CISA Q&As

Certified Information Systems Auditor

Pass Isaca CISA Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.passapply.com/cisa.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Isaca Official Exam Center

- **Instant Download** After Purchase
- **100% Money Back** Guarantee
- **365 Days** Free Update
- **800,000+** Satisfied Customers

QUESTION 1

The IS auditor has identified a potential fraud perpetrated by the network administrator. The IS auditor should:

A. issue a report to ensure a timely resolution

B. review the audit finding with the audit committee prior to any other discussions

C. perform more detailed tests prior to disclosing the audit results

D. share the potential audit finding with the security administrator

Correct Answer: B

QUESTION 2

Which of the following network management toots should an IS auditor use to review the type of packets flowing along a monitored link\\'?

A. Response time reports

B. Network monitors

C. Protocol analyzers

D. Online monitors

Correct Answer: B

QUESTION 3

To develop meaningful recommendations for findings, which of the following is MOST important for an IS auditor to determine and understand?

A. Criteria

B. Responsible party

C. Impact

D. Root cause

Correct Answer: C

QUESTION 4

Which of the following is an estimation technique where the results can be measure by the functional size of an information system based on the number and complexity of input, output, interface and queries?

A. Functional Point analysis

B. Gantt Chart

C. Time box management

D. Critical path methodology

Correct Answer: A

For CISA exam you should know below information about Functional Point Analysis:

Function Point Analysis (FPA) is an ISO recognized method to measure the functional size of an information system. The functional size reflects the amount of functionality that is relevant to and recognized by the user in the business. It is independent of the technology used to implement the system.

The unit of measurement is "function points". So, FPA expresses the functional size of an information system in a number of function points (for example: the size of a system is 314 fop\\'s). The functional size may be used:

To budget application development or enhancement costs To budget the annual maintenance costs of the application portfolio To determine project productivity after completion of the project To determine the Software Size for cost estimating

All software applications will have numerous elementary processes or independent processes to move data. Transactions (or elementary processes) that bring data from outside the application domain (or application boundary) to inside that

application boundary are referred to as external inputs. Transactions (or elementary processes) that take data from a resting position (normally on a file) to outside the application domain (or application boundary) are referred as either an

external outputs or external inquiries. Data at rest that is maintained by the application in question is classified as internal logical files. Data at rest that is maintained by another application in question is classified as external interface files.

Types of Function Point Counts:

Development Project Function Point Count

Function Points can be counted at all phases of a development project from requirements up to and including implementation. This type of count is associated with new development work. Scope creep can be tracked and monitored by

understanding the functional size at all phase of a project. Frequently, this type of count is called a baseline function point count.

Enhancement Project Function Point Count

It is common to enhance software after it has been placed into production. This type of function point count tries to size enhancement projects. All production applications evolve over time. By tracking enhancement size and associated costs a

historical database for your organization can be built. Additionally, it is important to understand how a Development project has changed over time.

Application Function Point Count

Application counts are done on existing production applications. This "baseline count" can be used with overall

application metrics like total maintenance hours. This metric can be used to track maintenance hours per function point. This is an

example of a normalized metric. It is not enough to examine only maintenance, but one must examine the ratio of maintenance hours to size of the application to get a true picture.

Productivity:

The definition of productivity is the output-input ratio within a time period with due consideration for quality.

Productivity = outputs/inputs (within a time period, quality considered)

The formula indicates that productivity can be improved by (1) by increasing outputs with the same inputs, (2) by decreasing inputs but maintaining the same outputs, or (3) by increasing outputs and decreasing inputs change the ratio

favorably.

Software Productivity = Function Points / Inputs

Effectiveness vs. Efficiency:

Productivity implies effectiveness and efficiency in individual and organizational performance. Effectiveness is the achievement of objectives. Efficiency is the achievement of the ends with least amount of resources.

Software productivity is defined as hours/function points or function points/hours. This is the average cost to develop software or the unit cost of software. One thing to keep in mind is the unit cost of software is not fixed with size. What

industry data shows is the unit cost of software goes up with size.

Average cost is the total cost of producing a particular quantity of output divided by that quantity. In this case to Total Cost/Function Points. Marginal cost is the change in total cost attributable to a one-unit change in output.

There are a variety of reasons why marginal costs for software increase as size increases. The following is a list of some of the reasons

As size becomes larger complexity increases.

As size becomes larger a greater number of tasks need to be completed.

As size becomes larger there is a greater number of staff members and they become more difficult to manage.

Function Points are the output of the software development process. Function points are the unit of software. It is very important to understand that Function Points remain constant regardless who develops the software or what language the

software is developed in. Unit costs need to be examined very closely. To calculate average unit cost all items (units) are combined and divided by the total cost. On the other hand, to accurately estimate the cost of an application each

component cost needs to be estimated.

Determine type of function point count

Determine the application boundary

Identify and rate transactional function types to determine their contribution to the unadjusted function point count.

Identify and rate data function types to determine their contribution to the unadjusted function point count.

Determine the value adjustment factor (VAF)

Calculate the adjusted function point count.

To complete a function point count knowledge of function point rules and application documentation is needed. Access to an application expert can improve the quality of the count. Once the application boundary has been established, FPA

can be broken into three major parts

FPA for transactional function types

FPA for data function types

FPA for GSCs

Rating of transactions is dependent on both information contained in the transactions and the number of files referenced, it is recommended that transactions are counted first. At the same time a tally should be kept of all FTR\\'s (file types

referenced) that the transactions reference. Every FTR must have at least one or more transactions. Each transaction must be an elementary process. An elementary process is the smallest unit of activity that is meaningful to the end user in

the business. It must be self-contained and leave the business in consistent state

The following were incorrect answers:

Critical Path Methodology - The critical path method (CPM) is an algorithm for scheduling a set of project activities

Gantt Chart - A Gantt chart is a type of bar chart, developed by Henry Gantt in the 1910s, that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project.

Terminal elements and summary elements comprise the work breakdown structure of the project. Modern Gantt charts also show the dependency (i.e. precedence network) relationships between activities. Gantt charts can be used to show

current schedule status using percent-complete shadings and a vertical "TODAY" line as shown here.

Time box Management - In time management, a time boxing allocates a fixed time period, called a time box, to each planned activity. Several project management approaches use time boxing. It is also used for individual use to address

personal tasks in a smaller time frame. It often involves having deliverables and deadlines, which will improve the productivity of the user.

Reference:

CISA review manual 2014 Page number 154

QUESTION 5

Which of the following is NOT an example of preventive control?

A. Physical access control like locks and door

B. User login screen which allows only authorize user to access website

C. Encrypt the data so that only authorize user can view the same

D. Duplicate checking of a calculations

Correct Answer: C

The word NOT is used as a keyword in the question. You need to find out a security control from given options which in not preventive. Duplicate checking of a calculation is a detective control and not a preventive control.

For your exam you should know below information about different security controls

Deterrent Controls

Deterrent Controls are intended to discourage a potential attacker. Access controls act as a deterrent to threats and attacks by the simple fact that the existence of the control is enough to keep some potential attackers from attempting to

circumvent the control. This is often because the effort required to circumvent the control is far greater than the potential reward if the attacker is successful, or, conversely, the negative implications of a failed attack (or getting caught)

outweigh the benefits of success. For example, by forcing the identification and authentication of a user, service, or application, and all that it implies, the potential for incidents associated with the system is significantly reduced because an

attacker will fear association with the incident. If there are no controls for a given access path, the number of incidents and the potential impact become infinite. Controls inherently reduce exposure to risk by applying oversight for a process.

This oversight acts as a deterrent, curbing an attacker\\'s appetite in the face of probable repercussions.

The best example of a deterrent control is demonstrated by employees and their propensity to intentionally perform unauthorized functions, leading to unwanted events. When users begin to understand that by authenticating into a system to

perform a function, their activities are logged and monitored, and it reduces the likelihood they will attempt such an action. Many threats are based on the anonymity of the threat agent, and any potential for identification and association with

their actions is avoided at all costs. It is this fundamental reason why access controls are the key target of circumvention by attackers. Deterrents also take the form of potential punishment if users do something unauthorized. For example, if

the organization policy specifies that an employee installing an unauthorized wireless access point will be fired, that will determine most employees from installing wireless access points.

Preventative Controls

Preventive controls are intended to avoid an incident from occurring. Preventative access controls keep a user from performing some activity or function. Preventative controls differ from deterrent controls in that the control is not optional and

cannot (easily) be bypassed. Deterrent controls work on the theory that it is easier to obey the control

rather than to risk the consequences of bypassing the control. In other words, the power for action resides with the user (or the attacker). Preventative controls place the power of action with the system, obeying the control is not optional. The

only way to bypass the control is to

find a flaw in the control\\'s implementation.

Compensating Controls

Compensating controls are introduced when the existing capabilities of a system do not support the requirement of a policy. Compensating controls can be technical, procedural, or managerial. Although an existing system may not support the

required controls, there may exist other

technology or processes that can supplement the existing environment, closing the gap in controls, meeting policy requirements, and reducing overall risk. For example, the access control policy may state that the authentication process must

be encrypted when performed over the Internet. Adjusting an application to natively support encryption for authentication purposes may be too costly. Secure Socket Layer (SSL), an encryption protocol, can be employed and layered on top of

the authentication process to support the policy statement. Other examples include a separation of duties environment, which offers the capability to isolate certain tasks to compensate for technical limitations in the system and ensure the

security of transactions. In addition, management processes, such as authorization, supervision, and administration, can be used to compensate for gaps in the access control environment.

Detective Controls

Detective controls warn when something has happened, and are the earliest point in the post-incident timeline. Access controls are a deterrent to threats and can be aggressively utilized to prevent harmful incidents through the application of

least privilege. However, the detective

nature of access controls can provide significant visibility into the access environment and help organizations manage their access strategy and related security risk. As mentioned previously, strongly managed access privileges provided to

an authenticated user offer the ability to reduce the risk exposure of the enterprise\\'s assets by limiting the capabilities that authenticated user has. However, there are few options to control what a user can perform once privileges are

provided. For example, if a user is provided write access to a file and that file is damaged, altered, or otherwise negatively impacted (either deliberately or unintentionally), the use of applied access controls will offer visibility into the

transaction. The control environment can be established to log activity regarding the identification, authentication, authorization, and use of privileges on a system. This can be used to detect the occurrence of errors, the attempts to perform

an unauthorized action, or to validate when provided credentials were exercised. The logging system as a detective device provides evidence of actions (both successful and unsuccessful) and tasks that were executed by authorized users.

Corrective Controls

When a security incident occurs, elements within the security infrastructure may require corrective actions. Corrective controls are actions that seek to alter the security posture of an environment to correct any deficiencies and return the

environment to a secure state. A security

incident signals the failure of one or more directive, deterrent, preventative, or compensating controls. The detective controls may have triggered an alarm or notification, but now the corrective controls must work to stop the incident in its

tracks. Corrective controls can take

many forms, all depending on the particular situation at hand or the particular security failure that needs to be dealt with.

Recovery Controls

Any changes to the access control environment, whether in the face of a security incident or to offer temporary compensating controls, need to be accurately reinstated and returned to normal operations. There are several situations that may

affect access controls, their applicability, status, or management. Events can include system outages, attacks, project changes, technical demands, administrative gaps, and full-blown disaster situations. For example, if an application is not

correctly installed or deployed, it may adversely affect controls placed on system files or even have default administrative accounts unknowingly implemented upon install. Additionally, an employee may be transferred, quit, or be on temporary

leave that may affect policy requirements regarding separation of duties. An attack on systems may have resulted in the implantation of a Trojan horse program, potentially exposing private user information, such as credit card information and

financial data. In all of these cases, an undesirable situation must be rectified as quickly as possible and controls returned to normal operations.

For your exam you should know below information about different security controls

Deterrent Controls

Deterrent Controls are intended to discourage a potential attacker. Access controls act as a deterrent to threats and attacks by the simple fact that the existence of the control is enough to keep some potential attackers from attempting to

circumvent the control. This is often because the effort required to circumvent the control is far greater than the potential reward if the attacker is successful, or, conversely, the negative implications of a failed attack (or getting caught)

outweigh the benefits of success. For example, by forcing the identification and authentication of a user, service, or application, and all that it implies, the potential for incidents associated with the system is significantly reduced because an

attacker will fear association with the incident. If there are no controls for a given access path, the number of incidents and the potential impact become infinite. Controls inherently reduce exposure to risk by applying oversight for a process.

This oversight acts as a deterrent, curbing an attacker\\'s appetite in the face of probable repercussions.

The best example of a deterrent control is demonstrated by employees and their propensity to intentionally perform unauthorized functions, leading to unwanted events.

When users begin to understand that by authenticating into a system to perform a function, their activities are logged and monitored, and it reduces the likelihood they will attempt such an action. Many threats are based on the anonymity

of

the threat agent, and any potential for identification and association with their actions is avoided at all costs.

It is this fundamental reason why access controls are the key target of circumvention by attackers. Deterrents also take the form of potential punishment if users do something unauthorized. For example, if the organization policy specifies that

an employee installing an unauthorized wireless access point will be fired, that will determine most employees from installing wireless access points.

Preventative Controls

Preventive controls are intended to avoid an incident from occurring. Preventative access controls keep a user from performing some activity or function. Preventative controls differ from deterrent controls in that the control is not optional and

cannot (easily) be bypassed. Deterrent controls work on the theory that it is easier to obey the control

rather than to risk the consequences of bypassing the control. In other words, the power for action resides with the user (or the attacker). Preventative controls place the power of action with the system, obeying the control is not optional. The

only way to bypass the control is to find a flaw in the control\\'s implementation.

Compensating Controls

Compensating controls are introduced when the existing capabilities of a system do not support the requirement of a policy. Compensating controls can be technical, procedural, or managerial. Although an existing system may not support the

required controls, there may exist other technology or processes that can supplement the existing environment, closing the gap in controls, meeting policy requirements, and reducing overall risk.

For example, the access control policy may state that the authentication process must be encrypted when performed over the Internet. Adjusting an application to natively support encryption for authentication purposes may be too costly.

Secure Socket Layer (SSL), an encryption protocol, can be employed and layered on top of the authentication process to support the policy statement.

Other examples include a separation of duties environment, which offers the capability to isolate certain tasks to compensate for technical limitations in the system and ensure the security of transactions. In addition, management processes,

such as authorization, supervision, and administration, can be used to compensate for gaps in the access control environment.

Detective Controls

Detective controls warn when something has happened, and are the earliest point in the post-incident timeline. Access controls are a deterrent to threats and can be aggressively utilized to prevent harmful incidents through the application of

least privilege. However, the detective nature of access controls can provide significant visibility into the access environment and help organizations manage their access strategy and related security risk.

As mentioned previously, strongly managed access privileges provided to an authenticated user offer the ability to

reduce the risk exposure of the enterprise\\'s assets by limiting the capabilities that authenticated user has. However, there are

few options to control what a user can perform once privileges are provided. For example, if a user is provided write access to a file and that file is damaged, altered, or otherwise negatively impacted (either deliberately or unintentionally), the

use of applied access controls will offer visibility into the transaction. The control environment can be established to log activity regarding the identification, authentication, authorization, and use of privileges on a system.

This can be used to detect the occurrence of errors, the attempts to perform an unauthorized action, or to validate when provided credentials were exercised. The logging system as a detective device provides evidence of actions (both successful and unsuccessful) and tasks that were executed by authorized users.

Corrective Controls When a security incident occurs, elements within the security infrastructure may require corrective actions. Corrective controls are actions that seek to alter the security posture of an environment to correct any deficiencies and return the environment to a secure state. A security incident signals the failure of one or more directive, deterrent, preventative, or compensating controls. The detective controls may have triggered an alarm or notification, but now the corrective controls must work to stop the incident in its tracks. Corrective controls can take many forms, all depending on the particular situation at hand or the particular security failure that needs to be dealt with.

Recovery Controls

Any changes to the access control environment, whether in the face of a security incident or to offer temporary compensating controls, need to be accurately reinstated and returned to normal operations. There are several situations that may

affect access controls, their applicability, status, or management.

Events can include system outages, attacks, project changes, technical demands, administrative gaps, and full-blown disaster situations. For example, if an application is not correctly installed or deployed, it may adversely affect controls

placed on system files or even have default administrative accounts unknowingly implemented upon install.

Additionally, an employee may be transferred, quit, or be on temporary leave that may affect policy requirements regarding separation of duties. An attack on systems may have resulted in the implantation of a Trojan horse program,

potentially exposing private user information, such as credit card information and financial data. In all of these cases, an undesirable situation must be rectified as quickly as possible and controls returned to normal operations.

The following answers are incorrect:

The other examples belong to Preventive control.

Reference:

CISA Review Manual 2014 Page number 44

and

Official ISC2 CISSP guide 3rd edition Page number 50 and 51