**Index for SANS GSEC Certification**

Humam Al-Shami

**Defensible Network Architecture:**

Stage 1: Conceptual design / closed-box diagramming (understand objectives & use of the network): B1, 17

Stage 2: Logical design (put design on paper-network diagram): B1, 19

Stage 3: Physical design (detailed design of network): B1, 20

Stage 4: Know where data is (supposed to be & not supposed to be): B1, 22

Stage 5: Understand communication flow: B1, 24

DLP (Data Loss Prevention) – Security control: B1, 24

Networks under attack (vulnerable routers & switches): B1, 27

Threat Enumeration (Tactics, Techniques, Procedures-of adversaries): B1, 29

Threat Agents (Who: opportunistic, organized, Advanced persistent): B1, 30-31

Attacks against routers (DOS, DDOS, sniffing, packet misrouting, routing table poisoning): B1, 32-33

Attacks against switches (CDP info disclosure – sniffing, MAC Flooding – downgrade attack, DHCP Manipulation – MiTM, STP Manipulation – switch reconfig, VLAN Hopping – Network segmentation bypass): B1, 34-36

**Network Design:**

Physical vs Logical Topologies: B1, 38-39

Approaches to network design (segmentation – VLAN, IPsec, firewall, SDN, air-gaps): B1, 30

SDN (Software designed networking) benefits with segmentation & Not Petya: B1, 41

VLAN (virtual separate LAN) – used for ease of management, not security (unless ACL used): B1, 42

802.1 X (authenticates user before giving access – wired & wireless – uses EAPOL): B1, 43

Network Design objectives 1 (web-apps = vulnerable, tiered architecture for segmentation, DiD): B1, 44

Network Design Objectives 2 (further breakdown of tiers – Public, semi-public DMZ), Middle (Proxy), Private): B1, 46

Network Design Objectives 3 (Key rules to manage network tiers + How to achieve rules): B1, 47

Final Design (Visual of network design): B1, 48

Benefits of Network Architecture understanding: (B1, 49)

**Protocol & Packet Analysis**

Protocol Stacks (purpose & general): B1, 55

ISO OSI Protocol Stack & layers: B1, 56-57

Protocol Stack Communication (How, headers, IP spoofing, encapsulation, decapsulation): B1, 58-59

OSI vs TCP/IP protocol (OSI more granular – more commonly referenced verbally. TCP/IP more used in real-world): B1, 60

TCP/IP Packet Generation procedure (what happens at each layer): B1, 62

IP (Internet Protocol) Basics – assist in determination of best route for communication & defines formation of IP addresses based on network settings. B1, 64

IPv4 (based on ARPANET) vs IPv6 – IPV6 is “better” since it accommodates more unique addresses; more space = more flexible deployment of addresses according to geographical regions + IPv6 has encryption integrated in protocol & authentication of endpoints (IPsec) + QoS features that give real-time sensitive applications priority over less critical streams: B1, 65-67

IPv4 header & key fields (Version – v4 or v6, Protocol – TCP/UDP/ICMP, TTL – hops before expiration, Fragmentation – how to reassemble big packet, Address & Destination, Options – indicate malicious activity): B1, 68-70

IPv6 header & key fields (version -v4 or v6, Traffic Class – QoS importance, Flow Label – QoS cluster packets together, Payload length – in bytes, Next Header – protocol field of IPv4, Hop Limit – TTL of IPv4, Source Address & Destination): B1, 71-72

IPV6 Features (Extra space, address auto-configuration using MAC, IPv4 co-existing with tunneling & translation, “options” in “Next Header” field not in header): B1, 73-74

ICMP Intro (report error conditions, network troubleshooting, ping & traceroute, not transfer data – supports IP & UDP): B1, 76

ICMP Header (Type – usually of error, Code – more detail of error, Checksum – validation): B1, 77-78

ICMP Common types & Codes (Type 0 & 8 – ping, Type 3 – Destination unreachable, Type 11 – TTL expired): B1, 79-80

TCP & its uses (determine if successful connection & sets rules for communication – key features are flow control, data receipt validation, & ease of use for devs) + Key Ports: B1, 82-84

TCP Connection (SYN, SYN/ACK, ACK Handshake – exchange of Acks allows for confirmation of delivery. Acks are piggy-backed onto packets sent to minimize volume of traffic): B1, 85-86

TCP Header (Source & Destination Port, Sequence Number – ISN [generated & sent in SYN] or accumulated sequence number, Ack number – next expected sequence number, Data Offset – length of the header, TCP Flags – type of packet [SYN, ACK, FIN, etc.], Window Size – max data to send back [flow control], Options): B1, 87-88

Closing a TCP session (Graceful – FIN/ACK & Abrupt RST/ACK): B1, 89

UDP intro & uses + common ports (efficient protocol without checking – used when re-sending data is unhelpful: calls, DNS, & more protocols): B1, 93-94

UDP Header (Source & Destination Port – server vs client ports, UDP Length, UDP Checksum): B1, 95

Sniffing & Sniffers + their use cases (need NIC in promiscuous mode, legit & non-legit uses, basic info on TCPDump, Wireshark, Snort, Kismet, BetterCAP): B1, 98-100

TCPDump analysis commands (tcpdump xprotocol – only dump xprotocol packets, dst port x – dump packets destined for port x, host x.com – dump packets to or from x.com): B1, 101

**Virtualization**Intro (Is Abstraction. Hypervisor is software, creates separation & abstraction of physical and virtual hardware. Allows for VMS because each OS Kernel has its own “hardware”): B1, 108 -109

Benefits (Isolation – reduces scope of damage from compromise & enables malware analysis & disaster recovery: B1, 111 + application-level abstraction: B1, 114 + Virtual Desktop Infrastructure [VDI] – allows for fast recovery + Malware [not perfect] & Forensic Analysis: B1, 117): B1, 111-117

Risks (Hypervisor is software, software has vulnerabilities. If exploited = very bad + Isolation Violation due to vulnerabilities – leads to “VM Escape”): B1, 118 & 119

Virtual Environment Defense (Guest OS minimal footprint, disable unnecessary services, check permissions, Hypervisors minimal footprint, disable unnecessary features, patches installed fast, host = same security considerations. + VMs can subvert network-based security devices. Need virtual versions of these security devices – virtual intrusion detection systems, firewalls, etc.): B1, 120

**Cloud**

Intro (Cloud = virtual hard drive/abstraction of storage): B1, 122

Types of Cloud Deployment (Public, Private, Hybrid – tradeoffs are control & visibility of data): B1, 123

Why Public Cloud (IT infra is costly, public can be rented): B1, 124

Benefits of Cloud Computing (Cost – variable cost according to need, Speed – can get access to resources instantly, Scale – can expand globally easily, Productivity – on-demand resources, Security – almost no vulnerabilities with major cloud providers): B1, 125-126

Types of Cloud Services (IaaS – rent servers & storage, PaaS – rent platform to host your code, no need to manage infra, SaaS – rent service like email access): B1, 127

Serverless Computing -Function as a Service - FaaS (Don’t rent hardware - you only pay when a function is called, ephemeral environment – code runs then is deleted [much easier for cleanup & maintenance], full server management by third party – saves time): B1, 128

Infrastructure as Code – IaC (create infrastructure from templates pre-setup with controls & checks – infra can be re-built every few hours for safety + constant state at large scale) & Continuous Integration / Continuous Delivery (CI/CD): B1, 130

Market Data on Growth & Market Share of Cloud (Amazon AWS is leader largely): B1, 131-132

Major Cloud Providers (Amazon, Microsoft, Google) – Strengths & weaknesses: B1, 133-134 + Services offered by each: B1, 135

Shared Responsibility concept (both cloud provider & subscriber are responsible for security, subscriber is liable if data breach occurs on cloud since they chose to place data on cloud, SLA can help set contract for responsibilities, CSA & NIST have additional guidance): B1, 137-138

AWS Shared Responsibility Model (AWS responsible for security OF the cloud, client responsible for security IN the cloud): B1, 139

Microsoft Azure Shared Responsibility Model: B1, 140

Major Security & Privacy Frameworks (having these tells you the CSP is trusted): B1, 141

Cloud Security Guidance (CSA) – CAIQ (Consensus Assessments Initiative Questionnaire), CCM (Cloud Controls Matrix): B1, 142-143

CSA Guidance Domain Matrix (each of the domains covered by the CSA guides): B1, 144

AWS Regions & Availability Zones (Regions are physical locations – separate from each over, Zones are data centers in a region connected to each other): B1, 146

Virtual Private Cloud – VPC (network inside AWS account, limited to single region, similar to on-premises data centers & contain route tables, gateways, net interfaces, subnets, & CIDR + others): B1, 147

Internet Gateway – IGW (connects VPC resources to the internet) + Network access control lists – NACL (secure traffic flow throughout cloud infra): B1, 148

Network Address Translation (NAT) Gateway (enables instances in private subnets to access the Internet): B1, 150

Subnet Network Access Control Lists – NACLs (control traffic flow using list of rules): B1, 151

Elastic Compute Cloud – EC2 (provides cloud virtual machine, configurable specs + AMI, AWS key pairs, security groups): B1, 152

Amazon Machine Images – AMIs (templates for EC2 instances, some free, some paid, downloaded from Amazon Marketplace): B1, 153

Management Subnet – place to host Bastion & RDP gateways (used to proxy SSH & Remote Desktop connections to other devices in the VPC – reduces attack surface & allows close monitoring of connections in one place): B1, 155

Serverless Security Benefits (attack surface is smaller externally, no servers to patch, if malware - next request brings up fresh server, harder to exfiltrate data): B1, 156

Serverless Security Concerns (attack surface is larger internally – anyone can deploy functions, makes it difficult to track & delete functions. Must have appropriate authentication controls, limit functionality given to each function, identify functions that handle sensitive data & ensure serverless service is compliant before storing sensitive data): B1, 157

Serverless & Application Security practices (Serverless = attack focus is on application. Input Validation/sanitization, Code review/testing, check libraries & dependencies): B1, 158

Cloud-Native Security Services (services offered by major CSPs): B1, 159

**Wireless Technologies**

IEE802.11 (Standards proposed by IEEE that enable LAN, WLAN, & MAN. Evolves through amendments that add capabilities or security features – e.g. 802.11ac. lock = encrypted network): B1, 171-173

IEE802.11 amendments (802.11b, 802.11a, 802.11g, 802.11n, 802.11ac, 802.11ax & new Wi-Fi Alliance “Generational Names”): B1, 174-183

IEE802.11 WEP (Wired Equivalent Privacy) & It’s security issues (deprecated in 2004): B1, 187-189

WPA - WEP replacement (based on 802.11i, *WPA1* – software upgrade to WEP using TKIP & deprecated 2009, *WPA2* – hardware upgrade using AES 128bit keys, *WPA3* – hardware upgrade using PSK/pre-shared keys & SAE: Dragonfly Handshake + cancel offline password guessing, encryption even on unsecure WLAN): B1, 191 &194-203

WPA2 & WPA3 vulnerabilities (WPA2 – KRACK research: key reinstallation attack, WPA3 – Dragon blood research: downgrade to WPA2 & weak ECDH attack, timing attack for offline password guessing & DoS): B1, 197-198 & 204-206

802.11 Rogue Access Points (initially – unauthorized access point connected for wireless access, today – masquerading evil twin AP with same name as legit AP: Both are bad): B1, 208-209

802.11 Denial of Service – DoS (“Jamming” most commonly perceived DoS – radio signal to interfere with our waves. Too many DoS methods, can’t stop all. To mitigate, respond quickly – find source, react fast. If can’t, fall back to wired before severe damage): B1, 210-211

PAN – personal network (Personal ≠ Secure, Bluetooth, NFC, Zigbee, RFID, based on IEE802.15): B1, 213

PAN: Bluetooth (not secure, evolved to have higher range, almost always on – very bad for security): B1, 215-216

Bluetooth vulnerabilities & protections (BlueBorne – remote code execution & info disclosure vulnerabilities, Billions of impacted devices. To mitigate: use latest version of BT, disable unnecessary BT profiles, don’t leave BT on, install patches, use Central Management systems): B1, 217-220

PAN: Zigbee (low-cost & low-power PAN alternative, runs on battery, collection of standards & technologies, security included in standards – vendor implementation optional): B1, 221-222

PAN: NFC – Near Field Communications (close proximity – 1 to 2 inches, NFC Forum in charge, many use cases – mandating security difficult, NFC Forum standards allow flexible security implementation, security often determined by vendor and industry): B1, 223

PAN: RFID – Radio Frequency Identification (unique identification of object with location & movement, security varies widely, tags can be read by anyone, cloning of tags is possible – thus identifying humans with RFID is controversial + RFID tags can be smaller than grain of dust): B1, 226

5G (next evolution of cellular communications – sub 1ms latency, higher bandwidth - multi gigabit, multiclient support – more devices connected, full implementation will take years, ITU – International Telecommunication Union mainly + 3GPP): B1, 228 – 229

IoT – Internet of Things (represent non-traditional devices on networks, danger comes when we forget they are computers – require same security as desktops & servers, important to know what devices are connected to your network to maintain them, can leverage any of wireless – Wi-Fi, Bluetooth, Zigbee, 5G, etc.): B1, 231-232

**Defense-in-Depth**

What is DiD (Any layer of protection can fail, So: Multiple levels of protection tied together. Preventative measures, Detective measures, Deterrent measures) + types of security (Perimeter, Network, Host, Application, Data security): B2, 5-6

Risk (Threats x Vulnerabilities. Focus: Confidentiality, Integrity, Availability - CIA) + Risk Balance – more functionality = less security: B2, 7-10 & 12

Filtering – foundation of DiD (Network Filtering – Firewalls, Anti-DDoS, Proxy, Mail relay & Host filtering – Anti-malware, Application control): B2, 14-15

Approaches to DiD – deploy measures to reduce, accept or transfer risk (Uniform protection, Protected Enclaves, Information-Centric, Vector-Oriented): B1, 16

Uniform Protection DiD (all parts of the organization receive equal protection – most common basic approach & weakest approach): B2, 17

Protected Enclaves DiD (Assets that require additional protection are segmented using VLANs to segment & (N)ACLs to restrict traffic between segments): B2, 18

Information-Centric DiD (Multiple layers of protection around confidential information): B2, 19

Vector-Oriented DiD (prevent threat from using vector to exploit vulnerability – remove or mitigate vector that threat needs to use): B2, 20

System Security Layer (configuration hardening to protect system layer & thus network layer): B2, 21-22

Configuration Management (establish baseline, change control to detect change from baseline, partition internal network – only accurate baseline can detect attacks with no known signatures): B2, 24

Recovering from virus (don’t trust again – after system analysis, reload don’t clean): B2, 25

Defense-in-Depth in the Cloud – easier! (Firewalls, Web Application Firewall – WAP, IPS/IDS, Email protection, VPN, Identity and Access Management – IAM, Centralized logging, Container security): B2, 26-29

Identity Access Management – IAM (on-premise: 30, Azure: 31, AWS: 32-33, GCP: 34): B2, 30-34

Network Segmentation (on-premise: 35, Azure: 36, AWS: 37, GCP: 38): B2, 35-38

Zero-Trust (every request must be authenticated & all must be encrypted, assume threat from inside & out, complements DiD) – Intro & Basic principles (Internal & external threats always present, every connection must be proven, inspect all network traffic): B2, 39-40

Zero-Trust – securing traffic (all traffic must be *Authenticated & Encrypted* + NIST standards*):* B2, 41

Zero-Trust – Variable Trust (dynamic change of access based on conditions – type of access requested, location, device, type of application. Point system): B2, 42-43

Zero-Trust – Log Inspection (logging is essential for zero-trust – logging can give valuable info to do analysis & create baseline for zero-trust model): B2, 44

Defense-in-Depth Summary (Security = understanding, managing, mitigating risk | risk management focus = confidentiality, integrity, availability. All based on filtering): B2, 45

**Identity & Access Management**

Definitions (Digital Identity & Identification, Authentication – relies on something you know or you have or something you are/biometrics, Authorization – what allowed to do after authentication, Accountability – who did what & when): B2, 48-49

Enrollment Process (becoming a subscriber: Identity proofing, Identity Assurance Levels, Credentials): B2, 51

Identity Proofing – proving who you claim to be (Resolution – Evidence of identity collected, Validation – Authenticity of Evidence checked, Verification – contents of evidence checked): B2, 52

Identity Assurance Levels (NIST outlines: IAL 1 – self-claimed, IAL 2 – Evidence based, IAL 3 – Physically check): B2, 53-54

Authentication – subject proves they have authenticators (subject = claimant, checker = verifier): B2, 55

Authenticator Assurance Levels – AAL (NIST outlines: AAL 1 – Single factor min, AAL 2 – any two factors & strong crypto, AAL 3 – selected two factors & strong crypto): B2, 56

Single Sign On – SSO (allows user to authenticate once with creds to access resources; does *not* mean single authentication – Centralized User Management, Directory based, Ticket based): B2, 57

SSO Protocols (SAML 2.0 – XML based & uses tokens + HTTP redirect binding to identity provider: 59; OAuth 2.0 – open standard 4 access delegation/granting websites access to their information on other sites without exchanging passwords – secure & supports SSO: 60): B2, 59-60

Controlling Access (least privilege – give only access they need, Need to Know – only give access for time it’s needed, Separation of Duties – split tasks across people / 2 needed to complete an action, Rotation of Duties – change jobs regularly to prevent getting comfortable & learning to cover their tracks & reduce chance of collaborating for bad in separation of duties): B2, 62-63

Access Control Techniques (Discretionary Access Control – access to certain resources via credential which can be shared & Mandatory Access Control – access to all resources via system-enforced credentials which cannot be shared): B2, 64

Access Control Techniques 2 (Role Based Access Control – discretionary or mandatory access control assigns roles based on organizational function, Lattice Based Access Control – mandatory access control that restricts access on interactions between subjects and objects. Subject only allowed access if security level of subject >= object): B2, 65

Managing & Monitoring Access (Account administration – enroll user+ assign authenticators & authorizations, Maintenance – reviewing data & checking for errors, Monitoring – successful & failed attempts to log on, Revocation – revoke privileges when no longer needed): B2, 66-67

Privileged access – admin access (privileged acc targeted & has high impact if breached): B2, 68

Privileged Access Management tools (regulate who can get privileged access to critical systems. Features: easier for user, Policy enforcement – allows strict rules on access & password policy, stronger passwords – auto generates strong password, Store passwords – acts as password vault, Rotate creds – auto change pass, Monitor logs – auto logs access to privileged accounts for review, Generate reports – auto audit reports of access to privileged accounts): B2, 69-70

Common Privileged Access Management Tools (CyberArk, HashiCorp, Azure Active Directory): B2, 71

Azure Privileged Identity Management (Features: least privilege, just-in-time privileged access, time bound access, approval to activate privileged role, multi-factor authentication, justification to understand why users activate, notifications when privileged role activated, conduct access reviews to ensure access needed, audit history): B2, 72

Tiered-Privileged Access Management (local admin sign-in = easy attacker lateral movement. To prevent: disable login with admin, keep privileged groups in Active Directory empty, restrict use of privileged AD accounts, Structure Active Directory to use tiered Admin model to prevent pass the hash attack: Tier 0/control of enterprise identities, Tier 1/control of enterprise servers & apps, Tier 2/control of user workstations & devices): B2, 73-74

Tiered Privileged Access Management 2 (separate admin accounts for Active Directory & Crown Jewels – Tier 0, Exchange Servers & Intranet Servers – Teir 1, User Workstations & Printers & Mobile Devices – Tier 2): B2, 75

**Authentication & Password Security**

Authentication Types (something you know – password, something you have – look up secrets / out of band devise / one time password tokens / cryptographic devices, something you are – biometrics): 78-80

Storing Passwords (must be hashed, not stored in clear text): B2, 81

Key Derivation Functions (based on irreversible hash function, capable of input transformation / key stretching, no two stored values are the same because of salt, difficulty factor to discourage cracking): B2, 82

Strength of Password Hash determinants (Key Derivation Function Quality, Character Set Support, Password & Derived Key Length, Difficulty Factor): B2, 84-85

PBKDK2 (generated Derived Key based on: Pseudo-Random Function for hashing like SHA, password, salt, cost factor – amount of times password will be hashed / at least 10000, desired length of derived key in bits): B2, 86

MD5 (hexadecimal hash, hash collisions problems, used in software world, usually 32-character alphanumeric string): B2, 87

SHA – Secure Hash Algorithm (SHA, SHA1, SHA2, SHA3, used in online databases mostly): B2, 88

Hash Identifier (online tool to try to identify hash type): B2, 88

LM/NLTM (used by Microsoft, LM weak for many, NTLM better but NTLMV2 is better): B2, 89-90

NTLMV2 (used when Kerberos unavailable for no service name, two-party authentication, three steps: Negotiate – Challenge – Response | + Nonce used): B2, 91

Password Dumps (cracking requires list of hashed passwords obtained from dumps from breaches – Collection #1 2019 & LinkedIn 2012): B2, 92

Password Cracking Approach (obtain hashed password list, determine hash/key derivation function, create possible password guesses, calculate hash for each function, match with collected hashes): B2, 93-94

Password Cracking Attacks (Brute Force – slowest but guaranteed, Dictionary/Wordlist – Fastest but only effective against weak passes, Hybrid attack – Extends dictionary attach with numbers & symbols, Pre-Computation attack - hashes for attach pre-computed and stored in “rainbow tables”: increases crack speed but disabled with salting): B2, 96-97

Speed up Cracking (parallelism with GPU or “rainbow tables”): B2, 98

Strong Password Policy (Do: > 8 length, check for recognizable words or sequences, block after x failed attempts, force change if breach. Don’t: truncate passes, allow hints, force composition rules, force periodic password changes, save passwords in clear text): B2, 99-100

Fighting Pre-Computation Attacks (Salting – unique per password but not secret & Peppering – unique per application & secret): B2, 101

Hashcat (password cracking with hardware acceleration / parallelization): B2, 104

Mimikatz (post-exploitation tool to extract info: sekurlsa – extract creds from Isass, kerberos – create Golden tickets, crypto – interact with cryptographic system, Isadump – dump system data from LSA & SAM): B2, 106

Multi-Factor Authentication (access granted only with more than one indicator – combines shared secret with OTP token or biometrics. 1 of Strongest ways to defend from breach when pass leaked): B2, 108

Adaptive Authentication (part requesting access needs to provide more authentication according to context of request & sensitivity of resource): B2, 109

**Security Frameworks**

Overview (CIS controls – 20 controls to block attacks, NIST Cybersecurity framework – risk-based strategies, MITRE ATT&CK – adversary tactics based on real-world observations): B2, 114-115

CIS Principles (defenses should be automated, specific technical activities should be taken to produce consistent defense, root problems must be fixed to ensure prevention or timely detection, guidelines should facilitate common ground for measuring security effectiveness): B2, 117

Pillars of the Controls (Offense Informs dense, Prioritization, Measurements & metrics, continuous diagnostics & mitigation, automation): B2, 119

CIS Controls (Basic, Foundational, Organizational): B2, 120

CIS Control #2 – Inventory & Control of Software Assets (only authorized software should be installed & all software must be recorded): B2, 122

Core Evaluation Test (install unauthorized software & check how long to be detected): B2, 123

Critical Control #2 Measures (no. of unauthorized software present, how long to remove unauthorized apps, percentage of systems not running app control software, no. of software recently blocked from executing, time to detect now software, time to remove unauthorized software): B2, 124

NIST Framework (Framework Core – references & controls for critical infrastructure, Framework Implementation tiers – contextualize risk & how it can be managed, Framework Profiles – used as target & provides guidelines to strengthen security): B2, 125

NIST Framework Core (Identify, Protect, Detect, Respond, Recover | + Ransomware example steps): B2, 126

NIST Implementation Tiers (Tier 1 - Partial, Tier 2 – Risk Informed, Tier 3 - Repeatable, Tier 4 – Adaptive): B2, 128

NIST Framework Profiles (align functions, categories, and subcategories based on needs. Profile based on risk appetite & risk tolerance. Security controls based on profile): B2, 130

MITRE ATT&CK (knowledge base of adversary tactics & techniques based on real-world observations + Tactics describe high-level attacks. Assumes breach, first tactic is initial intrusion): B2, 131-133

Mapping to Known Adversaries (If know which attackers do what, can map their technique to campaign and look for specific details in your environment): B2, 134

Other Frameworks (ASD, NIST 900-53, ISO 2700 1, GDPR, PCI, SOC 2 | All controls share similarities – defenses should focus on addressing most common and damaging attacks occurring today & in near future + Enterprise must ensure consistent controls across environment to negate attack): B2, 135-136

Summary – CIS vs NIST vs MITRE ATT&CK (CIS Technical & applicable – good first framework, NIST usually used in governance, MITRE ATT&CK – very technical usually used by security operations center): B2, 137

**Data Loss Prevention**

Data Loss vs Leakage (Loss = not readable, Leakage = accessed & shared unauthorized): B2, 141

Data Loss (Types, Cost, Prevention, Recover + Example: Ransomware): B2, 142-143

Prevention Strategies (Redundancy, Backups, Access Control): B2, 144

Redundancy (In-House – ECC memory, RAID, data redundant file systems: btrfs & zfs | Cloud – Availability, Integrity, Geo-replication): B2, 145-147

Backup Methods (System Imaging, incremental backups, differential backups, continuous backups): B2, 148

Backup common mistakes (DO: keep backups on separate networks, backup cloud data, use different technologies, verify backup systems and data frequently): B2, 149

Data Recovery (repair storage medium, Image data to another drive, logical data recovery, finally repair damaged data): B2, 150

Data Leakage (Types, Cost, Prevention, Insider Threats): B2, 151

Why Data Leakage Prevention Important (Business Losses, Reputational Damage, Fines): B2, 152

Regulations (GDPR – European & CCPA – Californian. GDPR Article 32 – security standards, Article 33 – notify supervisor of breach within 72 hours, Article 32 – notify people of breach clearly. CCPA 1798.150 – if breach due to organizational negligence: can sue for $100-$750): B2, 153-154

Breaches are expensive (British Airways $328mil fine, Equifax $575mil fine): B2, 155

Prevention Strategies (Secure Data Storage, Intrusion Detection, Exfiltration Detection): B2, 156

Storing Sensitive Data (must encrypt & delete if not needed – applies to digital & physical media): B2, 157

Selecting Data Loss Prevention Policies – aim to provide better way to protect digitally stored data in-transit & at-rest (policy defines where tool monitors, under which conditions should intervene, & which actions to take): B2, 158

DLP Classification Labels (data must be given labels to tell how sensitive it is & what security requirements needed + example DLP label configuration): B2, 159

DLP Policy for Data At-Rest (auto encryption of sensitive data, detect wrongly classified data & suggest other classification, detect sensitive data stored in wrong location & act): B2, 160

DLP Policy for Data In-Transit (Detect transmission with sensitive data & act, actions include escalate for approval, block, alert, or remove data): B2, 161

Data Loss Prevention Tools (Digital Guardian, Forcepoint, Office 365 DLP): B2, 162

Storing Data Offline (only online data can be stolen remotely. Store archived data offline): B2, 163

Intrusion Detection & Prevention (Network IDS – traffic of network: Online – real-time traffic analysis but basic detection rules. Offline – stored traffic analysis that is later but allows more thorough analysis): B2, 164 | Host IDS – activity of hosts logged combined with detection of malware signatures & snapshots of system state compared to previous states. Can detect suspicious system changes & attempts to circumvent security policies): B2, 164

Data Exfiltration (Step 1: search for interesting files – noisy. Step 2: gather data in one place & encrypt archive, Step 3: exfiltrate data – harder on segmented network): B2, 166

Defending against Exfiltration (Prevent: know what data you have & correctly classify, Limit user access, consider what you store where | Detect: system search is noisy, there will be false positives, access to network shares though can be monitored – look for repeated audit failures from one source): B2, 167

Insider Threats (Insiders pose big threat – 27% of e-crimes. Is a human issue – to mitigate: promote Insider Threat Program to deter, implement positive incentives, & put high risk people under increased monitoring): B2, 169-170

User Activity Monitoring – UAM (mandated by HIPAA, SOX, GDPR, + for auditing. Enables tracking of internal end-user activity & logs can be fed to ML to find anomalies & report): B2, 171

Digital Watermarking – to embed markers in data to track data source & ownership (Printer dots – non-visible & Document watermarks – visible & non-visible): B2, 172

**Mobile Device Security**

Mobile Operating System Market Share (Android no. 1, Apple no. 2): B2, 178

Android vs. iOS (Android – Open standard for modification, less locked down, recently integrated security. iOS – Closed software cannot be modified, locked down, focused on integrated security): B2, 179

Android Security (Android Oreo + significantly focused on security. More built-in security like iOS, still open OS but more embedded security, multiple layers of security out of box): B2, 180

Android Security Features (Applications tested before download, when apps run they are monitored for sus activity, Android Pay safe & doesn’t expose credit card, Critical data sandboxed or isolated, Device Manager allows wiping remote devices, Built-in encryption to protect data at-rest & in-transit): B2, 181

Android Key Info (Open Handset Alliance by google, Manufacturers modify OS, rapid hardware development thus wide disparity of hardware & software features. Fragmentation caused by competing hardware manufacturers who want profit, customization = incompatibilities, multi-level update cycle causes slow updates = more vulnerable time): B2, 182-183

Android Security Update Process (flaw disclosed to OHA, shared with vendors, vendors fix & send update to MOs, MOs decide if send update or not after testing): B2, 184

Apple iOS security (designed with security at core, closed model unless jailbroken, transparent to user, encryption that cannot be disabled): B2, 185

iOS Security Features (Hardware & Software security features, critical data protected at-rest if device stolen through encryption, network security provides security in-transit, apps run in secure environment): B2, 186

iOS Security Features 2 (Apple Pay – secure payments, cloud-services for backup & communication capabilities, Device Control for management including secure wipe, Privacy Controls enable or disable sharing of information with external services): B2, 187

Key Facts about iOS (massively popular for enterprise & end-user, most restrictive platform, apple end-to-end ownership model, mobile operator / MO software forbidden, evolving hardware capabilities dictate software features): B2, 188

Mobile Problems & Opportunities (commonly lacking functionality needed for secure use, immature or hampered enterprise controls | Best reaction is not to ban but to sanctioned by org & under supervision to limit exposure & mitigate risks): B2, 189

Threats we care about (TH1 – Loss of control & visibility caused by bring-your-own, TH2 – always on through multiple interfaces = more opportunity for attack, TH3 – Slow patching & extended vulnerability periods on Android, TH4 – potential theft or loss): B2, 190

Stolen Device Threat (access device resources, extract data, access stored authentication credentials, setup backdoor | Must enforce policies, locate devices remotely, update frequently, secure communications, diagnosis & troubleshooting + must report lost device fast): B2, 191-192

Unlocking, Rooting, and Jailbreaking (bypassing device restrictions – needed as analysts & requested by users): B2, 193

Mobile Malware (growing threat, new opportunities for exploiting users & financial gain): B2, 194

User Credential Theft (Zitmo variant of ZeuS trojan – controls SMS & phone functionality to block calls & intercept SMS messages. Used to re-direct SMS & enable bypass mTAN authorization): B2, 195

Mobile Malware Delivery (official app stores – short lived, Third-party app stores, Malicious websites – direct download, direct victim targeting – attachment or URL): B2, 196

Android Malware (most common, platform accommodates silent SMS delivery, untrusted apps, third party app stores, easy for attackers to repackage legitimate apps with malware + platform fragmentation makes exploits last longer since slow updates): B2, 197

iOS Malware (prevents unauthorizes executables from running, small number of malware targeting jailbroken devices, few questionable applications, very common phishing): B2, 198

**Vulnerability Assessments**

Need for Vulnerability Assessment at Scale (huge number of systems on networks, new vulnerabilities all the time, very hard to find & fix vulnerabilities at scale, need methodology with automation): B3, 6-7

Definitions (Vulnerability, Vulnerability Assessment – not lesser Pen-testing, Security Audit, Vulnerability Management): B3, 8-9

Vulnerability Assessment Framework – VAF (Engagement Planning, Threat Modeling, Discovery, Scanning, Validation, Remediation, Reporting): B3, 11

Modules (VAF workflow reflects modularity: 6 modules. But: no tool can perform all modules. Key is not scanning but analysis, correlation, documentation, problem-solving & database utilization): B3, 12

VA – Step 1: Engagement Planning (Rules of Engagement, Scoping, Process & procedures & checklists, Access & visibility): B3, 14

VA – Step 2: Intelligence & Threat Modeling (Gather intelligence, Identify likely threats, Identify likely targets/critical assets/functions, likelihood of mitigation controls success, sophistication & resourcing required): B3, 15

VA – Step 3: Discovery (Purpose is to identify live hosts & map network. Inputs: URLs, host names, domain names, Ips. Output: whois info, DNS, IPs) + Tools used: B3, 16

VA – Step 4: Scanning (Purpose is to find vulnerabilities in identified technologies using scanners: Nessus, NexPose, Burp, W3AF. + Concerns): B3, 17

VA – Step 5: Validation (assign confidence value & validate vulnerabilities): B3, 18

VA – Step 6: Remediation (assign risk & priority ratings + determine remediation/mitigation options): B3, 19

VA – Step 7: Reporting (findings not remediated in Step 6 will be reported as a risk. Report includes summary, analysis, findings, recommendations, appendix): B3, 20

Vulnerability Criticality (Ratings are subjective & provide different meanings to different organizations; Risk = Likelihood \* Impact): B3, 22

Heartbleed – example of high impact but low CVSS Score (risk is subjective & must be defined by your organization): B3, 23

Common Vulnerability Scoring System – CVSS (by USDH NIAC Vulnerability Disclosure working group – shows severity & urgency of vulnerability. CVSSv2 & CVSSv3): B3, 24

CVSSv3 National Vulnerability Database Calculation (Base Score, Temporal Score, Environmental Score): B3, 25

Calculating CVSS Scores (system is flexible – when environmental considerations manually evaluate more value can be derived from CVSS. Goal is to consider exploitability & impact): B3, 26

Dangers of Auto-Generated Risk Ratings (e.g., Microsoft DREAD. Automated risk ratings don’t account for data singularity. Risk requires customized usage per application.): B3, 27

Customized Risk Calculation (Quantitative – financial impact + others & Qualitative Risk Assessment – likelihood, difficulty of exploiting + others): B3, 28

MS17-010 (developed by NSA – exploits Microsoft SMB. Acquired by “Shadow Brokers”. Patch released March 2017. April 2017 FuzzBunch released including ETERNALBLUE. May 2017 WannaCry released using MS17-010 & ETERNALBLUE): B3, 29

Meltdown & Spectre (Meltdown – lower privilege process can read higher privilege process. Spectre – allow process to read from other processes at same privilege. Key point: patching Meltdown causes performance issues so things not always black & white): B3, 30

Not all Vulnerabilities Have CVSS Scores (Most are triggered by misconfigurations & abused features – hence no CVSS score. E.g., DTP & PowerShell): B3, 31

**Penetration Testing**

What is Pen-Testing (modeling techniques used by real-world attackers to exploit flaws under scope & rules of engagement with the purpose of helping to improve security practices): B3, 36

Red Team (emulates TTPs of adversaries to improve target environment. Goal is to train & measure blue teams’ detection & response): B3, 37

Adversary Emulation (red team exercise that emulates adversary operation): B3, 38

Purple Team (virtual team where red & blue team work together. Red team is not stealthy as normal – blue team knows & is watching + improving in real time): B3, 39

Why Pen-test? (allows for validation of vulnerability findings – shows effectives of security controls, finds gaps in network, and builds confidence in potential customers): B3, 40

Rules of Engagement (covers how the test is to be conducted – Dates, times of day, contact info, data allowed to be viewed, results submission & format, blocked exploit attempt response): b3, 41

Scoping (what systems and services can and cannot be targeted – too narrow of a scope reduces value of test): B3, 42

External Pen-test (focuses on perspective of attacker from outside the organization – relies heavily on OSINT & scanning): B3, 44

Internal Pen-test (what an attacker could do if starting from within the organization): B3, 45

Web-App Pen-test (focus on web apps & databases – most often exposed to the internet & provide access to sensitive data. If poorly coded can be exploited. OWASP offers tools & guidance on webapp security): B3, 46

Social Engineering Pen-test (exploiting human vulnerabilities – people are the weakest link. Types: Computer based – mail/pop-ups & human-based – Urgency /third-person authorization): B3, 47-48

Other types of pen-testing (Mobile device, Product Security – IoT/VOIP phones/printers/switches/ routers, Physical Penetration – badge readers/elevators/security gates/biometric devices/safes): B3, 49

Pen-testing phases (OSINT, Scanning & Enumeration, Vulnerability Identification, Exploitation, Post-exploitation. Report provided is most valuable part): B3, 52

OSINT (search engines, social media, dark web, Shodan, Maltego – collect data about target passively/undetected): B3, 53

Scanning & Enumeration (enumerate discovered systems to learn info about what services are tied to ports): B3, 54

Vulnerability Identification (select the best targets & services to use in the exploitation phase. Nessus & OpenVAS): B3, 55

Exploitation (chose the technique & exploit most likely to be successful and undetected. May require evading Anti-virus – AV evasion & ghost writing): B3, 56

Post-exploitation (demonstrate true impact – pivoting & lateral movement, privilege escalation, Data exfiltration, Maintaining access. Metasploit, Empire, Cobalt Strike, Covenant): B3, 57

Reporting (Executive Summary, Introduction, Methodology, Findings & Risk Assessment, Recommendations, Conclusion, Appendix): B3, 58

Port Scanning – Nmap (port scanning – TCP & UDP/ FYN/ FIN/ ACK/ ICMP ping sweeps, remote OS detection, stealth scanning, others – can determine version of app in some cases): B3, 61

Port Scanning with Nmap (power is in large number of users – constantly growing fingerprint database to identify more applications & Oss. + States – closed, open, filtered. + Scanning methods – listing hosts, wildcards, ranges, CIDR, combination)): B3, 62-64

OS Identification (done via comparing differences in responses to sent packets – kept up-to-date by user network): B3, 65

Metasploit (exploits vulnerability if present & provides access to the system. Useful in pen-testing & verifying vulnerability in system): B3, 66

Meterpreter (post-exploitation custom shell – payload specified when using Metasploit. Execution results in TLS connection. Custom menu including sniffing, screen capture, webcam & mic access, impersonating other users, spawning processes, & more): B3, 67

Command & Control implants – C2 (Empire – PowerShell, Cobalt Strike – commercial covert communication & post-exploitation, Covenant - .Net to do same as Empire, Sliver – DNS for covert communications): B3, 68

C2 Matrix (Documents capabilities of each C2 framework): B3, 69

Password Compromise (password cracking, spraying, reuse, & more. Easiest access method for attackers –Turn on MFA! Note: FIDO2 & YubiKeys better since can’t be bypassed with reverse proxy): B3, 70

Password Reuse & Stuffing (using your credentials from one website on many other sites): B3, 71

Password Spraying (commonly used passwords tried against many accounts. Targets web & cloud sites): B3, 72

Responder (poisoning tool that targets: LLMNR, mDNS, NBT-NS. Captures credentials by responding to requests coming from these services where DNS is unavailable or fails to successfully resolve a name): B3, 73

**Attacks & Malicious Software**

Marriott Data Breach (noticed four years later. 380+ million stolen records. Example of avoidable disaster if fundamental principles followed): B3, 80-83

After-math of Marriott (free credit monitoring & call center for customers + bad website): B3, 84-85

Equifax Breach (another case where detective controls & layered network could have prevented disaster): B3, 86

WannaCry Ransomware (few basic practices could have prevented – crippled hospitals, banks, & other industries): B3, 87-89

Root Cause & Tools (2017 in Asia, is a worm because it spreads itself, patch released two months prior to attack, utilizes Eternal Blue SMBv1 & DoublePulsar Backdoor Trojan): B3, 91

Gaining Access (Ping – test reachability, Kill – terminate running processes to disable antivirus, Exec – execute malware): B3, 92

Attack Summary (every attack has 1: system that is visible from internet, unchecked scanning & enumeration, unpatched vulnerability, weak authentication): B3, 93

Current Attacks (exploit known vulnerabilities although methods are changing. Best way to prevent is to backup frequently & separate systems): B3, 94

Ransomware as a Service – Raas (create custom version of ransomware for profit. Creator gets cut of ransom) + examples: B3, 95-96

Input Attacks (OS Command injection, Buffer overflows, SQL injection – treat all user-supplied input/“trust boundaries” as potential attack points): B3, 98

OS Command Injection (relies on dev using input to build calls back to the OS. To defend: avoid system calls in app especially system() function, process input, use built-in functions, strip OS commands and characters from input, define valid characters for input): B3, 99-100

Buffer Overflow (buffer is overrun, over-writing return pointer. – Sterpy() command + Defenses for buffer overflow): B3, 101-106

SQL Injection (user able to insert SQL due to poor input validation – can lead to OS compromise & accessing or replacing critical data) + Defenses: B3, 107-109

Input Attacks Defense Summary (validate input with allowlists – preferred - or denylist): B3, 110

Viruses & Worms (Virus requires interaction & host file. Worm spreads itself using vulnerabilities & exploits): B3, 112-113

Trojans (program + malware. Easiest way to construct is using source-code): B3, 114

Rootkits (subvert userland & kernel security controls to avoid detection. Goal is to provide ongoing system access to attacker. Utilize default system resources to blend in): B3, 115

Malware Analysis Stages (Automated analysis – quick & cheap: 117, Static Analysis – safer because no execution: 118, Behavior Analysis – run in controlled environment & monitor: 119, Code Reversing – hardest & most time-consuming + requires disassemblers & de-compilers: 120): B3, 116-120

**Web Application Security**

Web Communication Basics (HTTP protocol – transaction oriented. Request, header, body): B3, 128-129

Cookies (store data from browser session – used to keep state. Big lack of understanding. They cannot take info from your computer, only info you provided to the webpage that saved them. Real concern is sensitive data stored in cookies which can be vulnerable) – Persistent & Non-Persistent: B3, 130-132

SSL/TLS (encrypting network traffic, operates on port 443, encryption & server identity verification & data integrity – not a guarantee of security though + can be good for attackers sometimes): B3, 133-134

OWASP Top Ten (most critical web application security issues): B3, 136

Developing Secure Web Applications (security must be built into software dev life cycle!!! + steps): B3, 138-139

Basics of Secure Coding (Initialize all variables, validate user input, don’t make app require admin perms on server, handle errors & don’t display to user, least privilege, don’t store secrets in code, only tested & reliable libraries + monitor for vulnerabilities in them to patch ASAP): B3, 140-141

Web Application Vulnerabilities (Authentication: 143-145, Access Control: 146, Session Tracking: 147): B3, 143-147

Hacking Session Info (URL session tracking, Hidden form elements, Cookies): B3, 148

Protecting from Session Attacks (random & long IDs + encrypt them, new session ID upon authentication, expire IDs on logout & timeout): B3, 149

Web App Monitoring (SIEM correlation, track performance, examine logs, detect defacement): B3, 150-151

Web Application Firewall (monitor & rarely block HTTP/HTTPS traffic, network level & application level, most deployed in monitor mode only): B3, 152

Monolithic Architecture & Security Controls (all functionality required to perform duties exists on one tier – security controls are applied to each trust boundary): B3, 153

Microservice Architecture (many microservices instead of big service. Broken down by CQRS. Huge Attack Surface & number of trust boundaries): B3, 154-155

**Security Operations & Log Management**

Terms & Definitions (Message, Log File – collection of logs, Alert, Device, Logging, Auditing, Monitoring – real-time, Alerting, Debug traces): B3, 162

Lots of Logs (Every machine can write logs, each in different formats, growing amount of data to collect with more devices, can be overwhelming if not properly managed): B3, 163

Building Log Server (Prepare: build Linux server, deploy on network, allow SSH & syslog ports UDP514/TCP514. Operate: configure assets, configure log system, parse & augment logs): B3, 164

Accepted Log Standards & Lack Thereof (key-value – CEF & LEEF, JSON – Suricata IDS, Syslog – not true log but de facto especially in Linux yet being replaced by key-value & CSV, Windows logs – binary format w\_XML schema. Fast but requires agents to read & send): B3, 166

Managing Logs (Order: Firewalls, net security gear, servers, other server, Databases, Apps, Desktops): B3, 167

Key Points (logs are: key organizational records, produced by many devices, often look different, tackled in a logical sequence): B3, 168

Log Filtering (Input-driven, output-driven, & hybrid log collection strategies): B3, 169

Top Log reports (Order: Authentication, Changes, Network Activity, Resource access, malware, Failures, Analytics reports using “Never Before Seen”): B3, 170-172

Log Analysis Tools (Log Collection Tools – NXLog & Winlogbeat & Fluentd, Secure centralization - stunnel & OpenSSH & IPSec + VPNs, Pre-processing – NXlog, Log Storage – Elasticsearch & Hadoop, Analysis – SEC & OSSEC & OSSIM & Swatch/logwatch/longsentry & Elastic Stack): B3, 174-175

More tools & How to start using them **\*IMPORTANT\*** (collect, store, search, correlate & alert): B3, 176

Log Monitoring Strategy (workflow steps; not necessarily daily. If false alarm – change the rules or adjust controls): B3, 177

Setting up Log Monitoring: Phased Approach/step-by-step (Order: Decide logs to integrate, decide sections of network & systems to connect first, address new use cases for log data): B3, 178-179

Challenges to Organization Log Management Deployment (political boundaries, network & state boundaries, remote locations, custom applications, legal liability, compliance & regulations): B3, 180

Log Aggregation & SIEM (centralize, normalize, correlate, ensure central time server or UTC if multiple time zones, detect log deletion & corruption): B3, 181

Sigma (High-level generic language for writing SIEM rules – allows writing in Sigma format then convert to specific SIEM product’s rule using Sigmac. Advantage is portability and sharing potential): B3, 183

Sigma Format (YAML format. Each rule has: title, status, logsource, & info. Allows for ease of readability & customization): B3, 184

Converting Signatures to Alert Queries – Sigma to specific (write in Sigma, create file to map generic field names to platform-specific field names, convert Sigma to specific SIEM rule format using Sigmac) – process is simple & can be automated: B3, 185

Sharing rules – possible with Sigma & MISP (one’s treasure is another’s trash. Shared rules should go through a quality review): B3, 186

Orchestration (pull rule from MISP, convert with Sigmac, test rule, if match then manually assess if not validate performance & move into production): B3, 187

Real-time tasks requiring immediate response (Malware outbreaks – AV very likely to miss malware, Reliable Intrusion Evidence, policy violation, data theft, loss of service on critical assets, log deletion & corruption detection): B3, 189

Daily Tasks (unauthorized config changes, service disruption, intrusion evidence, suspicious login failures, minor to medium malware activity, summary of activities): B3, 190-191

Weekly Tasks (Review all log trends, creation & removal of accounts, device & network changes, summary of less critical probes & attacks): B3, 192

Monthly Tasks (long-term network & system log trends, Minor Policy violations, resource usage reports, security technology performance measurement): B3, 193

Quarterly Tasks (create audit reports, longer-term trends, infrastructure changes affecting log monitoring, log management system performance); B3, 194

Annual Tasks (review log policy, verify log retention & archival, longest-term trends, analyze & prepare next year’s security budget, review new regulations affecting logging): B3, 195

How Logs Help - & don’t (They figure out who, where, when, how, what & more BUT there are limitations & challenges): B3, 196

**Digital Forensics & Incident Response**

Digital Forensics in practice (InfoSec, DoD/Intelligence Community, Law Enforcement, Legal Support): B3, 202

Investigative Process (follow the evidence, be objective, let evidence speak for itself, don’t make assumptions, scientific method!): B3, 203

Follow the Evidence (User activity -> Creates Evidence -> Recovered by Forensics): B3, 204

Remaining Forensically Sound (Detailed Documentation: Chain of Custody, Start/End Times, Commands Typed. & Forensically Sound Methods: Write-blockers, read-only files, Protected storage environments): B3, 205

Digital Forensic Artifacts Examples (File Download, Program Execution, File/Folder Opening, Deleted File or File Knowledge, Physical Location, External Device/USB Usage, Account Usage, Browser Usage): B3, 207

Filesystem Timestamps (two groups of four timestamps for each file: $STDINFO & $FILENAME. Modified Time, Access Time, Metadata Change Time, Creation Time. “MACB” times): B3, 208

Must act fast (OS doesn’t store artifacts long. There are tricks to go “back in time” but act fast): B3, 209

DFIR subdisciplines (Endpoint, Endpoint Memory, Network, Threat Intel, Reverse Engineering): B3, 211

Digital Forensics Tools (Open-Source: Sleuth Kit/Autopsy, Perl/PowerShell/Python scripts, Log2timeline/Plaso. Virtual Machines: SANS SIFT, REMnux. Closed Source: EnCase, X-Ways): B3, 212

The SIFT Workstation/Virtual Machine: B3, 213-214

Incident Handling Fundamentals (action plan for intrusions, cybertheft, DoS, Malicious code, fire, floods, & others.): B3, 216

Incident vs Event (incident is bad event, event is observable occurrence in system or network): B3, 217

Incident Handling Process - PICERL (Preparation, Identification, Containment, Eradication, Recovery, Lessons Learned): B3, 218

Common Problems with PICERL (Issues in execution: Poor security, little scoping, not fixing vulnerabilities, no lessons learned): B3, 219

PICERL – Preparation (Know thy organization, internal visibility is key, have plans for rebuilding systems): B3, 221-222

Identification, Detection, Verification & Triage (sources for detecting an incident + motives for triage. First decision is always verification): B3, 223-224

Containment (Stopping the threat actor. Requires scoping. May help with eradication & occurs in multiple phases): B3, 225

Eradication (making sure that a new compromise using the same or similar vulnerability does not happen again & undoing the threat actor actions): B3, 227

Recovery (get back up & running. Rebuilding system is often most cost-effective. Try to bring systems online during off-hours to monitor easily) two options – re-install OS or recover from backup & patch: B3, 227

Remediation (identify & fix the root cause. Monitor any compromised systems – rootkit/backdoor suggests intent to return. Remediation involves taking in full story & fixing all gaps): B3, 228

Lessons Learned (final report after meeting with incident handling team for executive summary. + schedule follow-up review): B3, 229

Key Mistakes in Incident Handling (Failure to report or ask for help, no notes, destroying evidence, no backups, failed containment or eradication, failed reinfection protection, failure to apply lessons learned. + don’t place blame in follow-up meeting): B3, 230

Taking Notes – critical! (uses: basis of report, jog your memory, document damage for insurance, governor on your speed. Don’t suggest bias or make yourself look unprofessional!): B3, 231

Putting it all together (customize steps to your environment. Every incident is different. Planning is everything. Checklists & procedures! Practice!): B3, 232

Threat Hunting (proactive incident response – using knowledge of attacker activities to search for historical unknown events in an organization): B3, 233

**Cryptography**

Cryptosystem Fundamentals (The science of hidden writing. Communicate without revealing meaning to adversaries. Potentially validates to whom we are communicating. Must be a part of a larger defense-in-depth strategy): B4, 5

Cryptology, Cryptography, and Cryptanalysis (Cryptology = cryptography & it’s counterpart/opposite Cryptanalysis. Cryptanalysis is breaking ciphers): B4, 6-7

Components of Cryptography (Cryptography, decryption, encryption, plaintext, ciphertext): B4, 8

Cryptosystems (collection of all possible inputs, outputs, the algorithm, & keys. Note: humans are the weakest link. Loose the keys, lose the cryptosystem): B4, 9

Keys (permit existence of open algorithms. Longer key/key space = more secure): B4, 10-11

Enterprise Crypto (Symmetric = fast, Asymmetric = slow, Diffie-Hellman = secure key exchanges but not encryption, digitally sign = RSA/ECC with hashing + more): B4, 13

The Challenge/Goals of Cryptography (Confidentiality - symmetric, Integrity - Hash, Authentication - Asymmetric, Non-repudiation – Digital Signature): B4, 14-15

General Symmetric Encryption Techniques (XOR: 17, Rotation Substitution + Ceaser & Usenet: 18, Arbitrary Substitution: 20, Permutation: 22, Hybrid: mix of permutation & substitution): B4, 16-22

Frequency Analysis (breaks Monolithic & one-to-one substitution ciphers): B4, 21

Types of Cryptosystems (Symmetric – secret key, Asymmetric – public key, Hash – one way): B4, 24

Symmetric Key (Fast, requires secure key distribution channel – pre-shared key through asymmetric encryption like Diffie-Hellman): B4, 25

Secure Channel for Secret Key using Asymmetric (Asymmetric is slow, use it to securely transfer symmetric key to get best of both): B4, 26

Secure Channel for Secret Key using Diffie-Hellman (how it works! N, G, private & public key): B4, 27-28

Asymmetric Key Cryptosystems (slow, public key distributed, used as secure channel for symmetric key exchange, non-repudiation via digital signatures): B4, 29

Hash Functions (No key, irreversible, some algorithms have issues with predictable collisions – occur when inputs possible are greater than outputs possible. Primary use is message integrity): B4, 30-31

Collisions in Hashing (when two inputs give same hashed output. To reduce risk, ensure similar inputs result in drastically different outputs. Bit length makes the largest difference. Larger bit length = less collision. No way to predict when collision will occur): B3, 32

Integrity (original message, hash, & algorithm are transmitted to be checked by receiver): B4, 33

Digital Signatures (asymmetric to “sign” documents. Non-repudiable. Signed with private key associated with one person. RSA algorithm most common) + example: B4, 34 & 36

Steganography (conceals additional information. Hide in variety of file formats. Used also to disguise encrypted data to prevent attacks on encrypted data & where encrypted data is inappropriate for transmission): B4, 39

Crypto vs Stego (Crypto provides confidentiality but not secrecy. Easy to detect encrypted messages. Stego hides intent. Secrecy but not confidentiality): B4, 40

Detecting Cryptography (easy to detect encrypted text by human eye. Machines can use histograms to see variation in frequency of characters – encrypted is no variation in character frequency): B4, 41-42

How Steganography works (requires host – usually a file): B4, 43

General Types of Stego (Injection: 45, Substitution: 46, File generation: 47): B4, 44-47

Injection (hidden form elements/comments in HTML & GIF, word document holes. Increases size of data): B4, 45

Substitution (replaces original data – could result in degradation of the file. Usually insignificant data is replaced – LSB): B4, 46

Generate a New File (No host file needed, input text generates fractals/human-like text to be used as carrier): B4, 47

Detecting Steganography (works on tool-by-tool basis. Common way with images is to analyze LSB use. Tools: StegExpose, StegSecret. No universal way!): B4, 48-49

**Cryptography Algorithms & Deployment**

Concepts in Cryptography (math problem that is impossible to solve in polynomial time + branches of math involved): B4, 56

Computational Complexity (tractable or intractable. + big-O notation): B4, 57-58

Intractable problems (factoring large integer into two prime factors: 59, discrete logarithm for finite fields & elliptic curves: 60 & 61): B4, 59-61

DES (fast, widely used, symmetric algorithm. 64-bit cipher, small 56-bit key, not considered secure today. Weaknesses: small key space, crackable in short time; can be fixed by encrypting repeatedly since DES is not a group): B4, 63-65

2DES & 3DES (2DES susceptible to Meet-in-the-Middle attack so not safe. 3DES is secure with 112 or 168 bits depending on number of unique keys): B4, 66-67

DES Replacement: AES (how it was created. Three initial key sizes: 128, 192, 256): B4, 68

AES Algorithm – how it works (four transformations: AddRoundKey, SubBytes, ShiftRows, MixColumns) + Usage & Vulnerabilities: B4, 70-71 & 72

RSA – Asymmetric (Usage: webclients – SSL/TLS. Widespread due to patent expiration. Cracking means compromising poor implementations. No valid reports on cracking algorithm): B4, 73

Elliptic Curve (Usage: high speed, low power, low storage, high security at small key lengths; wireless, mobile, etc. + Vulnerabilities.): B4, 74

Comparing Key Length (Symmetric, DSA/DH, RSA, ECC. Asymmetric keys are longer): B4, 75

SHA-1/SHA-2/SHA-3 (SHA-I deprecated – 160 bit. SHA-2 & 3 output is 256 & 512 bit): B4, 76

Cryptanalytic Attacks (test crypto systems. Focused on: mathematics, key generation characteristics, collision probability, weaknesses in implementation, key protection +. Categories: Analytic, Statistical, Differential, Linear, Differential Linear): B4, 78

The Birthday Attack (Statistical probability problem. Much higher likelihood of birthday/hash matching in a bigger group): B4, 79

**Applying Cryptography**

Confidentiality in Transit: Private Networks (dedicated lines & equipment – expensive & underutilized. + private/leased lines provided by telco company. Still used today but now we think of telco as potential threat so protect data in transit over leased lines too): B4, 87

Confidentiality in Transit – VPN (data encrypted at one end & decrypted on other end): B4, 88

VPN Advantage: Flexibility (VPN tunnel over internet can be setup fast; private network takes weeks/months): B4, 89

VPN Breakdown (characteristics + other info. Note: 1-2 second delay with VPN. If time is critical use dedicated lines): B4, 90

Types of Remote Access (Client-to-site/transport – one gateway at border of host network & Site-to-site/tunnel – two gateways communicate. + client-to-client – more secure since no gateways but not used much since would require lots of config to setup between every device unless PKI is used): B4, 91

IPsec Overview (type of VPN. Enable encrypted communication between users & devices + replay attack prevention): B4, 93

Types of IPsec Headers (Authentication Header – integrity, origin authentication, no confidentiality/no encryption. Encapsulating Security Payload – integrity, origin authentication, confidentiality/encryption, does not factor in source & destination into ICV calculation): B4, 94-96

IPsec Modes (Tunnel mode – Tunnel to IP Traffic. site-to-site VPN. Transport mode – between two hosts. + Security Associations & Internet Key Exchange): B4, 98

SSL VPNs (Less operational problems compared to IPsec, cryptographically equivalent, applicationally not as secure. Only need browser for client. Cheaper. + Portal VPNs info): B4, 99-100

Security Implications & problems to be aware of with VPNS (1. restrict site-to-site VPN connection to minimum necessary to fulfill business needs through firewall if other side of connection is another organization. 2. Be careful where encrypted tunnels are setup to avoid bypassing security devices. 3. Clients that remote in should receive as much examination as possible): B4, 101

Full Disk Encryption (data encrypted before storing, decrypted before use. Protects against theft but not during regular system use/attack when signed in): B4, 103

GNU Privacy Guard – GPG (personal encryption via individual file/folder encryption & secure email abilities including digitally signed email & encrypted email): B4, 104-105

Establishing a Key & Choosing a Passphrase using GPG (generating public/private key process): B4, 106

Public Key Infrastructure – PKI (system for the creation, maintenance, and revocation of certificates. Certificates bind an identity to a public key. PKI consists of hierarchy of trusted authorities – chain of trust): B4, 109

Certificates (Digital document attesting the binding of an entity to a public key. Equivalent to passport or driver’s license): B4, 110

Certificate obtaining, sharing, & verifying process details (Certificate Authorities give certificate after identity & public key ownership are verified. Certificate is verified after certificate authenticity verification, identity claimed verification, and public key verification) + details of process: B4, 112-113

Operational Goals of PKI (registration, creation, distribution, revocation, & expiration steps): B4, 114-116

Certificate Revocation List – CRL (list of revoked certificates – downloaded & cert serial number checked. Several problems. Replacement is OCSP): B4, 117

Online Certificate Status Protocol – OCSP (replacement to CRLs. Client sends cert serial number to OCSP server & response is sent in regard to revocation status. Privacy concern with server knowing client is accessing resources belonging to certificate owner - addressed with stapling): B4, 118-119

Digital Certificates extra info (standard is X.509. Includes info on: X.509 version, serial number, info on owner, validity period, owner’s public key, & algorithm info, Signature by CA): B4, 120-121

HTTPS Communication Process (handshake explained): B4, 124-125

Problems with PKI (competing/incomplete standards, certification of CAs, do it yourself or outsource to specialized third party, added steps that users need to understand, high cost of establishment prior to receiving any benefits of PKI): B4, 126 -127

PKI other uses (disk encryption, code & driver signing, user authentication, IPsec & VPN authentication, Wireless authentication, Network Access Control, +. Assume that org will use it for more later): B4, 128

**Network Security Devices**

Network Security Devices (Deployed to provide security by monitoring traffic. Very scalable based on placement. New technologies regularly. Main categories: Firewalls – Prevention, Intrusion Detection – Detection, Intrusion Prevention – Prevention): B4, 134

Big Picture (defense-in-depth. Have multiple layers of protection & comprehensive protection – variety of preventive & detective technology. + Perform gap analysis to find exposures): B4, 135

Firewall pros & cons (preventive technology, reduces risks with filtering, increases privacy – hiders info gathering & encrypts communication + provides logs & saves bandwidth. Implemented as hardware or software. Placed between public internet & private internal network or between NIC & PC): B4, 137-138

Default rule (default deny is default. Default allow can be set but not recommended): B4, 139

Filtering (Ingress – rules for traffic flowing in. Egress - rules for traffic flowing out. Ingress always set. Egress recommended to prevent participating in DoS attack unknowingly or intrusion detection): B4, 140

Anti-Spoofing (prevent network addressing defined behind interface to source from another interface. Stops attacker from changing their source IP): B4, 141

Types of Firewalls (Packet filtering – packet by packet, Stateful filtering – collects & correlates packets then decides. More secure but slower, Proxy – sits between two connections & doesn’t allow them to communicate directly: slower. In referenced architecture, use all three): B4, 142

Stateless Packet Filter Firewall (minimal security, very fast, easily tricked, data content unchecked. Auto allows ACK packets – facilitates host discovery & ACK scan/finding unfiltered ports): B4, 143

Stateful Firewalls (keep record of state of traffic flows via state table for stateful protocols & timeout duration for non-stateful protocols. Issue is ICMP error packets which are blindly accepted. Alternative is payload traffic inspection + state table): B4, 144-145

Proxy/Application Gateway (Slowest, most inconvenient to manage, most secure. Sits between client & host. Default rule is deny if not allowed. Tears down each packet layer by layer & reconstructs if allowed. Maintains source to proxy & proxy to server connections): B4, 146

Intrusion Detection System – IDS (monitors & reports attacks against systems & networks. Mature Technology. Requires team for monitoring, alerting, & reaction. High cost to use & maintain. Not a replacement for Firewalls, strong policies, system hardening, fast patching, defense-in-depth. Deploy only after above defenses): B4, 148-149

IDS Alerts & Implications (True Positive, False positive, True negative, False negative): B4, 150-151

Network Intrusion Detection System – NIDS (passive sniffer at network aggregation points & detects events of interest using signature, anomaly, or application/protocol analysis. Can be software or hardware): B4, 152-153

How Signature Analysis Works (performs pattern matching. Rules indicate criteria that represent events of interest. Rules are applied as packets are received. Alerts are created when matches are found. Most implementations use a set of rules in lookup tables to optimize analysis): B4, 153

Rules & Signature Criteria (flexible rules language is valuable in an IDS. Allows classifying data on: protocol info, address info, port info, payload contents, strings, flags in headers, & traffic flow analysis): B4, 154-155

How Anomaly Analysis Works (Baseline of network must be performed to understand “normal”. Anomalous conditions are flagged. Can catch zero-day exploits. IDS vendor identifies anomalous conditions): B4, 156

How Application/Protocol Analysis Works (Individual Protocol Activity is understood by vendor. Use of protocol outside of “normal” is flagged/exclusive method. Difficult to implement. Few vendors can implement true protocol analysis): B4, 157-158

Deep vs. Shallow Packet Inspection (Shallow – fast, little fidelity, examines header info, Signature-based IDS. Deep – full analysis including variable length fields, deployed at application-level firewall, anomaly & protocol analysis IDS): B4, 159

Data Normalization (attackers try to de-normalize traffic to evade detection. IDS normalizes data to give consistent basis for analysis & rule generation) + MS SSQL server exploiting detection example: B4, 160

NIDS Advantages (Scalability, insight into traffic, detect problems, help org react swiftly to incidents, auditing, mor aggressive than preventive measures, identify misuse of change control policies, identify info leakage, + honeytoken info): B4, 162

NIDS Challenges (topology & access limitations, analyzing encrypted traffic, quantity not quality signatures, performance limitations, very costly): B4, 163

Snort as a NIDS (Open-source, low cost, can monitor multiple sites/sensors, efficient detection, low effort for reporting, very flexible ruleset): B4, 165

Snort Rule Flexibility (Big Plus. Rules to detect any type of pattern match. New rules for latest attacks published by community fast. Rules support honeytokens or other custom requirements): B4, 166

Writing Snort Rules (pre-loaded with hundreds of rules but to write, three options: Pass/Ignore, Log, Alert): B4, 167

Snort Rules (options must be separated by semicolon, Output: message, date/timestamp, source IP, destination IP, TCP information. Advanced rules check application layer data with “content field” parameter. + NOP/NOOP & hex 90 example. + flow:to\_server & established): B4, 169

Key Points for NIDS (Train staff in IDS analysis, employ sniffer in sec management console, get a SIEM, prepare incident response policies, perform ROI for in-sourced or outsourced IDS): B4, 170

Developments in NIDS (reductions of false positives through OS identification, vulnerability assessment before generating alert + active vs passive analysis, IDS hardware/“IDS blade” faster, wireless IDS/WIDS): B4, 171-172

Intrusion Prevention System – IPS (stops attacks on systems, network-based or host-based. Not alternative to defense-in-depth): B4, 174-175

NIPS deployments & benefits (deployed: between firewall & ISP router – protects firewall & DMZ, or behind firewall – protects internet network from VPN users & helps find infected internal hosts): B4, 176

NIPS Detail (deployed inline at aggregation points, custom Application specific Integrated Circuits/ASICs used to support high-speed analysis, data normalization & reassembly techniques, Hierarchical rule classification for efficiency / “multiresolution filtering”, cannot have false positives hence cannot identify as many attacks): B4, 177

NIPS Challenges (false positive = denial of service, must keep up with traffic demands, less-extensive rule base & more false negatives): B4, 179

Advancements in NIPS (improved throughput & response times, near-real-time analysis & forwarding, automated analysis/signature updates, environmental anomaly analysis, protocol “scrubbing”, rate limiting, policy enforcement, passive analysis): B4, 180-181

Passive Analysis (Correlates OS & vulnerability info with identified attacks. “Network learning” mode to identify network architecture & structure to classify attacks against internal systems & identify spooked packets from internal network): B4, 182

Example Architecture + tips (employ all above technologies! Make sure you have balance of prevention & detection. Pen testing to verify your architecture!): B4, 183

**Endpoint Security**

Components of Endpoint Security (goal is to control damage & reduce impact by: limiting attack surface & foundation for effective security): B4, 190

Enhancing Endpoint Security (enhance OS security by: better visibility, reducing attack surface, controlling the damage, early detection. With new tool always identify the overall risk that is going to be reduced & if solution is the most cost-effective way to reduce): B4, 191

Baselines (pictures of how a system normally looks & behaves. Determined through use of commands & logging output. Compare past to present for changes & act): B4, 192

Establishing a baseline (identify “normal”: type of traffic, amount of traffic, type of logs, number of logs, resource utilization, access times, length of access, current state & configuration of server – create snapshot & monitor changes through hashing): B4, 193

Detecting Anomalies (make baseline & check against it frequently. Use file checking tools to sift through background noise, combine file checkers with central log aggregation & graphical UI. Not always volume-based outliers. Often, outliers deviate slightly – use behavior-based anomaly detection): B4, 194 & 195

Antivirus Software (check for viruses, worms, or other malicious software. Typically integrated into suite of products. Scans files, attachments, emails, web content, etc. Now generic term for anti-malware & anti-adware. Called “endpoint security software” to avoid confusion): B4, 197-198

Endpoint Firewalls (*Network firewall* good to manage & filter traffic + provide boundary defense. For laptops outside of network need H*ost-based firewall* - hard to manage in large organization): B4, 199

Types of Endpoint Firewalls (Packet Filter/Stateful – only authorized services, application control– screen incoming packets & keep set of rules/allows granularity, operating system control – most flexible & will not allow a program to run until approved): B4, 200

File Integrity Checking (analyst defines list of critical files, files hashed by HIDS, hash regenerated frequently, if hash changes alert is sent): B4, 201

Steps for File Integrity Checking (define list of files, generate hashes, store hashes safely, confirm hashes can’t be modified, set intervals to rerun hashes, compare hashes, alert where hashes change, optionally alert when new files added within a directory): B4, 202

How Log Monitoring Works (inclusive analysis – list of keywords to raise alert, exclusive analysis – if event not listed raise alert. + Note: is cheap & easily implemented with tools): B4, 203-204

Application Control (only approved list of applications where hashes are valid can run. Blocking or alerting mode. Blocking recommended but won’t work without ready environment): B4, 205

Host-Based Intrusion Detection – HIDS (functionality of NIDS to each host, more granular: signature & anomaly analysis, unauthorized change monitoring, log monitoring, network monitoring. Local processing & alerting normally but benefit from centralized alerting for organizations): B4, 207

HIDS Network Monitoring (monitors network to & from the host. Listens on all interfaces: Ethernet, wireless, VPN. Signature analysis to find EOI. Outbound traffic can detect pivoting, internal reconnaissance, lateral movement, & C2): B4, 208

HIDS Advantages (additional information NIDS can’t see, pre/post-encrypted data, more monitoring/analysis capacity, detailed insight into network, finds inside attacks, more details about host to increase accuracy & reduce false positives. Last line of defense! If it alerts, all else failed): B4, 209

HIDS Challenges (updates can be complex, “tunnel vision” to individual hosts, requires centralized console to identify wide-scale events, full HIDS more costly than NIDS, requires resources on hosts & impacts system performance – more cost): B4, 210-211

Developments in HIDS & Recommendations (monitor change at application level, protect website with IDS, recently HIDS for networking devices too, make HIDS another data feed for SIEM, HIDS recently morphing into HIPS): B4, 212

HIPS details (stop common attacks – known & unknown, “system call interception” controls system calls like antivirus, Uses: signature & anomaly analysis by monitoring file integrity, network, & application behavior): B4, 214

HIPS Advantages (same advantages of HIDS but allows timely prevention. Anomaly Analysis stops unknown attacks – used to buy time in patching. Better defense for systems with expanding network perimeter. + protection for traveling laptops): B4, 215

HIPS Challenges (False positives – less on distributed scale, implementation & maintenance challenges, limited suite of applications, not replacement for system patching or AV, more system resources for in-depth anomaly analysis, cost for management consoles & labor to manage): B4, 216

Application Behavior Monitoring (Vendor Identifies intended app behavior, if unintended behavior found, blocked & alert generated. In practice hard to get right because apps constantly change. Most vendors use hybrid of application behavior monitoring & anomaly analysis): B4, 217

HIPS Recommendations (maintain requirements for testing procedure of HIPS software, policy for controlling HIPS client rules & updates, don’t blindly install updates without testing, don’t rely on the vendor to test for you, don’t rely on HIPS solely – use to buy more time to test patches): B4, 218

Developments in HIPS (protection against zero-day attacks, dynamic rule creation for custom apps based on behavior, application shielding – sandboxing apps, new target for attackers – must be updated & properly maintained): B4, 219

**Windows Security Infrastructure**

Windows Operating Systems (Client OS, Server, Embedded): B5, 5

Client Operating Systems (Personal Editions: Starter, Home, Ultimate. Work Editions: Business, Pro, Enterprise. Platforms: AMD/Intel, ARM. Licensing: Retail – more expensive, OEM – cheaper, Enterprise – organization-wide. Free on tablets & RPI. + X & S Modes): B5, 6-7

Server Operating Systems (Standard, Enterprise – no longer exists, Datacenter – supports more hardware & functionality. + special editions & R2 – new version. Core – little GUI vs Nano – no GUI. Licensing is hard): B5, 8-9

Windows Server Roles (role – major functionality. Feature – smaller component. + List of roles): B5, 10

Windows Embedded/IoT (industry-specific hardware, supports ARM & x86/x64, runs on RPI & Minnow & Arduino. Examples: IoT, Embedded Industry, Embedded Compact – ARM, Embedded Automative, & SQL Server for Embedded): B5, 11-12

Workgroups overview (Computers that share information in the absence of a domain controller. Local users are admins. Standalone computers only. Usually less than 50 devices): B5, 14-15

Workgroup benefits (simple, each computer manages itself, lower cost, users admins of their own machines – creative expression & joy): B5, 16

Workgroup drawbacks (Users are insane. Chaos & Anarchy. Difficult to manage large numbers. No central policy control or auditing. No single sign-on/standalones don’t trust each other’s SATs. No consistent permissions across machines): B5, 17

Managing Local Accounts (Computer Management Tool – CompMgmt.msc & Start screen): B5, 18

Security ID Numbers (Each user, computer, & group has its own SID. Standard SIDs such as: Everyone, Authenticated Users, Local Administrators. Win only cares about SID for permissions & privileges, not names. SAT is ID card to regulate your activities. + how to see your SID): B5, 19-20

Security Access Token – SAT (windows driver’s license. Attached to every process you start to check permissions & privileges before allowing attempted actions. + How to see SAT. Made possible by ACL): B5, 21

Achieving a more Perfect Workgroup (standalones don’t treat each other’s SATs. We need a central database for SID numbers for single sign-on & protocol to distribute. Need a domain controller): B5, 22

Active Directory Domains (AD is a shared registry for all users & computers on the network. AD is multi-master replicated – change on one AD database will sync with all others except on 2008 RODC which is ideal for remote branch offices. Notes: Account in AD = in domain; Local User stored on PC & only access that PC; Domain User stored in AD database & access any PC): B5, 24-26

Authentication Protocols (Kerberos & NTLM – send parts of SAT: SID number for user’s domain account & domain groups. SAT is constructed in memory of local machine with SID for local groups & privileges of user + info protocols send): B5, 27

Kerberos (Default – backup is NTLM. Faster & better in large environments that NTLM. Requires AD domain controller. Risks: if initial exchange packet is captured, brute force is possible. + If encryption keys stolen – rip) + example: B5, 29 & 30

NTLM (Kerberos predecessor. Used in workgroups – doesn’t require AD. “I’ll ask your mother” authentication. V1 password protection is bad. V2 is better. Should stop using for issues): B5, 31

Forests & Trusts (Domains linked with trusts for resource sharing & sign-on across domains. Inter-domain replication, two-way transitive trusts, Global Catalog, Global Catalog Servers): B5, 32-33 & 34

Cross-Forest Trusts (Entire forests can trust each other – 2003+. One-way or two-way, transitive, no Cross-Forest Replication of any AD data): B5, 35-36

Group Policy (Windows configuration management, Scales to thousands of PCs, limited to AD on domain controllers. + Group Policy Object – class applied to domain-joined computers. + Settings of GP): B5, 37

How Group Policy Works (config script that changes settings when run. GPOs stored & replicated on domain controllers unless local GPO. + How to access GPO & GPMC): B5, 38-39

Summary (OS systems, Workgroups, Local vs. Domain accounts, SID & SATs, Active Directory, Authentication, Forests & Trusts, Group Policy. – good summary): B5, 40-41

**Windows as a Service**

End of Support – Modern vs Fixed Lifecycle policy (Fixed lifecycle – end of sales, end of support, end of extended support, end of custom support. Modern lifecycle – newer versions supported for at least 12 months, after requires update to receive support) + what to do if OS expired: B5, 47-48

Windows Updates (Windows as a service = continuous updates: Feature updates – large new service every 180 days or 1 year, Quality Updates – big fix or patch every 30 days. + update vs upgrade): B5, 48-49

Security Updates Guide (cumulative updates = switch from bulletin to database + Using database): B5, 50

Updates for Channels (quality updates deferred up to 30 days for semi-annual & 35 for home. Feature updates deferred up to 18 months for both home & semi-annual. + Rings – PC group assigned a servicing channel): B5, 52-53

Channel configuration (through settings & GPO. + Deferring disabled if telemetry sharing off): B5, 54-55

Long-Term Channel & Insider Program (Long-term never gets feature updates, requires volume license agreement & special enterprise edition, only monthly quality updates. Insider Program gets access to upcoming features still in development – important for planning & testing; each organization should have few systems. Options: fast & slow): B5, 56-57

Windows Update (more control in GPO, “Update for Business” is just a marketing trick.): B5, 58-59

Windows Server Update Services – WSUS (local update server – free IIS web application which clients can download updates from. Saves bandwidth, controls updates, custom groups, scales to thousands of machines. Uses BITS non-obtrusive service. WSUS can be installed on 2003 & later): B5, 60-62

Windows Autopilot (Azure AD takes care of installing apps, updating, configuring automatically): B5, 63

Autopilot Requirements (10 Enterprise/Pro/Education, MDM solution like Intune, OEM that prints device ID on box, Azure AD domain. Options: on-premise AD domain, pre-provisioning, reset): B5, 64-65

Azure Virtual Desktop – VDI in the cloud (Windows running in the cloud accessed from any RDP capable device through a URL. Personal & shared VMs, optimized with OneDrive, encrypted with TLS on TCP port 443, no permanently listening & exposed TCP port. Access to: clipboard, audio, camera, printer, smart card, & USB): B5, 66-67

Azure Virtual Desktop Management & Requirements (setup pools with CPU, GPU, memory, disk speed options. Spin up as demand increases & drain/shutdown when decreases. Users connect through app or URL; organization must have Azure AD domain & on-premises AD domain. OS offered: 11 Enterprise, 10 Enterprise, 7 Enterprise with extended updates, Server 2012 R2 & 2016 & 2019): B5, 68-69

**Windows Access Controls**

Overview of Access Controls Purposes (selective privilege, “who” in logs for auditing, maintaining system integrity through BitLocker, least privilege to SATs through UAC): B5, 71-72

NTFS Overview (FAT & FAT32 comparison, NTFS characteristics, & ReFS comparison): B5, 74-75

NTFS Discretionary Access Control Lists - DACL (set of permissions of a folder or file. Individual permissions in the DACL are called access control entries – ACEs. ACEs accessed through security tab – standard/generic/custom permissions): B5, 76

Settings for ACE (Deny Overrides Allow, Explicit vs Inherited Permissions, Inheritance Settings): B5, 77-78

NTFS Owners & Owners Group (every object has an owner with all privileges; Ownership can be changed; creator group assigns permissions to whomever is the owner now. TAKEOWN.exe to take ownership of files recursively): B5, 79-80

Principle of Least Privilege (perform “needs analysis” & give minimum privileges needed. + good starter permissions): B5, 81-82

How to apply privileges & Permissions – AGULP model (Accounts -> Global Groups -> Universal Groups -> Local Groups <- Permissions. Inner groups inherit from outer – local group in this case): B5, 83-84

Active Directory Accounts & Groups (AD Users & Computers – primary tool to manage users, computers, & groups. Groups in Active Directory: global admin & enterprise admin + distribution group & security group + Domain Local Group – local group synced with AD. + How to create groups): B5, 85-87

Shared Folder Permissions (folders shared using Server service & SMP protocol. Shared folders accessed through: Network Places, drive letters, run line, shortcuts. Shared folder has its own DACL – simpler & no inheritance. + how to share folders): B5, 88-90

Hidden & administrative Shares (& at the end to make it hidden. IPC$ for inter-process communication. Get-SmbShare powershell cmd to see hidden shares or with Computer Management Tool. + Admin shares: root folder of drive-lettered volume & %SystemRoot% folder – can be disabled): B5, 91-92

Combining NTFS & Share DACLs (If user given different permissions to same file/folder through NTFS & Share permissions: most restrictive permission applies. But complexity is a vulnerability. Thus, many admins choose to focus only on NTFS permissions): B5, 93-94

REGEDIT.EXE (modify keys & values in registry – stores config settings. + Types of registry values & making your own keys & values): B5, 95

Remote Registry Service (allow remote access to registry. Disable for more security but needed for management tools – Instead set permissions in “winreg” key. Note: winreg subkey, “AllowedPaths” overrides share permissions on winreg): B5, 96-97

Modifying key permissions (which should have permissions changed? No simple answer. Use automation tools such as INF security templates & Group Policy): B5, 98

Active Directory Permissions (edit permissions with Active Directory Users: permissions inherited by users under the OU. Can delegate authority over everything in AD through permissions on properties of AD objects + can track who is doing what. Delegation of Control Wizard simplifies this): B5, 99-100

Privileges 1 (not related to a particular object – general capabilities. Types of Privileges: Take Ownership, Force Shutdown, Access from network, Act as OS, Add workstations, Allow remote logon, Back up files. + see what privileges you have): B5, 101-102

Privileges 2 (Rights: logon privileges. Log on locally, access from network, log on through RDP): B5, 103

Privileges 3 (Take Ownership: very dangerous & only admins have by default. Moderate it): B5, 104

Privileges 4 (Backup & Restore privileges: circumvent NTFS Permissions privileges – dangerous. + Tips on managing): B5, 105

Privileges 5 (Debug Privilege: allows debugging with apps like WinDbg or IDA pro but also allows DLL Injection. Cain can use DLL injection to steal pass hashes & LSA secrets. Metasploit can use privilege escalation to acquire Debug privilege. Don’t give admin to anyone – it has debug by default): B5, 106

BitLocker Overview (Disk encryption using AES. Benefits: Boot-up integrity check – needs TPM chip, sector-level encryption of hard drive. Supports USB & Thunderbolt drives, emergency recovery pin if needed, supports some self-encrypting hard drives. Requires Ultimate/Enterprise Edition or Server 2008+. Requires two NTFS volumes: Boot & System – can’t be encrypted. Performance loss = 5% non-self-encrypting drive & 1% self-encrypting. + turning on BitLocker): B5, 107-108

BitLocker with TPM (TPM performs: encryption, hashing, random key generation, secure key storage, +. Used also for smart card & biometrics encryption – not just BitLocker. fTPM for AMD, PTT for Intel, vTPM for VMs. + how to turn on & utilize): B5, 109-110

BitLocker Emergency Recovery (48-digit password. Used to decrypt volume even if TPM is damaged, Pin forgotten, or USB token lost. Group policy to force backup of recovery password to AD): B5, 113-114

BitLocker with UEFI Secure Boot (Unified Extensible Firmware Interface – replaces older BIOS. Checks digital signatures of binaries & firmware + early loads antivirus drivers. Allows detection of rootkits without TPM. + Allowed CA certificates & Disallowed hashes. + Requirements): B5, 115-116

**Enforcing Security Policy**

Security Templates (stamp containing settings that can be immediately applied to multiple machines for consistent settings. + What is in a template): B5, 122

Security Templates 2 (where stored, INF extension, edit: notepad & MMC Security Templates): B5, 123

Security Templates 3 (begin with templates from Microsoft, US Government – FCC & USCGB & DISA STIG, or CIS): B5, 124-125

SCA Snap-IN (easy way to apply template. Only works on local PC. No undo! + How to do): B5, 126-127

SECEDIT.EXE (CLI version of SCA tool. Scriptable with PowerShell to make applying templates much easier. Yet still can’t be used to apply templates across the network): B5, 128

Local Group Policy Object (set of config changes – every windows PC has local GPO. Computer config applies when no one logged on. User config applies when current user logged on): B5, 129

Local GPO Security Settings (most important security settings on system. Can import security template or edit by hand + instructions): B5, 130

Local GPO Scripts (many languages default like PowerShell, CBScript, Jscript, batch files. Also interpreters available for Perl & Python. Can be executed at startup/shutdown – run as OS, or logon/logoff – run as user. Can control every aspect of OS): B5, 131

Local GPO Administrative Templates (Custom Registry GUIs to make registry editing user-friendly. ADM/ADMX/ADML extensions – essentially text files): B5, 132-133

Domain GPOs/Group Policy Objects (security templates applied over the network. Downloaded on startup, shutdown, logon, & logoff. Stored in AD & replicated to all domain controllers. “Default Policy” applies to everyone & overrides local GPO): B5, 134

Group Policy Management Console (edit Domain GPOs. + How to open & manage GPMC. + How to import INF security template into GPMC): B5, 135

\*Recommended GPO Settings\* ENTIRE LIST OF RECOMMENDED SETTINGS (Security Options & Administrative Templates sections + accessing them): B5, 136-139

Password Policy (recommended: password history – 24 passwords, min password age – 1 day, min length – 8 characters, min complexity – three categories of characters but no longer recommended, max age – no longer recommended): B5, 140-141

Account Lockout Policy (to stop brute force. Min recommended: lockout duration – 120 mins, lockout threshold – 5 attempts, reset lockout after – 45 mins. Beware! Lockout policy can become DoS vulnerability!): B5, 142

Kerberos & NTLMv1 (Kerberos – default, faster, more secure, but requires AD membership & network access to domain controllers. NTLMv1 – no need for AD domain but can be sniffed with CAIN to reveal password hashes & cracked with RainbowCrack): B5, 143

Kerberos & NTLMv2 (NTLMv2 not vulnerable to sniff-and-crack if have passphrase, still vulnerable to SMB relay. GPO Security option: LAN Manager authentication level important settings: Level 3 & 5. Kerberos should still be used): B5, 144-145

Credential Guard (protects \*not all\* credentials in memory from kernel-mode malware & penetration testing tools like Mimikatz. Very specific hardware & software requirements but essential!): B5, 146-147

Protecting Admin Accounts (require MFA, strong passphrase, enable credential guard, require Kerberos & NTLMv2, two accounts or workstations to each admin – normal & admin, blank passwords used in console logons only, audit access to admin users & groups, rename default Admin account & create “honeypot” admin account): B4, 148-149

AppLocker (regulate processes & scripts allowed to run. Rules defined by: program hash, path of program, code certificate, user’s group. Rules in XML + How to create, manage, & test rules): B5, 151-152

User Account Control (run in low-privileged account then temporarily raise privileges when needed. How to turn off, how it works, folder/key virtualization – rare case when virtual admin privileges given without asking, & how to launch as admin. + UAC Group Policy Options): B5, 153-154

\*memorize\* Key Protocols ports & uses (SMB – TCP/139/445, RPC – TCP/135, LDAP – TCP/389/636/3268/3269, Kerberos – TCP/UDP/88): B5, 155-156

Key Protocols ports & uses 2 (DNS – TCP/UDP/53, RDP – TCP/UDP/3389, SQL – TCP/UDP/1433/1434, NetBIOS – TCP/UDP/137 & UDP/138 & TCP/139 & TCP/UDP/1512 & TCP/42, IPSec – UDP/500/4500 for IFE & Protocols 50 and 51 for ESP & AH): B5, 157-158

Windows Firewall Management Tool (Launching & containers): B5, 159

Windows Defender Firewall Pros & Cons: B5, 160

Network Location Types (Domain, Public, Private. Domain least strict, public most strict. How to edit category of connected network, default settings for each category, & logging info): B5, 161-162

Managing Firewall Rules (Secure Connection vs Encrypted Connection. Manage with GPO, PowerShell, or NETSH.EXE. Types of rules, rule order processing, & how to access rules): B5, 163-166

Internet Protocol Security – IPSec (Not just for VPN! + Uses – mutual authentication with Kerberos or certificates, AES encryption, digital signatures. & Benefits): B5, 168

CLI IPSec Tools (PowerShell & NETSH.EXE. + showing all rules & creating rules): B5, 169

IPSec & Group Policy (all IPSec settings can be managed in Group Policy & PowerShell – means possible custom configs for thousands of computers): B5, 170

IPSec Group Policy (How to: Request but not require IPSec, edit GPO assigned to an OU, +): B5, 171

Remote Desktop Services – RDS (remote server management, users control workstations from home, help desk, hundreds of remote users on one server without VM): B5, 173-174

Microsoft’s Thin-Client/RDS App – MSTSC.EXE (thin client app is used to control remote desktop, MSTSC.EXE is Microsoft’s default. + Clients for Linux, Apple, & Android): B5, 175-176

Remote Assistance & Remote Desktop (enabling & disabling RD, who can use RD. RA features – for remote help, uses encrypted file to connect, how to send invitation file. Easy Connect – alternative RA using password to connect instead of encrypted file. Invite file recommended): B5, 177-179

Remote Desktop Protocol – RDP (TCP/UDP/3389. Sends keyboard, mouse, touch input & returns graphics, sounds to thin client. Also transports user authentication to server. + RDP Encryption levels & Authentication levels. + Network Level Authentication – NLA. + Managing in Group Policy): B5, 180-181

Remote Desktop Protocol Services Best Practices: B5, 182-183

**Microsoft Cloud Computing**

Microsoft Is Going All-In on Cloud: B5, 189

Types of Cloud Computing (Infrastructure -IaaS, Platform -PaaS, Software - SaaS, Desktop -DaaS): B5, 190

The Big Three (Azure – identify & authentication, OneDrive – file storage, 365 – the Suite. Categories: Free, Hybrid, Full Cloud. + Other Microsoft Cloud Services Glossary): B5, 192-195

Microsoft Azure Active Directory – MAAD (user & device account database for authentication & access control. Not just for Microsoft offerings, also for third-party devs. + MAAD vs AADDS): B5, 196-197

Syncing Azure AD with On-Premises AD (Using Azure AD Connect – free tool allowing one-way or two-way syncing. If pass hashes can’t be synced can use ADFS instead. Users now have Azure AD/Microsoft Accounts & on-premises AD accounts.): B5, 198-199

Azure AD Single Sign-On (link domain or local account to Microsoft Account. Syncs your data to OneDrive giving planetary roaming user profile. Can be disabled & configured in Group Policy): B5, 200-201

Limit & Monitor Admin Roles in Azure! (Global Admin is All Powerful! + Other important roles to secure. & Most important actions to protect accounts. + PowerShell pros & cons): B5, 202-203

Enforce Endpoint Security (weakest link in Cloud. Traditional practices must still be applied! Endpoint Hardening, Remote Wipe lost or stolen devices, Mobile Device Management & Intune): B5, 204

Microsoft 365 Defender (Suite of products – Defender for: Endpoint, Office 365, Identity, Cloud App. + Terminology, Security Center Web portal, & Licensing issues): B5, 205-207

Endpoint Security: Require MFA for Mobile! (Four options: call, SMS, Pin to MS Authenticator App, or Push notification to Authenticator app. + details): B5, 208-209

Microsoft Intune (internet-friendly Group Policy & cross-platform. Requires Azure AD for users but not devices. + Device Compatibility list & Enrollment Process): B5, 210-211 & 212-213

Azure Conditional Access (restricts app & data access using custom rules. Requires Azure AD Premium & user authentication. MFA in mobile devices with conditional access will need lots of testing + with different browsers. & Conditional Access process): B5, 214-215

Conditional Access Selection Criteria (Block Criteria based on: group membership or roles, IP address, type of device, whether device registered in Azure AD, app used to access Azure, risk score): B5, 216

Additional Requirements (If not blocked, check for additional defined requirements: MFA, Intune-managed device that is compliant, Windows PC joined to an AD domain, approved client app): B5, 217

Testing (Step 1: Report-only mode test, Step 2: What-If Tool to simulate user activity, Step 3: apply policies to user & test as that user, Step 4: rollout to small portion of users & alert helpdesk. If goes well expand): B5, 218

Example actually creating policy – Steps: B5, 219

Azure Policy & Initiatives (used to audit existing Azure resources at large scale. Policies applied automatically through change detection or manually by scanning existing resources & comparing against policies. Initiatives are collections of policies – over 500 policies built-in. Initiatives exist for: PCI, ISO 27001, HIPAA, FedRAMP, & CIS, + more & can create your own): B5, 220-221

Azure Policy Assignment & Effects (assign initiative & choose action/effect for each policy: Deny, Audit, Modify, Append): B5, 223

Azure Policy Key Vault (service to secure secrets such as certificates, keys, or strings. Access occurs over HTTPS. + Azure Policies & Best Practices: Use Key Vault, don’t use one vault for everything, role-based access control, restrict access on IP address or Azure virtual network, use managed identities not manual authentication, enable purge protection for 90 days, regularly backup critical vaults, log access, use Azure Policy to audit & enforce proper use): B5, 224-226

Azure Monitor (Collect, Analyze, Visualize, & monitor log data in Azure. Source of log data: Azure VMs, On-Premises VMs & devices, Azure Services, Apps & Scripts, IoT devices. Data queried in KQL or exported & queried with SQL): B6, 227-228

Azure Monitor – Agents & Workspaces (Agent is what collects log data from VMs & PCs – log data sent over HTTPS. “Application Insight” must be installed to collect logs from apps. Workspaces are database tables to hold Azure Monitor log data. + workplace security & tables. + Terminology: metrics vs tracing vs telemetry vs logging): B5, 229-230

Activity Log (Alerts, Metrics, Logs, Service Health, Workbooks, Insights/agents): B5, 231

Azure Sentinel (SIEM - & connector examples + SOAR system – handler investigates manually or with playbooks/automations. Built on top of Azure Monitor) + portal: B5, 232 & 233-234 & 235 & 236

Azure Security Center (pane of glass for organization-wide security score. Azure Monitor & Policy alerts feed here. Built-in vulnerability scanning & security Initiative. + auto-provisioning, third party AV integration, auto vulnerability scanning in VMs, Role-Based Access Control + Requirements) + Portal: B5, 237-238 & 239

**Automation, Logging, & Auditing**

Automation Overview (many scripting & auditing tasks require scripting & command line skills. + Good for job security. Top for Linux: Bash & Python. Top for Windows: PowerShell): B5, 244

PowerShell Overview (Command Shell, easy to learn, replaced CMD, on Win7+, for windows & Linux & Mac, popular for Azure automation. + Requirements, Remoting, JEA, Transcription Logging): B5, 245-246

PowerShell & PowerShell Core (.NET vs .NET Core, only Windows vs Win & Mac & Linux, default vs not-default yet, closed source vs open source, more cmdlets vs less cmdlets. Frozen vs The Future): B5, 247-248

PowerShell Commands (Pipe Symbol: get-process, get-service, dir, get-winevent, get-ciminstance, get-help. + switches): B5, 249-250

Command Line Tools List (.EXE tools list HUGE. + How to change PATH variable to add tools): B5, 251-256

WMIC.EXE (get or set config data for settings using WMI service on local/remote. Show OS version, show shared folders, show programs that run at boot. + brief vs full. + PowerShell equivalent): B5, 257

Network Configuration Tools (WMIC, NETSH, GETMAC, IPCONFIG, ROUTE, NET, NETSTATE, NBTSTAT. & PowerShell cmdlets Replacements if have Win 8+): B5, 258-259

Windows Subsystem for Linux – WSL (Linux runs directly – not a VM, container, or emulator. Not everything works. Requires Win10 1709+. + How to enable, install, & commands to start. + WSL2 info): B5, 260-262

Scheduling Tasks – Task Scheduler (automate apps & scripts launching based on intervals: daily, weekly, monthly, once, at startup, logon, idle for X. + CLI management commands & Scheduled Task Accounts): B5, 263-264

Azure Tools (Azure Web Portal, PowerShell Az Module, Azure CLI, Cloud Shell, Azure Resource Manager Templates – ARM Templates): B5, 265-267

Azure Cloud Shell (can have multiple simultaneously – each connect to different container. Runs Ubuntu, persistent home directory, no default access to VMs but can be enabled, included script editor in toolbar): B5, 268-269

Azure Automation (PowerShell or Python scripts or Graphical Runbooks. Sandbox Worker Jobs – container in Azure. Vs. Hybrid Worker Jobs – run inside Windows/Linux VM anywhere & poll Azure for jobs using HTTPS TCP 443. + Runbook Workers/scripts, Role-Based Access Control, Azure Automate & Azure Functions): B5, 270-273

Windows Logging Config (Event Viewer to view logs across three categories: Application, Security, System. + Writing your own entries): B5, 274-275

Security Event Log & Audit Policies (Enable logging to security log, Types of events, & managing policies with Group Policy & ADITPOL.EXE): B5, 276-277

Interesting Event Log ID Numbers (5156, 4688, 4720, 4732, 4724, 4625, 4740, 1102. + Enabling policies to log them): B5, 278-279

System Access Control Lists – SACLs (NTFS & Registry: SACL defines which users & groups have their actions logged. How to enable object auditing for NTFS & Registry. SACL inherits from parent folders): B5, 281-282

Log Size & Wrapping Options (size of log determined by rate of new events & wrapping options. How to change max size & wrapping options: overwrite, archive, clear manually): B5, 283-284

What To Log (only necessary! Proprietary data, %SYSTEMROOT% & %PROGRAMFILES%, Take Ownership & Change Permissions, Printers. Use Templates especially for Registry): B5, 285-286

**Linux Fundamentals**

Market Share, Criticality of Linux, & Linux vs Windows (Linux is small share but used in critical systems so is important. + all appliances use Linux): B6, 5-7

Windows & Linux: Desktops & Servers (Win from desktop to server, Unix/Linux from server to desktop): B6, 8

Main Linux OSs (Ubuntu – dependability, Fedora – Security, WSL, macOS – BSD): B6, 10-11

WSL (uses virtualization to run Linux kernel inside light VM): B6, 13

Linux Vulnerabilities (Shellshock – privilege escalation, SambaCry – remote code execution, Baron Samedit – gain root access, Sudo Bypass – run as root without privileges): B6, 14-15

MacOS (second most popular OS, uses BSD, built on XNU Kernel, security built-in, many features cannot be disabled, most network services disabled by default): B6, 16

Security Advantages of each OS (Windows & Linux are essential. Windows for wide-spread use, Linux for the tools exclusive to it): B6, 17

OS Overview (Kernel, Userland, Hardware): B6, 18

Kernel (manages hardware & executing, loaded into memory at boot, brains of OS): B6, 19

Shell (Command line interpreter to run programs. Provides interface to the system. Vs Terminal Vs Console): B6, 20

Examples of Shells (Unix, DOS, & Windows. Difference is in scripting/programming language built in): B6, 21

Logical Linux Filesystem (main directories): B6, 22

Physical Filesystem (Logical filesystem composed of multiple physical partitions. 16 different partitions called “slices”. Possible to create single partition & ignore other slices but bad for: causing reliability problems with one subsystem, makes backups harder, disables possibility of setting different security options to different partitions. + Filesystem Security Goals): B6, 23-24

Disk Monitoring – df command (columns explained, reserve space feature – fake full, & info about swap partitions in use with swapon -s command): B6, 25

Filesystem Security Options & rules (read only, noSUID, noDev. RO or noSUID mostly. noDev everything but root): B6, 26-27

Linux Unified Key Setup – LUKS (hard disk encryption. Standardized, secure management of user passwords, easy transport/migration of data, platform independent. Cryptsetup utility to setup): B6, 28

File Attributes (ls -l & ls -ld to view attributes for a file. + details in each column): B6, 29

File Permission Modes (Symbolic – RWX-, & Absolute – 4210): B6, 30

Other Permission Bits (sUID – run as owner, sGID, Sticky – only owner can delete files. + representation in Symbolic & Absolute modes): B6, 32-33

Files vs Directories Permissions (Read, Write, Execute. Setuid. Setgd, Sticky. Different behaviors for files vs Directories): B6, 34-35

World-Writable Directories (used by programs to hold intermediate results. Compromise = Trojan. Avoid when possible. Always use sticky bit if must use): B6, 36

SUID/SGID Programs troubles (most common weakness to get privileged access. E.g., SUID root copy of UNIX Shell on USB drive. Monitor with “find” command & raise alarm if new SUID/SGID appear. + commands): B6, 37-38

Umask & Chmod (Umask – set default permissions for new files. Chmod – change permissions. + How to use commands Symbolic or Absolute mode. Note: Umask “subtracts” in absolute mode): B6, 39-40

Chown & Chgrp (Chown – file ownership & group ownership. Chgrp – group ownership. Chown – user & group ID or names. + commands. + superusers directories/files & normal users only files but without SUID & GUID permissions): B6, 41

User Accounts & Groups (UID & GID are what are important. Stored in /etc/passwd & shadow. Not always unique like Win SIDs – file sharing problematic): B6, 42

The Superuser vs Normal user – features & issues (many users can have UID0 – won’t know who did what in logs since all will log as UID0): B6, 43

Privilege Escalation Methods (Login as root – dangerous/block over network because no accountability, Su command – needs password but can log, Sudo – list of users allowed instead of password & more granularity + better logging compared to Su): B6, 44

Sudo & Sudoers (Sudo allows specified users to obtain root privileges for commands specified in Sudoers file. + details into Sudoers): B6, 45

“System” Accounts (accounts for services & apps. Have low UID numbers, attackers activate them as backdoors. If not using app or service, remove or block account + pros & cons for each): B6, 46

Passwd & Shadow File (Passwd – old & insecure, Shadow – new & secure): B6, 47-48

Pluggable Authentication Modules – PAM (Libraries that handle Authentication. Four management groups: Authentication, Password, Sessions, Accounts. + Config files in /etc/pam.d & format): B6, 49

Password Policies (/etc/pam.d/system-auth. Manage: password requirements, password history & reusage policy, Lock after x login failures, & unlocking. Note: don’t set login lock for root): B6, 51-52

Adversary Common Threats & Hunting (ls, ps, netstat: Baseline system & re-run frequently to find changes): B6, 53

Tail command (continuously monitor a file. + options): B6, 54

Automating scripting with | & grep (| - piping/sending output from command as input to other command. Grep – search for patterns in input file or input piped to grep. + example): B6, 55

Linux Services Overview (apps running in the background. Referred to as Daemons. + commands to list services, active services, & searching for services): B6, 56

INIT (first process – responsible for setup of user environment & is parent to all processes. Linear & slow, only on bootup & shutdown. + Runlevels – 6 or 7): B6, 57-58

Systemd (software suite replacement of INIT daemon. Supports parallel processing, monitors services after boot, faster boot, SELinux integration, no syslog – journald, No cron – calendar timers, BSD not supported, Not compatible with INIT managers, Daemons integrated. Includes: initd, journald, logind, networkd. + Commands): B6, 59-62

Cron service scheduling (start action in background at set time. Crontab file stores jobs – checked every minute. Works with system clock. Employable by system users): B6, 63

Linux Package Management (packages – more efficient way to install applications since include all the dependencies needed. Packet Management Tools – monitor updates & automatically upgrade apps, Features: Download Validation, Installation of dependencies, Binary format, Standard install locations, easier user experience, Verification of installation): B6, 65

APT – Advanced Packaging Tool (most common, used on Ubuntu & Debian, others exist. + commands): B6, 66-67

**Linux Security Enhancements & Infrastructure**

SELinux (loadable kernel module designed for security – controls what programs can do within the Linux system. MAC system built on least-privilege. Adds more security if default permission is “allow” & want more finite control over security): B6, 70-71

AppArmor (kernel module to restrict capabilities of programs using static analysis & ML-based tools. Can detect zero-days too. + access mode permissions list & example): B6, 72-73

Sysctl (interface to modify kernel settings at runtime. Used to harden Linux OS. Info + commands. + ways to edit variables): B6, 74

Sysctl Hardening – ASLR (Address Space Layout Randomization – prevent attackers by hiding memory address of functions including vulnerable functions. Effectiveness depends on randomized address space. Note: remove setarch): B6, 75

Modprobe – Disable Unneeded Kernel Modules (1. Find module with ‘find’, 2. ‘modinfo’ to check dependencies, 3. Disable with install option or /bin/false): B6, 76

Dynamic Loading After Boot & Disabling (gives kernel flexibility – risks rootkit injected after boot with root privileges. Disable is good but not guarantee since static kernel can be injected. + disabling): B6, 77

SSH Hardening (Steps to harden + uses for SSH): B6, 78

SSH Multifactor Authentication – must be used (options + info into Google Authenticator): B6, 79

Hardening Scripts Overview + Pros & Cons (default install is not always secure. Script is automated way to secure system. BUT too much security is dangerous! Must not blindly apply!): B6, 80-81

Lynis & Bastille (Bastille good for hardening & basic auditing. Lynis good for robust & detailed hardening audit. Audits software patch, vulnerability, & malware scanning. Lynis Collector for many systems): B6, 83

KEY LOG FILES – *IMPORTANT* (UTMP, WTM, BTMP, DMESG, Messages, MailLog, Secure): B6, 84-87

Syslog (standard for message logging – facility code to specific program logging. Commonly found in network devices. + Message Format: 89 – no standard for timestamp = hard to normalize & correlate. + Severity levels – not standardized = further confusion): B6, 88-90

Syslog Security Issues (Sender Authentication – not strong, Message Confidentiality – no encryption, Message Delivery & Replays – normal activity can be recorded & replayed to fool): B6, 91

Syslog Standards (BSD – legacy, 1024 bytes max, time precise in seconds. IETF – preferred & newer, independent of transport, not limited in size, timestamp precise in sub-seconds): B6, 92

Syslog-NG (replacement to syslog, additional filtering, send with TCP, additional security to remote system logging, advanced features like buffering & Syslog over TLS. + designed for multiple OS & drop-in replacement for Linux Syslog. + Commands to switch & enable in Linux & filtering): B6, 93

LogRotate (prevent filesystem from filling up – renames & compress logs once condition met. Configure in .etc/logrotate.conf – start with global options used as default. Later definitions overwrite earlier definitions. + example definitions/directives. + include command for reading commands in other files): B6, 94-96

Centralized Logging (Protects against log wiping & easy to search and scan. DOS possible – allow only recognized machines/rotate logs fast/have lots of storage, needs large storage, one machine holds sensitive info – becomes target: set syslog only receive/SSH daemon only accept login from few IPs): B6, 97-98

Auditd – access monitoring & accounting (monitor access to everything – kernel level. Logger for SELinux. Very detailed, granular filtering, no syslog so some may not work. Setup is easy if know files & system calls to monitor – find on web-docs like CIS audit.rules. Ausearch – granular search, Aureport – easier Ausearch, Autrace – trace process of binary & bind to Ausearch): B6, 99

Auditd creating rule (adding audit entry via auditctl + flags/options: -l,-s,-b,-f,-R,-a,-D,-e, -w = write & -W = delete): B6, 100

Auditd Search & Report Audit Logs (Ausearch options: -m, -start, -k, -a, -f. -i = interpret numeric entities as text. Aureport options: --summary, -f, -I): B6, 101-103

Auditd Example Rules (adding kernel watch, removing watch, monitor system calls always): B6, 104

Linux Firewalls (cheap network-based & host-based firewalls on same software. Allows scalable deployment, lower costs, and more firewalls. + difference between Network & host firewalls): B6, 105

IPtables – built-in Firewall (allows configuring tables, rules, & chains. Stateful firewall & NAT capabilities. + example commands): B6, 106-107

Rootkit Detectors (look for indicator of compromise in how RK operates. Works for file-level but better with kernel-level. Stealthier rootkit = less accurate detection. Known for false alarms so only initial indicator): B6, 108

RKHunter (detects rootkits, backdoors, & local exploits by comparing hashes of files online. Setup with crontab or timers to schedule checks. + commands): B6, 109

Chkrootkit (checks for suspicious processes & known bad files. Command: chkrootkit. RKs can target checking scripts & make them lie though. + Common false positive on mail server): B6, 110

Configuration Management Tools (configure systems the same way. Like Windows Group Policy. Set hardening standards, control packages & updates. Commercial & OSS options. + Ansible example tool – provisions servers/VMs locally or in-cloud, SSH between control & managed nodes, playbooks & inventory): B6, 111-112

**Containerized Security**Containers vs Virtual Machines (Containers – Kernel/OS virtualization, contains app & needed files, compact & portable. VM – hardware virtualization, entire OS kernel, large size, hypervisor as abstraction layer): B6, 116-117

Linux LXC (open standard for creating containers. Between chroot – isolates executable but hard with shared libraries - & VMs. Allows isolation at kernel level & contains app + requirements): B6, 118

Containers core- Cgroups &Namespaces (Cgroups – manages control & governance of resources mainly CPU & RAM. Namespaces – obtaining resources an application needs & perform isolation): B6, 119

Docker vs LXC (LXC – Original, OS level virtualization, multiple OS on single host, Cgroups & namespace. Docker – Single application LXC containers, more portable, easy to deploy/replicate/move, single process & state-less, open-standards, market-leader, apps run same in all environments): B6, 120-121

Docker Images (container is instance of image which is combination of filesystem & parameters defined in Dockerfile stored in public/Dockerhub or private Docker registries): B6, 122

Docker Swarm Mode (OSS container orchestration platform – native for & by Docker. Multiple hosts run in swarm mode & act as managers & workers. Built into Docker Engine): B6, 123

Kubernetes (OSS container-orchestration system for automated deployment, by google, maintained by CNCF, manages container deployments at scale & high availability across nodes. Complex): B6, 124

Docker Security Best Practices (use minimal trusted base images, do not run apps as root, cryptographically sign & verify your images, monitor images for known vulnerabilities, don’t store secrets in images, containers shouldn’t share network stack of host, don’t enable unused services, don’t allow privileged flags. + Docker Bench for Security Tool): B6, 126

Vulnerability management in Docker Images (images can contain vulnerabilities. Tools detect: *Clair* – API-driven & built into, Docker Hub alternative, Quay.io. & *Anchore* – CLI tool or docker container for CVE-based vulnerability reporting. Policies based on allowlists, denylists, credentials & file contents): B6, 127

CIS Docker & Kubernetes benchmarks for security: B6, 128

Terraform (declaratively define infrastructure then create, update, & manage resources using code – JSON. Known as Infrastructure as Code – IaC. Easy to re-create environment from one provider to another with Terraform. + Commands & Life-cycle: Init, plan, apply, destroy): B6, 129-131

Terraform Security Best Practices (.tf*State* file contains secrets, don’t commit! Encrypt when possible by backend. + run pre-apply & post-apply checks with tools mentioned. + use variables): B6, 132

**AWS Essentials, Controls, Best Practices**

AWS Well-Architected Framework (guide into best practices. 6 pillars: Operational Excellence, Reliability, Performance Efficiency, Cost Optimization, Sustainability, Security – focus: 7 principles): B6, 136

Security Principle 1: Implement a Strong Identity Foundation (IAM: 137-138, Identity Federation - temporary access granted by central Identity provider: 140, Amazon Cognito: 142, AWS Console: 143, AWS CLI: 144, AWS Access Keys: 145, implementing strong identity models: 146, Permissions Management: 148): B6, 137-149

Security Principle 2: Enable Traceability – monitoring & detection (Amazon CloudWatch: 152, AWS CloudTrail: 154, AWS Config: 156, GuardDuty & Security Hub: 158): B6, 150-158

Security Principle 3: Apply Security at All Layers (Networking – Infrastructure: 159, High Availability: 160, Defense-in-depth steps + Firewall Manager: 161-162, Network Firewall: 164, AWS Shield & Web App firewall - WAF: 165, CloudFront: 166): B6, 159-167

Security Principle 4: Automate Security best Practices (AWS CloudFormation): B6, 168

Security Principle 5: Protect Data in Transit & at Rest (Data Classification - Macie: 169-170, Ensure Protection – certificate manager + tokenization & encryption: 171-172, Key Management Service & CloudHSM + CMKs: 173): B6, 169-173

Security Principle 6: Keep People Away from Data (managed services, software integrity validation, automate compute protection): B6, 175

Security Principle 7: Prepare for Security Events (response phases: Educate, Prepare, Simulate, Iterate. + NIST SP 800-61): B6, 176-178

AWS Cloud Security Tools Summary (IAM: IAM/Directory Service/Organizations, Detective Controls: CloudWatch/TrustedAdvisor, Infrastructure Protection: GuardDuty/WAF/VPC/Shield, Data protection: Macie/KMS/CloudHSM/Elastic Load Balancer, Incident Response: CloudTrail/CloudWatch Alarm/SNS, Application Protection: Inspector): B6, 179

**macOS Security**

macOS Overview (ontop of XNU Kernel – Darwin + Aqua & Finder. Download using App Store or Apple Disk image files. Apps developed in XCODE – supports C, C++, Java, AppleScript, Python, Ruby, +. Code in your language & transform into XCODE): B6, 182

macOS Security Features 1 (Privacy controls – restrict app access, Keychain – password manager, auto strong passwords – suggests when signing up for services on safari): B6, 183-184

Security Features 2 (auto updates, Gatekeeper – verifies validity of applications, Safari anti-phishing & download protection): B6, 185

Security Features 3 (XProtect – anti-malware using denylists of signatures & YARA rules, Find My – lost devices service using IP address or Bluetooth network, Firewall – block ingress, not egress): B6, 187-188

Security Features 4 (FileVault – disk encryption using HFS+, Sandboxing & runtime protection – sandbox apps from AppStore + NOEXEC & ASLR, T2 Security Chip – separate processor/secure enclave to handle sensitive operations such as secure boot/biometric/crypto to prevent rootkit manipulation): B6, 189-190

Securing macOS (Turn on firewall, turn off unneeded services, limit service sharing, use secure file sharing, monitor access lists & update, use password assistant, use AV software + XProtect): B6, 191

macOS vulnerabilities (BuggyCow – privilege escalation by XNU mem bug in 2018, GateKeeper Bypass – run untrusted apps bypassed by GateKeeper & XProtect): B6, 192

Malware on macOS (FlashBack – trojan as Flash Player Installer, spread on websites exploiting Java vulnerability, 2012 600,000+ infections. CrescentCore – trojan as Flash Player Installer, signed with dev cert so passes Gatekeeper, detects virtual environments & AV, June 2019): B6, 193

**Workbook Index**

TCPDump (commands: 4, -X option: 10, -a option: 12, -XX: 15, -n: 15): WB, 4-15

Wireshark (filters – ftp, ftp-data, tftp. TCP stream – follow to see readable communication. UDP stream – same story): WB, 16-28

Aircrack-NG (aircrack, aireplay, airmon, airodump: 30. Cracking WEP: 32, Cracking WPA2: 34): WB, 29-36

Hashcat (Attack Modes: 38-39, Important files: 39, steps to crack: 40-42, custom rules: 43-44, Cracking Bitcoin Wallet: 46-47): WB, 38-47

Cain & Abel (Cracking SAM passwords: 52-63, Cracking Cisco Type 7 password – insecure: 64-66, Cracking MD5 router config: 66-71): WB, 52-71

AppLocker (Rules information, conditions, & modes: 73-74. Enabling required Application Identity Service: 74-78, Creating new user: 78-82, Configure AppLocker: 82-100): WB, 73-100

Nmap (options & flags: 103-109, Port Scanning: 109-112, OS & Application Version Scanning: 112-114, Nmap Scripting Engine - Lua, commands, & snmp-brute script + snmpcheck: 114-117): WB, 103-117

Malicious Software Analyzing with strings & buffer overflow assessment: WB, 120-128

Command Injection (restricted shell: 130, implementation: 132): WB, 130-134

Hping3 (options/IP options/TCP & UDP options: 137-139, Crafting & monitoring packets: 141-143, spoofing IP address: 144): WB, 137-144

Image Steganography (info on tool: 146, Hiding text: 148-157, Hiding files: 158-163): WB, 146-163

GNU Privacy Guard – GPG (replacement for GPG. Intro info: 166-172, Decrypt: 174-177, Validate signature: 178-180, Encrypt: 181-183): WB, 166-183

Snort (info & example rules: 185-188, Running Snort: 190, Reviewing Snort Logs: 192-195, Running snort on PCAP file: 196): WB, 185-197

Hashing (info on hash algorithms: 198-199, hashing algorithm tools: 200-201, automating file integrity check – Tripwire/Python example + diff – finds differences in files: 205-208): WB, 198-209

Process Hacker (info: 210, Tabs & color codes: 211-217, Filter/strings in virtual memory address – useful for finding purpose of malware: 218-219, Inject a DLL: 222-223, The Terminator: WB, 224): WB, 210-226

NTFS Permissions Reporter (Info: 227, Running: 228-230, Remediation & System32 danger info: 232-239): WB, 227-240

SECEDIT.EXE (Info + INF Sec templates: 241, Examine & Compare INF Sec Template to PC: 243-246, Applying INF Sec Template: 246): WB, 241-248

PowerShell Scripting (Processes/Services & Variables + Save CSV/HTML/Out-GridView: 252-260, Filesystem – dir/sort/ copy/GetHash/GetContent: 260-265, Select-String – PowerShell’s “Grep”/Find: 246, Select-Object – filters properties: 255-256, Windows Management Instrumentation Service – to query for info: 266-268, Querying Windows Event Logs: 268-270): WB, 249-272

**Attacks & Pen-Testing**

DOS & DDOS:

Attacks against routers (DOS, DDOS, sniffing, packet misrouting, routing table poisoning): B1, 32-33

802.11 Denial of Service – DoS (“Jamming” most commonly perceived DoS – radio signal to interfere with our waves. Too many DoS methods, can’t stop all. To mitigate, respond quickly – find source, react fast. If can’t, fall back to wired before severe damage): B1, 210-211

Packet Sniffing & Routing Table Poisoning:

Attacks against routers (Packet Sniffing): B1, 32-33

TCPDump and Wireshark use for packet sniffing: B1, 98-100

Attacks against routers (Routing Table Poisoning): B1, 32-33

Attacks against Switches:

Attacks against switches (*CDP info disclosure* – sniffing, *MAC Flooding* – downgrade attack, *DHCP Manipulation* – MiTM, *STP Manipulation* – switch reconfig, *VLAN Hopping* – Network segmentation bypass): B1, 34-36

WPA & Bluetooth Attacks:

WPA2 Vulnerabilities: B1, 197-198

WPA3 Vulnerabilities (Downgrade to WPA2 and Weak ECDH Attack): B1, 204-206

BlueBorne & Bluetooth Vulnerabilities: B1, 217-220

SQL Injection:

Web Application Attacks: B3, 107-109

Buffer Overflow:

Input Attacks: B3, 101-106

OS Command Injection:

Input Attacks: B3, 99-100

Cross-Site Scripting (XSS) & Cross-Site Request Forgery (CSRF):

Web Application Security Issues: B3, 136

Phishing:

Social Engineering Attacks: B3, 47-48

Credential Stuffing, Password Reuse, & Password Spraying:

Password Attacks: B3, 71-72

**Key Ports:**

TCP 20 - FTP Data (File Transfer Protocol Data)

TCP 21 - FTP (File Transfer Protocol)

TCP 22 - SSH (Secure Shell)

TCP 23 - Telnet

TCP 25 - SMTP (Simple Mail Transfer Protocol)

TCP/UDP 53 - DNS (Domain Name System)

TCP 80 - HTTP (Hypertext Transfer Protocol)

TCP 443 - HTTPS (Hypertext Transfer Protocol Secure)

UDP 67/68 - BOOTP/DHCP (Bootstrap Protocol/Dynamic Host Configuration Protocol)

UDP 69 - TFTP (Trivial File Transfer Protocol)

UDP 123 - NTP (Network Time Protocol)

UDP 161/162 - SNMP (Simple Network Management Protocol)

UDP 2049 - NFS (Network File System)

TCP 443 - SSL/TLS (Secure Sockets Layer/Transport Layer Security)

TCP/UDP 514 - Syslog

TCP 139/445 - SMB (Server Message Block)

TCP 135 - RPC (Remote Procedure Call)

TCP 389/636/3268/3269 - LDAP (Lightweight Directory Access Protocol)

TCP/UDP 88 - Kerberos

TCP/UDP 3389 - RDP (Remote Desktop Protocol)

TCP/UDP 1433/1434 - SQL (Structured Query Language)

TCP/UDP 137 & UDP 138 & TCP 139 & TCP/UDP 1512 & TCP 42 - NetBIOS (Network Basic Input/Output System)

UDP 500/4500 for IKE & Protocols 50 and 51 for ESP & AH - IPSec (Internet Protocol Security)