions based on user and group s/id numbers is called an access directory list. | 5 | 24 |
| **Active Directory Domains** | Active Directory (AD) is the name of the shared accounts database that gets installed on a Windows Server when it is promoted to become a domain controller. Again, what is Active Directory? It's a database of user accounts and other information that otherwise would have been in the local accounts databases of standalone computers. | 5 | 24 |
| **Domain Controllers** | A domain controller is just a server that helps to manage the AD database on behalf of the other computers and users in the organization. The AD database contains, among other things, all the SID numbers of the users, computers, and groups that have "outsourced" their authentication and account management work to the domain controller | 5 | 24 |
| **Multi-Master Replication** | You can make a change to the AD database on any domain controller in an AD domain, and this change is then automatically replicated to all other domain controllers. This is called multi-master replication. | 5 | 24 |
| **Active Directory Database** | Consider this analogy because it is the key to understanding what the Active Directory database is intended to be: Active Directory is like a registry for the entire network. Active Directory is a general-purpose database and can be accessed through an industry standard protocol, LDAP. Active Directory uses the same database engine as Microsoft Exchange Server and can store millions of objects. Its maximum size is 4,000 GB (4 TB). | 5 | 25 |
| **Domain** | All the users, computers, and groups that have (or rather, are) accounts in the AD database. | 5 | 25 |
| **Group Policy Objects** | Active Directory stores these settings in the form of Group Policy Objects that modify Registries and other things. | 5 | 25 |
| **In the Domain** | Strictly speaking, you are in the domain if you have an account in the Active Directory database. | 5 | 25 |
| **LDAP** |  | 5 | 25 |
| **Read-Only Domain Controller (RODC)** | An important exception to multi-master replication is when a domain controller is running Windows Server 2008 or later and the administrator has decided to install that controller as a Read-Only Domain Controller (RODC). A running RODC can authenticate users and fulfill search requests, but any changes made to the AD database on the RODC will never be replicated to any other controllers. Moreover, RODC controllers cache only the credentials of the groups an administrator specifies, and the list of cached credentials is tracked so that, in case of compromise, just those users can be immediately forced to change their passwords. | 5 | 25 |
| **Red-only Domain Controller** | * It will never be replicated to any other controllers. * RODC controllers cache only the credentials of groups an administration specifies. * The cached list is tracked so that if compromise, we would know who needs to change their password. * ROCD controllers may also be installed using the Server Core option to give it smaller disk and memory footprint to keep costs down. | 5 | 25 |
| **Registry** | Windows computers store all their configuration settings in a tiny database called the registry. It can be edited with REGEDIT.EXE. Active Directory can store many of the configuration settings for all users and computers too. | 5 | 25 |
| **Domain User, Computer or Group** | Now a distinction can be made. Local users and groups are accounts in the database of non-domain controllers. This is true whether or not that computer is a member of a domain. On the other hand, a domain user, computer, or group has its account in the AD database. These domain accounts are available for use by any computer that has joined the domain. | 5 | 26 |
| **Local Users and Groups** | Now a distinction can be made. Local users and groups are accounts in the database of non-domain controllers. This is true whether or not that computer is a member of a domain. On the other hand, a domain user, computer, or group has its account in the AD database. These domain accounts are available for use by any computer that has joined the domain. | 5 | 26 |
| **Authentication Protocols** | * The Sid numbers of domain accounts and groups in the Active Directory database are conveyed to domain-joined computers through authentication protocols. * The SAT is constructed “on the fly” at the remote computer where the user requests access to a resource, such as a shared file. * Privileges, local account SID numbers, and locally group SID numbers all come from the target computer’s own local registry. | 5 | 27 |
| **Security Access Tokens - 4 Parts** | * SID number for the user's domain account (AD) * SID numbers for the domain groups the user is a member of (AD) * SID numbers for the local groups on the server being accessed that the user is a member of (from the server's local accounts database) * The list of privileges the user has on the server being accessed (from the server's local registry) | 5 | 27 |
| **Kerberos - Authentication Protocols** | * Kerberos is the default authentication protocol in Active Directory environments. * Kerberos uses tickets to convey the user's account SID and group SID numbers to the target server to which the user desires access. * Tickets are encrypted with users' passwords and special keys shared only among the domain controllers. | 5 | 28 |
| **Kerberos Key Distribution Centers (KDCs)** | AD domain controllers are all Kerberos Key Distribution Centers (KDCs) because they hold every user's and computer's Kerberos master key. Your Kerberos key is derived from your password. | 5 | 28 |
| **Golden Tickets** | Mimikatz is best-known tool. With a Golden Ticket, the attacker can authenticate to any machine joined to the domain as a member of the Administrators local group on that machine, then install backdoors, rootkits, or any other malware. They grant access to everything. | 5 | 29 |
| **Kerberos Risks – Prevention** | 1. Protect the integrity of the domain controllers 2. Protect the backups of the domain controllers 3. Protect the credentials of the network administrators who manage the domain controllers, such as those in the Domain Admins and Enterprise Admis group in Active directory. | 5 | 29 |
| **Kerberos Risks.** | * If an attacker can capture the packets of the initial Kerberos exchange, the attacker can mount a brute force attack to get the user’s passwords. * There are specials encryptions keys that are replicated and shared among all domain controllers in the same domain. If the encryption keys are stolen, it could be catastrophic as we will lose control of our domain. | 5 | 29 |
| **Mimikatz** | Golden ticket tool | 5 | 29 |
| **Kerberos Authentication Protocols: Example** | A user with a laptop who needs to map a driver letter using the SMB protocol to a shared folder in a file server. The user has already authenticated to the domain controller using Kerberos, and now the user needs another Kerberos ticket for the file server. So the user send the name of the desired file server to the domain controller as part os a request for another ticket. The domain controller responds directly to the client, not to the target file server, the Kerberos ticket given to the client is encrypted so that only the target filer server can decrypt it, but the client can cache this ticket in memory and can send the ticket to the target file server when the client chooses. However, tickets have an expiration limit. | 5 | 30 |
| **NTLM - Authentication Protocols** | * Predecessor to Kerberos, but still supported for compatibility. * Used in workgroups because it doesn't require Active Directory. * Authentication defaults to NTML when Kerberos is not available. * Data encrypted with the user’s password hash is given to the server; the server passes it through to a controller, then the controller sends the user’s SID numbers to the server (if the hash is okay) * NTLMv1 is vulnerable to sniff-and-crack attacks * NTLMv2 is much less vulnerable * NTLM can be entirely disabled. | 5 | 31 |
| **Forests** | A forest is one of more AD domains that replicate special portions of their domain databases with each other and that all trust each other.   * One or more domains * Inter-domain replication * Two-way transitive trusts * The Global Catalog * Global Catalog Servers | 5 | 32 |
| **Global Catalog (GC)** | Special domain controllers that replicate across domain boundaries. | 5 | 32 |
| **Global Catalog Servers** | Portion of the AD database that is replicated everywhere in the forest. | 5 | 32 |
| **Inter-Domain Replication** | Because AD uses multi-master replication, a modified user or group on a domain controller will be replicated to all the other controllers in that domain. | 5 | 32 |
| **Trusts** | Domains can be linked with trusts to permit resource sharing and single sign-on across domains | 5 | 32 |
| **Two-Way Transitive Trusts** | Every domain in the forest trusts every other domain in the forest. | 5 | 33 |
| **Trust - Nature of** | Without a trust link between two domains:   * No single sign-on across domain boundaries * Can't assign permissions to users in the other domain * Can't log on to your desktop in the local domain with an account created in the other domain. | 5 | 34 |
| **Cross-Forest Trusts** | * These can be one-way or two-way, linked at the root domains * When two-way, every domain trusts every other domain * There is no replication of any AD data between the forests | 5 | 35 |
| **One-way or Two-way Forest Trust** | A two-way trust is just two one-way trusts going in opposite directions. Cross-forest trust are normally two-way. But you can make them one-way instead. The direction of the trust determines who can log on where and whether permissions and privileges can be granted. | 5 | 35 |
| **Transitive Forest Trusts** | When a cross-first trust is created, the trust link is transitive for all domains n both of the forests. To say that a trust link is transitive means that the trust passes through in a chain. | 5 | 35 |
| **Cross-Forest Replication** | Cross-Forests trust do not cause any replication of accounts (or any other data) between the two forests. Even with a two-way trust, there is no replication between the forests. | 5 | 36 |
| **Group Policy** | Group Policy is for Windows configuration management, can be used to configure evert security setting un windows so that you don’t have to touch every machine by hand. | 5 | 37 |
| **Group Policy - How it works** | * A GPO is like a configuration script that can change any setting when the script is run. * GPO are stored on and replicated between your domain controllers. * Clients download and apply GPOs at boot up, user logon, and roughly every 90 to 120 minutes after that. * A GPO can be applied to an entire domain or to just a selected site of organizational unit (OU) | 5 | 38 |
| **Windows as a Service** | OS updates must be obtained, tested, installed, and checked; this is a painful but absolutely necessary task when securing windows boxes. Implementing a patching management system is more important than having a perimeter firewall. | 5 | 45 |
| **End of Support**  **Modern Vs Fixed Lifecyle Policy** | Because of the critical importance of applying security updates, it is imperative to upgrade an OS before it becomes obsolete. Servers tend to be fewer in number and better managed, so upgrading a server is usually not a monumental task. Unless your environment is small or uses VDI technologies, your project planning for a mass client upgrade should begin at least 2 years before the expected end of support. | 5 | 46 |
| **Fixed Lifecycle Policy** | * End of Sales: When the product is no longer sold to retails or OEMs. * End of Mainstream Support: when warranties expire for the product, and it will no longer be improved, free incident support ends, and non-security updates become unavailable. * End of extended support: when security and paid support can no longer be purchases, except in special cases. * End of customer support: where there will be no further update or support options whatsoever unless they are negotiated with Microsoft. | 5 | 46 |
| **Fixed versus modern Lifecycle policy** | The fixed Lifecycle Policy is older and is itself being slowly phased out. The Modern Lifecyle Policy is newer and the future of Windows. When we talk about Windows as a server (WaaS), an important part of WaaS is the Modern Lifecycle Policy. | 5 | 46 |
| **Modern Lifecycle Policy** | Is for Microsoft products that are updated frequently and that must be updated in order to receive support. | 5 | 47 |
| **Expired Client OS** | Every month that it continues to run after expiration increased the probability to become infected. In this situation:   * Block all network connectivity to/from. * Switch to Edge or another browser that will get updates. * Keep all non-Microsoft applications up to date with patches. * Install a host-based intrusion prevention suite * Back up data more frequently. * Prepare for compromise because it will happen. | 5 | 47 |
| **Continuous Updates**  **(WaaS)** | * An update replaces existing program files with newer ones. * Most vulnerabilities are eliminated by applying updates. * Windows OS is continuously updating, and the changes are release as patches. | 5 | 48 |
| **Feature Updates** | Large new service or application changes that increment the Windows version (released roughly every 180 days, but possibly, only once a year. | 5 | 48 |
| **Quality Updates** | Smaller improvements to what currently exist, such as bug fixes and security patches (release every 30 days) | 5 | 48 |
| **Service Packs** | Feature updates were called service packs in the past. | 5 | 48 |
| **Windows Updates Vs Upgrades** | Upgrade is a change to a newer operating system version.  Update does not normally include an upgrade. It is also known as a patch. | 5 | 49 |
| **Critical for security**  **Windows patching** | * Most vulnerabilities are eliminated by applying updates. * Staying on top of the latest updates, testing them, rolling them out to boxes and auditing their correct installation consumes a great deal of time but it is vital for any organization. * Update management system is more important that a perimeter firewall. | 5 | 49 |
| **Microsoft Security Updates Guide** | * An update will roll up many changes into a single all-or-nothing package that fixes many bugs in one shot. This is called “Cumulative” update. * Update and vulnerability information is in a searchable online database called the security update guide”. | 5 | 50 |
| **Security Updates Guide** | Online database where you can search vulnerabilities. <Https://msrc.microsoft.com/update-guide> | 5 | 50 |
| **Windows Servicing Channels** | Are just options for delaying the installation of new updates instead of applying the updates immediately after release.   * Semi-Annual * Windows Insider * •Long-Term Channel | 5 | 51 |
| **Quality updates deferring –**  **Channel Update** | Semi-Annual channel: 30 days  Home Edition: 35 days | 5 | 52 |
| **Feature Updates deferring –**  **Channel Update** | Semi Annual & Home Edition: 18 months  18 months is the normal support period for a feature version. | 5 | 52 |
| **Branches –**  **Channel Update** | Microsoft used to call “channels” branches and people are still referring to channels as branches. | 5 | 53 |
| **Rings –**  **Channel Update** | A "ring" is just a group of computers to which you assign a particular servicing channel with a particular update deferral period, if any. How and why, you do this is totally independent of anything Microsoft does; it is unique and internal to your organization. | 5 | 53 |
| **Semi-Annual (Targeted)** | Used to be called the “current branch” | 5 | 53 |
| **Channel Configuration** | The distribution channel and deferral options are stored in the registry, and registry setting may be edited by hand or through Group policy. | 5 | 54 |
| **Telemetry Extortion** | Information about your computer, installed applications, application usage, system crashes, and other data are sent to Microsoft over TLS-encrypted channels on a regular basis. Microsoft calls this “Telemetry” but European nations call it spying. | 5 | 54 |
| **Long-Term Channel** | * Will never get feature updates (must upgrade entire OS) * Only monthly quality updates can be applied * Requires a Volume License Agreement with Microsoft | 5 | 56 |
| **Microsoft Volume Licensing Agreement** | licensing software in volume makes it easier and more affordable to run software on multiple computers within a single licensing organization. | 5 | 57 |
| **Windows Insider Program** | * Get access to upcoming feature updates while still in development * Important for early testing and feedback to Microsoft | 5 | 57 |
| **Windows Update** | Is a built-in Windows feature that connects on a scheduled basis to Microsoft’s download servers to download and install updates. It used to be called automatic updates. | 5 | 58 |
| **Windows Server Update Services (WSUS)** | * WSUS is your own local Windows Update server. * WSUS is built into Windows Server as an IIS web application. * You control which updates to deploy and when. * Clients can download from your WSUS server or from Microsoft. * Create your own custom groups of computers. * Scales up to thousands of machines when deployed in an array. * Can be managed using PowerShell. | 5 | 60 |
| **Background Intelligent Transfer (BITS)** | BITS “drizzles” files down to the client in the background so that other applications are not interrupted, and bandwidth is not monopolized by WSUS traffic. | 5 | 61 |
| **WSUS administration** | * WSUS is installed with the server manager tool like any other role of the feature * Users can be permitted to have full control over the update process. * It can work in the background at night to install and reboot machines. * Can create your own custom group of computers and then approve or deny different sets of updates for each group of computers. | 5 | 62 |
| **Microsoft Intune** | Is a configuration management system for devices anywhere in the world because Intune is hosted and run from Azure. | 5 | 63 |
| **Windows Autopilot** | 1. Device ID registered in Azure Ad credentials 2. OEM, MSP, or IT dept ships the device to the user. 3. User logs on with azure AD credentials 4. In the background, device joined to Azure AD, applications installed, latest updates applied, and configuration settings enforced | 5 | 63 |
| **Mobile Device Management (MDM)** | Works with Windows autopilot | 5 | 64 |
| **Windows Autopilot Requirements** | * Windows 10 Enterprise, Pro or Education edition (not Home) * Mobile Device Management (MDM) solution, such as Microsoft Intune * Computer OEM that will print a device ID on the shipping box * Computer must be joined to an Azure AD domain * Computer may optionally also be joined to an on-premises AD domain | 5 | 64 |
| **Autopilot Reset** | Allows a device to be transferred to another user without losing domain membership or Intune enrollment information. | 5 | 65 |
| **Azure Virtual Desktop** | Virtual Desktop Infrastructure (VDI) in the cloud.  Remote Desktop Protocol (RDP).  User device can run any OS.  Personal vs Shared VMs. | 5 | 66 |
| **Remote Desktop Protocol (RDP)** | To get Azure Virtual Desktop (AVD) the only requirement is that the client is able to run an RDP application and that I can reach the Azure cloud over a TLS-encrypted connection on TCP port 443. | 5 | 66 |
| **Azure Virtual Desktop**  **Management** | Create VM Pools:   * Shared versus personal VMs * CPU and GPU options * Memory and disk options * Spin up VMs for peak load * Drain VMs during off hours. Requirements: | 5 | 68 |
| **Azure Virtual Desktop**  **Management** | Requirements:   * Azure AD domain * Traditional AD domain * Domains synchronized * Low packet latency   Per-user licensing | 5 | 68 |
| **Packet Round Trip Time (RTT)** | Is the time between the user and the regional datacenter hosting the user’s VM. It should be less than 150ms. A RTT of 50 ms or less is preferred to avoid user complaints. | 5 | 69 |
| **Selective Access Control** | Permissions on files, folders, printers, registry keys, and other items allow you to regulate access to these objects. Some users only have read access to an object, whereas others are given full control. This kind of selective access control is possible only if users are authenticated first. | 5 | 71 |
| **User’s identity** | Represented by her security access token (SAT), which lists the Security ID number (SID) of the user’s account and all the SID numbers of the groups to which the user belongs A SAT also lists all of a user’s privileges on the computer where they sit. | 5 | 71 |
| **Windows Access Controls** | NTFS Permissions   * Shared Folder Permissions * Registry Key Permissions * Active Directory Permissions * Privileges * BitLocker Drive Encryption | 5 | 71 |
| **Windows Permissions and Privileges** | How permissions are applied in the Windows NT Filesystem, Shared Folders, Printers, Registry Keys, and Active Directory, and how Privileges are applied. | 5 | 73 |
| **CDFS** | CD Roms. | 5 | 74 |
| **NTFS** | * Filesystems supported by windows: * CDFS * FAT * FAT32 * exFAT * ReFS * NTFS | 5 | 74 |
| **FAT & FAT32** | Provide no auditing, access control, or fault tolerance. | 5 | 74 |
| **NTFS Overview** | The Windows NT File System (NTFS) should be considered the default filesystem. The exceptions are when you must retain the ability to boot into other operating systems, perhaps for disaster recovery, or when ReFS is chosen for large storage arrays. he NTFS filesystem has the following characteristics:   * Permissions * Auditing * Encryption (EFS and BitLocker) * Compression * Transaction-oriented processing * Theoretical maximum volume size: 8 PB (approximately 8000 terabytes) | 5 | 74 |
| **Windows Filesystem** | Windows supports a variety of filesystems, including: CDFS, FAT, FAT32, ReFS, and NTFS. CDFS is only for CD-ROMs. FAT and FAT32, although they can be used for hard drive partitions and are faster than NTFS on volumes smaller than 400 MB, provide no auditing, access control, or fault tolerance. | 5 | 74 |
| **Resilient File System (ReFS)** | Mainly intended for large storage volumes spread across multiple physical disks, such as in a RAID array. NTFS and ReFS are similar, both support permissions, BitLocker encryption and large volumes | 5 | 75 |
| **Access Control Entries (ACEs)** | Individual permissions in the DACL are called access control entries.  Permission ACEs are access through File Explorer. | 5 | 76 |
| **Discretionary Access Control List (DACL)** | A set of NTFS permissions on a folder or file is called a DACL | 5 | 76 |
| **NTFS DACLs** | Always enforced by the operating system. | 5 | 76 |
| **Standard ACEs (Security Tab)** | Standard, or generic permission ACEs. Each ACE consist of a user or group and the permissions assigned to that user/group as represented by the boxes checked. | 5 | 76 |
| **ICACLS.EXE** | It is a PowerShell cmdlet on manage NTFS permissions. | 5 | 76 |
| **ACEs - Advanced Security Settings** | The standard permissions are just collections of one of more individual permission ACEs. Individual permissions are the low-level, detailed, atomic ACEs that actually make up the DACL. A Deny permissions override Allow. Permissions can be inherited | 5 | 77 |
| **Permissions inheritance –**  **Scope** | * You can make it very explicit like: * This folder only * This folder, subfolders, and files * This folder and subfolders * This folder and files * Subfolders and files only * Subfolders only * Files only | 5 | 78 |
| **CREATOR OWNER group** | Thought of as the "WHOEVER THE OWNER HAPPENS TO BE RIGHT NOW" group. Example:   * **Folder, Subfolders,** and files: Authenticated Users: Read. * **Folder, Subfolders:** Authenticated Users: Create files * **Folder and Subfolders:** Authenticated Users: Create folders * **Subfolders and Files:** CREATOR OWNER: Full control. | 5 | 79 |
| **NTFS Owners** | * Every object has an owner. * Ownership can be changed. * Creator Owner group assigns permissions to whomever is the owner right now. * By default, whoever creates a file or folder becomes the owner. | 5 | 79 |
| **TAKEOWN.EXE** | From the command line you can use this command to take ownership of many files recursively on the local or remote computer if you have the take ownership privilege. | 5 | 80 |
| **DACL –**  **Default permissions** | **System:** Full Control  **Administrators:** Full Control  **CREATOR OWNER: Full** Control  **Authenticated Users:** Read and Execute (or Modify) | 5 | 81 |
| **Needs Analysis –**  **Principle of Least Privilege** | Before locking down NTFS permissions, you need to ask these questions:   * What is that you don’t want certain users to do? * What do you want to make sure other users can do? | 5 | 81 |
| **Principle of Least Privilege** | * Perform a “need analysis” based on job roles * Grant the minimum permissions that still permit users to get their legitimate work done. | 5 | 81 |
| **AGULP!** | Formal model on how privileges and permissions should be applied.   * ACCOUNTS: add user accounts to global groups account do users’ geographical locations, job descriptions, and shared needs. EX: participants in the company 401k plant since they all share a need to access 401k related resources. * GLOBAL GROUPS: add global groups to universal security groups whenever multiple global groups from multiple domains all need to be assigned the same privileges or permissions. can contain only members from the same domain where the group was crated, it can also contain members from any domain in the entire Active directory forest. * UNIVERSAL GROUPS: add global and universal groups to local groups on the computers with the resources that need to be secured. These local groups can be built in, or new ones can be created, but their purpose is to organize global and universal groups. * LOCAL GROUPS: assign permissions, logon rights, and privileges for these local groups to the resources/objects that need to be secured. * PERMISSIONS & RIGHTS: A local group can be assigned permissions, logon rights and privileges only to resources on the same computer where the local group exists. | 5 | 83 |
| **Local group – purpose** | To act as the bearer of a set of permissions, logon rights, and privileges so that these powers can be granted more easily to others. | 5 | 84 |
| **Universal group – purpose** | Is to gather together many global groups from multiple domains when those global groups all happen to need the same permissions/rights/privileges. | 5 | 84 |
| **Active Directory Accounts and Groups** | A Graphical tool (GUI) that manages users, computers, and groups in Active Directory is called “Active Directory Users and Computers”. | 5 | 85 |
| **Distribution Group** | Is like an email list. You cannot assign privileges or permissions to a distribution group. | 5 | 86 |
| **Groups in Active Directory** | In the Users container, you would also find the Domain Admins global group. The global Administration account is a member of Domain Admins by default. The Domain Admins group in a domain is ALL POWERFULL in that domain. In a forest with three domains, there would be three different global administrator user accounts and three different Domain Admins groups (one for each domain) | 5 | 86 |
| **Security Group** | Can have privileges or permissions to a distribution group. | 5 | 86 |
| **Domain Local Group** | Local groups can be granted permissions and privileges only on that one computer, and if you wanted something like a local group that was replicated through Active Directory and that could be used in ANY computers as well. | 5 | 87 |
| **Common Internet File System (CIFS) protocol** | It is just the Server Message Block (SMB) with a few enhancements. | 5 | 88 |
| **Server Message Block (SMB) protocol** | Used to share folder permissions | 5 | 88 |
| **Server service** | Also known as File and Print Sharing Service. Enforces shared permissions. | 5 | 88 |
| **Shared Folder Permissions** | A folder can be shared on the network by the File and Print Sharing Service known as Server service, and the Server Message Block (SMB) protocol.  A share folder has its own Discretionary Access Control List separated from any DACLS in the underlaying filesystem | 5 | 88 |
| **Share folders –**  **How** | ( for more information: get-help smbshare, net.exe help share) | 5 | 89 |
| **Hidden and Administrative Shares** | * ShareName$: The $ makes it hidden. * IPC$: for inter-process communications. * PowerShell: Get-SmbShare. * C$ and E$ are drive roots | 5 | 91 |
| **Administrative Shares** | * By default, the root folder of each drive-letter volume has a hidden share. * Its permissions are Administrations: full control and nothing else. * %SystemRoot% folder is also shared as ADMINS$ and permits read access to authenticated users. | 5 | 92 |
| **NTFS Permissions and Share DACLs - Combining** | NTFS permissions are always enforced, even when the user is remote. When a folder on an NTFS formatted volume, both the share permissions and the NTFS permissions must be considered when calculating a given user’s final, effective permissions. It has 3 steps:   1. Assemble a list of all the share permissions the user has to the folder for the user’s group memberships. 2. Assemble a list of all the NTFS permissions the user has to the file, this includes both explicit and inherited NTFS permissions. 3. Examine both the final share and the final NTFS permissions. | 5 | 93 |
| **Keys (REGEDIT.EXE)** | Yellow looking folders in REGEDIT GUI | 5 | 95 |
| **REGEDIT.EXE** | Graphical tool with access to the “Registry” where all configuration settings for the computer’s hardware, operating system, applications, and its users’ preferences are stored. | 5 | 95 |
| **Values (REGEDIT.EXE)** | File looking objects in these keys (yellow looking folders) | 5 | 95 |
| **winreg** | HKLM\System\CurrentControlSet\Control\SecurePipeServers\winreg\ | 5 | 96 |
| **REGSVC.EXE** | Remote Registry Service | 5 | 96 |
| **Remote Registry Service** | REGEDIT.EXE > File menu > Connect Network Registry. Disable the Remote Registry service to prevent network access Permissions on the winreg key are interpreted as the share DACL if the service is enabled | 5 | 96 |
| **Remote Registry Share Permissions** | Registry Key Permissions are always enforced against local or remote users, but you also can restrict who can access the registry remotely. | 5 | 96 |
| **AllowedPaths** | Winreg subkey that defines the registry paths that will still be remotely readable despite your share permissions on the winreg key. | 5 | 97 |
| **REGEDIT.EXE > Edit Menu > Permissions** | o manages the owner, auditing, and permissions of a registry key, open REGEDIT.EXE > highlight the key > Edit menu > Permissions. | 5 | 98 |
| **Active Directory Permissions** | In Active Directory, every object has an owner, permissions, and audit settings This includes users, computers, OUs, and entire domains | 5 | 99 |
| **Active Directory - Delegation of Authority in AD** | Just as you can delegate authority over a shared folder through its permissions, so you can delegate authority over user accounts, groups, computer accounts, and everything else in AD through the permissions on properties of AD objects. You can track precisely who is doing what because of each of these properties has its own audit settings. | 5 | 100 |
| **Force Shutdown from a Remote System** | It is another privilege, which does not refer to any particular object but is a dangerous capability one certainly would want to restrict. | 5 | 101 |
| **Privileges (1 of 5)** | Unlike permissions, privileges are not related to particular objects; they are general capabilities a user has on a computer.  Managed on each computer through Group Policy or PowerShell. whoami.exe /priv | 5 | 101 |
| **Rights** | Rights and privileges are synonyms. | 5 | 101 |
| **Take Ownership** | It is a privilege that allows one to take ownership of any object, and the owner of any object can change its permissions in any way the owner wants. | 5 | 101 |
| **whoami.exe /priv** |  | 5 | 101 |
| **NTRIGHTS.EXE** | Command tool used to manage privileges | 5 | 102 |
| **Privilege (User Right) - Descriptions** | Table of privileges with a short description of each | 5 | 102 |
| **Access this computer from the network** | Determines who is permitted to connect to the computer over the network using a protocol requiring user authentication. It does not apply to any packets whatsoever. | 5 | 102 |
| **Act as part of the operating system** | Allows a process to authenticate as any user or to create a SAT with any SID numbers and privileges desired. **This is dangerous.** | 5 | 102 |
| **Add workstations to domain** | Determines who can join workstations to the domain. Authenticated users have this privilege and can create up to ten computer accounts in the domain, but modification to an Active Directory property can increase or decrease this number. | 5 | 102 |
| **Allow logon through remote desktop services** | Determines, in part who can remotely log on to the computer using Remote Desktop Protocol (RDP( | 5 | 102 |
| **Back up files and directories** | Determines who can circumvent NTFS permissions fo the sale of making backups. | 5 | 102 |
| **Allow Log On Locally** | Restricts who can log on interactively at the keyboard | 5 | 103 |
| **Allow access to this computer from the network** | Restrict who can remotely authenticate to a computer. | 5 | 103 |
| **Allow log on through remote desktop services** | Restrict who can use Remote desktop protocol (RDP) | 5 | 103 |
| **auditd - Search and Report Audit Logs (3)** | Aureport provides human readable output on the audit events in audit.log  The command aureport -f -i –summary gives a nice overview on the number of audit records recorded per file  In this case, we wanted to search on our file sshd\_config so we added | grep sshd\_config, and with that option, we see the date and application that was used to access the file | 5 | 103 |
| **Privileges (2 of 5)** | To help isolate a server from certain users of groups you would assign the Deny access to this computer over the network, deny log on locally and deny log on through remote desktop services to users you don’t trust or without a need to log on.  If you want to deny everyone access to server except for a certain group, the assign the access this computer from the network, allow and log on locally, and or allow log on through remote desktop services rights to just those groups. EX: the database servers in the HR might be restricted only to admins and HR members. | 5 | 103 |
| **auditd Rules** | Audit Examples Rules | 5 | 104 |
| **Privileges (3 of 5)** | The owner of an object can change its permissions, no matter what the current permissions are.  Only administrators have this default.  Objects include files, folders, printers, AD containers, registry keys, processes, and threads. | 5 | 104 |
| **Take Ownership of Files or Other Objects** | The owner of an object can change its permissions in any way wanted. | 5 | 104 |
| **Backup Files and Directories** | “Circumvent NTFS permissions” privilege  Use group policy to delegate the backup files privilege | 5 | 105 |
| **Privileges (4 of 5)** | Ignore NTFS Permissions  Backup files and directories  Restore files and directories. | 5 | 105 |
| **Restore Files and Directories** | “Circumvent NTFS permissions” privilege  Reserve the Restore Files privilege for Domain admins only. | 5 | 105 |
| **Debug Programs privilege** | Used by developers for troubleshooting software and by security experts (and hackers) for reverse engineering. | 5 | 106 |
| **Windows "DLL Injection" Attack** | A new thread is injected into a victim process  Metasploit can use DLL injection to dump password hashes. | 5 | 106 |
| **Privileges (5 of 5)** | Debug Programs, Windows "DLL Injection" Attack | 5 | 106 |
| **The Cain Tool** |  | 5 | 106 |
| **BitLocker Overview** | Sectors encrypted with AES (128- or 256- bit)  Boot-up integrity checking with a TPM chip (TPM is optional)  Supports USB and thunderbolt drives  Emergency recovery PIN  Supports some self-encrypting hard drives (eDrive spec) | 5 | 107 |
| **Boot Volume – Bitlocker** | Contains the operating system files | 5 | 107 |
| **System Volume – Bitlocker** | Contains the files used during the very beginning of the boot-up process. | 5 | 107 |
| **Full Volume Encryption Key (FVEK)** | At this point, the boot-up environment is considered healthy, so the TPM gives a key to Windows to decrypt another key on the hard drive called the Full Volume Encryption Key (FVEK). The FVEK is the key that encrypts/decrypts all the sectors of the BitLocker-protected volume. The FVEK is 128-bit AES by default, but 256-bit AES can be used instead. With a self-encrypting drive, this process is different because it relies on UEFI Secure Boot (discussed later) | 5 | 108 |
| **Hardware-based Encryption Drives** | Some hard drives perform encryption internally inside the drive hardware itself, not using Windows or the motherboard CPU. These drives are called "self-encrypting drives" or "hardware-based encryption drives", and there is almost no performance penalty for the encryption. | 5 | 108 |
| **Self-encrypting drives** | Some hard drives perform encryption internally inside the drive hardware itself, not using Windows or the motherboard CPU. These drives are called "self-encrypting drives" or "hardware-based encryption drives", and there is almost no performance penalty for the encryption. | 5 | 108 |
| **BitLocker with a Trusted Platform Module (TPM)** | TPM is a chip in the motherboard.  TPM performs encryption, hashing, random key generation, secure key storage, and other crypto tasks.  A physical or virtual TPM can be used as a smart card or to encrypt biometric logon data, so it's not just for BitLocker | 5 | 109 |
| **Platform Trust Technology** | This feature of the Intel CPU is called "Platform Trust Technology" (PTT). For both AMD fTPM and Intel PTT, you may need to enable it in the firmware of the motherboard before it may be used. | 5 | 109 |
| **Trusted Platform Module (TPM)** | TPM is a chip in the motherboard. TPM performs encryption, hashing, random key generation, secure key storage, and other crypto tasks. A physical or virtual TPM can be used as a smart card or to encrypt biometric logon data, so it's not just for BitLocker. | 5 | 109 |
| **Initializing TPM** | Setting an owner's password | 5 | 110 |
| **BitLocker TPM Options** | **BitLocker with TPM:**   * TPM + USB Drive + PIN • TPM + USB Drive * TPM + PIN * TPM Only (vulnerable to cold boot attack).   **BitLocker with no TPM:**   * Pre-boot passphrase (Windows 8 or later) * USB drive with the key inserted at boot up * No boot-up integrity protection to detect malware, though | 5 | 111 |
| **Cold Boot Attack** | Why have these extra PIN and USB options on top of the TPM? If someone steals your laptop and boots it into Windows, the computer is now vulnerable to a "cold boot attack" in which the BitLocker decryption key (and other sensitive data, such as passwords) can be directly extracted from RAM by pulling the batteries and power, then rebooting from another USB or DVD drive to allow a search or dump of raw memory. By requiring a pre- boot PIN or USB token, it helps to defeat cold boot attacks under the right circumstances. | 5 | 112 |
| **BitLocker Emergency Recovery** | * Back up your BitLocker recovery password! * Recovery password is actually a 48-digit number * Recovery password can decrypt a volume even if the TPM is damaged, the user PIN is forgotten, or the USB token is lost. * Use Group Policy to force backup of the recovery password to Active Directory for scalable mass deployments. | 5 | 113 |
| **Recovery Procedure –**  **BitLocker** | If the TPM chip is damaged or cleared, f the boot-up environment is modified, or if the user forgets their PIN or loses their USB token, then emergency recovery is required. | 5 | 113 |
| **BitLocker - Force Backup to Active Directory** | If the recovery password has been lost, or if it wasn't created in the first place, then the last hope is for a recovery key to have been stored in Active Directory. Using Group Policy, the plaintext BitLocker recovery password can be stored in the Active Directory database, as well as an encrypted copy of the FVEK key itself. If multiple volumes on Windows Server are encrypted, recovery information will be stored for each volume in AD. | 5 | 114 |
| **BitLocker with UEFI Secure Boot**  **(Unified extensible firmware interface)** | UEFI Secure boot:   * digitally signed binaries that load the operating system, * digitally signed UEFI firmware, early load of antivirus scanners before any other third-party software or drivers are loaded, and * optional Trusted Platform Module (TPM) integration to measure other aspects of the computer's boot-up components to detect changes. * Secure Boot checks the digital signatures of motherboard firmware, operating system binaries, and option ROM firmware in supported peripherals from the time of power-on until the antivirus scanner is running, at which point the AV scanner takes over the job of combating malware. | 5 | 115 |
| **Disallowed File Hashes** |  | 5 | 115 |
| **Secure Boot** |  | 5 | 115 |
| **UEFI Secure Boot** | * Checks digital signatures of boot-up binaries and the firmware itself * Early load of antivirus drivers during boot * Secure Boot + BitLocker | 5 | 115 |
| **Unified Extensible Firmware Interface (UEFI)** | UEFI replaces the older BIOS firmware  UEFI cannot be added later; it must be built into the motherboard | 5 | 115 |
| **Enforcing Security Policy** | This module discusses one of the best tools for automating security configuration changes, Microsoft's Security Configuration and Analysis (SCA) snap-in, and some of the most important changes to make with it, such as password policy, lockout policy, and null user session restrictions. It also briefly discusses Group Policy Objects (GPOs) and the many security configuration changes that can be made through them throughout the domain. | 5 | 120 |
| **Security Templates (1 of 3)** | A security template is a plaintext configuration file that can store hundreds of security settings. A computer can be stamped with a template and reconfigured in one shot to match the settings in the template. | 5 | 122 |
| **Windows Security Templates** | A security template is a plaintext configuration file that can store hundreds of security settings. A computer can be stamped with a template and reconfigured in one shot to match the settings in the template. | 5 | 122 |
| **Security Template –**  **What is in a template?** | * Password policy * Lockout policy * Audit policy * Privileges * Event log settings NTFS permissions * Group memberships * Service startup * Registry permissions | 5 | 122 |
| **Security Templates (2 of 3)** | Security templates are kept by default in %SystemRoot%\Security\Templates\ and also %SystemRoot%\Inf\, and they end with the .INF filename extension. Templates can be edited with Notepad, but a much easier method is to use a Microsoft Management Console (MMC) snap-in named Security Templates. | 5 | 123 |
| **Security Baseline Guides** | There are individual guides for the older products, too, if you have not yet upgraded. | 5 | 124 |
| **Security Templates (3 of 3)** | Don't create a new template from scratch; start with a template with good, tested settings, then edit for your environment. Several sources of security guidance for Windows:   * Microsoft's security baseline guides * Guides from the United States Government (DoD, NIST, NSA) * Center for Internet Security (CIS) | 5 | 124 |
| **Center for Internet Security (CIS)** | Has security templates available but also configuration guides to go with them. | 5 | 125 |
| **Defense Information Systems Agency (DISA)** | These guidelines come in the form of Security Technical Implementation Guides (STIGs), which include security templates, checklists. Scripts, SCAP XML specifications and other documents. | 5 | 125 |
| **Federal Desktop Core Configuration (FDCC)** | These templates were incorporated into the Federal Desktop Core configuration (FDCC) standards | 5 | 125 |
| **Security Technical Implementation Guides (STIGs)** |  | 5 | 125 |
| **SCA Snap-In (1 of 2)**  **Security Configuration and Analysis Tool.** | Security Configuration and Analysis Tool. Warning: There is no undo feature!   * Applies a template to a computer (reconfigures the computer) * Compares a template to a computer's actual settings (audit only) * We cannot apply a template to a computer across the network with this tool, but that's why we have Group Policy and PowerShell | 5 | 126 |
| **SCA Snap-In (2 of 2)**  **Security Configuration and Analysis** | Security Configuration and Analysis. | 5 | 127 |
| **SECEDIT.EXE** | SECEDIT.EXE is the command line version of the graphical SCA tool. Use PowerShell to script the application of a security template with SECEDIT.EXE from a shared folder, portable drive, or DVD | 5 | 128 |
| **Group Policy Object (GPO)** | * A set of configuration changes. A giant, comprehensive template. * GPO > Computer Configuration Applies even when no one is logged on. * GPO > User Configuration Applies to current user's desktop | 5 | 129 |
| **Local Group Policy Object** | * Every Windows computer has a local GPO * Add the "Group Policy Object Editor" snap-in to an MMC window | 5 | 129 |
| **INF Template** | INF files are most commonly used for installing device drivers for hardware components. | 5 | 130 |
| **Local GPO Security Settings** | Import an INF template into the "Security Settings" container of the GPO. | 5 | 130 |
| **GPO Scripts** | Start/Shutdown script run as the operating system  Logon/Logoff scripts run as the console user.  PowerShell, VBScript, Jscript and other languages are supported. | 5 | 131 |
| **Local GPO Scripts** | You can assign scripts to be executed through a GPO as well.  These options are found under both Computer and User Configuration > Windows Settings > Scripts | 5 | 131 |
| **Administrative Templates** | Many aspects of the user's graphical interface can be configured through Administrative Templates. For example, under User Configuration > Administrative Templates > Control Panel, there is an option to restrict which Control Panel applets the user is permitted to open (and another to restrict access to the Control Panel entirely). In that same section is a subcontainer (Display) that can be used to require a password-protected screensaver. | 5 | 132 |
| **INF Template** | An INF template can be directly applied to a machine with SECEDIT.EXE or imported into a GPO. An INF template is a text file that contains the settings to be applied, such as registry value changes or NTFS permissions change | 5 | 132 |
| **Local GPO Administrative Templates** | * A user-friendly registry editor * Registry controls almost everything * Hundreds of settings are available * You can import more * ADM templates You can edit these * ADM templates to configure any registry value | 5 | 132 |
| **Default Domain Policy** | * There is a GPO named "Default Domain Policy" stored on domain controllers. * This GPO applies to all users and computers throughout the entire domain by default. The Default Domain GPO applies to everyone in the domain, whereas a local GPO applies only to the computer on which it is found. * The settings in this domain GPO override any conflicting settings in the local GPO on each computer; hence, domain administrators always have the final say. | 5 | 134 |
| **Domain Group Policy Objects** | * Domain GPOs are stored in Active Directory. * Downloaded automatically at startup, shutdown, logon, and logoff. * Refreshed on clients roughly every 90-120 minutes * Only downloaded by domain-joined computers, not standalones. * Domain GPO’s can be applied to thousands of computers joined to the domain without touching each machine | 5 | 134 |
| **Organizational Unit (OU)** |  | 5 | 134 |
| **Group Policy Management Console (GPMC)** | * The GPMC is your primary tool for creating, editing, and managing domain GPOs. * Use the GPMC tool to edit GPOs on domain controllers * GPMC installed on windows server by default, can be added to workstations too. | 5 | 135 |
| **GPO Settings - Best Practices** | * Password Policy. * Account lockout policy * Kerberos and NTLMv2 * Credential guard * Protecting administrative accounts * User account control | 5 | 136 |
| **Password Policy** | * Enforce password history: 24 passwords * Minimum password age: 1 day * Minimum password length: 8 characters. Maximum length: 127 characters * Think passphrases, not passwords. | 5 | 140 |
| **Account Lockout Policy** | * Account lockout duration: 120 minutes * Account lockout threshold: 5 attempts * Reset account counter lockout after: 45 minutes * Thwart brute force password guessing attacks. | 5 | 142 |
| **Kerberos** | * Best way to support authentication in a Windows environment. * Default protocol * Faster, more secure * Requires Active Directory domain membership | 5 | 143 |
| **NTLMv1** | * Still supported for standalones and backward compatibility * Susceptible to password sniffing attacks, but ... | 5 | 143 |
| **LAN Manager Authentication Level** | * Send NTLMv2 Response Only (when negotiating) * Refuse NTLMv1 and LanManager | 5 | 144 |
| **NTLMv2** | * Not vulnerable to sniff-and- crack attacks if you have a passphrase * Still vulnerable to SMB relay attacks | 5 | 144 |
| **Credential Guard** | Credential Guard protects many (but not all) credentials in memory from kernel-mode malware and penetration testing tools like Mimikatz.  Requirements:   * UEFI firmware * UEFI secure boot enabled * Windows 10 or later * Server 2016 or later * Hyper-V hypervisor enabled * Must have enterprise edition, not pro. | 5 | 146 |
| **Administrative Accounts - Protection** | Require smart card authentication (if you have a PKI)  Enforce a strong passphrase policy  Enable Credential Guard  Require Kerberos and NTLMv2 (forbid LanManager and NTLMv1)  Give each admin at least two user accounts:   * Regular account (regular activities) * Administrative account (only as needed) | 5 | 148 |
| **Run as a Different User** |  | 5 | 149 |
| **Administrative Accounts - Protection** | * Admins should have two workstations (or a jump server) * Limit local account use of blank passwords to console logons only * Audit all access to administrative users and groups in AD * Rename the built-in Administrator account * Create a honeypot "Administrator" account as an IDS canary | 5 | 150 |
| **AppLocker** | * Regulate the processes and scripts users are permitted to launch * Helps fight malware and unauthorized apps. * Requires Enterprise or Education Edition for central management (no Home or Pro) | 5 | 151 |
| **AppLocker –**  **Rules** | * SHA256 hash of program * Local path to program * Network path to program * Code signing certificate * User’s group membership. | 5 | 151 |
| **AppLocker –**  **Rules apply to** | * Executables (EXE and DDL) * MSI software packages * APPX software packages * Scripts (many types) | 5 | 151 |
| **User Account Control** | User Account Control (UAC) in Windows Vista and later enables users to conveniently install and run programs as low-privileged accounts and then temporarily raise privileges on an as-needed basis without logging on and off or resorting to RUNAS.EXE or DROPMYRIGHTS.EXE in order to do so.  St**andard user process (default):**   * SAT stripped of dangerous privileges.   **Administrative user process:**   * Standard STA for a member of the administrators local group. * Right-click > Run As administrator.   **UAC can be managed via group policy:**   * Admin approval modes: prompting options: Prompting options * Standard User approval modes: Fail or prompt for credentials. | 5 | 153 |
| **User Account Control –**  **Group policy options** | UAC: Behavior of the elevation prompt determines how UAC allows the execution of commands that require administrative privileges.  **Administrator-level users:**   1. No prompt (if UAC is disabled 2. Prompt to consent (to be alerted) 3. Prompt for credentials: which requires a password or smart card.   **Standard users:**   1. No prompt or simply fail 2. Prompt for credentials, which requires the password or smart card of an administrative-level account. | 5 | 154 |
| **Key protocols** | Perimeter and host-based firewalls should filter out unwanted traffic.  Protocols to memorize:   1. SMB: TCP/139/445 2. RPC: TCP/135 3. LDAP: TCP/389/636/3268/3269 4. Kerberos: TCP/UDP/88 | 5 | 155 |
| **Remote Procedure Call (RPC): TCP/135** | Remote Procedure Call (RPC) networking is used extensively on Windows networks. | 5 | 155 |
| **Server Message Block (SMB): TCP/139/445** | Server Message Block (SMB) is the file and printer sharing protocol. When using NetBIOS, SMB operates on TCP/139; without NetBIOS, it uses TCP/445 and is sometimes referred to as the Common Internet File System (CIFS) protocol. All SMB/CIFS packets should be blocked going to or coming from the internet unless they are being tunneled through IPsec or a VPN. | 5 | 155 |
| **Windows Network Traffic** | There are certain network traffic flows that are characteristic of Windows networks. You should be familiar with their signatures and purposes so that you can recognize them in your packet traces and firewall logs. This list will also be important later when conducting audits. Hackers can use these port numbers to help "fingerprint" your boxes too. These essential protocols and their port numbers should be memorized. | 5 | 155 |
| **Global Catalog** | It is a special portion of the Active Directory database. It is LDAP accessible. | 5 | 156 |
| **Kerberos: TCP/UDP/88** | Kerberos is the default authentication protocol on Active Directory networks. It uses UDP/88 primarily; however, when tickets get too large, TCP/88 is used. The Kerberos change password port (TCP/UDP/464) is listening on domain controllers, but Windows clients still prefer to use an RPC session to change their passwords. Neither the Kerberos administration port (TCP/749) is used nor the Kerberos demultiplexor (TCP/2053) | 5 | 156 |
| **Lightweight Directory Access Protocol (LDAP): TCP/389/636/3268/3269** | The Lightweight Directory Access Protocol (LDAP) is the default protocol for searching and editing the Active Directory database. Cleartext LDAP uses TCP/389, whereas SSL-encrypted LDAP goes over TCP/636. A special portion of the Active Directory database called the global catalog also is LDAP-accessible over TCP/3268 (cleartext) and TCP/3269 (SSL-encrypted) on domain controllers. LDAP uses Kerberos for authentication, so it is not the case that the cleartext channels send passwords in the clear too. | 5 | 156 |
| **Key Protocols –**  **More** | DNS: TCP/UDP/53  RDP: TCP/UDP/3389  SQL Server: TCP/UDP/1433/1434  NetBIOS: TCP/UDP/137, UDP/138, TCP/139, TCP/UDP/1512, TCO/42.  IPSec: UDP/500/450 for IKE, Protocols 50 and 51 for ESP and AH. | 5 | 157 |
| **Internet Protocol Security (IPsec): UDP/500/4500, Protocols 50 and 51** | Internet Protocol Security (IPsec) is supported natively on Windows; it does not have to be installed. It is used for authenticating and encrypting packet data, including Layer 2 Transport Protocol (L2TP) VPNs. | 5 | 157 |
| **NetBIOS and WINS: TCP/UDP/137, UDP/138, TCP/139, TCP/UDP/1512, TCP/42** | Indeed, NetBIOS is like a constant background noise that you have to exclude from your logs. (It's mind-boggling how chatty Windows machines can be.) | 5 | 157 |
| **Remote Desktop Protocol (RDP): TCP/UDP/3389** | Windows Remote Desktop Services (formerly known as "Terminal Services") uses the Remote Desktop Protocol (RDP) to provide remote control of desktops and hosted applications. Windows clients use the same protocol for the Remote Assistance feature. RDP operates on TCP and UDP 3389. (Citrix ICA uses TCP/1494, which is a different protocol with a similar purpose.) | 5 | 157 |
| **Microsoft SQL Server: TCP/UDP/1433/1434** | SQL Server listens for queries on TCP/UDP/1433 and is monitored on TCP/UDP/1434. If you use client/server applications with SQL Server, you see a ton of this traffic. | 5 | 158 |
| **Point-to-Point Tunneling Protocol (PPTP): TCP/1723, Protocol 47** | Point-to-Point Tunneling Protocol (PPTP) is another VPN protocol. PPTP uses both TCP/1723 and the Generic Routing Encapsulation (GRE) protocol. GRE operates on protocol ID number 47. This protocol is completely obsolete, however, and should never be used. It is mentioned here only because it refuses to die. | 5 | 158 |
| **Secure Sockets Layer (SSL) and Transport Layer Security (TLS): TCP/443** | SSL and TLS are widely used for web browsing, email, VPNs, and many other protocols intended to cross over the internet. For good or bad, we place an immense amount of trust (or just hope) in these protocols. Strictly speaking, SSL is obsolete and only TLS should be used going forward, but SSL nonetheless is still widely deployed, and the term "SSL" is entrenched in the minds of the nontechnical public, so if only as a marketing term, SSL will be mentioned for many years to come. | 5 | 158 |
| **Windows Defender Firewall:**  **WF.MSC to launch** | There are various ways to launch the WF management tool, as shown in the slide above:   * Execute "wf" in PowerShell or in the Run dialog box to launch the WF.MSC console (without the quotes).-[ * Open Control Panel > View By: Large Icons > Windows Defender Firewall > Advanced settings link on the left. * Find Administrative Tools in either the Start menu or in Control Panel > Windows Defender Firewall. | 5 | 159 |
| **Windows Defender Firewall:**  **Cons** | * No centralized logging or alerting capabilities * No IDS features * No user behavior monitoring features complex. | 5 | 160 |
| **Domain Network** | Selected automatically when AD is available. | 5 | 161 |
| **Network Location: Types** | Each live network interface (physical, virtual, Wi-Fi, or VPN) is assigned a profile type.  You can have different firewall rules for each profile.  You can have different firewall rules for each profile:   * Domain (selected automatically when AD is available) * Public (coffee shops, hotels, airports, and such) * Private (home and office) | 5 | 161 |
| **Private Network** | Home and office. | 5 | 161 |
| **Public Network** | Coffee shops, Hotels, Airports, and Such. | 5 | 161 |
| **Firewall Defaults:**  **Network Locations** | For each profile, you can enable the firewall, block/allow inbound our outbound connections. Configure logging options and specify whether the user is notified when a program is prevented from receiving inbound connections (giving administrative users the opportunity to allow them in for that program). | 5 | 162 |
| **Firewall Rules - Managing** | Manage the firewall with Group Policy, PowerShell or NETSH.EXE  Can be organized by the network profile(s) in which they are activated (a rule is only enforced when its associated profile is currently active) | 5 | 163 |
| **Firewall-IPsec Integration** | * Secure Connection Mutual authentication and packet signing. * Require Encryption Mutual authentication, and encryption. | 5 | 163 |
| **Secure Connection** | Firewall-IPSec integration, Mutual authentication and packet signing. | 5 | 163 |
| **Secure Connection with Encryption Required** | Firewall-IPSec integration, Mutual authentication, and encryption. | 5 | 163 |
| **Firewall: exceptions** | * Keep blocking * Unblock | 5 | 165 |
| **Firewall Rules - Order of Rule Processing** | 1. Rules that allow/block traffic for particular services.  2. Rules that allow traffic only if it is secured with IPsec (AH or ESP).  3. Rules that block traffic, inbound or outbound.  4. Rules that allow traffic, inbound or outbound, with or without IPsec. 5.  Default behavior for the active network profile (allow or block). | 5 | 165 |
| **NETSH.EXE** | The Windows Firewall can be managed with NETSH.EXE from the command line. NETSH.EXE isn't just for the firewall, it's a general-purpose tool for managing networking-related settings. | 5 | 165 |
| **PowerShell (Win 8, Server 2012 and later)** | The older binaries, such as NETSH.EXE, have been deprecated in favor of PowerShell. There are a large number of cmdlets for IPsec and firewall rules in Server 2012, Windows 8, and later. To see a listing of these cmdlets, run this command: Get-Command -Module NetSecurity | 5 | 166 |
| **IPSec: Internet Protocol Security** | * Not just for VPNs * Mutual authentication with Kerberos certificates * 256-bit AES packet encryption * Packet digital signatures for integrity and proof of origin * IPSec driver built-in: no extra software must be installed. | 5 | 168 |
| **IPsec: Benefits** | * Zero-trust: limit access to TCP/UPD ports based on group memberships in Active Directory (similar to shared folder permissions, but for ports) * Encrypt RPC,SMB, DNS, LDAP, RDP, VNC or PowerShell Remoting over the internet without a VPN. | 5 | 168 |
| **IPsec: Command Line Tools** | PowerShell:  • Get-Help \*IPsec\*  • Get-Help -Full New-NetIPsecRule.  NETSH.EXE:  • netsh.exe advfirewall consec /? | 5 | 169 |
| **IPsec: and Group Policy** | 100% of IPsec settings can be managed through Group Policy and PowerShell | 5 | 170 |
| **IPsec: Deployment example** | * Enable IPsec on all computers in the domain, but don’t require IPsec, only **request** IPsec. * **Require** IPsec only for the servers in one OU, perhaps only for SMB | 5 | 170 |
| **IPsec: Connection Security Rule** | In the screenshot in the slide, a Connection Security Rule is an IPsec rule, and this rule can be set to require IPsec or to merely request IPsec. The rule in the screenshot is requesting only IPsec, meaning that a plaintext connection would still be allowed if the IPsec negotiation failed for some reason. | 5 | 171 |
| **IPsec: Group Policy Example** | To request but not require IPsec on all computers in the domain, edit the Default Domain Policy Group Policy Object. Because it is the Default Domain Policy GPO, it applies to all systems. Of course, this is just a brief example, in real life, this requires more careful planning and testing. | 5 | 171 |
| **Remote Desktop Services** | * RDS permits remote control of a graphical desktop. * Admins can remotely manage servers (LAN or cloud) * Users can access their own workstations from home * Help desk personnel can remotely assist users * Hundreds of remote users can share one server | 5 | 173 |
| **Remote Assistance** | IT person helping you remotely. It also uses RDS or Remote Desktop Services. | 5 | 174 |
| **MSTSC.EXE - Microsoft's Built-In Thin Client App** | In order to connect:   1. You must be able to authenticate successfully, 2. You must be authorized for remote desktop access, and 3. There cannot be a host-based or perimeter firewall blocking access to the RDS port (TCP/3389). Once connected, the MSTSC dialog box disappears, and a new rectangular window is displayed showing the desktop of the remote computer. It looks just like the window in which a local VM is running, but it happens to be the remote system instead. | 5 | 175 |
| **Remote Desktop Connection** | The main thin client application built into Windows is MSTSC.EXE also known as Remote Desktop. | 5 | 175 |
| **Thin Client** | The local application running on a user’s computer that displays a graphical desktop from another remote computer. | 5 | 175 |
| **Thin Client: Linux, Apple iOS, macOS, and Android** | There are other think client apps available besides MSTSC.EXE, including free and commercial apps for Linux, Apple iOS, macOS, and android. | 5 | 175 |
| **Desktop as a Service (DaaS)** | Think of what having a thin client means for cross-platform compatibility and bring your own device (BYOD) to work. A user might prefer using ubuntu at home and Mac at work, but needing to run mission-critical Windows applications remotely using RDS. When the applications and data are hosted in the cloud, this opens up the possibility of Desktop as a Service or DaaS. | 5 | 176 |
| **Remote Assistance** | The intent of Remote Assistance is to use the Remote Desktop capabilities to allow a trusted remote person to help troubleshoot problems on the local user's desktop. | 5 | 177 |
| **Remote Desktop** | Remote Desktop Services (RDS) is a role for Windows Server. When this role is not installed on server or workstation, but you can still access the desktop of that machine over the network, the feature is called Remote Desktop (without the word "Services" at the end). You enable or disable Remote Desktop by going to the System applet in Control Panel, clicking the Remote Settings link on the left, then going to the Remote tab. | 5 | 177 |
| **Easy Connect** | A password is displayed on the user’s screen, then the user conveys this password to the helper by phone, email, text message or instant message app. The remote helper enters the password when connecting. | 5 | 179 |
| **RDP Encryption Levels** | There are four possible RDP encryption levels:   * **Low**: Client determines encryption strength and only data sent to the server is encrypted. Data received from the server is cleartext. * **Client Compatible:** Encryption strength is determined by the RDP client, but all data is now encrypted, both to and from the server. TLS is permitted if requested. * **High:** Encryption is set to the server's highest level of encryption possible. Any client that cannot support it is rejected. TLS is permitted if requested. * **FIPS Compliant:** Similar to High, but the algorithms used must be 3DES, AES, RSA, and/or SHA. RC4 is not permitted. TLS is permitted if requested. | 5 | 180 |
| **Remote Desktop Protocol (RDP)** | * TCP and UDP port 3389. * RDP sends keyboard, mouse, and touch input to server. * RDP returns graphics changes and sounds back to the thin client application on the user's computer. * RDP also transports the user's authentication data to server | 5 | 180 |
| **Terminal Services** | RDS (Remote Desk Services) was called Terminal Services long time ago. | 5 | 180 |
| **Network Level Authentication (NLA)** | * Authenticates the client and server before a session is even created in the memory of the RDS server. * NLA helps to prevent DoS attacks against the server and potential credentials-stealing attacks against the client. | 5 | 181 |
| **RDP Authentication Levels** | You have three choices for RDP authentication:   * **RDP:** Uses native RDP encryption and authentication * **Negotiate:** Tries to use TLS but falls back to native RDP if necessary * **SSL:** It's actually TLS, not SSL. Native RDP encryption is no longer supported. * Users do not have to be authenticated with a password. * Users may also authenticate to the RDS server with a smart card. | 5 | 181 |
| **Remote Desktop Services - Best Practices** | * Perimeter firewall to restrict TCP/UDP 3389 traffic. * Set up a VPN or RDS gateway for roaming users. * Host-based firewalls, especially for mobile devices. * Require a smart card or impose a strong passphrase policy. * Use TLS or IPsec instead of the default encryption   More on the book. | 5 | 182 |
| **Microsoft Cloud Computing** | Just as Apple integrates iCloud and iTunes into its phones and tablets, so Microsoft integrates Windows into its own cloud services. Windows security now must include Microsoft cloud security. Understanding Windows requires at least a basic understanding of Microsoft's cloud services, which requires an understanding of the essentials of cloud computing in general | 5 | 189 |
| **Cloud Computing - Types** | Desktop as a Service (DaaS) + SaaS + PaaS + IaaS.  Software as a Service (SaaS) + PaaS + IaaS.  Platform as a Service (PaaS) + IaaS.  Infrastructure as a Service (IaaS).  More is outsourced to the provider as you move upwards. | 5 | 190 |
| **Infrastructure as a Service (IaaS)** | Infrastructure as a Service (IaaS) is what a cloud provider offers when the provider will host the virtual machines or containers of the customer, but then do little else beyond this bare hosting. | 5 | 190 |
| **Platform as a Service (PaaS)** | Platform as a Service (PaaS) is what a cloud provider offers when the provider makes available virtual machines or containers with licenses, plus most of the additional applications and services developers need to build and run their own applications hosted on the PaaS provider's network. | 5 | 190 |
| **AutoPilot** | Microsoft Service that allows a user to power up an OEM-customized device, have that device automatically upgraded to Windows Enterprise over the internet, joined to the organization’s Azure AD domain in Microsoft’s cloud, automatically install Microsoft 365 applications, enroll for Intune configuration management, and permit long with a Microsoft Account in the cloud. | 5 | 191 |
| **Desktop as a Service (DaaS)** | Desktop as a Service (DaaS) is what a provider offers when a user-oriented operating system, such as Windows 11 or Linux Mint, is run as a VM, container, or thin-client session on the provider's hardware, then accessed by users over the internet so that the user can interact with the graphical desktop of that OS. | 5 | 191 |
| **Software as a Service (SaaS)** | Software as a Service (SaaS) is what a cloud provider offers when the provider manages the applications with which users interact and everything else down the stack. Microsoft 365, OneDrive, Outlook.com, and SharePoint Online are examples of SaaS offerings. | 5 | 191 |
| **Azure (and Azure Active Directory)** | Azure is Microsoft’s global collection of datacenters to implement Microsoft’s cloud-based services. Azure consists of over 1000 million virtual machines and containers, plus all the physical servers, storage arrays, network devices, and other hardware necessary to host these VMs and containers. | 5 | 192 |
| **Microsoft Cloud Computing: The Big Three** | * Azure (https://azure.microsoft.com) * OneDrive (<https://onedrive.com>) * Office 365 (https://office.com) | 5 | 192 |
| **Free Cloud** | There are free services to entice consumers to start them down the path toward cloud computing (the free versions of OneDrive, Outlook.com, Skype, Teams, Office Online, and Office Mobile), and these services are indeed useful, but they also lack the full functionality needed by enterprises. These services are teasers. They train and familiarize the public on the use of Microsoft's cloud services. | 5 | 193 |
| **Hybrid Cloud** | There are various paid subscription versions of a few essential applications, such as email and document collaboration, and these do have most of the functionality wanted, but they are sold and accessed separately (Exchange Online, SharePoint Online, Teams, and OneDrive for Business). These offerings are aimed at those organizations that require a hybrid approach to cloud computing, doing a mixture of both on-premises and cloud-based services simultaneously. To Microsoft, these are transition services to help ease the way toward eventually doing everything in Microsoft's cloud | 5 | 193 |
| **Microsoft 365** | Is the biggest of the “big three”. Azure and OneDrive really are supporting services to make Microsoft365 possible. | 5 | 193 |
| **Microsoft Cloud Services - Types** | Free, Hybrid, and Full. | 5 | 193 |
| **Office 365** | When “Office 365” subscription includes Windows Enterprise licenses and device management features, it is called Microsoft 365 instead of just Office 365 | 5 | 193 |
| **Exchange Online** | Exchange online, SharePoint online, etc are aimed to organizations that require a hybrid approach to cloud computer. | 5 | 194 |
| **Full Cloud** | There is the full cloud experience (Microsoft 365, Azure AD, Intune, etc.), sold as a per-user monthly subscription price, which aims to move almost all of your data and applications up to Microsoft's servers. Microsoft 365 comes in a wide array of bundles (Personal, Home, ProPlus, Small Business, Small Business Premium, Midsize Business, and Enterprise), and there are special programs for governments, schools, and nonprofits too. Microsoft 365 is also compatible with a hybrid solution, but a hybrid in which most data and applications are hosted in Azure, not on premises. Microsoft 365 is Microsoft's ultimate goal. It is the cloud-based Rome toward which all roads lead. Microsoft 365 is how Microsoft plans to thrive as a corporation even if Android or Apple iOS were to somehow supplant Windows on all endpoint devices. | 5 | 194 |
| **Intune** | Is for the inventory and management of hardware devices, including Android, Apple iOS, and Windows. It can enforce MDM security polices, inventory operating systems versions, inventory the apps installed and remotely install and update apps. Intune can also perform remote data wipe, remote lock and remote PIN reset. | 5 | 194 |
| **Microsoft Office** | “Local Install” or “Desktop Installed” is the traditional bundle of Microsoft Office applications that have existed for decades, including Word, Excel, and PowerPoint. | 5 | 194 |
| **Microsoft Teams** | Is for group collaboration, messaging, group chat, real-time presence status, VoIP, video teleconferencing, file sharing, project-oriented calendars, large audience broadcasts including deep integrations with Microsoft 365 and Azure AD. | 5 | 194 |
| **Outlook.com** | Previously known as Hotmail and it not the same thing as Exchange Online. Includes free email, calendar, contacts, and OneDrive-integration features. | 5 | 194 |
| **SharePoint Online** | Is for cloud file storage for multiple teams or projects, content management, live collaboration, enterprise social networking, and e-discovery for regulatory compliance. | 5 | 194 |
| **Skype** | Is for cloud chat, voice over IP, video conferencing and desktop sharing. | 5 | 194 |
| **Azure Virtual Desktop** | Is the Azure DaaS offering. With Azure virtual desktop, a roaming user can always access a window client OS in the cloud pre-configured with all the apps that user requires. The user’s data will also presumably be stored in Azure too, such as in OneDrive, but this is not mandatory. | 5 | 195 |
| **Office Mobile** | Includes miniaturized versions of some of the office apps for installation on Apple iPhone and Android. | 5 | 195 |
| **Office Online** | Is for browser-based free, cloud versions of Microsoft Word, Excel, PowerPoint, sway, Publisher, and OneNote. Client platforms include Windows, Android, C | 5 | 195 |
| **Microsoft Azure Active Directory (MAAD)** | * A planetary-scale user and device accounts database for authentication and access control to Microsoft 365, Outlook.com, OneDrive, and more. * Not just for Microsoft's own SaaS offerings, but a PaaS authentication and identity platform for third-party developers too. * It doesn’t support Kerberos, NTLM, Group Policy, or LDAP * Supports Web-based authentication like SAML, Oauth, OpenID Connect, and WS-Federation. It is integrated in Intune. | 5 | 196 |
| **Azure Active Directory Domain Services (AADDS)** | It provides traditional domain controller serves in Azure to ones’ other VMs hosted in Azure for the sake of Kerberos, NTLM, Group Policy and LDAP.  AADDS is Microsoft maintaining traditional controllers for you so that you don’t have to upload and maintain your own VM controllers. | 5 | 197 |
| **Azure AD Connect Tool** | * Free Microsoft tool * Syncs your domain controllers with Azure AD * Allows SSO to Microsoft 365 and to on- premises servers with the same password * Sync passwords one-way or two-way, or use pass-through authentication | 5 | 198 |
| **Active Directory Federation Services (ADFS)** | If Syncing password hashes to Azure AD is unacceptable, another route is to use Active Directory Federation services. When you install an ADFS server in your local DMZ, it can forward authentication request from Azure AD to your internal on-premises domain controllers. You will have more control and potentially between security if you use ADFS, but ADFS requires more work. | 5 | 199 |
| **Microsoft Accounts** | Is a user account in Azure AD for consumer-oriented services, such as Outlook.com and OneDrive. | 5 | 199 |
| **Organizational Accounts** | Is a user account in Azure AD intended for business-oriented services such as Exchange Online and Microsoft 365. It is also referred as a “Work or School Account” | 5 | 199 |
| **Azure D Single Sign-On** | Link your domain or local user account to your Microsoft Account in Azure AD.  Your roaming user profile in OneDrive can sync some of your settings across all your Windows machines in the world. | 5 | 200 |
| **Planetary Roaming User Profile** | When you connect your logged-on user account to your Microsoft Account, you also choose which data, if any should be synced to OneDrive can be downloaded and synced to any other computer where you log on with the same Microsoft Account. | 5 | 200 |
| **Administrative Roles** | An Azure "role" is similar to an administrative global group in on- premises Active Directory. | 5 | 202 |
| **Global Administrator** | The most important is the Global Administrator role, which is the only role that can assign all the other roles and reset the password of any other account. (In some web and PowerShell interfaces, the Global Administrator role is sometimes referred to as the "Company Administrator" role, but it's the same thing.) | 5 | 202 |
| **Endpoint Security - Enforce in Cloud** | * Endpoint devices are the Achilles' Heel of the cloud. * Secure endpoints against malware infection and theft (Defender antivirus). * Remote wipe lost or stolen devices and change passwords. * Use multifactor authentication | 5 | 204 |
| **Endpoint Hardening** | Azure and Microsoft 365 security begins with good endpoint security. Our users’ workstations, laptops, tablets, and phones are more easily hacked than Azure. Our users are far easier to socially engineer than IT admins. | 5 | 204 |
| **Mobile Device Management (MDM) and Microsoft Intune** | Use MDM policies to require a passcode or biometric authentication to unlock mobile devices, require full disk encryption. | 5 | 204 |
| **Remote Wipe** | Mobile devices will inevitable be lost or stolen. Plan for this by requiring whole drive encryption, passcode, or biometrics to unlock the device, and automatic backups or real-time data sync over the internet. Remotely wipe these devices, change the user’s password, issue new user’s certificates, and revoke their old certificates. | 5 | 204 |
| **Microsoft 365 Defender** | Microsoft 365 Defender is a suite of disparate products lumped together under the "Defender" branding | 5 | 205 |
| **Microsoft Defender for Endpoint** | Defender for Endpoint is also for servers. This product aims to reduce OS attack surfaces, block new live attacks, detect attacks in real time, generate alerts detailed enough for incident response, and accelerate the correct response to those alerts. | 5 | 205 |
| **Microsoft 365 Security Center Web Portal** | The portal surfaces alerts and incidents from the various Defender products, provides security recommendations, and calculates an overall Microsoft Secure Score. Having a single portal is important for security because the lifecycle of an incident may span multiple products, machines and users. | 5 | 206 |
| **Microsoft Defender for Cloud App Security** | Cloud App Security acts as a special-purpose HTTPS proxy server than can examine and/or block the HTTPS web requests routed through it. | 5 | 206 |
| **Microsoft Defender for Identity** | Watches over user accounts in Azure Active Directory and on-premises Active Directory to detect outside hacker compromise and malicious insider abuse. | 5 | 206 |
| **Microsoft Defender for Office 365** | Blocks and detects attacks to users through email and collaboration applications such as outlook and Microsoft Teams on the client’s side and exchange Online on the cloud side. | 5 | 206 |
| **Azure Security Center** | Is mainly for resources in Azure while Microsoft 365 Security Center is for most everything else. These 2 portals will likely be merged together someday. | 5 | 207 |
| **Multifactor Authentication - Endpoint Security** | In addition to your password, choose an MFA option for your Android or Apple iOS phone:  Receive a phone call with a recorded message (no app required)  Receive an SMS text message with a PIN (no app required)  PIN in Microsoft Authenticator app (changes every 30 seconds)  Push notification to the Microsoft Authenticator app (yes/no) | 5 | 208 |
| **Phone Enrollment** | The user must go through an enrollment procedure on their phone. This binds the identity of the user to the identity of that particular phone. The user is prompted to enroll the first time they attempt to log onto Azure of Microsoft 365 after the Azure AD administrator has enable multifactor authenticator for that user. | 5 | 208 |
| **Azure Conditional Access** | Since not all data and application in azure are equally important, an administrator can require multifactor authentication when users attempt to run certain apps or only when accessing certain data. | 5 | 209 |
| **Microsoft Intune - Endpoint Security** | Microsoft Intune is a mobile device configuration management system hosted in Azure:  Endpoint Configuration Management:   * Like internet-friendly Group Policy using HTTPS * Supports Windows and non-Windows devices * Requires an Azure Active Directory domain for users, but devices do not have to be joined. | 5 | 210 |
| **Intune Device Compatibility** | Intune Compatible Devices:   * Google Android * Apple iOS and macOS * Windows 10 and later, including Home edition * Both personally-owned and company-owned devices. | 5 | 212 |
| **Windows Device Enrollment** | Windows Autopilot  • All Settings > Accounts > Access work or school > Connect  • Rules define to which Azure AD groups the device is added | 5 | 212 |
| **Azure Conditional Access Policy: (1 of 5)** | 1. Azure AD domain user attempts to access a resource 2. Does the user or user's device match selection criteria defined in the Conditional Access policy to block access? If so, block. 3. Does the user or user's device meet the additional requirements to grant access? If not, then block. 4. Requirements satisfied, grant access. | 5 | 214 |
| **Azure Conditional Access Policy: Selection Criteria (2 of 5)** | Does the user or user's device match selection criteria defined in the Conditional Access policy to block access? If so, block.   * User group memberships or roles in Azure AD * User's device source IP address * Type of user device, such as iPhone or Windows PC * Whether the user's device is registered in Azure AD * Which app the user is using to access the Azure service * Sign-in risk score determined by Microsoft | 5 | 216 |
| **Signals** | User logon or action | 5 | 216 |
| **App Protection Policies (APP)** | Are separate from, but compatible with, Conditional Access and Intune. APP rules are mainly for Microsoft applications such as Word, Excel, and these APP rules will most likely be manage through Intune. | 5 | 217 |
| **Azure Compliant Devices** | Must use Intune or a compatible third-party Mobile Device Management (MDM) system that can mark those device objects Azure AD complaint. | 5 | 217 |
| **Azure Conditional Access Policy: Additional Requirements (3 of 5)** | Does the user or user's device meet the additional requirements to grant access? If not, then block.  These additional requirements, when the action is Grant, include one or more of the following:   * Using multifactor authentication * Using an Intune-managed device that is compliant with Intune's configuration settings Using a Windows computer joined to an on-premises AD domain * Using an approved client app Using an approved client app that enforces App Protection Policies (APP) | 5 | 217 |
| **Azure Conditional Access Policy: Testing (4 of 5)** | Because of the potential for self-inflicted problems, thorough testing Conditional Access policies is essential for success.   * Report-Only Mode * What-If Tool * Lab Testing | 5 | 218 |
| **Lab Testing** | Next, configure the policy to apply to a test user, then actually log on as that real user with multiple real devices in the lab, such as an Android phone, iPad tablet and a Windows laptop. | 5 | 218 |
| **Report-Only Mode** | First, when creating a new policy or editing an existing one, set that policy to "report-only" mode. In report-only mode, a policy will not block access, but the triggering of that policy can be logged for review. This logging is extremely useful for debugging false positives and false negatives. | 5 | 218 |
| **What-If Tool** | Next, use the What-If Tool built into the Azure web portal to simulate the logon and activity of a test user. The What-If Tool can simulate how the user authenticates, such as with or without multifactor authentication, user group memberships, user device characteristics, and more. This is faster than using a real user and a real device from the lab. | 5 | 218 |
| **Azure Conditional Access Policy Example: MFA (5 of 5)** | See the screenshot on the book. | 5 | 219 |
| **Azure Event Hooks** | When Azure detects a change, it will cause any of your Azure policies to be evaluated by Azure and this is called an Azure Event hook or an Azure Kernel Trigger. | 5 | 220 |
| **Azure Kernel Triggers** | When Azure detects a change, it will cause any of your Azure policies to be evaluated by Azure and this is called an Azure Event hook or an Azure Kernel Trigger. | 5 | 220 |
| **Azure Policy (1 of 4)** | Azure Policy is a collection of services to help tenants perform audits at scale, to assist with the discovery and remediation of security vulnerabilities, and to comply with industry and government regulations | 5 | 220 |
| **Azure Policy Initiatives (2 of 4)** | * Over 500 built-in policies. * Collection of policies is called an "initiative" in Azure. * Initiatives for PCI, ISO 27001, HIPAA, FedRAMP and CIS. * Create your own custom policies and initiatives | 5 | 221 |
| **Initiative** | A group of one or more policies is called an "initiative" | 5 | 221 |
| **Azure Policy Assignment and Effects (3 of 4)** | Assign an initiative and choose the action for each policy.   * **DENY**: Block the attempted creation or change * **AUDIT**: Do not block, but log the non-compliance * **MODIFY:** Modify an existing resource property tag * **APPEND:** Add a new resource property tag | 5 | 223 |
| **Tags (Azure)** | A VM could have various “Tags” applied to it for resource tracking purposes. There tags are visible in the Azure web portal and can be used in automation roles for Lifecyle management of the asset. | 5 | 223 |
| **Azure Policy Example: Key Vault (4 of 4)** | Azure Key vault is a server to streamline and secure the entire Lifecyle of secrets, both for VMs in Azure and for applications anywhere in the world that can authenticate to Azure over HTTPS. | 5 | 224 |
| **Hardware Security Module (HSM)** | Support in the Azure datacenter for maximum key protection.  There is an Azure Policy for HSM use | 5 | 224 |
| **Key Vault** | Hardware Security Module (HSM) support in the Azure datacenter for maximum key protection. There is an Azure Policy for HSM use. | 5 | 224 |
| **Azure Policy - Key Vault:**  **Vault Access Control** | * HTTPS access to secrets * Per-vault permissions * Limit by source IP or vNet * Purge protection * Azure Monitor logging * Allow encryption/decryption without exposing the key | 5 | 224 |
| **Purge Protection** | If the contents of a vault are accidentally or deliberately deleted, the loss of the vault’s secrets could cause application failures, user work disruptions and permanent data loss. Enable purge protection on a vault to cause Azure to keep deleted secrets for a period of days that choose. | 5 | 224 |
| **Secret** | A "secret" can be any certificate, key or string of text | 5 | 224 |
| **Vault** | A "vault" is a named container for secrets in Azure. A vault requires authentication and authorization in order to interact with the secrets in that vault | 5 | 224 |
| **Key Vault - Azure Policy** | Here are the names of some of these policies, which are hopefully self-explanatory enough for this overview:   * Key vaults should have purge protection enabled * Firewall should be enabled on Key Vault * Private endpoints should be configured for Key Vault * Resource logs in Key Vault should be enabled * Key Vault keys should have an expiration date * Keys using RSA cryptography should have a specified minimum key size * Certificates should expire within the specified number of days * Keys should be backed by a Hardware Security Module (HSM) | 5 | 225 |
| **Key Vault - Security** | * Chose the geographical region where the secrets will be kept. This is important for regulatory, privacy, and government access reasons. * Some vault permissions regulate overall management of the vault, such as who can create, delete, import, backup and restore the vault as a whole. * Other vault permissions regulate interaction with the secrets in the vault without actually exposing the raw secrets to the user or VM that needs access. * Access to a vault can also be restricted by the caller’s source IUP address. * Access to secrets is via HTTPS * When extra security is desired, some types of secrets (not all) can be stored in a Hardware Security Module (HSM) for a fee. | 5 | 225 |
| **Key Vault - Best Practices** | * Tell developers and admins to use Key Vault! Don't store secrets in unprotected files, database tables or registry keys. * Don't use one, big vault for everything. Use separate vaults for damage control, manageability, auditing, and security. * Use role-based access control to regulate access to vault data. Do not grant more permissions than are needed. * Restrict access to vaults based on the caller's source IP address or Azure virtual network. * Use the managed identities built into Azure services to authenticate to Key Vault instead of authenticating manually. If authentication credentials are stored in unprotected files, tables, or registry keys, then those can be stolen too. This would undermine the purpose of using Key Vault in the first place. * Enable purge protection on your vaults and allow deleted secrets to be recovered for up to 90 days. * Even with purge protection enabled, regularly export and back up your mission-critical vaults (and then secure those backups too). * Enable Key Vault log data to be sent to Azure Monitor. Remember, Azure Monitor feeds into Sentinel and Azure Security Center as well. Be on the lookout for unauthorized access attempts. * Use Azure Policy to audit and enforce proper use of Key Vault. | 5 | 226 |
| **Azure Monitor (1 of 3)** | Azure Monitor is a Microsoft cloud service for collecting, analyzing, visualizing, and monitoring log data.  Azure monitors: Reports, Alerts, Dashboards  Sources: Azure VMs, on premises VMs, azure services, custom apps & scripts, IoT devices. | 5 | 227 |
| **Kusto Query Language (KQL)** | Azure Monitor uses the friendly Kusto Query Language for querying Data. | 5 | 228 |
| **Agent** |  | 5 | 229 |
| **Application Insight** | Is similar in concept to an “agent” but the insight is generally not a service, it’s more likely to be a module or script added to the configuration of the app being monitored. The term “Application Insight” refers to both the agent like thing running as a part of the app as well as the part of the azure portal where this data is analyzed. | 5 | 229 |
| **Azure Monitor: Agents and Workspaces (2 of 3)** | * Each VM requires an agent service to forward log data. * Web applications require a few extra lines of code for gathering tracing information. * All log data sent over HTTPS to Microsoft datacenters. | 5 | 229 |
| **Tracing** | The logging provided by Application Insights is sometimes referred to as "tracing" | 5 | 229 |
| **Distributed Tracing** | "distributed tracing" is gathering tracing data from multiple processes, containers, and virtual machines in a microservices architecture web application. | 5 | 230 |
| **Metrics** | Numerical log data gathered at fixed intervals, such as CPU utilization, are called "metrics". | 5 | 230 |
| **Resource Logs** | Tracing logs are often called resource logs or diagnostic logs | 5 | 230 |
| **Telemetry** | When log data is passively received from remote sources that actively send that data to Azure Monitor, this type of logging is called "telemetry", especially when that data is received over the internet. Hence, roughly speaking, metrics are actively gathered, while telemetry is passively received. | 5 | 230 |
| **Tracing** | Textual log data produced by sporadic events or triggered by user requests at a low level inside an app is called "tracing", and tracing is typically done by Application Insights. (Tracing logs are also sometimes called "resource logs" or "diagnostic logs" in the Azure Monitor documentation.) | 5 | 230 |
| **Workspace: Azure Log analytics** | A workspace is a set of zero or more database tables to hold Azure Monitor log data. You must have at least one workspace, but you may have many. A monitored resource for Azure Monitor, such as group of VMs or a web application, must be configured with a destination workspace. Multiple resources can write to a single workspace. One workspace can hold petabytes of data. | 5 | 230 |
| **Workspace Security** | Each workspace has its own permissions to control who can query, modify, or delete data in that workspace. Workspace tables are always encrypted. You may generate your own encryption keys for encrypting workspace data instead of using Microsoft's keys. Your keys are stored in Azure Key Vault. | 5 | 230 |
| **Azure Monitor Example: Activity Log (3 of 3)** | This slide is mainly intended as an example of the Azure web portal and to show some of the Azure Monitor categories. | 5 | 231 |
| **Azure Sentinel (1 of 4)** | Sentinel is Microsoft's SIEM and SOAR in the Azure cloud, built in part on top of Azure Monitor | 5 | 232 |
| **Logic Apps** | Sentinel's SOAR features rely on Azure Logic Apps. Logic Apps is an Azure service for workflow automation of business processes, such as connecting events in Microsoft 365 with internal accounting systems and third- party procurement web services. This is done with a graphical "playbook" editor to reduce the need for custom programming as much as possible. While Log Apps is a general-purpose orchestration platform, Sentinel uses Logic Apps specifically for security. | 5 | 232 |
| **Playbook** | Graphical editor that reduces the need for custom programming as much as possible. | 5 | 232 |
| **Security Information Event Management (SIEM)** | As a Security Information Event Management (SIEM) system, Sentinel can be connected to a variety of log data sources to consolidate mountains of log data into a searchable database for security purposes. | 5 | 232 |
| **Security Orchestration Automated Response (SOAR)** | As a Security Orchestration Automated Response (SOAR) system, Sentinel assists with the automated scanning, reporting, alerting and incident response reactions to all of that consolidated security log data. | 5 | 232 |
| **Sentinel SIEM** | Sentinel is Microsoft's SIEM and SOAR in the Azure cloud, built in part on top of Azure Monitor: | 5 | 232 |
| **Connectors** | Sentinel uses "connectors" to ingest data from a wide variety of Microsoft and non-Microsoft sources. Here is a list of example connectors, but there are several additional ones beyond these: | 5 | 233 |
| **Sentinel SIEM: Data Connectors (2 of 4)** | Sentinel uses "connectors" to ingest data from a wide variety of Microsoft and non-Microsoft sources. Here is a list of example connectors, but there are several additional ones beyond these: | 5 | 233 |
| **Sentinel SIEM: Data Connectors Examples** | * Amazon web services * Barracuda Firewall * Palo Alto networks * Azure Active directory * Microsoft 365 defender * Microsoft office 365 * Windows firewall * Syslog | 5 | 233 |
| **Incident Playbooks** |  | 5 | 235 |
| **Sentinel SOAR: Incident Playbooks (3 of 4)** | When Sentinel detects a potential security incident, it creates an alert. One of your Sentinel administrators will assign an owner to each alert, who is then responsible for handling the overall incident to which the alert is related.  **Handler can drill down into the details of the incident by hand as necessary:**   * Users & Groups * VMs & IP addresses * Files & Registry Keys * Processes & Ports   **Handler can run playbooks to automate many of the steps of the response:**   * Create a ticket * Send Slack Message * Disable Account * Reboot VM | 5 | 235 |
| **Sentinel Example: Overview (4 of 4)** | This slide is mainly intended as an example of the Sentinel web portal. | 5 | 236 |
| **Azure Security Center: The Top of the Pyramid (1 of 2)** | * Provides a "single pane of glass" for an organization-wide Secure Score. * Azure Monitor and Azure Policy security alerts feed up to the top here. * Over 150 security recommendations to improve your Secure Score. * Built in vulnerability scanning and an Azure Policy security initiative. * Provides a framework for third-party security products too | 5 | 237 |
| **Security Center** | Security Center includes an overall Secure Score, like a report card, and the ability to graph that score over time. Executives will love this. | 5 | 237 |
| **Auto-Provisioning** | A nice automation feature of Security Center is called "auto-provisioning".  Auto-provisioning is the automatic installation of one or more agents, such as for Azure Monitor, upon which Security Center depends; for example, if there isn't an AppLocker policy configured on a Windows VM, auto-provisioning will install the necessary AppLocker policy on that VM to start logging operating system activity at a low level. | 5 | 238 |
| **Azure Security Center: Web Portal (2 of 2)** | This slide is mainly intended as an example of the Security Center web portal. | 5 | 239 |
| **Automation, Logging, and Auditing** |  | 5 | 241 |
| **Automation** | 95% of what can be done with graphical tools can be done from the command line with your own scripts. There will be many security and auditing tasks that require command line tools and your scripting skills. | 5 | 244 |
| **PowerShell** | * Command shell * Easy-to-learn scripting language * Replaces the old CMD.EXE * Installed on Windows 7 and later systems by default * For Windows, Linux and macOS * Popular for Azure automation | 5 | 245 |
| **Windows PowerShell** | * Command shell * Easy-to-learn scripting language * Replaces the old CMD.EXE * Installed on Windows 7 and later systems by default * For Windows, Linux and macOS * Popular for Azure automation | 5 | 245 |
| **Just Enough Admin (JEA)** | An optional feature of PowerShell remoting is something called "Just Enough Admin (JEA)". Once configured on a target host, when a user remotes into that host, the JEA feature will c 1) block all commands by default and 2) only allow the commands that are explicitly permitted. This is similar to application control but for PowerShell commands instead. JEA can even limit the allowed command line arguments to a command by blocking all arguments that do not match a regular expression pattern defined by the administrator of the target box. JEA requires at least Windows PowerShell 5.0. | 5 | 246 |
| **Transcription Logging** | It is also possible to log in great detail what commands are executed in PowerShell. This transcription logging is compatible with remoting and JEA but does not require either one. Transcription logging even captures commands run entirely in memory or when the commands are executed from obfuscated scripts. The transcription logs are text files, not Event Viewer logs. Transcription logs can be fed into a SIEM for analysis and alerting, but this is not the default. | 5 | 246 |
| **PowerShell Core vs Windows PowerShell** | PowerShell Core runs on top of the .NET Core Framework. Windows PowerShell runs on top of the Full .NET Framework. PowerShell Core is cross-platform because it runs on Windows, Linux, and macOS, but Windows PowerShell runs only on Microsoft Windows. PowerShell Core is an open-source project in GitHub, but Windows PowerShell is closed source and comes pre-installed on Microsoft Windows alone. | 5 | 247 |
| **PowerShell Examples** | Commands | 5 | 249 |
| **PowerShell.exe** | get-process, get-process | export-csv outputfile.csv, get-service, get-help invoke-command -full, get-help \*net\* | 5 | 249 |
| **Windows PowerShell Examples** | Commands | 5 | 249 |
| **How to Change Your PATH** | On most versions of Windows, you change your PATH environment variable by opening Control Panel > System applet > Advanced System Settings link (or Advanced tab) > Environment Variables button > select the PATH system variable at the bottom > Edit > append the full path(s) to your script or program folder(s), separating each full path with a semicolon. When you finish, close PowerShell or CMD.EXE, then relaunch it. | 5 | 251 |
| **Microsoft Command Line Tools and Documentation** |  | 5 | 251 |
| **.EXE Tools** | CMD Line, Windows | 5 | 252 |
| **Command Line Tools - Microsoft** | Very extensive list. Please refer to the book. | 5 | 252  253  254 255 256 |
| **WMIC.EXE** | Show operating system version and edition: wmic.exe os get caption. Show shared folders on a remote server: wmic.exe /node:Server52 share list brief | 5 | 257 |
| **Command Line Tools - Network Configuration** | Virtually every setting related to networking can be queried and/or reconfigured through one of the following command line tools:   * WMIC.EXE * NETSH.EXE * GETMAC.EXE * IPCONFIG.EXE * ROUTE.EXE * NET.EXE * NETSTAT.EXE * NBTSTAT.EXE * NETSH.EXE   was modeled on the Cisco command line interface in both purpose and feel. | 5 | 258 |
| **Network Configuration Tools** | Other than PowerShell, these are your best command line tools for managing networking:   * WMIC.EXE * NETSH.EXE * GETMAC.EXE * IPCONFIG.EXE * ROUTE.EXE * NET.EXE * NETSTAT.EXE * NBTSTAT.EXE * NETSH.EXE | 5 | 258 |
| **Windows PowerShell Replacements** |  | 5 | 259 |
| **Windows Subsystem for Linux (WSL)** | Run Linux executables and scripts directly on Windows without a virtual machine or emulator. WSL is not a virtual machine, container, or emulator. WSL implements many low-level Linux Application Programming Interface (API) functions on Windows. WSL includes filesystem drivers to allow Linux tools to directly access the physical drives on the computer. Windows and WSL tools can access the same NTFS folders and files simultaneously on the physical drive. On Windows 11 and later, even graphical Linux apps can be run! | 5 | 260 |
| **Windows Subsystem for Linux (WSL) - Commands** |  | 5 | 261 |
| **Scheduling Tasks** | Through command line tools and custom scripts, you can automate your work. What you need now is a way to run these tools/scripts automatically on a recurring basis. Enter the Task Scheduler. | 5 | 263 |
| **Task Scheduler** | Through command line tools and custom scripts, you can automate your work. What you need now is a way to run these tools/scripts automatically on a recurring basis. Enter the Task Scheduler. | 5 | 263 |
| **Task Scheduler - Command Line Management** | You also can manage scheduled tasks from the command line on local or remote systems with PowerShell or the SCHTASKS.EXE tool. The old AT.EXE utility is obsolete and should not be used. | 5 | 263 |
| **Task Scheduler - Scheduled Task Accounts** | Keep in mind that if you change the password for an account used to run scheduled tasks, the scheduled tasks themselves are not updated with the new password. This must be done manually in the GUI, through a script, or with a third-party tool designed for this. If you need to schedule tasks running with administrative privileges, consider creating a special account for this purpose and assigning it a long passphrase, perhaps 50+ characters. No one is logging in to his desktop with this account, so it's not inconvenient to use a long passphrase. Make sure to audit all access to this account. | 5 | 264 |
| **Azure Tools** | * PowerShell A module * Azure CLI * Azure Cloud Shell * Azure Resource Manager Templates | 5 | 265 |
| **Azure Web Portal** | The main graphical tool is the Azure web portal for your browser: | 5 | 265 |
| **PowerShell Az Module** | PowerShell Az Module can be used for scripting and automation. It includes over 3000 commands | 5 | 265 |
| **Windows Azure Tools** | * PowerShell A module * Azure CLI * Azure Cloud Shell * Azure Resource Manager Templates | 5 | 265 |
| **Windows Defender Firewall: Good and Bad** |  | 5 | 265 |
| **Azure CLI** | Azure CLI is a more traditional text-oriented tool for Azure scripting (Commands) | 5 | 266 |
| **Azure Cloud Shell** | You don't want to use a command shell on your local computer? No problem, use Azure Cloud Shell! | 5 | 267 |
| **Azure Resource Manager (ARM) Templates** | An Azure Resource Manager template (ARM template) is a text file that defines what end result you want created or configured in Azure. An ARM template is not a script; rather, your own scripts can deploy ARM templates | 5 | 267 |
| **Azure Cloud Shell** | Choose PowerShell or Bash, Az is the Azure CLI tool, Buttons to upload and download files, or to open a script editor inside the Cloud Shel | 5 | 268 |
| **Az** | PowerShell module | 5 | 269 |
| **az** | Command line tool that has no dependencies on PowerShell | 5 | 269 |
| **Azure Automation (1 of 2)** | With Azure Automation, you can run PowerShell or Python scripts on Windows or Linux machines anywhere in the world as long as those machines have HTTPS access to Azure: | 5 | 270 |
| **Azure Sandbox Worker jobs** | Run PowerShell or Python scripts in a container in Azure to manage your Azure resources. | 5 | 270 |
| **Azure Hybrid Worker Jobs** | Run PowerShell or Python scripts inside Windows or Linux VMs, anywhere in the world, with jobs downloaded over HTTPS | 5 | 270 |
| **Job** | A "job" is a running instance of a runbook. There can be many simultaneous jobs from one runbook. | 5 | 270 |
| **Runbook** | A PowerShell or Python script executed by Azure Automation is called a "runbook". The script can be created with a source code editor, such as Visual Studio Code, or with a graphical designer application in the Azure web portal using one's browser. The graphical designer generates the necessary code and makes it possible for those who dislike coding to still create (simple) runbooks. | 5 | 270 |
| **Sandbox Worker Jobs** | Run PowerShell or Python scripts in a container in Azure to manage your Azure resources | 5 | 270 |
| **Webhook** | Scripts can be executed just once, on a recurring schedule, in response to Azure Logic Apps events, in response to Azure Monitor events, triggered with a simple HTTPS POST request from any tool or source (this is called a "webhook"), or manually by any user with the Automation Operator role in your Azure subscription. One runbook can call other runbooks, hence, each runbook can be treated like a reusable module or building block for other runbooks. Over time, you will develop your own library of runbooks customized for your organization. | 5 | 270 |
| **Worker** | One or more jobs will be executed by a "worker". There are two types of workers. A worker can be an Azure Sandbox Worker or a Hybrid Runbook Worker | 5 | 270 |
| **Hybrid Runbook Worker** | The Hybrid Runbook Worker is for scripts that will need access to the filesystem, registry, services, drivers, or other items inside of your VMs. Your VMs can be in Azure, on premises, or on non-Azure clouds. This type of worker is also required when your script will need to run for more than 3 hours, when the script requires OS- level privilege escalation, when the script launches other EXE programs, when the script queries the WMI service, when the script requires more than 400 MB of private working memory, or when the script requires more than 1 GB of disk space while running. In short, use a Hybrid Runbook Worker for everything other than 100% pure Azure resource management | 5 | 271 |
| **Microsoft Flow** | Power Automate (previously known as Microsoft flow) is built on top of Azure Logic Apps. Power Automate is for regulate users (not developers or IT people) who prefer a graphical design application (not source code editor) | 5 | 271 |
| **Nxautomation** | On Windows, Hybrid Runbook Worker jobs run as the local System identity in a service. On Linux, these jobs run as a special sudo-elevated account named "nxautomation" that was created when the Hybrid Runbook Worker agent was installed. On both platforms, a job can be given "run as" credentials when that job needs to authenticate to other machines or services. These credentials, as well as the private key of the Automation account and other secrets, are encrypted with Microsoft-managed keys and stored in Azure Key Vault | 5 | 271 |
| **Power Automate** | Power Automate (previously known as "Microsoft Flow") is built on top of Azure Logic Apps. Power Automate is for regular users (not developers or IT people) who prefer a graphical design application (not source code editor) to create relatively simple automation workflows, such as for SharePoint and OneDrive. Power Automate is easier to use than Azure Automation, is mainly for interacting with resources only in Azure, and is aimed at end users. | 5 | 271 |
| **Azure Functions** | Azure Functions is a service for developers to have their custom code executed without creating or maintaining any virtual machines or containers themselves. Developers upload the source code of their functions and define when those functions are run, then Microsoft handles everything else. This is for "serverless" and "microservices" web applications. There are servers and containers involved in this, of course, it's just that Microsoft handles them invisibility for the developers. Azure Functions is built on top of Azure App Service. While IT people can use Azure Functions too, it is mainly aimed at developers. | 5 | 272 |
| **Azure Automation: Graphical Runbooks (2 of 2)** | Built into the Azure web portal is a graphical runbook designer. The designer allows novice coders to get started without writing PowerShell scripts from scratch | 5 | 273 |
| **Graphical Runbooks** | Writing complex Python or PowerShell scripts is not everyone's cup of tea. When you create a runbook, you may choose between a traditional Python or PowerShell runbook (source code editing required) or a graphical runbook (source code available but hidden by default). Graphical runbooks only use PowerShell. | 5 | 273 |
| **Configure Windows Logging** | All Windows systems have at least three Event Logs: Application, Security, and System. Administrators can view the contents of these logs with the Event Viewer app in Administrative Tools > Event Viewer. | 5 | 274 |
| **Event Logs** | All Windows systems have at least three Event Logs: Application, Security, and System. | 5 | 274 |
| **Event Viewer** | In PowerShell, run "Show-EventLog" to launch the Event Viewer. | 5 | 274 |
| **Security Event Log and Audit Policies** | Manage audit policies with Group Policy and AUDITPOL.EXE | 5 | 276 |
| **AUDITPOL.EXE** | you'd rather manage audit policies from the command line, use the AUDITPOL.EXE tool. It's also built into Windows Vista and later by default. Starting with Vista, by the way, you can enable or disable special subcategories of audit policies using AUDITPOL.EXE too. | 5 | 277 |
| **Group Policy** | Almost all audit policy settings on both servers and workstations can be remotely configured through Group Policy. Audit policy is set under Computer Configuration > Windows Settings > Security Settings > Local Policies > Audit Policy. Hence, when you need to configure audit settings throughout your domain, Group Policy is the way to do it. | 5 | 277 |
| **Event Log ID Numbers** | EventID 5156 4688 4720 4732 4724 4624 4625 4740 1102 | 5 | 278 |
| **Security Log Events** | EventID & Descriptions | 5 | 278 |
| **Audit Object Access** | Even if you do enable Audit Object Access for both success and failure, nothing extra is logged. | 5 | 280 |
| **Enable Object Auditing** | 1. Enable the Audit Object Access policy  2. Configure the object SACLs wanted | 5 | 280 |
| **NTFS and Registry SACLs** | Auditing access to NTFS files/folders, registry keys, and printers is a two-step process. | 5 | 280 |
| **System Access Control List (SACL)** | An object's SACL defines exactly which users and groups should have their interaction with the object logged. Moreover, you can choose exactly what types of interaction will be logged  SACLs can be inherited from parents’ folders, just like permissions (DACLs)  Different SACLs for different types of objects, like files and registry key. | 5 | 280 |
| **Log Size and Wrapping Options** | A security template also can be used to set the maximum size and wrapping options for your Event Logs.  Each log can be sized and moved separately from the other logs.  Appropriate log size will be determined by the rate of new events and your wrapping options (maximum size = gigantic). | 5 | 283 |
| **Maximum Log Size** |  | 5 | 283 |
| **Flushing the Logs** | Overwriting events as needed is the most convenient, but the danger is that hackers or malware will deliberately fill the logs with useless data in order to cover their tracks. This is called "flushing the logs" and is often preceded by the attacker or malware disabling all audit policies first. | 5 | 284 |
| **Log Wrapping Options** | When a log file fills to its maximum capacity, the wrapping options engage for that log. There are three wrapping options:   * Overwrite events as needed (oldest events first) * Archive the log when full (do not overwrite events) * Do not overwrite events (clear log manually) | 5 | 284 |
| **Logging - What Should Be Logged?** | The more you log, the slower your machine's performance, so avoid flooding logs with useless data that no one will ever examine or need.  Try to anticipate how an attacker might leave a trail of clues, then audit to collect that data. | 5 | 285 |
| **Security Templates (Again)** | These security templates can be applied, of course, with the Security Configuration and Analysis snap-in, the SECEDIT.EXE command line tool, or Group Policy. | 5 | 286 |
| **File & Printing Sharing (SMB-In): Tab properties** | * General * Programs and services * Users and Computes * Protocols and Ports * Scope * Advanced | 5 | 163 164 |
| **Security Event Log: Types** | * Audit Account Logon Events (Success, Failure): On domain controllers this tracks authentication request processed by the domain controllers. * Audit Account Management (Success, Failure): this monitors user and group tasks, such as account creating, etc. * Audit Directory Services Access (Success, Failure): On domain controllers, this is requiring logging access to Active Directory objects as defined by those objects’ individual audit settings. * Audit Logon Events (Success, Failure): this tracks interactive and over-the-network logons to the computer itself. * Audit Object Access (Failure) this is required to begin logging access to NTFS folders and files, registry keys, and share printers. * Audit Policy Change (Success, Failure) tracks changes to the audit policies themselves and changes to privileges assignments. * Audit Privilege Use (Failure): Monitors the exercise of certain privileges on the machine. * Audit Process Tracking (Not defined): this is rarely enabled and usually only by programmers who are debugging their own code. * Audit System Events (Success, Failure): Tracks system startup, shutdown, and other system-wide events. | 5 | 276 277 |
| **Operating System Market Share** | The key takeaway is that even though Windows typically dominates from an install base, Linux systems are typically the most mission critical systems within an environment. It is also important to note that most security appliances that are installed in a data center are running Linux. | 6 | 5 |
| **Linux Systems - Criticality of** | It is, therefore, important to identify the critical data, map it to critical processes, and determine what systems they reside on. The question is, "Which operating system is running on those systems?" In many cases, a significant percentage of them are probably Linux systems. In my experience, there are typically a lot fewer Linux systems, but they are often the most critical systems.  What percentage of those systems are Linux-based?   * Make a list of the most critical data to your organization. * Determine the most critical processes that access that data. * Map back to what servers that data resides on (hint: consider some of the appliances) | 6 | 6 |
| **Linux vs. Windows** | * The install base of Windows is higher; however, Linux is usually installed on the mission critically systems, network infrastructure, and security components (Firewalls, IDS, Application proxies). * For security analysis, assessment, and penetration testing, many security teams use Linux in addition to other operating systems. * If you sort by price, the most expensive systems are often UNIX -or Linux-based. * Most of the cloud services are sunning Linux on the back-end, even Microsoft Azure is based on a UNIX environment. | 6 | 7 |
| **UNIX** | UNIX was conceived in 1969 as an OS that was going to run on a server with many users all doing different things at the same time | 6 | 8 |
| **Windows and Linux: Desktops and Servers** | 1. Windows server:   DESKTOP > SERVER  Old: Single-user platform  New: Multiple Processes for multiple users at the same time.   1. SERVER>DESKTOP   Old: installed on servers with many simultaneous users.  New: Used as a desktop OS. | 6 | 8 |
| **Advanced Packaging Tool (APT)** | Debian is probably best known for its package management system, the Advanced Packaging Tool (APT). | 6 | 10 |
| **Debian** | also known as **Debian GNU/Linux**, is a Linux distribution composed of free and open-source software by the community-supported Debian Project. its first stable version (1.1) was released on June 17, 1996. The Debian Stable branch is the most popular edition for personal computers and servers. Debian is also most notably for Ubuntu | 6 | 10 |
| **Linux - Types of Operating Systems** | Main Linux Distributions: Ubuntu (Debian), Fedora (RedHat) | 6 | 10 |
| **Ubuntu** | * Linux distribution * Based on revered Debian distribution (Debian was created in 1993) * “Humanity to others” in African. * Focus on education, open licensing, multimedia work, or specific desktop environments. * Stable environment, and use of apt- installation packages. | 6 | 10 |
| **Fedora** | * Linux distribution based on Red Hat. * 5 different editions currently available:   Workstation: personal use.  Server: for servers in a datacenter  CoreOS Focused on cloud: minimal Fedora image, provides the bare essentials.  Silverblueu: Created for container-based workflows and IoT devices. | 6 | 11 |
| **RedHat** | Linux distribution. | 6 | 10 |
| **Security-Enhanced Linux (SEL)** |  | 6 | 11 |
| **Linux** | * Open-source operating system with many variants. * Originally developed for personal computers but has been ported to other platforms. * Many security tools are freely available and often only run-on Linux. * Runs the Linux kernel as the "brains" of the operating system | 6 | 12 |
| **The GNU Project** | The GNU Project was already up and running as a repository and a distributing point for such ideas. It was at this time that Torvalds' idea—called Linux—took root and was made available to anyone who was interested. | 6 | 12 |
| **Windows Subsystem for Linux (WSL)** | * Windows Subsystem for Linux (WSL) is a Windows 10 feature that provides a full Linux kernel allowing users to use native Linux tools. * Optimized Linux Kernel in Windows 10 * Runs inside lightweight VM * Uses native Linux tools from a shell prompt. * Access to windows file system * Linux Distributions installed through the Microsoft App Store. * Multiple Linux Distributions available. | 6 | 13 |
| **Baron Samedit** | * Discovered in January 2021 by Qualys research team * Gain root access without a password * Has been hiding for nearly 10 years | 6 | 14 |
| **Linux Vulnerabilities (1)** | Shellshock, SambaCry, Baron Samedit | 6 | 14 |
| **SambaCry** | * Discovered in 2017 and considered Linux's EternalBlue/WannaCry * Arbitrary code can be executed remotely by abusing writeable samba shares | 6 | 14 |
| **Shellshock** | * Discovered in 2014 * Unintentional handling of environmental variables leads to privilege escalation * Can be triggered remotely | 6 | 14 |
| **Linux Vulnerabilities (2)** | Sudo Bypass | 6 | 15 |
| **sudo -u#-1 id -u** | * **You change the user ID to -1** * System Calls setresuid(2) and setreuid(2), these functions will be called before running the command. Used by Sudo to change the user ID. * These system calls will return 0. This is because the sudo command is already running as user ID 0. | 6 | 15 |
| **Sudo Bypass** | * CVE-2019-14287 * Discovered in October 2019 * Sudo supports running a command with a user-specified username or user ID, if permitted by the sudoers policy * Users can run any command as root by using sudo and providing -1 as user ID * Exploiting the bug requires that the user have Sudo privileges that allow them to run commands with an arbitrary user ID * The bug is fixed in Sudo 1.8.28. * More details about Sudo and root later in this course | 6 | 15 |
| **BSD** | Berkeley Software Distribution, discontinued OS based on research UNIX. | 6 | 16 |
| **macOS (BSD)** | * Second most popular OS. * Uses BSD as its underlying system. * Built on top of Apple's XNU kernel. * Designed to have security built in. * Many features cannot be disabled. * Most network services disabled by default | 6 | 16 |
| **OS best for security** | * All three: Windows, Linux and macOS * To be proficient in security, it is recommended that you have experience with windows and Linux systems. * Most of the popular security distros are all based on Linux. * Many security tools are only available on Linux. | 6 | 17 |
| **Computer Hardware** | A collection of physical components such as the actual CPU and memory chips | 6 | 18 |
| **Kernel** | **“Brains”**   * The memory-resident part of an operating system that directly interfaces with the hardware is called the kernel * Computer program at the core of a computer's operating system * Complete control over everything in the system | 6 | 18 |
| **Operating System Overview** | The three key elements that make up Linux are:   * **Kernel:** The core component of the operating system that is often referred to as the "brains" of the operating system * **Userland:** The portion of the operating system with which users and the process interact directly * **Hardware:** A collection of physical components such as the actual CPU and memory chips | 6 | 18 |
| **Userland** | The portion of the operating system with which users and the process interact directly | 6 | 18 |
| **Kernel - What is a Kernel?** | The kernel of any operating system is generally loaded into memory at boot up. This allows faster interaction between all components, and yet it must have a dedicated space in memory to reside without shifting. This is protected space in memory and users cannot directly interact with it | 6 | 19 |
| **Kernel services:** | * Filesystem * Low-level network protocol support (for example, IP) * Memory and process management. | 6 | 19 |
| **Console** | physical terminal | 6 | 20 |
| **Shell** | * Shell is the command line interpreter used to run programs on the computer. * It provides the user with an interface to the system: * Listen to the terminal * Translate requests into action by the kernel and programs | 6 | 20 |
| **Terminal** | Terminal: text input/output environment | 6 | 20 |
| **Shells - Examples of** | UNIX:   * Bourne Shell (sh) * C Shell (csh) * Bourne-Again Shell (bash) * Korn Shell (ksh) * exTended C Shell (tcsh) * Zshell (zsh).   DOS:   * Command.com.   Windows:   * Powershell.exe | 6 | 21 |
| **/bin** | Executable programs; some SUID/SGID | 6 | 22 |
| **/dev** | The /dev directory contains the special device files that programs running on the system use to communicate with the physical hardware devices controlled by the operating system kernel. UNIX systems derived from the AT&T System V (SYSV) UNIX standard, such as Solaris and HP-UX, often put these device files into the /devices directory, but also usually maintain a /dev directory for compatibility with other systems. | 6 | 22 |
| **/home** | User home directories | 6 | 22 |
| **/usr** | usr is the place where most of the critical components of the operating system live, including system binaries, programming libraries and tools, and online documentation. You can think of this directory structure as being read-only after the operating system is loaded; not many changes under /usr should occur unless the operating system is upgraded, or patches are installed. /var is the place where the system keeps frequently changing data, such as log files and temporary queues for system services such as email and printing | 6 | 22 |
| **/var** | Contains log files, queues, and so on | 6 | 22 |
| **Logical Filesystem** | The top of the filesystem tree is the root directory, /. Below this directory are various important subdirectory trees. | 6 | 22 |
| **Root Filesystem** | The top of the filesystem tree is the root directory, /. Below this directory are various important subdirectory trees. | 6 | 22 |
| **Root filesystem: Top of directory hierarchy** | **/usr:** Primary OS directory; read-only  **/home:** user home directories  **/export/home:** user home directories  **/dev:** Directory containing files used to talk to system devices  **/devices:** Directory containing files used to talk to system devices.  **/var:** Contains log files, queues, and so on.  **/bin:** Executable programs: some SUID/SGID  **/usr/bin:** Executable programs: some SUID/SGID  **/user/local:** Executable programs: some SUID/SGID  **/opt:** Executable programs: some SUID/SGID | 6 | 22 |
| **/etc/fstab** | Disk partitions, their mount points, and the options applied to each partition generally are described in a file called /etc/fstab. Solaris uses the name /etc/vfstab apparently just to be perverse. These filesystem options in the /etc/fstab file enable the administrator to specify different security settings for various partitions. | 6 | 23 |
| **Overlap Partition** | One slice generally is reserved as the "overlap" partition, which represents the entire disk geometry. By convention, this is usually the third slice of the disk because early UNIX systems required the first slice to be the root filesystem and the second slice to be the swap space for the virtual memory system. This "overlap" slice should not be used for filesystems. Note that not all possible disk partitions need to be used. In fact, some administrators just create a single partition that spans an entire disk and ignores the other slices. It turns out that this is a bad idea for several reasons. | 6 | 23 |
| **Partitions** | Although the UNIX filesystem appears to be a single logical entity to the users on the system, it actually is made up of several pieces (called partitions) that correspond to physical sections of the machine's disk drives. | 6 | 23 |
| **Physical Filesystem** | Logical Linux filesystem is made up of multiple physical disk partitions. Disk partitions are mounted at various points in the filesystem. Different security options can be set on each mount point. This is a key component of Linux: Security can be integrated into the installation process | 6 | 23 |
| **Slices** | From a UNIX perspective, disk drives have 16 different partitions called "slices", either numbered 0–15 or lettered a–p. For example, Linux would use the device named hda7 to specify slice 7 (which technically is the eighth slice because the numbering starts at 0 of the first disk a hard disk hd] on the system). Note that older UNIX systems such as Solaris allow only 8 physical partitions per drive | 6 | 23 |
| **Filesystem Security Goals** | * Protecting the /user filesystem is important. * The administrator also should attempt to stop people from creating or bringing unauthorized set-UID and set-GID programs to the machine. * System administrators need to be able to apply patches to the OS and update software that has been installed on the machine. * Avoid denial-of-services (DoS) attacks by partition the system carefully. | 6 | 24 |
| **Rootkit** | An attacker who compromises a system likely wants to install a rootkit, a set of binaries that gives the attacker a backdoor into the system and helps him or her escape detection by the system administrator. Typically, the binaries that the attacker replaces are OS programs in the /usr/bin and /usr/sbin directories, so protecting the /usr filesystem is important. | 6 | 24 |
| **Disk Monitoring (df)** | Administrators can display the currently mounted partitions with the df command. df stands for disk free because the command actually was created to help administrators find partitions that were running out of disk space. | 6 | 25 |
| **Raw Disk** | Note that df shows only currently mounted filesystems. Typically, the machine also has a partition devoted to "swap space" for the virtual memory system. This is a "raw disk" partition that is not mounted into the logical UNIX filesystem. However, this swap partition has an entry in the /etc/fstab file. Information about swap partitions currently in use can also be obtained using the swapon -s command. | 6 | 25 |
| **Swap Space** | Note that df shows only currently mounted filesystems. Typically, the machine also has a partition devoted to "swap space" for the virtual memory system. This is a "raw disk" partition that is not mounted into the logical UNIX filesystem. However, this swap partition has an entry in the /etc/fstab file. Information about swap partitions currently in use can also be obtained using the swapon -s command. | 6 | 25 |
| **Filesystem Security Options** | * **ro:** Filesystem is mounted read- only (files and directories cannot be modified) * **nosuid:** SUID/SGID bits are ignored on all programs in the filesystem. * **nodev:** Special device files don’t work. | 6 | 26 |
| **nodev:** | Similarly, nodev means that special "device files" are ignored in the filesystem. These are the kinds of files usually found in /dev and /devices and are used to communicate with the system hardware devices. Device files appearing in other directories usually are a problem, the only exception being special filesystems used by anonymous FTP servers and the like | 6 | 26 |
| **nosuid:** | The nosuid option means that the operating system simply ignores the set-UID and set-GID bits on executables in the filesystem. So, if a user attempts to execute a set-UID program out of a nosuid filesystem, the program runs as the user and not as the owner of the program. | 6 | 26 |
| **ro:** | The ro (read-only) option causes the UNIX operating system kernel to prevent writes or updates to the given filesystem. When a filesystem is in read-only mode, nobody can update files or directories in that partition, add new files, or delete files. | 6 | 26 |
| **SGID** | SGID: Temporarily grant user file group permissions | 6 | 26 |
| **SUID** | SUID: File executes with privileges of file's owner | 6 | 26 |
| **Filesystem - Goal of Security** | * Filesystems should either be mounted nosuid or read-only. * /usr and /usr/local contain SUID/SGID programs but can be read-only. * Most other filesystems must be writeable but have no SUID/SGID programs * /filesystem contains /dev, but all other filesystems can be mounted nodev. | 6 | 27 |
| **Cryptsetup** | Cryptsetup is the utility used to conveniently set up disk encryption based on the dm-crypt kernel module. These include plain dm-crypt volumes, LUKS volumes, loop-AES, and TrueCrypt format. | 6 | 28 |
| **Hardening Guides - CIS** | * CIS has benchmarks for several distributions: Ubuntu CentOS and Red Hat * Provides details on how to properly secure a system * Scoring tools (CIS-CAT) are available; they can also automatically harden your system | 6 | 28 |
| **LUKS:**  **Linux Unified Key Setup** | * Is the standard for Linux hard disk encryption * Compatibility via standardization * Secure against low entropy attacks * Support Multiple Keys * Effective passphrase revocation * Free | 6 | 28 |
| **SGID - find files with permissions** | Older distros: # find / -perm +2000 Newer distros: # find / -perm /2000 | 6 | 38 |
| **File Attributes** | The easiest way to view the attributes for a file is with the ls –l command, which gives a detailed file listing: | 6 | 29 |
| **ls -l** | The easiest way to view the attributes for a file is with the ls –l command, which gives a detailed file listing: | 6 | 29 |
| **ls -ld** | The -d option forces the ls command to display the file attributes for the actual directories listed—in this case, /etc.—rather than showing the attributes for all the files in that directory, which is the default. | 6 | 29 |
| **Permissions** | See screenshot in the book. | 6 | 29 |
| **Permissions: Absolute** | Represented by numbers (4, 2, 1):  4: Read access  2: Write access  1: Execute access  0: No permission | 6 | 30 |
| **chmod** | change permission on a Linux system, you would use the chmod command | 6 | 30 |
| **File Permissions Overview - Linux** | 1. Symbolic: Represented by letters (RWX):   R: read access  W: Write access  X: Execute, run access  - : No Permission.   1. Absolute: Represented by numbers (4, 2, 1):   4: Read access  2: Write access  1: Execute access  0: No permission. | 6 | 30 |
| **Permissions Symbolic** | Represented by letters (RWX):  R: read access  W: Write access  X: Execute, run access  - : No Permission | 6 | 30 |
| **File Permissions - UNIX** | Read, write, and execute permissions can be set for three different categories of people: the file's owner, people belonging to the UNIX group listed as the group owner of the file, and everybody else (or other) | 6 | 31 |
| **Permission Bits - Other** | setuid, setgid, sticky (s, s, t). Because there is no room display these extra three bits in the ls -l output, the ls command shows these settings replacing the Xs in the normal output. | 6 | 32 |
| **set-GID** | Set Group ID: run program as group | 6 | 32 |
| **set-UID** | Set User ID: run program as owner | 6 | 32 |
| **Sticky Bit** | Sticky Bit: Only the owner can delete file in directory | 6 | 32 |
| **Absolute Mode - Other Bits in** | A fourth octal digit representing these values can be put in front of the three octal digits used to represent the read/write/execute permissions for the various ownership categories. | 6 | 33 |
| **Directory - Permissions** | Read: Can get directory listing,  Write: Can create/remove files,  Execute: May cd to the directory and access its files  Setuid: N/A  Setgid: Group ownership of new files is inherited.  Sticky: Only owner may remove or rename files. | 6 | 34 |
| **File - Permissions** | Read: Can read file contents,  Write: Can modify file contents,  Execute: May execute file  Setuid: Files executes with privileges of file’s owner.  Setgid: As above but for group owner  Sticky: N/A | 6 | 34 |
| **Files versus Directories** | Because the UNIX file permissions model is so limited, the meaning of each of the permission flags we have seen so far is "overloaded". In other words, the permission flags have different behaviors, depending on whether or not they are being applied to files or directories. | 6 | 34 |
| **World-Writable Directories** | * /tmp, used by programs to hold intermediate result * What if an attacker clobbers your program's temporary file and substitutes his or her own?   Golden Rules:   1. Avoid world-writable directories if possible   2. Always set the sticky bit for world-writable directories | 6 | 36 |
| **SUID - find files with permissions** | Older distros: # find / -perm +4000 Newer distros: # find / -perm /4000 | 6 | 37 |
| **SUID/SGID Programs** | SUID/GUID - the double-edged sword:  1. You can't run UNIX without SUID/SGID programs  2. However, rogue SUID/SGID programs can easily compromise a machine.  Golden Rules:  1. Keep track of the SUID/SGID programs provided with your operating system.  2. Raise an alarm if new or unexpected SUID/SGID programs appear. | 6 | 37 |
| **Permissions: Umask and Chmod** |  | 6 | 39 |
| **Umask** | * Umask sets default permissions for newly created files. * Umask reads existing Umask setting. * Umask: u-x,g=r,o-w , umask 132 | 6 | 39 |
| **chmod (change mode) command** | The chmod command, which stands for "change mode", is used to define the way a file can be accessed by modifying the permissions of files. The chmod command has options permissions and then the filename of the file to change the permissions. Chmod can also be written in symbolic and absolute mode.  Two examples follow:   * chmod u=rwx,g=rx,o=r filename * chmod 754 filename   Chmod has an option for recursion, –R, which is useful to change the permissions of many files in a directory tree with one command | 6 | 40 |
| **chgrp** | chgrp changes group ownership | 6 | 41 |
| **chown** | Chown changes file ownership (Only UID 0 can use chown). Both user and group ownership can be changed at the same time with chown. Chown will use both user and group IDs or their names | 6 | 41 |
| **Permissions: chown and chgrp** | chown user1 /home/user1 , chgrp support /home/user1 | 6 | 41 |
| **/etc/passwd** | The /etc/passwd file stores the username, UID, and other various information such as path and default group | 6 | 42 |
| **/etc/shadow** | The /etc/shadow is the password database | 6 | 42 |
| **User Accounts and Groups - UNIX** | * Usernames and passwords are case sensitive * User and group names are for the convenience of humans * Linux systems store ownership information with user IDs and group IDs | 6 | 42 |
| **Security model:** | There are 2 classes of security model.  Superuser and Normal user | 6 | 42 |
| **Superuser** | * The superuser is usually called root. * Superusers control all files, processes, and devices * Superusers accounts always have UID of 0 | 6 | 43 |
| **Normal user** | * Multiple UID 0 accounts can exist besides the root account * Only Username with the first uid 0 account appears in the log entries * UID 0 account creation or access is any attacker’s goal. | 6 | 43 |
| **Anonymous Login** | Login as Root. Dangerous! | 6 | 44 |
| **Privilege Elevation** | * You can legitimately elevate your access to superuser in three ways. * The first way is to simply log in to the system as the root account. * The second way to gain superuser access is by using the su command, which is located in the /bin/su or the /usr/bin/su paths. * The third way to gain superuser access is through the use of the sudo command. * The sudo command enables a normal user to run a command with root privileges. * The normal user can execute the sudo command only if he or she is in the /etc/sudoers file. | 6 | 44 |
| **Su command** | Switch user | 6 | 44 |
| **sudo command** | Enables a regular user to run a command with root privileges. | 6 | 44 |
| **Sudo and Sudoers File** | Sudo gives granular control of execution:   * Users only know their own password * Allowed to execute only the commands specified in sudoers file | 6 | 45 |
| **System Accounts** | * Linux operating system tend to come with many service accounts for various apps. * Typically, these accounts have low UID numbers (UID <100? 500? 1000?). * Attackers sometimes activate these accounts as “backdoors” into the system. * If you are not using a particular service or app, remove (or block) the account. | 6 | 46 |
| **Passwd File** | Originally, UNIX systems kept a user's one-way encrypted password along with user account information in the /etc/passwd file. Each field in the password file is separated by a colon (:), as follows:   * 1st: User ID * 2nd: Location for the encrypted password (normally an x to indicate the system is shadowed) * 3rd: User identification number * 4th: User's primary group * 5th: GECOS fiel65d (remarks) * 6th: User's home directory * 7th: What happens upon login (normally this will be a process) | 6 | 47 |
| **Shadow File** | * Username: Name matching the entry in the /etc/passwd file * Password: Encrypted password along with the salt and type of encryption * Last: Number of days since the last password change based on 1 Jan 1970 * Minimum: Number of days before the password can be changed again * Maximum: Number of days after which the password must be changed Warn: * Number of days before the password must be changed, so the user gets a warning message * Expire: Number of days before the account becomes disabled after its password expired * Disable: Number of days before the account becomes disabled Reserved: Not used at this time | 6 | 48 |
| **Pluggable Authentication Modules (PAM)** | The Pluggable Authentication Modules (PAM) are system libraries that handle Linux authentication and the many tasks that go along with it. PAM employs four management groups that handle specific types of authentication requests. The four groups perform the following functions:   * Authentication (auth): Used to establish a user identity and possibly membership in a group * Passwords (password): Used when a user provides authentication credentials such as a password * Sessions (session): Aids in implementing services and tasks associated with user authentication * Accounts (account): Used to perform actions not based on authentication. Configuration files are in /etc/pam.d | 6 | 49 |
| **Password Enforcement - PAM** | System-wide password enforcement is mainly controlled by the system-auth configuration file. To set minimal password requirements, you must set the following variables:   * minlen= $: Minimum length of password must be $. * lcredit=-$: Minimum number of lowercase letters must be $. * ucredit=-$: Minimum number of uppercase letters must be $. * dcredit=-$: Minimum number of digits must be $. * ocredit=-$: Minimum number of other characters must be $. * difok=$: Number of character changes between the old and new password that are enough to accept the new password must be $. | 6 | 51 |
| **PAM pam\_pwhistory** | Is used to set a policy and prevent the reuse of older passwords.  Difok = $: number of characters that must be different for the old password $.  Pam\_pwhistory: Modules remember old password uses.  Remember=$: this value is the number of old passwords remembered | 6 | 51 |
| **Locking User Accounts After Too Many Login Failures - PAM** | pam\_tally module counts fail login  onerr=fail: tells the system what to do when reaching a set number of fail-lock accounts.  no\_magic\_root: tells the system not to lock the root account  per\_user keeps an account of each individual use.  deny=: is the number of attempts made before locks $ | 6 | 52 |
| **IoC - Indicators of Compromise** | This is one method of performing basic threat hunting in your environment. In threat hunting, you proactively look for indicators of compromise (IoC) to find an adversary early in the attack process. Because this method is run on a regular basis, it can find new infections quickly and allow for timely response to minimize the overall impact. | 6 | 53 |
| **ls** | Shows you a list of files on the system and helps identify modification or changes to files | 6 | 53 |
| **netstat** | Shows network connections, including outbound connections and the services that are making the connections | 6 | 53 |
| **ps** | Provides a list of running services and details about the services | 6 | 53 |
| **Threat Hunting (ls, ps, netstat)** | This is one method of performing basic threat hunting in your environment. In threat hunting, you proactively look for indicators of compromise (IoC) to find an adversary early in the attack process. Because this method is run on a regular basis, it can find new infections quickly and allow for timely response to minimize the overall impact.  The objective is to find new files that have been loaded or installed, are running as a service and are making an outbound connection., a high indication of malicious activity. | 6 | 53 |
| **System compromise: after the fact** | Most adversaries once a system gets compromised, are going to:   1. install programs 2. run daemons or services 3. make outbound connections. | 5 | 53 |
| **tail in Action** | Many examples:  Use to option -f to output appended file updates.  The sleep option -s can be set to control the refresh rate.  tail -f -s 5 /var/log/messages  Using the -c option, you can control the numbers of bytes displayed.  tail -c 500 /var/log/messages | 6 | 54 |
| **| - Piping** | A great way to do this is to do on-the-fly scripting by piping (that is, using the |) to send the output from one command as input to another command. This tool enables a security professional to take steps that are normally manually performed and allows them to be automated | 6 | 55 |
| **grep** | Although automation is important, often an analyst wants to be able to look for or identify certain activity. This is the point at which the grep command comes into play. The grep command can be used to search for certain patterns in an input file or input piped to the grep command. As output, the command will show the "hits" in the input where the pattern matched. | 6 | 55 |
| **ssh alerts/log messages - find** | tail /var/log/auth.log | grep sshd | 6 | 55 |
| **tail /var/log/auth.log | grep sshd** | Find all alerts related to ssh logins | 6 | 55 |
| **Daemons** | Services | 6 | 56 |
| **Linux Services** | A Linux service is an application (or set of applications) that runs in the background waiting to be used or carrying out essential tasks.   * Services are also referred to as daemons * •Sometimes the name of these services or daemons ends with the letter d. (sshd) | 6 | 56 |
| **Services status: Linux** | * Enabled: services are currently running. They usually have no problems. * Disabled: services are not active but can be activated at any time without a problem. * Masked: services won't run unless we take that property away from them. * Static: services will only be used in case another service or unit needs it. | 6 | 56 |
| **sudo systemctl | grep httpd** | Search for httpd service on the system | 6 | 56 |
| **sudo systemctl | grep running** | List all active services on the system | 6 | 56 |
| **sudo systemctl list-unit-files --type service --all** | List all the services on the system | 6 | 56 |
| **init** | * first process to start. * Also knows as system V style init. * Init was the original boot time service started * Follows a linear process * Unable to detect and handle sudden changes to the system. * Tracks services only during boot or shutdown. * Uses runlevels (6) | 6 | 57 |
| **Runlevels** | All init-based solutions use runlevels, which tell services how a process will start as the init process starts. The various runlevels with init are as follows:   * Runlevel 0 means shut down the systems * Runlevel 1 means single-user mode * Runlevel 2 means multi-user mode * Runlevel 3 means multiuser mode with networking * Runlevel 4 means undefined / special use case (customizable) * Runlevel 5 means starting the system with appropriate display manager and graphics * Runlevel 6 means system reboot | 6 | 57 |
| **System V** | Init is also known as system Vxs | 6 | 57 |
| **Systemd (1)** | Systemd is the name of a software suite; among many other functions, it replaces the init daemon, which is responsible for starting and running processes and services in Linux.  Features:   * Supports parallel processing * Monitors services after boot * Supports device hot plugging | 6 | 59 |
| **Units** | Resources the system is configured to manage | 6 | 59 |
| **Segmentation** |  | 6 | 60 |
| **systemctl list-units --type service --all** | Systemd common commands | 6 | 60 |
| **Systemd (2)** | Common commands. Systemd plays a key role on the hardening of Linux systems because a key component of security is to reduce the attack surface | 6 | 60 |
| **Systemd (3)** | **Pros:**  • Faster booting process  • Configuration of device dependencies  • SELinux integration  • Able to restart crashed services  • Parallel processing.  **Cons:**  • No syslog; binary logging instead  • No cron; replaced by calendar timers  • Not compatible with other init managers  • BSD does not support systemd  • Daemons are bundled and integrated | 6 | 61 |
| **Systemd (4)** | 3 core functions:  First, systemd provides an integrated method for managing both systems and services. Second, systemd provides a software platform for managing, maintaining, and creating software. You can use this platform not only to make changes to the existing services but also to create and develop new services.  Third, systemd provides an interface between the applications and the kernel.  This interface is dangerous because the kernel is the brain of the OS and, if compromised, can cause a significant negative impact on the system. | 6 | 62 |
| **Cron** | * Scheduling daemon * Starts an action (in the background) at preset time * Can also be employed by users * Uses crontab file to store the jobs that are going to run * Cron daemon (crond) works in sync with the system clock | 6 | 63 |
| **Linux Package Management: Overview** | * Software is distributed in precompiled packages * Packages contain binaries and dependencies that are required for the software to run * Package management tools monitor updates and changes so that software upgrades can automatically happen | 6 | 64 |
| **Package Management: Features** | 1. Download validation: Verify the download is valid and from a trusted source. 2. Installation of dependencies: Determine dependencies and make sure they are installed. 3. Binary format: Install precompiled applications to minimize dealing with source code. 4. Standard locations: Use standards to install files in common locations to enable reuse of code such as libraries. 5. User experience components: Add features to make installation easier and enhance the user experience. 6. Verification of installation: Verify the installation was done correctly and make sure it is a stable installation. | 6 | 65 |
| **Advanced Packaging Tool (APT)** | Debian uses dpkg as the tool for its package management; the main piece of software that implements its package management is the advanced packaging tool known as APT. It is important not to confuse Linux APT with advanced persistent threats. They use the same acronym, but they have a completely different meaning | 6 | 66 |
| **dpkg** | * Debian package management is based on the tool dpkg. * The common package management solution is APT * Packages often have the extension .deb * Often compatible with Ubuntu ( be careful with declared dependencies) * Easy-to-use syntax and commands. | 6 | 66 |
| **apt - Using** | * **apt-get update:** * Updates the system database of software updates that are available. * **apt-get install package-name**: Installs the specified software with all associated dependencies. * **apt-get remove package-name**: Removes the specified software but does not remove dependencies. * **apt-get update:** Updates the system database of software updates that are available. * **apt-get upgrade**: Upgrades all the software with the available updates. | 6 | 67 |
| **Mandatory Access Control (MAC)** | SELinux uses Mandatory Access Controls, which enable administrators to control all software interactions on the system. The security model is based on least privilege and starts with users having no rights at all. Administrators must grant users access through the use of security policies. | 6 | 70 |
| **Multi-Category Security (MCS)** | is an enhancement to SELinux and allows users to label files with categories. These categories are used to further constrain *Discretionary Access Control* (DAC) and *Type Enforcement* (TE) logic | 6 | 70 |
| **Multilevel Security (MLS)** | The Multi-Level Security technology refers to a security scheme that enforces the Bell-La Padula Mandatory Access Model. Under MLS, users and processes are called *subjects*, and files, devices, and other passive components of the system are called *objects*. Both subjects and objects are labeled with a security level, which entails a subject's clearance or an object's classification. Each security level is composed of a *sensitivity* and a *category*, for example, an internal release schedule is filed under the internal documents category with a confidential sensitivity. | 6 | 70 |
| **SELinux** | Security enhanced Linux or SELinux is a loadable kernel module specifically designed for security. This module has a set of kernel configuration and provides support access control policies. Most modern Linux Releases support the SELinux kernel module. In essence, it controls what programs can do within the Linux system | 6 | 70 |
| **Bell-LaPadula Mandatory Access Model** | MLS enforces the Bell-LaPadula Mandatory Access Model, and it is used in Labeled Security Protection Profile (LSPP) environments. | 6 | 71 |
| **Labeled Security Protection Profile (LSPP)** | LSPP conformant products support access controls that are capable of enforcing access limitations on individual users and data objects. Two classes of access control mechanisms are provided: those that allow individual users to specify how resources (e.g., files, directories) under their control are to be shared; and those that enforce limitations on sharing among users. | 6 | 71 |
| **AppArmor** | * Includes many default policies. * Linux kernel module to restrict the capabilities of programs. * Behavior-based protection and dynamic protection. * Restricts program's resource access and privilege level, controls can be defined per application. * Combines static analysis and learning-based tools | 6 | 72 |
| **Password Spraying** | Password spraying is a simple concept which takes a commonly used password and tries it across a very large number of accounts, typically web and cloud resources. | 6 | 72 |
| **Sysctl Hardening** | Sysctl is an interface used to modify kernel settings at runtime; within the /etc/sysctl.conf file you can modify network and system settings to harden your Linux. Sample configuration files are available. Use the command sysctl –a to show all the variables for your system | 6 | 74 |
| **Address Space Layout Randomization**  **ASLR** | Address Space Layout Randomization: This technique is being used to prevent attackers, who previously knew that a program function was vulnerable and could be exploited, from getting access to it by preventing him from knowing the memory address where that function is allocated. | 6 | 75 |
| **Buffer Overflow - Hardening** | Address Space Layout Randomization: This technique is being used to prevent attackers, who previously knew that a program function was vulnerable and could be exploited, from getting access to it by preventing him from knowing the memory address where that function is allocated. | 6 | 75 |
| **/bin/false** | Module will not be loaded | 6 | 76 |
| **Loadable Kernel Modules (LKMs)** | Is an object file that contains code to extend the running kernel, or so-called base kernel, of an operating system. LKMs are typically used to add support for new hardware (as device drivers) and/or filesystems, or for adding system calls. | 6 | 76 |
| **Modprobe** | Being used to add or remove loadable kernel modules (LKM) from your kernel. By disabling unneeded kernel modules, you are reducing exposure to exploits and code complexity. | 6 | 76 |
| **Modules - Disabling** | 1. Find modules that you want to disable 2. Use 'modinfo' to see dependencies with other modules and validate what a module does 3. Disable the module | 6 | 76 |
| **Rootkit - Defense** | Modprobe | 6 | 76 |
| **Dynamic Loading After Boot - Disabling with sysctl** | The risks of having dynamic loading after boot is that a rootkit with any functionality can be inserted into the kernel with root privileges.  Disabling loadable kernel modules (kernel.modules\_disable=1) severely limits the vector or a malicious actor injecting a malicious kernel module.  To disable loadable kernel modules:  # sysctl -a | grep modules  kernel.modules\_disabled = 0  # echo 1 > /proc/sys/kernel/modules\_disabled | 6 | 77 |
| **Sysctl** | Disabling Dynamic Loading After Boot | 6 | 77 |
| **SSH** | The SSH Protocol (also referred to as Secure Shell) is a method for remote login from one computer to another in many Linux environments, this is being used to centrally manage these systems and is considered as a critical service, it is crucial to harden this protocol as attackers are actively abusing flaws in this protocol. |  |  |
| **SSH Hardening** | Some hardening examples you can apply on SSH via SSH config:   1. TCP Wrappers, only allow specific hosts 2. Disable Root Login 3. Set Idle Timeout Interval 4. Disable Empty Passwords 5. Set a custom SSH warning banner 6. Block SSH brute force attacks | 6 | 78 |
| **SSH Multifactor Authentication** | Multifactor authentication adds another layer of authentication, it is based on the concept on “something you know” and something you have”. For critical services such as SSH, multifactor authentication must be used. EX:   * SSH keys * Google Authenticator * FreeOTP * Authy * Duo. Some are free; others are commercial but have minimal cost. | 6 | 79 |
| **Hardening Scripts: Overview** | Hardening scripts are programs that you run on a system with the goal of properly securing or reducing the number of vulnerabilities that exist on a system | 6 | 80 |
| **Hardening Scripts:**  **PROS** | * Scalable way to secure a system. * Security can be defined once and scaled across the enterprise. * Automation of the hardening process. * Can be used as a validation method in auditing. | 6 | 81 |
| **Hardening Scripts:**  **CONS** | * Blindly applying security can be dangerous. * Security with no clear metrics makes validation difficult. * Person applying the script does not necessarily learn or understand security. * Different system require different security. | 6 | 81 |
| **Hardening Guides:**  **CIS Benchmarks for Linux** | * CIS (Center for internet Security) has benchmarks for several distributions: Ubuntu, CentOS, and Red Hat. * Provides details on how to properly secure a system. * Scoring tools (CIS-CAT) are available; then can also automatically harden your system. | 6 | 82 |
| **Lynis** | Lynis performs an extensive health scan of your systems to support system hardening and compliance testing. Lynis runs on almost all UNIX systems and will only test components that it can find. The main purpose is to perform audits for compliance frameworks such as Basel II, GLBA, HIPAA, PCI DSS, and SOX   * Available authentication methods. * Expired SSL certificates. * Outdate Software. * User accounts without passwords. * Incorrect file permissions * Firewall auditing. | 6 | 83 |
| **Log Files - Key** | * **utmp:** Gives a complete picture of users' logins including which terminals were used, logouts, system events, current status of the system, and system boot time. * **wtmp:** Gives historical data of utmp. * **btmp:** Records only failed login attempts. * **dmesg:** Is a display or driver message. It is used to examine or control the kernel ring buffer. * **messages:** Contains global system messages, including the messages that are logged during system startup. Several things are logged in /var/log/messages, including mail, cron, daemon, kern, and auth. * **maillog**: Contains the messages from the mail server running on the system, such as sendmail. * **secure**: Contains all security-related messages on the system. This includes authentication failures, possible break-in attempts, SSH logins, failed passwords, sshd logouts, and invalid user accounts. | 6 | 84 |
| **btmp** | Records only failed login attempts. | 6 | 85 |
| **last (command)** | The last command, however, can read from utmp, wtmp, and btmp. Running the command last by itself will give you who logged in, when they logged in, and when they logged out, among other useful info on the screen, and it is historical data | 6 | 85 |
| **utmp** | Utmp and wtmp are the two main log files; they cannot be read with any text editor utility or text-reading program such as cat. They must be read with the who, last, or utmpdump command. | 6 | 85 |
| **utmpdump** | Utmp and wtmp are the two main log files; they cannot be read with any text editor utility or text-reading program such as cat. They must be read with the who, last, or utmpdump command. | 6 | 85 |
| **who command** | The who command only reads from the utmp file and shows who is logged in right now. | 6 | 85 |
| **dmesg** | Dmesg is both the special log file and the command to read the log file itself. It does have some data that is similar to the messages log file, but the dmesg log contains strictly the information about the boot process of the system from the point of initial startup all the way through the loading of the entire OS and kernel. If you experienced unusual activity during the boot process of your Linux system, reviewing the dmesg logs would be your best bet in finding the cause for the unusual boot of the system. | 6 | 86 |
| **messages** | The messages log stores valuable, non-debug, noncritical messages, and should be considered the general system activity log, which is a fantastic resource for troubleshooting and anomaly hunting. The messages log file is generally located in /var/log/messages. | 6 | 87 |
| **Syslog (1)** | * The workhorse of the Linux logging system is the system logging daemon, or syslogd. * Standard for message logging. * Any part of the system including applications, drivers, and other daemons, can make log entries. * A facility code is used to specify the type or program that is logging the message. * Messages with different facilities may be handled differently. | 6 | 88 |
| **Syslog (2)** |  | 6 | 89 |
| **Syslog (3)**  **Severity levels** | Records only failed login attempts  A picture containing text, font, screenshot, receipt  Description automatically generated | 6 | 90 |
| **Syslog Security (1)**  **Syslog Security considerations** | Syslog does not, by default, encrypt the traffic, so syslog traffic is inherently not confidential, and all traffic is sent in cleartext.  Security considerations:   * Sender Authentication: Syslog does not provide strong sender authentication * Message Confidentiality: Syslog by default does not encrypt traffic and it is sent in cleartext. * Message Delivery and Replays: An attacker could record a set of messages that indicates normal activity of a machine, and the those messages can be later replayed back to the servicer without any knowledge or verification that those log messages are truly from the source server. | 6 | 91 |
| **BSD Syslog** | * Message truncation with BSD syslog (1024 bytes maximum per packet) * Time precision up to second (Insufficient for many attacks) | 6 | 92 |
| **IETF Syslog** | * IETF Syslog should be preferred as it is independent of the transport * Not limited in message size, and allows sub seconds timestamps * New standard | 6 | 92 |
| **Syslog Security (2)** | Currently there are two standards for syslog. We have the BSD Syslog standard, which is also referred to as legacy, and the newer standard called IETF Syslog, From a security perspective, the preferred syslog standard is IETF, however, a lot of appliances tend to support BSD Syslog and not the new IETF standard. | 6 | 92 |
| **Syslog-NG**  Syslog Next Generation (syslog-NG) | * The syslog-NG application is a drop-in replacement for the traditional syslog that is included with Linux and UNIX systems. * Additional filtering * Sends data with TCP * Developed to add additional security to remote system logging * Supports advanced features such as buffering and syslog over TLS. | 6 | 93 |
| **Logrotate** | Logrotate is a useful utility that will rename and compress logs once a condition is met in order to prevent the filesystem for filling up. | 6 | 94 |
| **HUP Signal** | signal hang up, referring to the telephone industry when terminating a connection | 6 | 95 |
| **missingok** | If the log file is missing, go on to the next one without issuing an error message. | 6 | 95 |
| **Centralized Logging** | * Protects against log wiping * Denial-of-service possibility: allow only recognized machines * Needs a lot of disks for large environments * One machine holds a lot of sensitive information * Easy to search and scan. | 6 | 97 |
| **auditd** | Auditd subsystem is an access monitoring and accounting solution for Linux that is developed and maintained by Red Hat. The information in the log may prove useful in debugging security-related issues. For example, you can view the SELinux logs via auditd. | 6 | 99 |
| **Auditd subsystems:** | * File access logging * Directory access logging * Network connection logging * Lots of documentation for rules specially with CIS (center for internet security) * Rules controlled by audit.rules and/or the use of auditctl command. | 6 | 99 |
| **ausearch** | The **ausearch** utility allows you to search Audit log files for specific events. | 6 | 99 |
| **aureport** | Is a tool that produces summary reports of the audit system logs. The aureport utility can also take input from stdin as long as the input is the raw log data. | 6 | 99 |
| **autrace** | **autrace** is a program that will add the audit rules to trace a process similar to strace. It will then execute the *program* passing *arguments* to it. | 6 | 99 |
| **auditd – Create Rule** | * **auditctl –l:** Lists all the currently running rules within auditd. * **auditctl –s:** Shows the current status of the auditd daemon. * **auditctl –b:** Sets maximum number of outstanding audit buffers allowed. If all buffers are full, the kernel consults the failure flag for action. * **auditctl –f:** Sets the failure flag as 0, 1, or 2. This option enables you to determine how you want the kernel to handle critical errors. If set to 0, audit messages that could not be logged will be silently discarded. If set to 1, messages are sent to the kernel log subsystem. If set to 2, it will trigger a kernel panic. * **auditctl –R:** Enables you to read audit rules from the file specified. This capability is useful when you are testing some temporary rules and want to use the old rules again from the audit.rules file. * **auditctl –a:** Is used in conjunction with –S for system call auditing. The options for –a are action and filter, which are set to specify when a certain event is logged. Action can be either always or never. Filter specifies which kernel rule-matching filter is applied to the event. The rule-matching filter can be one of the following: task, exit, user, or exclude. Action,filter will be always,exit in most cases, which tells auditctl that you want to audit this system call when it exits. * **auditctl –D:** Deletes all the rules running in the auditd daemon. | 6 | 100 |
| **auditd - Search and Report Audit Logs (1)**  **Step 1**  **Step 2**  **Step 3**  **Step 4**  **Step 5**  **Output for Step 5** | Let’s use auditd to monitor any changes on its configuration file and see how to detect potential malicious activities.  Add and audit entry via auditctl to validate if the SSH config is accessed or changed  Sudo auditctl -w /etc/ssh/sshd\_config -p rwxa -k SANSsschange  Write audit rule Path to file. Permissions name    Let’s generate a long entry by access our sshd configuration file  SANS@SANS401: ~$ nano /etc/ssh/ssh config  Check if that even is in our audit logs  A picture containing text, screenshot, font, businesscard  Description automatically generated  Let's run the following command:  Sudo ausearch -f /etc/ssh/sshd\_config -i  A screen shot of a computer screen  Description automatically generated with low confidence | 6  6  6  6  6  6  6 | 100  100  100  101  101  101  102 |
| **ausearch** | Ausearch enables you to look through the logs based on the following:  • Message type, –m,  • When the logs occurred, –start  • Key phrases that mark logs by the rule, –k  • The audit event ID, –a  • The filename, –f | 6 | 101 |
| **autrace** | Autrace enables you to trace an executable upon execution so that you can learn what system calls it makes. Consequently, you can build auditd rules for interesting system calls you feel you should audit and track | 6 | 101 |
| **auditd - Search and Report Audit Logs (2)** |  | 6 | 102 |
| **sudo aureport -f -i –summary** |  | 6 | 102 |
| **sudo aureport -f -i | grep sshd\_config** |  | 6 | 102 |
| **Aureport** | * Aureport provides human readable output on the audit events in audit.log. * The command aureport -f -I summary gives a nice overview on the numbers of audit records recorded per file. | 6 | 103 |
| **Auditd Rules** | * Auditctl -w /etc/sysconfig/ -p rwa – k sysaccess:   Adds a watch to the /etc/sysconfig only and all files and directories beneath it for any read, write or attribute change access.   * Auditctl -w /sbin/Modprobe -p x -k kernel\_modules   adds a rule to watch for execution of the /sbin/Modprobe command   * Auditctl -W /sbin/Modprobe -p x -k kernel\_modules:   Removes Modprobe rule by using the -W.   * Auditctl -a always,exit -F arch=b64 -F “auid<=1000” -S rename -S renameat -k rename:   Monitors for system calls always and when they exit, it looks at 64-bit processes only | 6 | 104 |
| **Firewalls - Linux** | In addition to having network firewalls, it is also critical to deploy host-based firewalls. The beauty of Linux is that the same software that is used to protect a network can also be used for host-based protection. One of the valuable benefits of doing this is that you now have a more scalable solution in building, maintaining, and deploying rulesets. | 6 | 105 |
| **Host-based Firewalls** | * Receive packets * Inspect packets * Denied by the ruleset, drop the packet * Example: Iptables | 6 | 105 |
| **Network-based Firewalls** | * Receive packets * Inspect packets * Denied by the ruleset, drop the packet * Example: Pfsense | 6 | 105 |
| **Iptables – Built-in Firewall** | Iptables allows you to configure the tables, rules, and chains provides by the Linux kernel firewall. It is a powerful and customizable tool that provides firewalling and NAT capabilities on your Linux system. | 6 | 106 |
| **Rootkit Detectors** | * Look for indicators of compromise in how a rootkit operates. * The stealthier the rootkit, the less accurate the detection. * Often used for a kernel level rootkit. * Will work with file-level rootkits but often most effective with file integrity checking * Have been known to generate false alarms, so useful as an indicator but not a conclusive decision. | 6 | 108 |
| **rkhunter** | * Detects rootkits, backdoors, and some local exploits * Compares hashes of important files to online databases of known good hashes * Capable of allowlisting * Mails you if an alert is found * Advisable to set up with crontab or timers for scheduled checks | 6 | 109 |
| **chkrootkit** | * Checks system for suspicious processes and known bad files * The command to run a scan is chkrootkit * Common false positive on a mail server is: Bind shell infected on port 465 | 6 | 110 |
| **Configuration Management Tools** | Are an important aspect of increasing the overall security of your Linux environment.   * Keeps systems configured the same. * Many different players in the market. * Both open-source and commercial options. | 6 | 111 |
| **Ansible - Configuration Management** | Is a configuration management tool that is open-source that allows to automate system administration tasks related to installation, account setup, configuration and so on. It can be used to automate the provisioning of servers or virtual machines, both in the cloud and on-premises. The Ansible infrastructure consists of a control node (in blue on the drawing) where Ansible is installed and of managed nodes which represent the network devices or servers (also known as hosts) that you want to manage with Ansible and are represented in green on the drawing. It is from the control nodes that you can run commands or playbooks on the managed nodes. For that matter, there is an SSH trust between the control nodes and the managed nodes, no agent is required in order to run Ansible. | 6 | 112 |
| **Playbook** | Defines the desired state of these client systems and are written in YAML. These playbooks are designed to be reusable to allow the execution of a tasks more than once on multiple hosts. | 6 | 112 |
| **YAML** | Human readable format | 6 | 112 |
| **Containers** | * Isolate and perform resource allocation * Kernel/OS virtualization * Contains the application and associated files * Compact and portable * Built on top of the host OS's kernel | 6 | 116 |
| **Containers vs. Virtual Machines (1)** | If someone wanted to just move a specific application and have it run in a consistent manner, virtual machines did not work very well. The introduction of containers solved this problem. Containers contain the application and all associated dependencies to enable the application to run in a consistent manner on different platforms. In the case of containers, the physical hardware is not virtualized, the abstraction layer is done at the OS level. | 6 | 116 |
| **Virtual Machines** | * Isolate and perform resource allocation * Hardware virtualization * Contains the entire operating system's kernel * Large disk size * Hypervisor as abstraction layer | 6 | 116 |
| **Containers vs. Virtual Machines (2)** | * Virtual machines perform isolation at an OS level * Containers perform isolation at an application level. * Virtual machines are bigger and not as compact as containers. * Each container shares the same kernel, the applications run as isolated processes, but they run on a single host operating system. * Virtual machines each have different guest operating systems that run on one host operating system. * In containers, each application is running on a single host operating system. * In virtual machines each application in each guest uses a different separate kernel. * In containers, all the applications share the same kernel | 6 | 117 |
| **chroot** | “chroot" isolates an executable to a given directory or location. The problem is this becomes difficult with shared libraries or components. LXC allows for the application to run in an isolated environment, but it contains all the required components, so it can overcome the limitations of chroot. | 6 | 118 |
| **Containers** | LXC is often viewed as a middle ground between chroot and virtualized environments. | 6 | 118 |
| **Linux LXC:**  **Kernel modules** | * Kernel namespaces (ipc, uts, mount, pid, network, user) * Cgroups * Apparmor and SELinux profiles. * Seccomp policies * Chroots (using pivot\_root) * Kernel capabilities | 6 | 118 |
| **Linux LXC/Containers** | * LXC is an open standard for creating containers in a Linux environment. It provides a powerful way to create a container that contains all the components needed to run an application. * Allows for creation and distribution of containers * Middle ground between chroot and virtualized environments. * Creates an isolated environments running on a single kernel. * Operating system level containers. | 6 | 118 |
| **cgroups** | * Developed by Google * Control the isolation of system resources * Govern the system resources * Key resources are CPU and memory * Control resources for a group of processes | 6 | 119 |
| **Containers: cgroups and namespaces** | Cgroups and namespaces are the core of the kernel that allows for the isolation of different applications. It is what provides the foundation for security. | 6 | 119 |
| **namespaces** | * Developed by IBM * Obtain system resources that are needed * Present the resources to the application * Appear to be dedicated to the application * Isolation for a single process | 6 | 119 |
| **Containers: LXC to Docker - Comparison of** | Docker is an evolution of LXC | 6 | 120 |
| **Docker** | * Single application LXC containers * More portable and flexible containers * Easy to deploy, replicate, and move application workloads * Single process and stateless | 6 | 119 |
| **LXC** | * Original Linux containers * Operating system level virtualization * Allows for multiple operating system virtualizations on a single host * Key features cgroups and namespace | 6 | 120 |
| **Docker** | * Docker uses OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries, and configuration files. * Based on open standards. * Creates a filesystem that contains the application and all components needed to run it. Market leader in container technology. * Application will run the same in all environments. * Applications are isolated to achieve a level of security | 6 | 121 |
| **Docker Hub** | * Public container repository available at hub.docker.com * Official images from major software vendors * Can be used to publicly store personal images | 6 | 122 |
| **Docker Images** | A Docker container is an instance of a Docker image which is a combination of a file system and startup parameters. Docker images are defined in a Dockerfile which contains the instructions for building the image. | 6 | 122 |
| **Docker Registry** | * Custom Docker images can be stored in privately hosted registries. * Docker registry server image available in Docker Hub * Public cloud providers have private registry services | 6 | 122 |
| **Dockerfile** | * Contains instructions to build Docker image * Can use publicly available images as a base * Images can be built at runtime * Images can be stored in public or private registries | 6 | 122 |
| **Docker Swarm Mode** | * Docker Swarm Mode is an open-source container orchestration platform and is the native clustering engine for and by Docker. * Create a service where you define its optimal configuration state. * Multiple Docker hosts which run in swarm mode and act as managers and workers. * Built into the Docker engine. | 6 | 123 |
| **Borg** | Borg is Kubernetes’ predecessor. Most of the features in Kubernetes, originated from Borg, with some little modifications. Their basic structure is similar. | 6 | 124 |
| **k8s** | K8s is short for Kubernetes. | 6 | 124 |
| **Kubernetes (k8s)** | * Kubernetes (k8s) is an open-source container-orchestration system for automating application deployment, scaling, and management. * Originally designed by Google based on their internal container-orchestration platform “Borg” * Maintained by the cloud native computing foundation. * Manages container deployments at scale and high availability across multiple nodes. * Complex system of building blocks to deploy, maintain and scale containerized applications. | 6 | 124 |
| **Pods** | Are the smallest deployable units of computing that can be created and managed in Kubernetes. | 6 | 124 |
| **Container Security** | Docker Best Practices, Vulnerability Scanning Tools, Secure Configuration Baselines | 6 | 125 |
| **United States Government Configuration Baseline (USGCB)** |  | 6 | 125 |
| **Docker Security Best Practices** | 1. Use minimal trusted base images to limit vulnerabilities is system libraries.  2. Do not run applications as root. Create a dedicated user with minimal permissions.  3. Cryptographically sign and verify your Docker images to mitigate MITM attacks.  4. Monitor custom or base Docker images for known vulnerabilities.  5. Do not store secrets such as connection strings or API keys in Docker images.  6. Docker containers should not share the network stack of the host by using the --net=host flag.  7. Don't enable unused services such as SSH or expose unused ports.  8. Do not allow containers to be run with the –privileged flags as this allows full access to the Docker host. | 6 | 126 |
| **Anchore** | * Command-line took or Docker container to do CVE-based security vulnerability reporting. * Defines customer security policies based on allowlist or denylists, credentials or file contents. | 6 | 127 |
| **Clair** | * API-driven static container analysis * Consumes multiple CVE databases such as Debian Security Bug Tracker, Ubuntu CVE Tracker, and Red Hat Security Data * Build by CoreOS, which is now a part of Red Hat * Built into Quay.io, a private cloud container registry alternative to Docker Hub | 6 | 127 |
| **Docker Images - Vulnerability Management in** | Just like an OS, Docker images contain software packages that could contain known vulnerabilities. Open-source scanning tools such as Clair and Anchore can be used to detect and manage them. | 6 | 127 |
| **CIS Benchmarks** | To secure Docker or Kubernetes, the Center for Internet Security (CIS) has created secure configuration benchmarks containing security best practices. |  |  |
| **CIS Benchmarks: Docker** | * Ensure auditing is enabled on Docker host. * Docker daemon configuration file permissions need to be set correctly * Docker images need to be securely configured. * Do not run privileged Docker runtime | 6 | 128 |
| **Kubernetes CIS Benchmarks** | * Configure the management API securely. * Kubernetes configuration file permissions need to be set correctly * Ensure key-value store is securely configured. * Ensure worker nodes are securely configured. | 6 | 128 |
| **Infrastructure as Code (IaC)** | To make this process even easier, you could write the provisioning process in code (similar to JSON). This concept is known as Infrastructure as Code (IaC) and can be created with tools such as Terraform. With IaC, teams can automatically manage and provision the technology stack for their needs by writing a piece of code rather than manually configuring and deploying the whole infrastructure. | 6 | 129 |
| **Terraform** | * Terraform allows you to declaratively define what your infrastructure should look like and then create, update, and manage your infrastructure resources. * Write declarative configuration files. * Plan and predict changes * Create reproducible infrastructure. | 6 | 129 |
| **terraform init** | The terraform init command is used to initialize a working directory containing Terraform configuration files. This is the first command that should be run after writing a new Terraform configuration or cloning an existing one from version control. It is safe to run this command multiple times | 6 | 130 |
| **Terraform Life Cycle** | INIT: the terraform init command is used to initialize a working directory containing Terraform conf files.  PLAN: the terraform command plan is used to create an execution plan.  APPLY: is used to apply the changes required to reach the desired state of the configuration, or the pre-determined set of actions generated b a terraform plan execution plan.  DESTROY: the terraform destroy command is used to destroy the terraform-managed infrastructure. | 6 | 130 |
| **terraform plan** | Used to create an execution plan. | 6 | 130 |
| **terraform refresh** | If you happen to be out of sync (your existing infrastructure no longer matches with your intended Terraform state), you can run the terraform refresh command to sync everything. | 6 | 130 |
| **terraform apply** | The terraform apply command is used to apply the changes required to reach the desired state of the configuration, or the pre-determined set of actions generated by a terraform plan execution plan | 6 | 131 |
| **terraform destroy** | The terraform destroy command is used to destroy the Terraform-managed infrastructure. | 6 | 131 |
| **Terraform Security Best Practices** | 1. Don't commit the state file. The .tfstate file is automatically generated when you run a terraform command that alters your state (e.g., terraform apply). This file stores all the secrets, keys, configuration files, generated passwords, etc. It is crucial that this file does not fall into the wrong hands. 2. To follow-up on the previous point, a Terraform backend is a remote location that is used to store the state file. This allows the state file to be locked to prevent errors when multiple people are working on the same environment at the same time. Some backends allow you to encrypt your state as well (e.g., GCS backend). 3. Run pre-apply and post-apply checks with tools such as Sentinel, InSpec, Serverspec, and Forseti to ensure that the new build does not violate any policies. 4. Terraform allows you to use variables. This not only allows your code to be more maintenance friendly, but it also increases your security. You are no longer required to store sensitive information in the config files. | 6 | 132 |
| **Amazon Web Services (AWS)** |  | 6 | 135 |
| **AWS Well-Architected Framework: Security Pillar** | The AWS Well-Architected Framework is provided by Amazon to give guidance in the application of best practices and recommendations for the design, delivery, and maintenance of secure AWS workloads. It helps customers design and operate reliable, secure, efficient, and cost-effective workloads in the cloud. The framework is divided into six pillars: | 6 | 136 |
| **AWS Security Pillars** | See book | 6 | 136 |
| **AWS IAM** | Allows you to restrict and to control access to your resources within your AWS account. IAM is one of the biggest threat areas in the cloud, as improper identity and access management can lead to unauthorized access and data leakage. | 6 | 137 |
| **Identity and Access Management (IAM)** | AWS IAM provides different features:   * Ability to grant access to use and administer resources in your AWS account. * Set granular permissions for different resources to ensure that only authorized users can perform actions. * Integrate with other AWS services so that you can implement specific policies, permissions, temporary access, and service roles. * Federate an identity trust to an outside IdP so that you can maintain a more singular set of users, if you are already using an IdP (you don't have to create users in your AWS account). | 6 | 137 |
| **Identity and Access Management (IAM)** | Purpose:  Restrict and control access to resources  Why:  Important perimeter as IAM users are exposed to the internet.  Features:  Many different features: grant access, granular permissions, temporary access, identity federation, analysis of access.  Components:  Users, Groups, Policies and Roles | 6 | 137 |
| **AWS IAM Key Concepts** | An account, Principals, User, Roles, Policies | 6 | 138 |
| **AWS IAM Key Concepts:** Account. | * Strong isolation barrier * Administrative capabilities * Container for resources * Isolate IAM * Billing & cost management | 6 | 138 |
| **AWS IAM Key Concepts:** Principals. | * Person/app that makes a request for an action * Users/services/roles | 6 | 138 |
| **AWS IAM Key Concepts:** User | * Entity with credentials. * Actual person or application * Not authorized to perform actions by default * Groups. | 6 | 138 |
| **AWS IAM Key Concepts:** Roles | * Not associated to an entity. * Assumable by anyone who needs it. * No long term credentials. * Specific permissions | 6 | 138 |
| **AWS IAM Key Concepts:** Policies | * Set of permissions to control access. * Who and what (+ conditions) * Managed policies. * Inline policies | 6 | 138 |
| **IAM - Key Concepts AWS** | An account, Principals, User, Roles, Policies | 6 | 138 |
| **Inline Policies** | Inline policies are policies that are directly embedded into one single entity. | 6 | 139 |
| **Managed Policies** | There are two types of policies: managed policies and inline policies. Managed policies can be attached to multiple entities within an AWS account. | 6 | 139 |
| **IAM Access Analyzer** | IAM Access Analyzer is a free tool provided by AWS to identify resources (e.g., S3 or roles) that are shared with an external entity (access from the outside of the zone of trust). | 6 | 140 |
| **Identity Federation** | * System of trust between two parties. * Authenticate users * Pass information and authorize access/actions * Assign permissions to external user identities * No local IAM users to manage. | 6 | 140 |
| **Identity Provider (IdP)** | A SAML 2.0 identity provider is an IAM resource that describes an identity provider (IdP) service that supports the SAML 2.0 (Security Assertion Markup Language 2.0) standard. | 6 | 140 |
| **Service Provider (SP)** | A cloud service provider is the company where you get your cloud services, such as AWS. | 6 | 140 |
| **Amazon Cognito** | Amazon Cognito is an authentication service that allows users to sign-up, sign-in, and access your web and mobile applications. The authentication process relies on social identity providers (IdP) such as Google, Facebook, and Amazon, or on enterprise identity providers such as Azure Active Directory. Cognito can be leveraged to control access to applications running on AWS resources, through the mapping of users with roles (RBAC) or with Attribute-Based Access Control (ABAC) using attributes from the IdP. | 6 | 142 |
| **Identity Pool** | An identity pool is used to grant temporary AWS credentials to access AWS resources and services such as S3 and DynamoDB (a fully managed NoSQL database service). An identity pool supports anonymous guest users as well as a federated third-party IdP (Facebook, Google, Apple, Amazon, etc.) | 6 | 142 |
| **User Pool** | A user pool is a directory in Amazon Cognito. It allows users to sign-in and sign-up to your web or mobile application through Amazon Cognito or a federated third-party IdP. All users will have a directory profile that can be accessed through an SDK (Software Development Kit). | 6 | 142 |
| **AWS Console** | * AWS Management Console (Web Interface): manage and access AWS cloud resources. * Supports browser and mobile application * Access to the services and the resources of an AWS account * Gives information related to billing and cost management and provides health monitoring. * Sing-in as the root account or as an IAM user. Possible to use and switch roles in the AWS console | 6 | 143 |
| **AWS Access Key ID** | Is generated and downloaded from the IAM section of the AWS management console. | 6 | 144 |
| **AWS CLI** | * The AWS Command Line Interface (CLI) is an open-source tool that allows you to access and manage your AWS services and resources programmatically. * The CLI runs on Linux, Windows, and macOS. * Programmatically manage AWS services and resources. * Use Access Key to log in. * Different outputs: JSON, TEXT or TABLE. | 6 | 144 |
| **Secret Access Key** | Secret access keys are—as the name implies—secrets, like your password. | 6 | 144 |
| **AWS Access Keys** | Access keys allow IAM principals to make programmatic calls to AWS. Access keys are composed of an access key ID and a secret access key. As previously mentioned, these keys can be generated by the root user (or another IAM principal) and downloaded from the AWS Management Console. An important point to note is that it is only possible to download an access key at the time of creation, after which it will not be possible to download again. | 6 | 145 |
| **Temporary Access Keys** | Additionally, there are temporary access keys. With temporary access keys, in addition to the access key ID and the secret access key, there is a temporary security token that must be sent to AWS when the temporary security credentials are to be used. An advantage to that, as the name suggests, is that these are like short term credentials that expire after a certain amount of time. These temporary access keys can be used to grant temporary access for users to resources in an AWS account | 6 | 145 |
| **Programmatic Access** | Generate and download access keys.   * Access Key ID. * Secret Access Key | 6 | 145 |
| **Key Management** | * Up to two keys per user. * Rotate Access Keys and Disable root user keys. | 6 | 145 |
| **Human Identities** | * Human Identities: these identities access AWS resources via the AWS Management Console, the AWS CLI, a client application, or a mobile application. * Machine Identities: these identities make requests to other AWS services | 6 | 146 |
| **Machine Identities** | * Human Identities: these identities access AWS resources via the AWS Management Console, the AWS CLI, a client application, or a mobile application. * Machine Identities: these identities make requests to other AWS services | 6 | 146 |
| **Identity Federations** | * Centralize identity management. * Use temporary credentials. * Cognito to authenticate, authorize and manage end-users. * ABAC with session tags or groups to manage permissions. | 6 | 146 |
| **Long-term Credentials** | * Change passwords periodically. * Use temporary credentials instead of access keys. * Use AWS secret manager for non IAM related credentials. | 6 | 146 |
| **AWS Organizations** | * Account separation (Workloads and OUs) * Service control Policies. | 6 | 148 |
| **Share Resources** | * Policies or assume role. * AWS Resource Access Manager. * IAM Access Analyzer. | 6 | 148 |
| **Identity-based Policies** | Identity-based policies are attached to users, groups, and roles. | 6 | 148 |
| **Permissions Boundaries** | Allow for the defining of a maximum set of permissions that an identity-based policy can be granted. | 6 | 148 |
| **Principle of Least Privilege (AWS)** | * Identity-based and resource-based policies. * Permissions boundaries. | 6 | 148 |
| **Resource-based Policies** | Resource-based policies are attached to resources. | 6 | 148 |
| **Service Control Policies** | SCP can be applied to applied at different levels (root, OU level or single account level of an organization hierarchy and can define security controls that all IAM principals must adhere to. | 6 | 148 |
| **AWS Resource Access Manager (RAM)** | Provides for access management by automatically granting or revoking access to accounts as they are moved in and out of the Organization/OUs. | 6 | 149 |
| **Enable Traceability** | This design principle is all about the implementation of monitoring, alerting, and auditing of the actions and changes that happen in an AWS environment. | 6 | 150 |
| **Traceability (AWS)**  **Collect log and metric data** | * CloudTrail, Config, Security Hub, GuardDuty. * AWS service metrics. * AWS service logging capabilities. | 6 | 150 |
| **Traceability (AWS)**  **Automate** | * Automated actions and notifications for remediation. * Automate analysis and log searching. * Investigate and remediate. | 6 | 150 |
| **Traceability** | Is really about monitoring AND detection. | 6 | 150 |
| **Amazon CloudWatch** | Amazon CloudWatch is a monitoring solution for AWS services as well as applications running on AWS.   * Metric repository with alarms and automated actions. * CloudWatch Logs is the central repository to gain visibility into all logs. * CloudWatch events describes operational changes and performs automated actions. | 6 | 152 |
| **Enable Traceability: Amazon CloudWatch** | Amazon CloudWatch is a monitoring solution for AWS services as well as applications running on AWS.   * Metric repository with alarms and automated actions. * CloudWatch Logs is the central repository to gain visibility into all logs.   CloudWatch events describes operational changes and performs automated actions. | 6 | 152 |
| **AWS CloudTrail** | AWS provides different security controls for CloudTrail such as: Data protection, Monitoring and Log aggregation.   * Logging of management plane and data plane events. * Management plane events are enabled by default and retained for 90 days. * **Data Protection:** encryption, lifecycle rules and integrity validation. * CloudWatch can monitor for specific API calls to detect unusual activities. * Aggregate log files from multiple accounts into one centralized monitoring account. | 6 | 154 |
| **CloudTrail** | AWS CloudTrail: logging of management plane and data plane events. Management plane events are enabled by default and retained for 90 days. Data Protection: encryption, lifecycle rules and integrity validation. CloudWatch can monitor for specific API calls to detect unusual activities. Aggregate log files from multiple accounts into one centralized monitoring account | 6 | 154 |
| **Enable Traceability: AWS CloudTrail** | AWS provides different security controls for CloudTrail such as: Data protection, Monitoring and Log aggregation.   * Logging of management plane and data plane events. * Management plane events are enabled by default and retained for 90 days. * **Data Protection:** encryption, lifecycle rules and integrity validation. * CloudWatch can monitor for specific API calls to detect unusual activities.   Aggregate log files from multiple accounts into one centralized monitoring account. | 6 | 154 |
| **AWS Config** | AWS Config is a service that continuously monitors and records service configurations in order to provide a view on configuration changes over time. | 6 | 156 |
| **Enable Traceability: AWS Config** | AWS Config is a service that continuously monitors and records service configurations in order to provide a view on configuration changes over time. EX:   * Resource administration: Detects resource misconfigurations. * Compliance and auditing: Having a view into historical resource to ensure you are compliant with your best practices. * Troubleshoot configuration changes: Identifies unintended configurations changes. * Security analysis: Gives a view on IAM policies assigned to roles, users, and groups and EC2 security groups to analyses potential sec. weaknesses. * Centralized view: Aggregate configuration and compliance data from multiple accounts and regions into one centralized account to help monitor. | 6 | 156 |
| **AWS GuardDuty** | * Monitors, analyzes, and processes different log sources * Identifies unexpected or malicious activity * Detects exposed credentials, privilege escalations | 6 | 158 |
| **AWS Security Hub** | * Gives a comprehensive view of the security state * Performs checks against industry standards and best practices * Collects and aggregates findings from security services | 6 | 158 |
| **Enable Traceability: GuardDuty and Security Hub** |  | 6 | 158 |
| **GuardDuty (AWS)** | * Monitors, analyzes, and processes different log sources * Identifies unexpected or malicious activity * Detects exposed credentials, privilege escalations | 6 | 158 |
| **Security Hub (AWS)** | * Gives a comprehensive view of the security state * Performs checks against industry standards and best practices * Collects and aggregates findings from security services | 6 | 158 |
| **Availability Zones** | Regions are composed of multiple Availability Zones (AZ), where an AZ represents an isolated and physically separate data center. Each AZ has its own redundant power, networking, and connectivity which aids in making the AWS Infrastructure more fault tolerant with high availability. | 6 | 159 |
| **AWS Global Infrastructure** | The AWS global infrastructure is composed of multiple layers. 24 Regions. 77 Availability Zones and 5 Local Zones. Data Centers. 400+ Point of Presences | 6 | 159 |
| **Local Zone** | A relatively new concept in AWS is a Local Zone. A Local Zone is an extension of an AWS Region that places AWS services closer to large population centers. This can help provide AWS services with lower latency to the services' end-users. | 6 | 159 |
| **Point of Presence** | Finally, there is a Point of Presence. A Point of Presence is connected to the AWS network to help provide cache capabilities for services such as Amazon CloudFront (content delivery network - CDN) and Route 53 (cloud Domain Name System - DNS). Amazon CloudFront will be discussed in greater detail shortly | 6 | 159 |
| **Regions** | Regions represent physical locations situated all around the world. Regions are isolated from each other. Each region provides fully redundant networking connectivity. As of this writing, there are 24 regions globally, with 8 more in construction. Regions can be found in North America, South America, Europe, Asia Pacific, and the Middle East | 6 | 159 |
| **Availability Zone (AZ)** | Multiple Availability Zones in case of a natural disaster or other outage. Is it enough? Sometimes no:   * In November 2020, AWS services were impacted in the Northern Virginia region * Happened a few times in last few years. Cross-region deployment and replication can provide further resiliency | 6 | 160 |
| **High Availability** |  | 6 | 160 |
| **Networking - DiD (1)** | Networking defense-in-depth to protect workloads against unauthorized and unintended access.   * Implement isolation and boundaries. * Separate public and private workloads. * Control Network traffic at all layers. | 6 | 161 |
| **AWS WAF** | * HTTP-based: AWS WAF * AWS Network Firewall to inspect inbound and outbound traffic * Transit Gateway to centralize traffic * One Inspection VPC with a firewall | 6 | 162 |
| **Networking - DiD (2** | How to implement traffic filtering and inspection at all layers in a centralized approach?   * HTTP-based: AWS WAF * AWS Network Firewall to inspect inbound and outbound traffic * Transit Gateway to centralize traffic * One Inspection VPC with a firewall | 6 | 162 |
| **AWS Network Firewall** | AWS Network Firewall is a highly available, managed firewall service that protects VPCs with stateful inspection, intrusion prevention and detection, and even some web filtering to block unauthorized or malicious traffic. Furthermore, AWS Network Firewall can integrate with other AWS services to store logs and can also integrate with products from AWS Partners to further integrate into central third-party policy orchestration and analytics solutions. This includes integration for the sending of logs and security event information to a SIEM. | 6 | 164 |
| **AWS Shield** | AWS Shield is a managed DDoS protection service that safeguards applications running on AWS. 2 tiers: Standard, Advanced   * Standard tier: Always -on network flow monitoring with inspection of incoming traffic to AWS services. * Advanced: Offers additional detection and mitigation against larger and more sophisticated DDoS attacks. | 6 | 165 |
| **AWS Web Application Firewall (WAF)** | AWS Web Application Firewall protects applications and APIs against the most common web exploits that can affect availability, security, or the consumption of excessive amounts of resources | 6 | 165 |
| **Amazon CloudFront** | Security controls to ensure Web application security with amazon CloudFront.   * Is a Content Delivery Network (CDN) capability which can be used to deliver web content to customers. * AWS WAF and AWS Shield to protect against web exploit and DDoS attacks. * HTTPS enforcement and field-level encryption to encrypt traffic and sensitive data. * Logging and security monitoring to detect and respond to (security) incidents. | 6 | 166 |
| **CloudFront** | Amazon CloudFront is a Content Delivery Network (CDN) capability which can be used to deliver web content to customers. But it is also much more than CDN. It can integrate with the various other technologies and capabilities from Amazon that we have discussed, on top of its own built-in mechanisms for the security of web applications and communications. | 6 | 166 |
| **Content Delivery Network (CDN)** | A Content delivery network is a distributed group of servers that caches content near end users. | 6 | 166 |
| **HTTP Flood Attack** | Utilizes what appear to be legitimate HTTP GET or POST requests to attack a web server or application. | 6 | 167 |
| **Automate Security Best Practices**  **AWS CloudFormation** | * Define the infrastructure as code and automate security controls. * Provision AWS resources quickly and consistently. * Handles the provisioning and the configuration. * Implements security controls and best practices in templates. * AWS CloudFormation registry for extension. | 6 | 168 |
| **AWS CloudFormation** | Define the infrastructure as code and automate security controls. Provision AWS resources quickly and consistently. Handles the provisioning and the configuration. Implement security controls and best practices in templates. AWS CloudFormation registry for extensions (versioning/monitoring). | 6 | 168 |
| **Amazon Macie** | Is a data security service that discovers sensitive data using machine learning and pattern matching, provides visibility into data security risks, and enables automated protection against those risks. | 6 | 169 |
| **Data at Rest** | Is data while in non-volatile storage such as block storage (S3 buckets) or in databases.  Protection of data at rest can be enforces through encryption and tokenization. | 6 | 172 |
| **Lifecycle strategies** | Should be based on the sensitivity level and other consideration such as legal and organizational requirements. The goal is to determine how to data is going to be managed over time. | 6 | 169 |
| **Data Classification** | * Identify the type and the classification of data * Map security controls and levels of protection to the sensitivity levels * Classification done through tagging in AWS * Define data lifecycle * Automate sensitive information discovery, classification, and protection with Amazon Macie | 6 | 169 |
| **Data in Transit** | Refers to any data moving from one system, to another, including communication between resources or between services and end users. The goal here is to ensure we are protecting for data confidentiality and data integrity. | 6 | 171 |
| **Ensure Protection:**  **Data in transit** | * Protect data integrity and confidentiality when sent from one system to another. * User encryption keys or TLS certificated and rotate them on appropriate tie intervals. * AWS certificate manager to provision, manage, renew, and deploy certificates. * Enforce encryption in transit in AWS services (VPC security groups, HTTP to HTTPS redirection) | 6 | 171 |
| **AWS Certificate Manager** | Can integrate with many different AWS resources such as Elastic Load Balancers, CloudFront distributions, and API Gateway. | 6 | 171 |
| **Data in Transit / at Rest - Protection** | Here, we will focus on the fact that encryption keys and certificates should be rotated at appropriate time intervals. | 6 | 171 |
| **Cryptographic Keys** | "Cryptographic keys play an important part in the operation of cryptography. These keys are analogous to the combination of a safe. If a safe combination is known to an adversary, the strongest safe provides no security against penetration. The proper management of cryptographic keys is essential to the effective use of cryptography for security. Poor key management may easily compromise strong algorithms." - The National Institute of Standards and Technology (NIST) in Special Publication 800-57 part 1, rev. 5 | 6 | 172 |
| **Data in Transit and at Rest: Ensure Protection (2)** | * Tokenization: use tokens to represent sensitive pieces of information * Encryption at rest: make the data unreadable without the encryption key * AWS KMS and AWS CloudHSM | 6 | 172 |
| **Tokenization** | use tokens to represent sensitive pieces of information | 6 | 172 |
| **AWS Key Management Service (KMS)** | * AWS Key Management Service (KMS) allows customers to centrally manage their cryptographic keys (and associated permissions) for more than 70 different AWS services * Create, store and control keys to encrypt or sign data. * Support for customer managed, AWS managed, and AWS owned keys. * AWS CloudHSM to generate and manage our own cryptographic keys for use in custom apps. | 6 | 173 |
| **Customer Master Keys (CMKs)** | A CMK is a logical representation of a master key and its related metadata. CMKs can take the form of both symmetric and asymmetric keys. 3 kinds:   * Customer Managed CMKs * AWS Managed CMK * AWS Owned CMK | 6 | 173 |
| **Elliptic Curve (ECC)** |  | 6 | 173 |
| **Key Management Service** | Allows customers to centrally manger their cryptographic keys (and associated permissions) for more than 70 different AWS services. | 6 | 173 |
| **Keep People Away from Data** | * Use managed service to reduce the burden of administrative tasks and automate them * Implement software integrity validations with code signing certificates and hash values. * Automate compute protection to reduce human errors and shift the focus to other security aspects. | 6 | 175 |
| **AWS Incident Response** | Incident response in AWS is divided into phases:   1. Educate: Learn cloud technologies. Refine tools and techniques. 2. Prepare: Document and test incident response capabilities 3. Simulate: SIRS to train incident responders and validate procedures and automation. 4. Iterate: Create a code- based solution to automate response. Trigger an AWS Lambda function | 6 | 176 |
| **Security Events (1) - Prepare** | Processes to respond to and mitigate the impact of security incidents should be defined even if there already are appropriate preventive and detective controls in place  Incident response in AWS is divided into 4 phases.   1. Educate: learn cloud technologies. Refine tools and techniques. 2. Prepare: Document and test incident response capabilities. 3. Simulate: SIRS to train incident responders and validate procedures and automation. 4. Iterate: create a code-based solution to automate response. Trigger an AWS lambda function. | 6 | 176  178 |
| **macOS - What is macOS?** | * macOS, previously known as Mac OS X, is the traditional operating system developed by Apple. * This operating system is built up on top of the XNU kernel, with standard Unix facilities available from the command line interface. * Apple disk image files (\*.dmg). * Development environment: XCODE. * Darwin subsystem uses Linux-like file permissions | 6 | 182 |
| **XNU Kernel** | Is the computer operating system (OS) kernel developed at Apple Inc. since December 1996 for use in the Mac OS X (now macOS) operating system and released as free and open-source software as part of the Darwin OS, | 6 | 182 |
| **Automatic strong passwords - macOS** | Automatic strong passwords are a feature that suggests randomly generated strong passwords when signing up for a new service | 6 | 183 |
| **Keychain** | Keychain is Apple's built-in password and credential manager | 6 | 183 |
| **macOS Security Features (1)** | Privacy Controls, Keychain, Automatic Strong Passwords | 6 | 183 |
| **Privacy Controls - macOS** | * Privacy controls: allow the user to restrict application access to system and personal resources. * Keychain: Is Apple’s built-in password and credential manager. * Automatic strong passwords: Is a feature that suggest randomly generated strong passwords when signing up for a new service. | 6 | 183 |
| **Anti-Phishing & Download Protection - macOS** | The included Safari web browser has built-in anti-phishing and download protection (anti-malware) features | 6 | 185 |
| **Automatic Updates** | Every macOS device can be configured to receive automatic updates | 6 | 185 |
| **Gatekeeper - macOS** | Gatekeeper is a feature that verifies the validity of applications prior to execution | 6 | 185 |
| **macOS Security Features (2)** | Automatic Updates, Gatekeeper, Anti-Phishing & Download Protection | 6 | 185 |
| **Find My - macOS** | Find My is the name of the lost/stolen device service from Apple. It requires the user to register their device using an iCloud account | 6 | 187 |
| **macOS Security Features (3)** | XProtect, Find My, Firewall | 6 | 187 |
| **XProtect - macOS** | XProtect is an anti-malware module working closely with the Gatekeeper functionality | 6 | 187 |
| **Firewall - macOS** | Every macOS has a basic system firewall installed which allows users to block incoming connections. It is disabled by default. | 6 | 188 |
| **File Vault - macOS** | FileVault is the tool that enables disk encryption on macOS devices  Protection with disk encryption in case system is stolen | 6 | 189 |
| **macOS Security Features (4)** | File Vault, Sandboxing & Runtime Protection, Security Chip | 6 | 189 |
| **Sandboxing & Runtime Protection - macOS** | Sandboxing and runtime protection features protect macOS from malicious executables (Mac App Store apps only) | 6 | 189 |
| **Security Chip - macOS** | Recent macOS devices are shipped with a Security Chip that can execute sensitive operations in a protected environment (protected enclave) | 6 | 189 |
| **eficheck - macOS** | Runs once a week and checks Mac’s firmware against **Apple's** database of what is known to be good. | 6 | 190 |
| **macOS - Securing** | * Turn off unneeded services. * Turn on firewall to control access to services. * Limit service sharing. Carefully monitor access lists. * Set up secure file sharing. * Use password assistant for stronger authentication | 6 | 191 |
| **BuggyCow** | * BuggyCow is a privilege escalation vulnerability caused by a bug in XNU's memory management code * Discovered by Google's Project Zero in 2018 * Unprivileged processes can change shared data used by privileged processes without notifying the file management system | 6 | 192 |
| **GateKeeper Bypass** | GateKeeper Bypass is a vulnerability in macOS that allows attackers to run untrusted applications (without being verified by Gatekeeper and Xprotect) • Symlinks in ZIP archives are not checked by the default • Accessing a specific path result in the automatic mounting of that location • The attacker simply has to trick the user into opening the zipped symlink and run an application (Gatekeeper trusts all network shares) | 6 | 192 |
| **macOS - Vulnerabilities in** | BuggyCow, GateKeeper Bypass | 6 | 192 |
| **CrescentCore** | * CrescentCore is another trojan that disguises itself as a Flash Player installer * Discovered in June 2019 * Tries to detect virtual environments and anti-virus software * Upon installation, user infected by a backdoor | 6 | 193 |
| **FlashBack** | * FlashBack is a trojan that presents itself as a Flash Player installer * Upon installation, the user would be infected by a backdoor that connected to a botnet • Later spread on websites that exploited a vulnerability in Java * In 2012, it infected more than 600,000 Mac devices worldwide | 6 | 193 |
| **macOS - Malware on** | FlashBack, CrescentCore | 6 | 193 |
| **Log Analytics Workspaces** |  | 6 | 230 |