Study Guide

Cloud Application Security for the CCSP ®

# Checklist of Exam Objectives: Areas to Study

## ❏ 4.1 Advocate training and awareness for application security

### Cloud development basics

### Common pitfalls

### Common cloud vulnerabilities (e.g., Open Web Application Security Project (OWASP) Top-10, SANS Top-25)

## ❏ 4.2 Describe the Secure Software Development Life Cycle (SDLC) process

### Business requirements

### Phases and methodologies (e.g., design, code, test, maintain, waterfall vs. agile)

## ❏ 4.3 Apply the Secure Software Development Life Cycle (SDLC)

### Cloud-specific risks

### Threat modeling (e.g., Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege (STRIDE), Disaster Reproducibility, Exploitability, Affected Uses, and Discoverability (DREAD), Architecture Threats, Attack Surfaces, and Mitigations (ATASM), Process for Attack Simulation and Threat Analysis (PASTA)

### Avoid common vulnerabilities during development

### Secure coding (e.g., Open Web Application Security Project (OWASP) Application Security Verification Standard (ASVS), Software Assurance Forum for Excellence in Code (SAFECode)

### Software configuration management and versioning

## ❏ 4.4 Apply cloud software assurance an validation

### Functional and non-functional testing

### Security testing methodologies (e.g., blackbox, whitebox, static, dynamic, Software Composition Analysis (SCA), Interactive Application Security Testing (IAST)

### Quality assurance (QA)

### Abuse case testing

## ❏ 4.5 Use verified secure software

### Securing application programming interfaces (API)

### Supply-chain management (e.g., vendor assessment)

### Third-party software management (e.g., licensing)

### Validated open-source software

## ❏ 4.6 Comprehend the specifics of cloud application architecture

### Supplemental security components (e.g., eb application firewall (WAF), Database Activity Monitoring (DAM), Extensible Markup Language (XML), firewalls, application programming interface gateway (API)

### Cryptography

### Sandboxing

### Application virtualization and orchestration (e.g., microservices, containers)

## ❏ 4.7 Design appropriate identity and access management (IAM)) solutions

### Federated identity

### Identity Providers (IdP)

### Single Sign-on (SSO)

### Multi-factor Authentication (MFA)

### Cloud Access Security Broker (CASB)

### Secrets management

# Exam Essentials: What you need to know

#### Golden Keys:

##### Developers need training to ensure adequate knowledge of application security practices

##### Security must be built into the *entire* Systems Development Lifecycle – not just at the beginning and not just at the end

##### Threat modeling and coding standards can identify application security requirements

##### Thorough application security testing is required to ensure that the security controls are working correctly

##### Secure software can only be built using secure components

##### Secure applications require secure architecture

##### Applications are on the front line of Identity and Access Management

# Important Terminology

#### Data Owner — the individual that is responsible for the protection of data throughout the data lifecycle. An organization may have more than one data owner, but there must always be one individual that is held accountable for each data element

#### Encryption — The process of rendering sensitive data unreadable through substitution and transposition using a mathematical function (algorithm)

#### Confidentiality — the protection of sensitive data from unauthorized disclosure

#### Due Care — the actions taken by a reasonable, prudent person to protect others from unreasonable harm

#### Due Diligence — the enforcement of the actions of due care

#### Integrity — The measure of accuracy or precision of an entity or process

#### Availability — The measure of the criticality of an entity and the value of the entity to supporting a business process

#### Hashing Algorithms — a mathematical function used to detect changes to data and thereby support integrity

#### Non-repudiation — The ability to link actions to an individual entity

#### Threat — Any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, individuals, other organizations, or the Nation through an information system via unauthorized access, destruction, disclosure, or modification of information, and/or denial of service. CNSS 4009

#### Attack — Any kind of malicious activity that attempts to collect, disrupt, deny, degrade, or destroy information system resources or the information itself. CNSSI 4009

#### Asset — an entity with value to its owner

#### Vulnerability — Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited by a threat source CNSSI 4009

#### Residual Risk — Portion of risk remaining after security measures have been applied CNSSI 4009

#### Risk Acceptance — The level of risk within the limits set by the risk owner

#### Information Security Risk — The risk to organizational operations (including mission, functions, image, reputation), organizational assets, individuals, other organizations, and the Nation due to the potential for unauthorized access, use, disclosure, disruption, modification, or destruction of information and/or information systems. NIST SP800-30r1

#### Chain of Custody — a documented record of all actions related to evidence throughout the evidence lifecycle

#### Incident — an adverse event with the potential to affect business mission

#### Social Engineering — the manipulation of a person to induce them to do something they should not do

#### Compliance — proven adherence to standards

#### **API –** Application Programming Interface

#### **DAM -** Database Activity Monitoring

#### **CASB -** Cloud Access Security Broker

#### **STRIDE –** Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Escalation of Privilege

#### **DREAD -** Disaster, Reproducibility, Exploitability, Affected Users, Discoverability

#### **OWASP -** Open Web Application Security Project

#### **WAF -** Web Application Firewall

# Self-Assessment Questions: Test your Understanding

###### The organization has a website that allows customers to purchase products and set up personal accounts. Which process would use a third-party identity provider?

###### Discretionary Access Control (DAC)

###### Mandatory Access Control (MAC)

###### Single Sign-on (SSO)

###### Federated Identity Management (FIM)

###### The organization is conducting a risk assessment that determines the risk of using a cloud provider for hosting a custom application. What type of cloud deployment is most common for this purpose?

###### Private Cloud

###### Infrastructure as a Service (IaaS)

###### Community Cloud

###### Platform as a Service (PaaS)

###### A hidden attack vector that may not have been identified may be due to an insecure::

###### Input

###### API

###### Requirement

###### Cloud Access Security Broker (CASB)

###### The organization is conducting a risk assessment that determines the risk of using a password for remote access. What should the organization do to provide more secure access?

###### Multi-factor authentication (MFA)

###### Single Sign-on (SSO)

###### Service Level Agreements (SLAs)

###### Sandboxing

###### The organization is concerned that excessive network traffic from an application that may have been manipulated by a malicious user. What device may identify this type of problem?

###### Database Activity Monitor (DAM)

###### Sandboxing

###### Application Programming Interface (API)

###### Rogue microservice

###### An application may be insecure due to its use of open-source code that was not adequately tested. How can such code be identified?

###### Buffer overflows

###### Excessive database calls

###### Software Composition Analysis (SCA)

###### Black box testing

###### The resilience of an application to an attack can be determined through:

###### Quality Assurance

###### Abuse case testing

###### Sociability testing

###### Non-functional testing

###### What is the risk of re-introducing old vulnerabilities often related to?

###### Configuration management

###### Version Control

###### Lack of documentation

###### Black box testing

1. What is a key requirement to perform a white box test?
   1. Source code
   2. Executables
   3. Databases
   4. Threat modeling
2. What security principle should an application enforce follow when handling the users of different privilege levels?
   1. Dual control
   2. Encryption
   3. Least privilege
   4. Logging

# Answers to Self-Assessment Questions:

###### D – While FIM is a type of single sign on, answer D is more precise than answer C.

###### D– An argument can be made for several answers here but the most common for hosting a custom application in Platform as a Service

###### B – Especially the use of older APIs may be an attack vector not easily identified

###### A – using MFA increases the security of access control

###### A – a DAM can identify if an application has been manipulated to make excessive database calls

###### C – SCA can identify code that should not have been used due to licensing problems or lack of testing

###### B – in addition to testing for use cases, the tests should include abuse case testing as well

###### B – Version control is often at the root of regression problems

###### A – white box testing allows the examination of the source code

###### A – a common problem with application threads is the loss of granularity in access and allowing a lower level user to access higher level data.