Domain 8 - Software Development Security

Test ID: 178966575

Pregunta #1 de 67

Id. de pregunta: 1105582

¿Qué instrucción define correctamente el modelo de base de datos orientado a objetos?

- ✓ **A)** Puede almacenar datos que incluyen clips multimedia, imágenes, vídeo y gráficos.
- ✗ **B)** Interconecta lógicamente bases de datos ubicadas remotamente.
- ✗ **C)** La relación entre los elementos de datos tiene la forma de un árbol lógico.
- ✗ **D)** Es un híbrido entre bases de datos relacionales y basadas en objetos.

Explicación

Una base de datos orientada a objetos se utiliza para almacenar varios tipos de datos, como imágenes, audio, vídeo y documentos. Los elementos de datos y los diferentes componentes se conocen como objetos. Estos objetos se utilizan para crear componentes de datos dinámicos. En una base de datos orientada a objetos, los objetos se pueden crear dinámicamente según los requisitos y las instrucciones ejecutadas. El modelo orientado a objetos proporciona facilidad para reutilizar código, análisis y mantenimiento reducido.

El modelo de base de datos distribuida implica varias bases de datos que están situadas en ubicaciones remotas y están conectadas lógicamente. En un modelo de base de datos distribuida, las bases de datos están conectadas lógicamente entre sí para garantizar que la transición de una base de datos a otra sea transparente para los usuarios. Las bases de datos conectadas lógicamente aparecen como una sola base de datos para los usuarios. El modelo de base de datos distribuida permite que diferentes bases de datos situadas en ubicaciones remotas sean administradas individualmente por diferentes administradores de bases de datos. Este modelo de base de datos proporciona características de escalabilidad, como el equilibrio de carga y la tolerancia a errores.

Una base de datos relacional de objetos es un híbrido entre una base de datos basada en objetos y una base de datos relacional, y hereda las propiedades de ambas. Una base de datos relacional de objetos permite a los desarrolladores integrar la base de datos con sus propios tipos de datos y métodos personalizados.

En una base de datos jerárquica, los datos se organizan en una estructura de árbol lógico en lugar de utilizar filas y columnas. Los registros y los campos están relacionados entre sí en una estructura de árbol primario-secundario. Una estructura de árbol de base de datos jerárquica puede tener ramas y hojas donde las hojas son los campos de datos y se accede a los datos a través de rutas de acceso bien definidas mediante grupos de registros que actúan como bifurcaciones. Se utiliza una base de datos jerárquica donde existen relaciones de una a varias.

Objetivo:
Seguridad del desarrollo de software

Subobsecución:
Identificar y aplicar controles de seguridad en entornos de desarrollo

Referencias:

CISSP Cert Guide (3rd Edition), Capítulo 8: Seguridad de desarrollo de software, programación orientada a objetos

---

Pregunta #2 de 67

Id. de pregunta: 1105595

¿Qué afirmación es cierta de un ataque de salami?

- ✗ **A)** Es una técnica de ingeniería social.
- ✗ **B)** No es un ejemplo de diddling de datos.
- ✓ **C)** Se trata de robar pequeñas cantidades de dinero de múltiples cuentas.
- ✗ **D)** Es un tipo de ataque pasivo.

Explicación

Un ataque de salami implica cometer numerosos crímenes pequeños varias veces para garantizar que nadie se dé cuenta del crimen más grande. Por ejemplo, un ataque de salami restará una cantidad insignificante de fondos de varias cuentas durante un gran período de tiempo y dirigirá estos fondos a la cuenta del atacante. Este proceso equivale a cometer fraude porque la cantidad transferida cada vez es insignificante por naturaleza y pasa desapercibida.

Un ataque de salami es un ataque activo. Es una técnica de diddling de datos.

Un ataque de salami no es una técnica de ingeniería social. Una técnica de ingeniería social es la práctica de obtener información confidencial, ya sea engañando o manipulando a usuarios legítimos.

Objetivo:
Seguridad del desarrollo de software

Subobsecución:
Identificar y aplicar controles de seguridad en entornos de desarrollo

Referencias:

¿Qué es un Ataque Salami?, https://ajmaurya.wordpress.com/2014/03/27/what-is-a-salami-attack/

Pregunta #3 de 67

Id. de pregunta: 1111816

¿Qué afirmación es cierta de un ciclo de vida de desarrollo de software?

✗ **A)** Las pruebas de carga de trabajo se deben realizar mientras se diseñan los requisitos funcionales.

✓ **B)** Las pruebas unitarias deben ser realizadas por el desarrollador y el equipo de control de calidad.

✗ **C)** Un programador de software debe ser la única persona para desarrollar el software, probarlo y someterlo a producción

✗ **D)** Las pruebas en paralelo comprueban si hay más de un sistema disponible para la redundancia.

Explanation

Unit testing should be performed by the developer and by the quality assurance team. Unit testing refers to the debugging performed by the programmer while coding instructions. The unit testing should check the validity of the data format, length, and values. After writing the instructions, the developer might run tools to detect errors.

A software programmer should not be the only person to develop the software, test it, and submit it to production. Therefore, distinction of duties ensures checks by using formal procedures adopted by the quality assurance team. After the software program is submitted, it is again verified by the quality assurance team by using formal procedures and practices before sending it to the program library.

Parallel testing is the process of feeding test data into two systems, which are the altered system and another alternative system, and comparing the results. The original system can serve as the alternative system. It is important to perform testing by using live workloads to observe the performance and the bottlenecks present in the actual production environment. Parallel testing ensures that the system fulfills the defined business requirements. It does not involve testing for redundancy.

Designing the functional requirements is a part of the system design specifications stage and does not involve workload testing.

The SDLC includes the following phases:

1. Plan/Initiate Project
2. Gather Requirements
3. Design
4. Develop
5. Test/Validate
6. Release/Maintain
7. Certify/Accredit
8. Change Management and Configuration Management/Replacement

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Test/Validate

---

Pregunta #4 de 67

Id. de pregunta: 1105561

Un atacante está en el proceso de realizar un cambio no autorizado en algunos datos de la base de datos. Debe cancelar los cambios de la base de datos de la transacción y devolver la base de datos a su estado anterior. ¿Qué operación de base de datos debe utilizar?

- ✗ **A)** punto de control
- ✗ **B)** cometer
- ✓ **C)** reversión
- ✗ **D)** punto de salvaguarda

Explanation

You should use a rollback operation. A rollback operation cancels any database changes from the current transaction and returns the database to its previous state. It prevents a transaction from updating the database with partial or corrupt data. Rollbacks occur during the operations/maintenance phase of the SDLC.

A commit operation finalizes any database changes from the current transaction, making the changes available to other users. A savepoint operation creates a logged point to which the database can be restored. It allows data to be restored to a certain point in time. A checkpoint operation saves data that is stored in memory to the database. It allows the memory to be cleared. When a database detects an error, a checkpoint enables it to start processing at a designated place.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

Rollback, https://www.techopedia.com/definition/9229/rollback

Pregunta #5 de 67

Id. de pregunta: 1105576

El sitio web de una organización incluye varios subprogramas Java. Los applets de Java incluyen una característica de seguridad que limita el acceso del applet a ciertas áreas del sistema del usuario web. ¿Cómo lo hace?

✗ **A)** mediante códigos de objeto

✓ **B)** mediante el uso de espacios aislados

✗ **C)** Mediante certificados digitales y de confianza

✗ **D)** mediante lenguajes de macros

Explanation

Java applets use sandboxes to enforce security. A sandbox is a security scheme that prevents Java applets from accessing unauthorized areas on a user's computer. This mechanism protects the system from malicious software, such as hostile applets, by enforcing the execution of the application within the sandbox and preventing access to the system resources outside the sandbox.

A hostile applet is an active content module used to exploit system resources. Hostile applets coded in Java can pose a security threat to computer systems if the executables are downloaded from unauthorized sources. Hostile applets may disrupt the computer system operation, either through resource consumption or through covert channels.

Object code refers to a version of a computer program that is compiled before it is ready to run in a computer. The application software on a system is typically in the form of compiled object codes and does not include the source code. Object codes are not related to the security aspects of Java. They represent an application program after the compilation process.

Macro programs use macro language for the automation of common user tasks. Macro languages, such as Visual Basic, are typically used to automate the tasks and activities of users. Macro programs have their own set of security vulnerabilities, such as macro viruses, but are not related to Java security.

Digital and trust certificates are used by the ActiveX technology of Microsoft to enforce security. ActiveX refers to a set of controls that users can download in the form of a plug-in to enhance a feature of an application. The primary difference between Java applets and ActiveX controls is that the ActiveX controls are downloaded subject to acceptance by a user. The ActiveX trust certificate also states the source of the plug-in signatures of the ActiveX modules. Java applets are short programs that use the technique of a sandbox to limit the applet's access to specific resources stored in the system.

Objective:

Software Development Security

Sub-Objective:

Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Java Applets

---

Pregunta #6 de 67

Id. de pregunta: 1111815

Su empresa ha adquirido un sistema experto que utiliza el razonamiento if-then-else para obtener más datos de los que están disponibles actualmente. ¿Qué técnica de procesamiento de sistemas expertos se está implementando?

- ✓ **A)** técnica de encadenamiento hacia adelante
- ✗ **B)** técnica de encadenamiento hacia atrás
- ✗ **C)** modelo de cascada
- ✗ **D)** modelo espiral

Explanation

The expert system processing technique that is being implemented is the forward-chaining technique. The forward-chaining technique is an expert system processing technique that uses if-then-else rules to obtain more data than is currently available. Forward chaining is the reasoning approach that can be used when there are a small number of solutions relative to the number of inputs. The input data is used to reason forward to prove that one of the possible solutions in a small solution set is the correct one.

An expert system consists of a knowledge base and adaptive algorithms that are used to solve complex problems and to provide flexibility in decision-making approaches. An expert system uses artificial intelligence to extract new information from a set of information, and exhibits reasoning similar to that of humans knowledgeable in a particular field to solve a problem in that field. An expert system operates in two modes: forward chaining and backward chaining.

Backward chaining is the process of beginning with a possible solution and using the knowledge in the knowledge base to justify the solution based on the raw input data. Backward chaining works backwards by analyzing the list of the goals identified and verifying the availability of data to reach a conclusion on any goal. Backward chaining starts with the goals and looks for the data that justifies the goal by applying if-then-else rules.

The spiral model is a software development model that is based is on analyzing the risk and building the prototypes and the simulation during the various phases of the development cycle.

The waterfall model is a software development model that is based on proper reviews and on documenting the reviews at each phase of the software development cycle. This model divides the software development cycle into phases. Proper review and documentation must be completed before moving on to the next phase. The modified waterfall model was reinterpreted to have phases end at project milestones. Incremental development is a refinement to the basic waterfall model that states that software should be developed in increments of functional capability.

Other software development models include the cleanroom model and the capability maturity model (CMM).

- The cleanroom model follows well-defined formal procedures for development and testing of software. The cleanroom model calls for strict testing procedures and is often used for critical applications that should be certified.
- The capability maturity model (CMM) describes the principles, procedures, and practices that should be followed by an organization in a software development life cycle. The capability maturity model defines guidelines and best practices to implement a standardized approach for developing applications and software programs.

Knowledge-based system (KBS) or expert systems include the knowledge base, inference engine, and interface between the user and the system. A knowledge engineer and domain expert develops a KBS or expert system. Expert systems are used to automate security log review to detect intrusion.

A fuzzy expert system is an expert system that uses fuzzy membership functions and rules, instead of Boolean logic, to reason about data. Thus, fuzzy variables can have an approximate range of values instead of the binary True or False used in conventional expert systems. An example of this is an expert system that has rules of the form "If w is low and x is high then y is intermediate," where *w* and *x* are input variables and *y* is the output variable.

A software process is a set of activities, methods, and practices that are used to develop and maintain software and associated products. Software process capability is a means of predicting the outcome of the next software project conducted by an organization.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Forward and Backward Chaining Techniques of Reasoning in Rule-Based Systems (PDF), http://i-rep.emu.edu.tr:8080/jspui/bitstream/11129/2325/1/mzoribareen.pdf

---

Pregunta #7 de 67

Id. de pregunta: 1114016

¿Cómo aplica la seguridad un componente ActiveX?

- ✗ **A)** mediante códigos de objeto
- ✓ **B)** mediante Authenticode
- ✗ **C)** mediante lenguajes de macros
- ✗ **D)** mediante el uso de espacios aislados

Explanation

Authenticode is used by the ActiveX technology of Microsoft to enforce security. ActiveX refers to a set of controls that users can download in the form of a plug-in to enhance a feature of an application. The primary difference between Java applets and ActiveX controls is that the ActiveX controls are downloaded subject to acceptance by a user. The ActiveX trust certificate also states the source of the plug-in signatures of the ActiveX modules.

Java applets use sandboxes to enforce security. A sandbox is a security scheme that prevents Java applets from accessing unauthorized areas on a user's computer. When a user accesses a Web page through a browser, class files for an applet are downloaded automatically, even from untrusted sources. To counter this possible threat, Java provides a customizable sandbox and enforces the execution of the application within the sandbox. This prevents Java applets from accessing unauthorized areas on a user's computer or system resources outside the sandbox. Sandbox protections include preventing reading and writing to a local disk, prohibiting the creation of a new process, preventing the establishment of a network connection to a new host, and preventing the loading of a new dynamic library and directly calling a native method. The sandbox security features are designed into the Java Virtual Machine (JVM). These features are implemented through array bounds checking, structured memory access, type-safe reference cast checking, checking for null references, and automatic garbage collection. These checks are designed to limit memory accesses to safe, structured operations

A hostile applet is an active content module used to exploit system resources. Hostile applets coded in Java can pose a security threat to computer systems if the executables are downloaded from unauthorized sources. Hostile applets may disrupt the computer system operation either through resource consumption or through the use of covert channels.

Object code refers to a version of a computer program that is compiled before it is ready to run in a computer. The application software on a system is typically in the form of compiled object codes and does not include the source code. Object codes are not related to the security aspects of Java. They represent an application program after the compilation process.

Macro programs use macro language for the automation of common user tasks. Macro languages, such as Visual Basic, are typically used to automate the tasks and activities of users. Macro programs have their own set of security vulnerabilities, such as macro viruses, but are not related to Java security.

Java applets are short programs that use the technique of a sandbox to limit the applet's access to specific resources stored in the system.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, ActiveX

Pregunta #8 de 67

Id. de pregunta: 1105623

Un hacker ha utilizado un defecto de diseño en una aplicación para obtener acceso no autorizado a la aplicación. ¿Qué tipo de ataque se ha producido?

- ✗ **A)** desbordamiento de búfer
- ✓ **B)** escalamiento de privilegios
- ✗ **C)** gancho de mantenimiento
- ✗ **D)** puerta trasera

Explanation

An escalation of privileges attack occurs when an attacker has used a design flaw in an application to obtain unauthorized access to the application. There are two type of privilege escalation: vertical and horizontal. With vertical privilege escalation, the attacker obtains higher privileges by performing operations that allow the attacker to run unauthorized code. With horizontal privilege escalation, the attacker obtains the same level of permissions as he already has but uses a different user account to do so.

A backdoor is a term for lines of code that are inserted into an application to allow developers to enter the application and bypass the security mechanisms. Backdoors are also referred to as maintenance hooks.

A buffer overflow occurs when an application erroneously allows an invalid amount of input in the buffer.

Objective:
Software Development Security

Sub-Objective:
Define and apply secure coding guidelines and standards

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Escalation of Privileges and Rootkits

privilege escalation attack, http://searchsecurity.techtarget.com/definition/privilege-escalation-attack

Pregunta #9 de 67

Id. de pregunta: 1111817

¿Qué es un agente en un entorno informático distribuido?

- ✗ **A)** Un protocolo que codifica mensajes en una instalación de servicio Web
- ✗ **B)** El middleware que establece la relación entre objetos en un entorno cliente/servidor
- ✗ **C)** un identificador utilizado para identificar de forma única a los usuarios, recursos y componentes dentro de un entorno
- ✓ **D)** Un programa que realiza servicios en un entorno en nombre de una entidad de seguridad de otro entorno

Explanation

In a distributed computing environment, an agent is a program that performs services in one environment on behalf of a principal in another environment.

A globally unique identifier (GUID) and a universal unique identifier (UUID) uniquely identify users, resources, and components within a Distributed Component Object Model (DCOM) or Distributed Computer Environment (DCE) environment, respectively.

Simple Object Access Protocol (SOAP) is an XML-based protocol that encodes messages in a Web service setup.

Object request brokers (ORBs) are the middleware that establishes the relationship between objects in a client/server environment. A standard that uses ORB to implement exchanges among objects in a heterogeneous, distributed environment is Common Object Request Broker Architecture (CORBA). A distributed object model that has similarities to CORBA is DCOM.

The Object Request Architecture (ORA) is a high-level framework for a distributed environment. It consists of ORBs, object services, application objects, and common facilities.

The following are characteristics of a distributed data processing (DDP) approach:

- It consists of multiple processing locations that can provide alternatives for computing in the event that a site becomes inoperative.
- Distances from a user to a processing resource are transparent to the user.
- Data stored at multiple, geographically separate locations is easily available to the user.

Objective:

Software Development Security

Sub-Objective:

Identify and apply security controls in development environments

References:

Why, When, and Where to Use Software Agents, http://www.agentbuilder.com/Documentation/whyAgents.html

---

Pregunta #10 de 67

Id. de pregunta: 1114019

¿Cuál es una característica del mantenimiento de registros en un sistema?

✗ **A)** El registro proporciona pistas de auditoría, pero mejora las infracciones de seguridad.

✗ **B)** El registro evita las infracciones de seguridad, pero solo se ocupa de la supervisión pasiva.

✓ **C)** El registro ayuda a un administrador a detectar infracciones de seguridad y puntos vulnerables en una red.

✗ **D)** El registro proporciona control de acceso mediante la autenticación de credenciales de usuario.

Explanation

Logging helps the administrator to detect vulnerable points in a network, specify changes that can enhance the system's security, log suspicious activity from a specific user or a system, and identify a security breach.

Logging does NOT enhance security violations.

Logging is not only a passive but also an active process of assimilating information about various aspects, such as performance and security of an infrastructure.

Logging as a part of the access control system provides accountability services and does not provide authentication and authorization services to legitimate users.

Logging is the process of collecting information that is used for monitoring and auditing purposes. Logging establishes user accountability by providing audit trails and system logs related to system resource usage and activities. If an intrusion occurs, logging helps find the potential source of an attack. Therefore, logs must be secured properly. Logs should be periodically archived and reviewed for any suspicious activity. The period of log retention depends on the security requirements of the organization. Logs can also be used for security evaluation of a company during the course of information security audits.

An infrastructure can be monitored by performing activities, such as log analysis and intrusion detection by using the IDS. An organization can also periodically deploy countermeasure testing to ensure that the infrastructure devices comply with the security policy and meet the security needs of the organization. Countermeasure testing is not a monitoring technique, but it ensures that an organization meets its security objectives.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Auditing and Logging

---

Pregunta #11 de 67

Id. de pregunta: 1105594

¿Qué se utiliza en la computación evolutiva?

- ✓ **A)** algoritmos genéticos
- ✗ **B)** modelos matemáticos o computacionales
- ✗ **C)** características de los organismos vivos
- ✗ **D)** conocimiento de un experto

Explanation

Genetic algorithms are used in evolutionary computing. Evolutionary computing is a type of artificial intelligence.

Biological computing uses the characteristics of living organisms. Knowledge-based or expert systems use knowledge from an expert. Artificial neural networks (ANNs) use mathematical or computational models.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

What is an Evolutionary Algorithm? (PDF), https://www.cs.vu.nl/~gusz/ecbook/Eiben-Smith-Intro2EC-Ch2.pdf

---

Pregunta #12 de 67

Id. de pregunta: 1114013

Debe asegurarse de que los tipos de datos y las reglas se aplican en la base de datos. ¿Qué tipo de integridad se debe aplicar?

✗ **A)** integridad referencial

✗ **B)** supresión celular

✗ **C)** integridad de la entidad

✓ **D)** integridad semántica

Explanation

Semantic integrity should be enforced. Semantic integrity ensures that data types and rules are enforced. It includes checking data types, values, data constraints, and uniqueness rules. Semantic integrity protects the data by ensuring that data values follow all the rules.

Entity integrity ensures that each row is identified by a unique primary key. Referential integrity ensures that each foreign key references a primary key that actually exists.

Cell suppression is not a type of integrity. It is a technique used to hide certain cells.

The system design specification phase of the software development life cycle (SDLC) focuses on providing details on which kind of security mechanism will be a part of the software product. The system design specification phase also conducts a detailed design review and develops a plan for validation, verification, and testing. The organization developing the application will review the product specifications together with the customer to ensure that the security requirements are clearly stated and understood, and that the planned functionality features are embedded in the product. Involving security analysts at this phase ensures maximum benefit to the organization. It also enables you to understand the security requirements and features of the product and to report existing loopholes.

The SDLC includes the following phases:

1. Plan/Initiate Project
2. Gather Requirements
3. Design
4. Develop
5. Test/Validate
6. Release/Maintain
7. Certify/Accredit
8. Change Management and Configuration Management/Replacement

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Semantic integrity,
https://www.ibm.com/support/knowledgecenter/en/SSGU8G_12.1.0/com.ibm.sqlt.doc/ids_sqt_254.htm

---

Pregunta #13 de 67

Id. de pregunta: 1105614

Su empresa implementa varias bases de datos. Le preocupa la seguridad de los datos de las bases de datos. ¿Qué instrucción es correcta para la seguridad de la base de datos?

- ✗ **A)** Las variables de enlace proporcionan control de acceso mediante la implementación de restricciones granulares.
- ✗ **B)** El lenguaje de manipulación de datos (DML) implementa el control de acceso a través de la autorización.
- ✓ **C)** El lenguaje de control de datos (DCL) implementa la seguridad a través del control de acceso y las restricciones granulares.
- ✗ **D)** El lenguaje de identificación de datos implementa la seguridad en los componentes de datos.

Explanation

Data control language (DCL) implements security through access control and granular restrictions. DCL is used to configure which DML statements users can use.

None of the other statements is true.

Data identification language is not a valid language used in databases.

A bind variable is a placeholder in a SQL statement that must be replaced with a valid value or value address for the statement to execute successfully.

Data manipulation language (DML) is used to change the values of data within a database.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

Data Control Language, https://www.lifewire.com/data-control-language-dcl-1019477

Pregunta #14 de 67

Id. de pregunta: 1105586

Su organización ha implementado recientemente una red neuronal artificial (ANN). La ANN permitió a la red tomar decisiones basadas en la experiencia que se les proporcionó. ¿Qué característica de la ANN se describe?

- ✗ **A)** capacidad de retención
- ✓ **B)** adaptabilidad
- ✗ **C)** integridad neuronal
- ✗ **D)** tolerancia a fallos

Explanation

Adaptability is the artificial neural network (ANN) characteristic that is described. Adaptability refers to the ability of an ANN to arrive at decisions based on the learning process that uses the inputs provided. It is important to note that the ability of ANN learning is limited to the experience provided to them. An ANN is an adaptive system that changes its structure based on either external or internal information that flows through the network by applying the if-then-else rules.

ANNs are computers systems where the system simulates the working of a human brain. A human brain can contain billions of neurons performing complex operations. An ANN can also contain a large number of small computational units that are called upon to perform a required task. A neural network learns by using various algorithms to adjust the weights applied to the data. The equation Z = f [wn in], where *Z* is the output, *wn* are weighting functions, and *in* is a set of inputs, scientifically describes a neural network.

Fault tolerance refers to the ability to combat threats of design reliability and continuous availability. ANNs do not provide fault tolerance.

Retention capability and neural integrity are generic terms and are invalid options.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Neural networks, https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html

Pregunta #15 de 67

Id. de pregunta: 1111820

¿Qué afirmación es cierta de los canales encubiertes?

✗ **A)** Un canal encubierto actúa como una ruta de confianza para la comunicación autorizada.

✗ **B)** Un canal encubierto regula el flujo de información e implementa la política de seguridad.

✗ **C)** Un canal encubierto es direccionada por una calificación C2 proporcionada por TCSEC.

✓ **D)** Un canal encubierto no está controlado por un mecanismo de seguridad.

Explanation

A covert channel is not controlled by a security mechanism. A covert channel is a communication path that accesses information in an unauthorized manner and violates the security policy. A covert channel is not a regulated path of the information flow and is an effect of a software bug or a compromised system.

Covert channels are addressed by the Trusted Computer System Evaluation Criteria (TCSEC) rating B2 and above. Covert storage channels are addressed in level B2, and covert timing channels are addressed in level B3.

Unlike the overt channel that is specifically designed as an authorized communication channel, the covert channel is used by the attackers to violate the security policy of a system. Therefore, the covert channel is avoided for communication because it lacks the mandatory control.

The two types of covert channels are as follows:

- Covert timing channel: In a covert timing channel, a process sends information to another process but modulates the use of system resources. For example, the process enables you to access a hard disk and the information regarding the number of CPU cycles. When the second process is completing a job, the first process waits for the signal and then performs the unauthorized job. Covert timing channels convey information by modifying the timing of a system resource in some measurable way.
- Covert storage channel: In a covert storage channel, the security risk arises due to the storage location. For example, a problem may arise when a process writes data to a specific location and another process is able to read this information either directly or indirectly, irrespective of the security level it occurs in. A covert storage channel is an information transfer that involves the direct or indirect writing of a storage location by one process and the direct or indirect reading of the storage location by another process.

Both the covert timing channel and the covert storage channel violate the security policy of a system. A Loki attack is an example of a covert channel.

Objective:

Software Development Security

Sub-Objective:

Define and apply secure coding guidelines and standards

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Covert Channel

---

Pregunta #16 de 67

Id. de pregunta: 1163832

Un desarrollador ha solicitado un cambio determinado en la configuración de un servidor de archivos. ¿Qué paso debe producirse a continuación en el proceso de cambio si existe una directiva de control de cambios?

✗ **A)** Aprobar el cambio.

✗ **B)** Pruebe el cambio.

✓ **C)** Documente el cambio.

✗ **D)** Implemente el cambio.

Explanation

The formal change control process includes four steps:

1. Define the change requests. - This includes documenting the change request.
2. Submit and review the change request. - This includes testing the change.
3. Define options, and create response document. -
4. Final decision and approval.

A well-structured change management process ensures that changes follow a certain process before they are implemented in the live environment.

The software development life cycle (SDLC) includes the following phases:

- Plan/Initiate Project
- Gather Requirements
- Design
- Develop
- Test/Validate
- Release/Maintain
- Certify/Accredit
- Change Management and Configuration Management/Replacement

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Change Management and Configuration Management/Replacement

---

Pregunta #17 de 67

Id. de pregunta: 1111805

Recientemente, un atacante inyectó código malintencionado en una aplicación web del sitio web de la organización. ¿Qué tipo de ataque ha experimentado su organización?

✗ **A)** desbordamiento de búfer
✗ **B)** recorrido de ruta de acceso
✓ **C)** secuencias de comandos entre sitios
✗ **D)** Inyección SQL

Explanation

Your organization experienced a cross-site scripting (XSS) attack. A XSS attack occurs when an attacker locates a vulnerability on a Web site that allows the attacker to inject malicious code into a Web application.

A buffer overflow occurs when an invalid amount of input is written to the buffer area.

A SQL injection occurs when an attacker inputs actual database commands into the database input fields instead of the valid input.

Path traversal occurs when the characters are entered into the URL to traverse directories that are not supposed to be available from the Web`../`

Some possible countermeasures to input validation attacks include the following:

- Filter out all known malicious requests.
- Validate all information coming from the client, both at the client level and at the server level.
- Implement a security policy that includes parameter checking in all Web applications.

The system design specification phase of the software development life cycle (SDLC) focuses on providing details on which kind of security mechanism will be a part of the software product. The system design specification phase also

conducts a detailed design review and develops a plan for validation, verification, and testing. The organization developing the application will review the product specifications with the customer to ensure that the security requirements are clearly stated and understood, and that the planned functionality is embedded in the product. Involving security analysts at this phase maximizes the benefit to the organization. It also enables you to understand the security requirements and features of the product and to report existing loopholes.

The system development phase of the SDLC includes coding and scripting of software applications. The system development stage ensures that the program instructions are written according to the defined security and functionality requirements of the product. The programmers build security mechanisms, such as audit trails and access control, into the software according to the predefined security assessments and the requirements of the application.

The SDLC includes the following phases:

1. Plan/Initiate Project
2. Gather Requirements
3. Design (including system design)
4. Develop (including system development)
5. Test/Validate
6. Release/Maintain
7. Certify/Accredit
8. Change Management and Configuration Management/Replacement

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

Cross-site scripting, http://www.google.com/about/appsecurity/learning/xss/

Pregunta #18 de 67

Id. de pregunta: 1111813

Todos los siguientes son contramedidas para ataques de administración de sesiones, EXCEPTO:

✗ **A)** Implementar identificadores de sesión aleatorios.

✗ **B)** Implementar marcas de tiempo o validación basada en el tiempo.

✗ **C)** Cifrar las cookies que incluyen información sobre el estado de la conexión.

✓ **D)** Implementar controles previos y posteriores a la validación.

Explanation

You should not implement pre- and post-validation controls as a countermeasure for session management attacks. Pre- and post-validation controls are countermeasures to use in parameter validation attacks.

Countermeasures for session management attacks include the following:

- Implement randomized session IDs.
- Implement time stamps or time-based validation.
- Encrypt cookies that include information about the state of the connection.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Session Management Cheat Sheet, https://www.owasp.org/index.php/Session_Management_Cheat_Sheet

---

Pregunta #19 de 67

Id. de pregunta: 1105597

Durante una evaluación de seguridad reciente, descubrirá que un equipo de la red se ha visto comprometido. Una aplicación se ha instalado inadvertidamente en el equipo. Esta aplicación permite a un criminal utilizar el ordenador comprometido para llevar a cabo un ataque. ¿Cuál es el término para este equipo comprometido?

✗ **A)** víctima
✓ **B)** zombi
✗ **C)** bot
✗ **D)** botnet

Explanation

The compromised computer is called a zombie. A zombie is a computer on which an application is installed that will be used to attack another computer or network at a later date.

A bot is the application that is installed on the compromised computer. A botnet is a group of compromised computers that are used to carry out an attack using the bot.

The compromised computer is not referred to as the victim. The zombie is usually not considered a victim. The victim is the computer that is attacked by the bot hosted on the zombie.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Botnet

---

Pregunta #20 de 67

Id. de pregunta: 1105615

Se le ha encomendado el desarrollo de una nueva aplicación para su organización. Estás involucrado en la fase de inicio del proyecto. ¿Qué actividad debe implementar durante esta fase?

- ✗ **A)** definir la línea base funcional formal
- ✓ **B)** identificación de amenazas y vulnerabilidades
- ✗ **C)** pruebas de funcionalidad y rendimiento
- ✗ **D)** certificación y acreditación

Explanation

Identification of threats and vulnerabilities takes place during the project initiation phase of an application development life cycle. The project initiation phase involves obtaining management approval and the performing an initial risk analysis. Risk analysis identifies the potential threats and vulnerabilities based on the environment in which the product will perform data processing, the sensitivity of the data required, and the mechanisms that should be a part of the product as a countermeasure.

Certification and accreditation are the processes implemented during the implementation of the product. Certification is the process of technically evaluating and reviewing a product to ensure that it meets the stated security requirements. Accreditation is a process that involves a formal acceptance of the product and its responsibility by management. Accreditation is the final step in authorizing a system for use in an environment.

Defining formal functional baseline is included in the functional design analysis stage and not in the project initiation stage. A formal functional baseline can include security tasks and development, as well as testing plans to ensure that the security requirements are defined properly.

Functionality and performance tests are conducted in an environment during software development to assess a product against a set of requirements.

In a product development lifecycle, it is important that security be a part of the overall design and be integrated at each stage of product development. The security of an application is most effective and economical when the application is originally designed

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Security in the System and Software Development Life Cycles

---

Pregunta #21 de 67

Id. de pregunta: 1105604

¿Qué tipo de ataque malintencionado utiliza secuencias de comandos de Visual Basic?

- ✗ **A)** un ataque de ingeniería social
- ✓ **B)** un ataque de caballo de Troya
- ✗ **C)** un ataque de buceo de contenedor de basura
- ✗ **D)** un ataque de denegación de servicio

Explanation

Visual Basic scripting (VBS) is typically used to code Trojan horses. Trojan horses are digital pests hidden in seemingly benign programs. Trojan horses are designed to perform malicious actions, such as retrieve passwords or erase files. VBS programs typically have .vbs file name extensions, and are often transmitted through e-mail messages. An administrator can protect a network from VBS scripts by preventing e-mail clients from either downloading or running attachments with .vbs file name extensions.

A social engineering attack occurs when a hacker poses as a company employee or contractor to gain information about a network from legitimate company employees. A hacker typically uses social engineering to gain user names and passwords or sensitive documents by non-technical means, such as posing as an employee or dumpster diving.

A denial of service (DoS) attack occurs when a hacker floods a network with requests so that legitimate users cannot gain access to resources on a computer or a network.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Identity and Access Management, Malicious Software

---

Pregunta #22 de 67

Id. de pregunta: 1105605

¿Qué tipo de virus está diseñado específicamente para infectar programas a medida que se cargan en la memoria?

- ✗ **A)** transeúnte
- ✓ **B)** residente
- ✗ **C)** replicación del sector de inicio
- ✗ **D)** compañero

Explanation

A resident virus is specifically designed to infect programs as they are loaded into memory.

A companion virus is designed to take advantage of the extension search order of an operating system. A nonresident virus is part of an executable program file on a disk that is designed to infect other programs when the infected program file is started. A boot sector replicating virus is written to the boot sector of a hard disk on a computer and is loaded into memory each time a computer is started.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Identity and Access Management, Malicious Software

---

Pregunta #23 de 67

Id. de pregunta: 1105621

¿Qué función proporciona la llamada a procedimiento remoto (RPC)?

✗ **A)** Identifica componentes dentro de un entorno de computación distribuida (DCE).

✗ **B)** Proporciona un sistema de archivos integrado que todos los usuarios del entorno distribuido pueden compartir.

✓ **C)** Permite la ejecución de rutinas individuales en equipos remotos a través de una red.

✗ **D)** Proporciona código que se puede transmitir a través de una red y ejecutar de forma remota.

Explanation

Remote procedure call (RPC) allows the execution of individual routines on remote computers across a network. It is used in a distributed computing environment (DCE).

Globally unique identifiers (GUIDs) and universal unique identifiers (UUIDs) are used to identity components within a DCE. They uniquely identify users, resources, and other components in the environment. A UUID is used in a Distributed Computing Environment.

Mobile code is code that can be transmitted across a network and executed remotely. Java and ActiveX code downloaded into a Web browser from the World Wide Web (WWW) are examples of mobile code.

A distributed file service (DFS) provides an integrated file system that all users in the distributed environment can share.

A directory service ensures that services are made available only to properly designated entities.

Objective:
Software Development Security

Sub-Objective:
Define and apply secure coding guidelines and standards

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Security of Application Programming Interfaces

---

Pregunta #24 de 67

Id. de pregunta: 1105599

¿Qué amenaza de seguridad es una aplicación de software que muestra anuncios mientras se ejecuta la aplicación?

✓ **A)** adware

- ✗ **B)** virus
- ✗ **C)** spyware
- ✗ **D)** gusano

Explanation

Adware is a software application that displays advertisements while the application is executing. Some adware is also spyware that monitors your Internet usage and personal information. Some adware will even allow credit card information theft.

A worm is a program that spreads itself through network connections.

Spyware often uses tracking cookies to collect and report on a user's activities. Not all spyware is adware, and not all adware is spyware. Spyware requires that your activities be monitored and tracked; adware requires that advertisements be displayed.

A virus is malicious software (malware) that relies upon other application programs to execute itself and infect a system.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security. Spyware/Adware

Adware, http://searchcio-midmarket.techtarget.com/sDefinition/0,%20sid183_gci521293,00.html

Unexplained computer behavior may be caused by deceptive software, http://support.microsoft.com/kb/827315

Pregunta #25 de 67

Id. de pregunta: 1105612

¿Qué afirmaciones son ciertas con respecto a las evaluaciones de procesos de software? (Elija todo lo que se aplique).)

- ✗ **A)** Identifican a los contratistas que están calificados para desarrollar software o para monitorear el estado del proceso de software en un proyecto de software actual.

- ✓ **B)** Determinan el estado del proceso de software actual de una organización y se utilizan para obtener soporte desde dentro de la organización para un programa de mejora de procesos de software.
- ✗ **C)** Desarrollan un perfil de riesgo para la selección de fuentes.
- ✓ **D)** Desarrollan un plan de acción para la mejora continua de los procesos.

Explanation

Software process assessments determine the state of an organization's current software process and are used to gain support from within the organization for a software process improvement program. In addition, they develop an action plan for continuous process improvement.

Software capability evaluations identify contractors who are qualified to develop software or to monitor the state of the software process in a current software project. In addition, they develop a risk profile for source selection.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

Rethinking the Concept of Software Process Assessment,
http://www.iscn.at/select_newspaper/assessments/sintef_ass.html

---

Pregunta #26 de 67

Id. de pregunta: 1114792

Durante un proyecto de desarrollo de software, debe asegurarse de que el progreso del período del proyecto se supervisa adecuadamente. ¿Qué técnica(s) se pueden utilizar?

un. Diagramas de Gantt

B. Pruebas unitarias

c. Técnica delphi

d. Gráficos de la técnica de revisión de la evaluación del programa

e. Gráficos de técnicas de revisión de evaluación de prototipos

- ✗ **A)** opción b
- ✗ **B)** opción e

- ✗ **C)** Sólo las opciones C y D
- ✗ **D)** Sólo las opciones A y B
- ✗ **E)** Opción d
- ✗ **F)** Opciones C y E Solamente
- ✗ **G)** opción A
- ✗ **H)** opción c
- ✓ **I)** Sólo las opciones A y D

Explanation

Periodical progress of a project can be monitored by using Gantt charts and the Program Evaluation Review Technique (PERT) charts.

Gantt charts are bar charts that represent the progress of tasks and activities over a period of time. Gantt charts depict the timing and the interdependencies between the tasks. Gantt charts are considered a project management tool to represent the scheduling of tasks and activities of a project, the different phases of the project, and their respective progress. Gantt charts serve as an industry standard.

A PERT chart is a project management model invented by the United States Department of Defense. PERT is a method used for analyzing the tasks involved in completing a given project and the time required to complete each task. PERT can also be used to determine the minimum time required to complete the total project.

Unit testing refers to the process in which the software code is debugged by a developer before it is submitted to the quality assurance team for further testing.

The Delphi technique is used to ensure that each member in a group decision-making process provides an honest opinion on the subject matter in question. Group members are asked to provide their opinion on a piece of paper in confidence. All these papers are collected, and a final decision is taken based on the majority. Delphi technique is generally used either during the risk assessment process or to estimate the cost of a software development project.

A prototype is a model or a blueprint of the product and is developed according to the requirements of customers. There is no process known as the Prototype Evaluation Review Technique charts.

Cost-estimating techniques include the Delphi technique, expert judgment, and function points.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Program Evaluation and Review Technique chart, http://searchsoftwarequality.techtarget.com/definition/PERT-chart

Top Ten Benefits of Gantt Charts, http://www.brighthubpm.com/templates-forms/2434-top-ten-benefits-of-a-gantt-chart/

Pregunta #27 de 67

Id. de pregunta: 1111810

Durante el ciclo de vida de desarrollo de la aplicación, el equipo realiza pruebas para depurar las instrucciones de código. ¿Qué método de prueba de software está utilizando el equipo?

✓ **A)** pruebas unitarias

✗ **B)** pruebas perpendiculares

✗ **C)** pruebas de caja azul

✗ **D)** pruebas verticales

Explanation

A part of the application development lifecycle, unit testing is an internal testing performed to debug the code instructions. Unit testing is performed by the developer rather than by the quality assurance team. After the code is developed, it is sent to the quality assurance team for evaluation and detection of anomalies, functional errors, and security loopholes. Unit testing can use test design methods, such as white box and black box. Keep the unit testing guidelines in mind:

- The test data is part of the specification.
- Correct test output results should be developed and known beforehand.
- Testing should check for out-of-range values and other bounds conditions.

Perpendicular testing, blue-box testing, and vertical testing are not valid categories of software test approaches in an application development life cycle.

Black-box testing does not explicitly use the knowledge of the internal structure. The black-box test design typically focuses on testing functional requirements. Black-box testing implies that the selection of test data and the interpretation of test results are performed on the basis of the functional properties of software rather than its internal structure.

The white-box technique focuses only on testing the design and internal logical structure of the software product rather than its functionality. In general, the software testing should be planned, and the results of the tests should be documented throughout the software development life cycle as permanent records.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Test/Validate

---

Pregunta #28 de 67

Id. de pregunta: 1105620

Su empresa decide que se debe comprar un nuevo producto de software para ayudar al personal de marketing a administrar sus campañas de marketing y los recursos utilizados. ¿Durante qué fase del proceso de adquisición de software documenta los requisitos de software?

- ✗ **A)** Fase de supervisión
- ✗ **B)** Fase de mantenimiento
- ✓ **C)** Fase de planificación
- ✗ **D)** Fase de contratación

Explanation

During the planning phase, the software requirements are documented. You should also create an acquisition strategy during this phase and develop the evaluation criteria.

During the contracting phase, you should issue the request for proposal (RFP), evaluate the proposals, and complete final contract negotiations with the selected seller.

During the monitoring phase, you should ensure that the supplier completes the contract and formally accept the final product.

In the maintaining phase, you should maintain the software, including possibly decommissioning the software at some future date.

Objective:
Software Development Security

Sub-Objective:
Assess security impact of acquired software

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Security Impact of Acquired Software

---

Pregunta #29 de 67

Id. de pregunta: 1105606

¿Qué tipo de virus está diseñado específicamente para aprovechar el orden de búsqueda de extensiones de un sistema operativo?

- ✗ **A)** residente
- ✗ **B)** replicación del sector de inicio
- ✗ **C)** transeúnte
- ✓ **D)** compañero

Explanation

A companion virus is specifically designed to take advantage of the extension search order of an operating system. In Microsoft Windows, the extension search order is .com, .exe, then .bat. For example, when a user starts a program named calc on a Windows operating system, Windows first looks for a program named calc.com in the current folder. If a virus is named calc.com, and the actual program file is named calc.exe, then the virus will be started instead of the calc.exe program because Windows will stop searching after it finds calc.com.

A resident virus is loaded into memory and infects other programs as they in turn are loaded into memory. A nonresident virus is part of an executable program file on a disk and infects other programs when the infected program file is started. A boot sector replicating virus is written to the boot sector of a hard disk on a computer and is loaded into memory each time a computer is started.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Identity and Access Management, Malicious Software

---

Pregunta #30 de 67

Id. de pregunta: 1111818

¿Qué afirmación es cierta de la información?

- ✗ **A)** Un ataque de salami no es un ejemplo de diddling de datos.
- ✗ **B)** El diddling de datos se utiliza para extraer información confidencial sobre los empleados.
- ✓ **C)** La diddling de datos hace referencia a la manipulación de los datos de entrada en una aplicación.

✗ **D)** El diddling de datos está asociado con los forasteros de una organización.

Explanation

Data diddling is an active attack that involves manipulation of data while the data is being entered into an application. Data diddling techniques, such as a salami attack, involve alteration of small amounts of data while it enters an application. Therefore, data diddling is considered an active attack rather than a passive attack. A data diddling attack occurs when an employee has been shaving off pennies from multiple accounts and depositing the funds into his own bank account.

A salami attack involves committing numerous small crimes multiple number of times to ensure that no one notices the larger crime. For example, a salami attack will subtract an insignificant amount of fund from multiple accounts over a large period of time, and direct these funds to the attacker's account. This process amounts to committing fraud because the amount transferred each time is insignificant in nature and goes unnoticed.

Data diddling is typically attributed to employees who are aware of the internal working of the processes of the organization and the corresponding input and process controls. Therefore, employees are able to perform acts of data manipulation through techniques, such as salami.

Other techniques affecting the confidentiality and integrity of business operations of an organization are as follows:

- Scavenging involves an attacker seeking sensitive information without knowing the format of the information.
- Sniffing involves extracting confidential information, such as user credentials, bank account numbers, and personal identification numbers.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Data Diddling, http://cybercrimeandforensic.blogspot.com/2009/02/data-diddling.html

---

Pregunta #31 de 67

Id. de pregunta: 1105558

¿Qué afirmación es cierta de los lenguajes de programación?

✗ **A)** Un lenguaje de programación de alto nivel requiere más tiempo para codificar instrucciones.

✗ **B)** El compilador traduce un comando a la vez.

✗ **C)** La alta cohesión y el alto acoplamiento representan la mejor programación.

✓ **D)** Los ensambladores traducen el lenguaje ensamblador al lenguaje de máquina.

Explanation

Assemblers translate assembly language into machine language.

Interpreters translate one command at a time. Compilers translate large sections of program instructions.

The cohesive module refers to a piece of software code that either does not depend on or depends less on other software modules to be executed. High cohesiveness of a software program represents best programming due to reduced dependency levels. Coupling refers to the level of interconnection required between various software modules in a software program to perform a specific task. A lower coupling indicates lesser dependence on other programs and higher performance.

High-level languages require less time to code a program compared to low-level programming languages. This is because high-level languages use objects that act as independent functional modules having a specific functionality and reduce the number of programmers involved in coding application instructions.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Assembly Languages and Assemblers

---

Pregunta #32 de 67

Id. de pregunta: 1114793

¿Qué extensiones se usan para asignar nombres a los archivos por lotes en un entorno Microsoft?

a.bat

b. cmd

c.dll

d.exe

✗ **A)** Sólo opciones B y C

✗ **B)** Opción d

- ✓ **C)** Sólo las opciones A y B
- ✗ **D)** opción b
- ✗ **E)** opción c
- ✗ **F)** Sólo las opciones C y D
- ✗ **G)** opción A

Explanation

The .bat and .cmd extensions are used for naming batch files in a Microsoft environment.

The .dll file extension indicates a dynamic link library file. These are generally used in device drivers and for other configuration purposes.

The .exe file extension indicates an executable file. Executable files are used to start programs and applications.

Batch files are very similar to script files in the Unix environment.

Scripts and batch files are created to decrease administrator workload. The files contain the commands to perform certain tasks. Common usage of these file types include file manipulation, text and report printing, and program execution. A batch file or script contains all the commands needed to execute and complete the tasks. It reduces administrative effort because the administrator simply starts the batch file, instead of having to execute each of the commands within the batch file separately.

Batch files and scripts may contain login credentials. For this reason, they should be stored in a protected area.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Guide to Windows Batch Scripting, http://steve-jansen.github.io/guides/windows-batch-scripting/

Pregunta #33 de 67

Id. de pregunta: 1105613

Su organización usa una base de datos relacional para almacenar la información de contacto del cliente. Debe modificar el esquema de la base de datos relacional. ¿Qué componente identifica esta información?

- ✗ **A)** Lenguaje de control de datos (DCL)
- ✗ **B)** lenguaje de manipulación de datos (DML)

✗ **C)** lenguaje de consulta (QL)

✓ **D)** lenguaje de definición de datos (DDL)

Explanation

The data definition language (DDL) identifies the schema of the database. The schema of a database defines the type of data that the database can store and manipulate. The schema is the description of a relational database. The schema also defines the properties of the type of data that a database can store as valid data objects. DDL is also used to create and delete views and relations between tables.

The query language (QL) is used to generate a request for information from a user in the form of query statements to obtain relevant output.

The data control language (DCL) manages access control to records in a database. DCL defines the granular user permissions to various data objects and implements database security. Examples of DCL commands are grant, deny, revoke, delete, update, and read.

The data manipulation language (DML) refers to a suite of computer languages used by database users to retrieve, insert, delete, and update data in a database. DML provides users the ability to store, retrieve, and manipulate the data according to the relevant instructions issued.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

Data definition language, https://whatis.techtarget.com/definition/Data-Definition-Language-DDL

Pregunta #34 de 67

Id. de pregunta: 1114017

¿Qué declaración describe correctamente un caballo de Troya?

✗ **A)** Para ser ejecutado, depende de otros programas.

✗ **B)** Es una técnica de ingeniería social.

✓ **C)** Incrusta código malicioso dentro de utilidades útiles.

✗ **D)** Modifica las direcciones IP en un paquete IP para imitar una fuente autorizada.

Explanation

A Trojan horse is a malware that is disguised as a useful utility but is embedded with malicious code. When the disguised utility is run, the Trojan horse carries out malicious operations in the background and provides the useful utility on the front end. Trojan horses use covert channels to carry out malicious operations. Malicious activities may include deleting system files or planting a backdoor into a system for later access. Trojan horses are typically installed as a rogue application in the background to avoid suspicion by the user.

A Trojan horse is not a social engineering technique. Social engineering involves tricking another person into sharing confidential information by posing as an authorized individual. Social engineering is a non-technical intrusion that relies heavily on human interaction and typically involves tricking other people into break normal security procedures.

To be executed, a Trojan horse does not depend on other application programs. A virus is the malicious software (malware) that relies upon other application programs to execute and infect a system. The main criterion for classifying a piece of executable code as a virus is that it spreads itself through applications running on a host system. A virus infects an application by replicating itself.

A Trojan horse does not modify the IP address in an IP packet to imitate an authorized source. IP spoofing refers to modification of a source IP address in an IP datagram to imitate the IP address of the packet originating from an authorized source. This results in the target computer setting up communication with the attacker's computer. This process provides access to restricted resources in the target computer. In a spoofing attack, also referred to as a masquerading attack, a person or program is able to masquerade successfully as another person or program. A man-in-the-middle attack is an example of a spoofing as well as a session hijacking attack. Other types of spoofing attacks are e-mail spoofing and Web spoofing.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Identity and Access Management, Malicious Software

---

Pregunta #35 de 67

Id. de pregunta: 1105579

¿Qué es un ejemplo de escalada de privilegios?

- ✗ **A)** obtener acceso a un archivo restringido mediante un caballo de Troya
- ✗ **B)** Obtener acceso a un sistema mediante las credenciales de otro usuario
- ✗ **C)** obtener acceso a un sistema suplantando a un usuario para obtener sus credenciales

✓ **D)** Obtener acceso a un archivo restringido cambiando los permisos de su cuenta válida

Explanation

An example of privilege escalation is gaining access to a file you should not have rights to access by changing the permissions of your valid account. Privilege escalation describes logging in to a system using your valid user account and then finding a way to access files that you do not have permissions to access. This usually involves invoking a program that can change your account permissions, such as Set User ID (SUID) or Set Group ID (SGID), or by invoking a program that runs in an administrative context.

There are several methods of dealing with privilege escalation, including using least privilege accounts and privilege separation. Privilege escalation can lead to denial of service (DoS) attacks.

Gaining access to a system by using another user's credentials is a form of hacking.

Gaining access to a system by impersonating a user to obtain his credentials is a form of social engineering.

Gaining access to a file by using a Trojan horse is not privilege escalation.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Escalation of Privileges and Rootkits

---

Pregunta #36 de 67

Id. de pregunta: 1105573

¿Cuál es otro nombre para un ataque asincrónico?

✓ **A)** ataque de tiempo de comprobación/tiempo de uso (TDC/CDU)

✗ **B)** condición de carrera

✗ **C)** desbordamiento de búfer

✗ **D)** gancho de mantenimiento

Explanation

A TOC/TOU attack is another name for an asynchronous attack. This attack happens when an attacker interrupts a task and changes something to affect the result. The tasks occur in the correct order but the data transmitted by the

tasks is changed in some manner.

None of the other options is considered another name for an asynchronous attack. A race condition differs from a TOC/TOU attack in that a race condition actually makes processes execute in a different order to affect the result.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Time of Check/Time of Use (TOC/TOU)

Pregunta #37 de 67

Id. de pregunta: 1105569

¿Qué término describe la capacidad de un módulo para realizar su trabajo sin usar otros módulos?

- ✗ **A)** alto acoplamiento
- ✓ **B)** acoplamiento bajo
- ✗ **C)** alta cohesión
- ✗ **D)** baja cohesión

Explanation

Low coupling describes a module's ability to perform its job without using other modules.

High coupling would imply that a module must interact with other modules to perform its job.

Cohesion reflects the different types of tasks that a module carries out. High cohesion means a module is easier to update and does not affect other modules. Low cohesion means a module carries out many tasks, making it harder to maintain and reuse.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Coupling

Pregunta #38 de 67

Id. de pregunta: 1114010

¿Qué declaración define correctamente el modelo de madurez de la capacidad en el contexto del desarrollo de software?

✗ **A)** Es un modelo basado en la realización de revisiones y la documentación de las revisiones en cada fase del ciclo de desarrollo de software.

✗ **B)** Se trata de un modelo basado en el análisis del riesgo y la construcción de prototipos y simulaciones durante las distintas fases del ciclo de desarrollo de software.

✗ **C)** Es un modelo formal basado en la capacidad de una organización para atender proyectos.

✓ **D)** Es un modelo que describe los principios, procedimientos y prácticas que se deben seguir en el ciclo de desarrollo de software.

Explanation

The capability maturity model (CMM) describes the principles, procedures, and practices that should be followed by an organization in a software development life cycle. The capability maturity model defines guidelines and best practices to implement a standardized approach for developing applications and software programs. It is based on the premise that the quality of a software product is a direct function of the quality of its associated software development and maintenance processes. This model allows a software development team to follow standard and controlled procedures, ensuring better quality and reducing the effort and expense of a software development life cycle. The CMM builds a framework for gap analysis and enables a software development organization to constantly improve their processes.

A software process is a set of activities, methods, and practices that are used to develop and maintain software and associated products. Software process capability is a means of predicting the outcome of the next software project conducted by an organization.

Based on the level of formalization of the life cycle process, the five maturity levels defined by the CMM are as follows:

- Initial: The development procedures are not organized, and the quality of the product is not assured at this level.
- Repeatable: The development process involves formal management control, proper change control, and quality assurance implemented while developing applications.
- Defined: Formal procedures for software development are defined and implemented at this level. This category also provides the ability to improve the process.

- Managed: This procedure involves gathering data and performing an analysis. Formal procedures are established, and a qualitative analysis is conducted to analyze gaps by using the metrics at this level.
- Optimized: The organization implements process improvement plans and lays out procedures and budgets.

Other software development models include the cleanroom model, the waterfall model, and the spiral model:

- The cleanroom model follows well-defined formal procedures for development and testing of software. The cleanroom model calls for strict testing procedures and is often used for critical applications that should be certified.
- The waterfall model is based on proper reviews and the documenting of reviews at each phase of the software development cycle. This model divides the software development cycle into phases. Proper review and documentation must be completed before moving on to the next phase.
- The spiral model is based on analyzing the risk, building prototypes, and simulating the application tasks during the various phases of development cycle. The spiral model is typically a metamodel that incorporates a number of software development models. For example, the basic concept of the spiral model is based on the waterfall model. The spiral model depicts a spiral that incorporates various phases of software development. In the spiral model, the radial dimension represents cumulative cost.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, CMMI

Pregunta #39 de 67

Id. de pregunta: 1105624

¿Qué técnica de spyware inserta una biblioteca de vínculos dinámicos en la memoria de un proceso en ejecución?

✗ **A)** Galletas
✓ **B)** Inyección de DLL
✗ **C)** desbordamiento de búfer
✗ **D)** Retransmisión abierta SMTP

Explanation

DLL injection is a spyware technique that inserts a dynamic link library (DLL) into a running process's memory. Windows was designed to use DLL injection to make programming easier for developers. Some of the standard

defenses against DLL injection include application and operating system patches, firewalls, and intrusion detection systems.

SMTP open relay is an e-mail feature that allows any Internet user to send e-mail messages through the SMTP server. SMTP relay often results in an increased amount of spam. SMTP relay is designed into many e-mail servers to allow them to forward e-mail to other e-mail servers.

Buffer overflow occurs when the length of the input data is longer than the length processor buffers can handle. Buffer overflow is caused when input data is not verified for appropriate length at the time of the input. Insufficient bounds checking causes buffer overflows. Buffer overflow and boundary condition errors are examples of input validation errors.

Cookies store information on a Web client for future sessions with a Web server. It is used to provide a persistent, customized Web experience for each visit and to track a user's browser habits. The information stored in a cookie is not typically encrypted and might be vulnerable to hacker attacks.

Objective:
Software Development Security

Sub-Objective:
Define and apply secure coding guidelines and standards

References:

DLL Injection and Hooking, http://securityxploded.com/dll-injection-and-hooking.php

---

Pregunta #40 de 67

Id. de pregunta: 1105603

¿Qué tipo de código malicioso está oculto dentro de un programa benigno cuando se escribe el programa?

- ✗ **A)** un gusano
- ✗ **B)** un virus
- ✓ **C)** un caballo de Troya
- ✗ **D)** una bomba lógica

Explanation

A Trojan horse is a type of malicious code that is embedded in an otherwise benign program when the program is written. A Trojan horse is typically designed to do something destructive when the infected program is started. Trojan horses, viruses, worms, and logic bombs are all examples of digital pests. Software development companies should consider reviewing code to ensure that malicious code is not included in their products.

A virus is added to a program file after a program is written. A virus is often associated with malicious programs that are distributed in e-mail messages. A worm creates copies of itself on other computers through network connections. A logic bomb is designed to initiate destructive behavior in response to a particular event. For example, a logic bomb might be programmed to erase a hard disk after 12 days.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Identity and Access Management, Malicious Software

---

Pregunta #41 de 67

Id. de pregunta: 1111811

Durante el reciente desarrollo de una nueva aplicación, el cliente solicitó un cambio. Debe implementar este cambio según el proceso de control de cambios. ¿Cuál es el primer paso que debe implementar?

✗ **A)** Registre la solicitud de cambio.
✗ **B)** Obtener la aprobación de la gerencia.
✓ **C)** Analice la solicitud de cambio.
✗ **D)** Envíe los resultados del cambio a la administración.

Explanation

You should analyze the change request. The change control procedures ensure that all modifications are authorized, tested, and recorded. Therefore, these procedures serve the primary aim of auditing and review by the management. The necessary steps in a change control process are as follows:

1. Make a formal request.
2. Analyze the request. This step includes developing the implementation strategy, calculating the costs of the implementation, and reviewing the security implication of implementing the change.
3. Record the change request.
4. Submit the change request for approval. This step involves getting approval of the actual change once all the work necessary to complete the change has been analyzed.
5. Make changes. The changes are implemented and the version is updated in this step.
6. Submit results to management: In this step, the change results are reported to management for review.

A stringent change management process ensures that all the changes are implemented and recorded related to production systems, and enforces separation of duties. For instance, in a software development environment, changes made to production software programs are performed by operational staff rather than the software programmers, who are responsible for coding software applications for clients. Such a process ensures that the changes are implemented in the proper manner and the process is documented. Change management is about the decision to make the change.

Configuration management is not the same as change management. Configuration management is about tracking the actual change. It is the discipline of identifying the components of a continually evolving system for the purposes of controlling changes to those components and maintaining integrity and traceability throughout the life cycle. Configuration management controls the changes that take place in hardware, software, and operating systems by assuring that only the proposed and approved system changes are implemented. In configuration management, a configuration item is a component whose state is to be recorded and against which changes are to be progressed. In configuration management, a software library is a controlled area accessible only to approved users who are restricted to the use of an approved procedure.

Configuration control is controlling changes to the configuration items and issuing versions of configuration items from the software library. Configuration management includes configuration control, configuration status accounting, and configuration auditing.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Change Management and Configuration Management/Replacement

---

Pregunta #42 de 67

Id. de pregunta: 1105575

¿Qué tipo de virus se instala en el sistema antivirus e intercepta las llamadas que el sistema antivirus realiza al sistema operativo?

- ✗ **A)** virus de secuencia de comandos
- ✓ **B)** virus de túnel
- ✗ **C)** virus meme
- ✗ **D)** virus del sector de inicio

Explanation

A tunneling virus installs itself under the anti-virus system and intercepts any calls that the anti-virus system makes to the operating system.

A script virus includes lines of instructions that are written in a scripting language, like VBScript or JavaScript. The code in the script carries out malicious activities, such as copying itself to everyone in your contact list.

A meme virus is not really a virus. Any e-mail message that is continually forwarded around the Internet is considered a meme virus. While meme viruses do not truly "infect" a system, they cause performance degradation just from the number of times they are forwarded.

A boot sector virus infects the boot sector of a hard drive. They usually move data within the boot sector or overwrite the sector with new information.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Tunneling Virus, https://www.techopedia.com/definition/4147/tunneling-virus

---

Pregunta #43 de 67

Id. de pregunta: 1105580

¿Qué declaración define correctamente los ataques de spam?

- ✗ **A)** Envío de varios paquetes falsificados con el indicador SYN establecido en el host de destino en un puerto abierto
- ✓ **B)** Enviar repetidamente correos electrónicos idénticos a una dirección específica
- ✗ **C)** uso de mensajes de eco ICMP de gran tamaño para inundar el equipo de destino
- ✗ **D)** enviar paquetes falsificados con la misma dirección de origen y de destino

Explanation

A spamming attack involves flooding an e-mail server or specific e-mail addresses repeatedly with identical unwanted e-mails. Spamming is the process of using an electronic communications medium, such as e-mail, to send unsolicited messages to users in bulk. Packet filtering routers typically do not prove helpful in such attacks because packet filtering

routers do not examine the data portion of the packet. E-mail filter programs are now being embedded either in the e-mail client or in the server. E-mail filter programs can be configured to protect from spamming attacks to a great extent.

A ping of death is a type of DoS attack that involves flooding target computers with oversized packets and exceeding the acceptable size during the process of reassembly. This causes the target computer to either freeze or crash. Other DoS attacks, named smurf and fraggle, deny access to legitimate users by causing a system to either freeze or crash.

In a SYN flood attack, the attacker floods the target with the spoofed IP packets, causing it to either freeze or crash. The Transmission Control Protocol (TCP) uses the synchronize (SYN) and acknowledgment (ACK) packets to establish communication between two host computers. The exchange of the SYN, SYN-ACK, and ACK packets between two host computers is referred to as handshaking. Attackers flood the target computers with a series of SYN packets to which the target host computer replies. The target host computer then allocates resources to establish a connection. The IP address is spoofed. Therefore, the target host computer never receives a valid response in the form of ACK packets from the attacking computer. When the target computer receives many SYN packets, it runs out of resources to establish a connection with the legitimate users and becomes unreachable for the processing of valid requests.

A land attack involves sending multiple spoofed TCP SYN packets with the target host's IP address and an open port as both the source and the destination to the target host on an open port. The land attack causes the system to either freeze or crash because the computer replies to itself.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Email spam, https://searchsecurity.techtarget.com/definition/spam

---

Pregunta #44 de 67

Id. de pregunta: 1114014

¿Qué herramienta ayuda en el diseño del diseño de desarrollo de aplicaciones como parte del ciclo de vida del desarrollo de aplicaciones?

- ✗ **A)** Espiral
- ✗ **B)** Agregación
- ✓ **C)** CASO
- ✗ **D)** Delfos

Explanation

Computer-aided software engineering (CASE) refers to the use of software tools to assist in the development and maintenance of application software: CASE business and functional analysis, system design, code storage, compilers, translation, and testing software. Middle CASE products are used for developing detailed designs, such as screen and report layouts.

Aggregation is a database security concern that arises when a user does not have access to sensitive data, but can access portions of it. This loophole enables a user to aggregate the data, and use it to deduce a fact.

Delphi is a technique of expert judgment that ensures each member in a group decision-making process provides an honest opinion on the subject matter in question. Group members are asked to provide their views on the subject in writing. All these papers are collected, and a final decision is taken based on the majority. Delphi technique is generally used either during the risk assessment process or to estimate the cost of a software development project.

Spiral is a software development model based on analyzing the risk, building prototypes, and simulating the application tasks during the various phases of development cycle. The spiral model is typically a meta model that incorporates a number of software development models. The basic concept of the spiral model is based on the waterfall model.

Software engineering is defined as the science and art of specifying, designing, implementing, and evolving programs, documentation, and operating procedures whereby computers can be made useful to man.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Computer-aided Software Engineering, use artificial intelligence to extract new information from a set of informationhttp://www.selectbs.com/analysis-and-design/computer-aided-software-engineering-case-tool

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Computer-Aided Software Engineering (CASE)

Pregunta #45 de 67

Id. de pregunta: 1105592

¿Cuál es la mejor descripción de CAPI?

- ✗ **A)** una interfaz de programación de aplicaciones que utiliza Kerberos
- ✗ **B)** Una interfaz de programación de aplicaciones que utiliza la autenticación en dos fases
- ✓ **C)** Una interfaz de programación de aplicaciones que proporciona cifrado

✗ **D)** Una interfaz de programación de aplicaciones que proporciona responsabilidad

Explanation

Cryptographic application programming interface (CAPI) is an application programming interface that provides encryption.

None of the other options is a description of CAPI.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

Microsoft CyptoAPI, https://en.wikipedia.org/wiki/Microsoft_CryptoAPI

Pregunta #46 de 67

Pregunta con ID: 1105600

¿Qué amenaza de seguridad utiliza a menudo cookies de seguimiento para recopilar e informar sobre las actividades de un usuario?

✗ **A)** Caballo de Troya
✗ **B)** gusano
✓ **C)** spyware
✗ **D)** virus

Explanation

Spyware often uses tracking cookies to collect and report on a user's activities to the spyware programmer.

None of the other options is correct. A virus is malicious software (malware) that relies upon other application programs to execute itself and infect a system. A worm is a program that spreads itself through network connections. A Trojan horse is malware that is disguised as a useful utility, but embeds malicious code in itself.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security. Spyware/Adware

Spyware, http://searchsecurity.techtarget.com/sDefinition/0,%20sid14_gci214518,00.html

---

Pregunta #47 de 67

Id. de pregunta: 1163833

¿Cuál es el proceso para garantizar que las políticas de seguridad corporativas se lleven a cabo de manera consistente?

- ✓ **A)** auditoría
- ✗ **B)** Escaneo
- ✗ **C)** huellas
- ✗ **D)** ingeniería social

Explanation

Auditing is the process of ensuring the corporate security policies are carried out consistently.

Social engineering is an attack that deceives others to obtain legitimate information about networks and computer systems. Footprinting is the process of identifying the network and its security configuration. Scanning is the process that hackers use to identify how a network is configured

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

Conducting a Security Audit: An Introductory Overview, https://www.symantec.com/connect/articles/conducting-security-audit-introductory-overview

---

Pregunta #48 de 67

Id. de pregunta: 1105560

¿Qué lenguaje de interfaz es una interfaz de programación de aplicaciones (API) que se puede configurar para permitir que cualquier aplicación consulte bases de datos?

✗ **A)** XML

✓ **B)** ODBC

✗ **C)** JDBC

✗ **D)** OLE DB

Explanation

Open Database Connectivity (ODBC) is an application programming interface (API) that can be configured to allow any application to query databases. The application communicates with the ODBC. The ODBC translates the application's request into database commands. The ODBC retrieves the appropriate database driver.

Java Database Connectivity (JBDC) is an API that allows a Java application to communicate with a database. Extensible Markup Language (XML) is a standard for arranging data so that it can be shared by Web technologies. Object Linking and Embedding Database (OLE DB) is a method of linking data from different databases together.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Asset Security, Database Interface Languages

---

Pregunta #49 de 67

Id. de pregunta: 1111807

¿Qué par de procesos deben separarse entre sí para gestionar la estabilidad del entorno de prueba?

✗ **A)** validez y producción

✗ **B)** pruebas y validez

✗ **C)** validez y seguridad

✓ **D)** pruebas y desarrollo

Explanation

The testing and development processes should be separated from each other to manage the stability of the test environment. Separating the test environment and the development environment is an example of separation of duties.

The responsibilities of the test and development staff in the software development life cycle (SDLC) process should be clearly distinguished. For example, debugging is performed by the programmer while coding the instructions. This

process is known as unit testing. After the software program is submitted, it is again verified by the quality assurance team by using formal procedures and practices. It is recommended that a software programmer develop the software, test it, and submit it to production. Separation of duties ensures that the quality assurance team conducts checks by using formal procedures. Software should be tested thoroughly before it is sent to the production environment. This will ensure that the software does not adversely affect the business operations of the organization.

All the other options are invalid in the contexts of the software development life cycle and separation of duties.

The SDLC includes the following phases:

- Plan/Initiate Project
- Gather Requirements
- Design
- Develop
- Test/Validate
- Release/Maintain
- Certify/Accredit
- Change Management and Configuration Management/Replacement

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Test/Validate

---

Pregunta #50 de 67

Id. de pregunta: 1114012

¿Qué virus está escrito en Visual Basic (VB) y es capaz de infectar sistemas operativos?

✗ **A)** virus polimórfico
✗ **B)** virus de sigilo
✗ **C)** virus auto-garbling
✓ **D)** virus de macro

Explanation

Macro viruses are programs written in Word Basic, Visual Basic, or VBScript. Macro viruses pose a major threat because their underlying language is simple, and they are easy to develop. Macro viruses can infect operating systems and applications, but most often affect Microsoft Office files. They do not rely on the size of the packet. The ability of macro viruses to move from one operating system to the other allows them to spread more effectively than other types of viruses. Macro viruses are typically used with Microsoft Office products.

A stealth virus hides the changes it makes to system files and boot records, making it difficult for antivirus software to detect its presence. A stealth virus keeps a copy of a file before infecting it and presents the original copy to the monitoring software. The stealth virus modifies the actual file and makes it difficult to detect the presence of the virus.

A self-garbling virus can hide itself from antivirus software by manipulating its own code. When a self-garbling virus spreads, it jumbles and garbles its own code to prevent the antivirus software from detecting its presence. A small part of the virus code later decodes the jumbled part to obtain the rest of the virus code to infect the system. The ability of the self-garbling virus to format its own code makes it difficult for an antivirus to detect its presence.

A polymorphic virus produces different operational copies of itself to evade detection by the antivirus software. There are usually multiple operational copies to ensure that in the event of an antivirus detection, only few copies are caught. A polymorphic virus is also capable of implementing encryption routines that will require different decryption routines to avoid detection.

Macro viruses written in Visual Basic for Applications almost exclusively affect operating systems.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Virus

---

Pregunta #51 de 67

Id. de pregunta: 1105578

Una aplicación personalizada se utiliza para administrar los archivos de recursos humanos de su empresa. Un administrador informa de que ciertos usuarios pueden realizar acciones que no deberían permitirse. Al investigar este problema, descubre que a los usuarios se les ha concedido un permiso inadecuado.

¿Qué tipo de amenaza de seguridad se ha producido?

- ✗ **A)** virus
- ✗ **B)** bomba lógica

✗ **C)** gusano

✓ **D)** elevación de privilegios

Explanation

Privilege escalation has occurred. Privilege escalation can be the accidental assignment of too high a permission set to a user or group of users. It can also be the result of bugs or back doors left in an application. The highest level of operator privilege is Access Change. Read Only is the lowest level of operator privilege.

A worm is a program that spreads itself through network connections.

A logic bomb implies a dormant program that is triggered following a specific action by the user or after a certain interval of time. The primary difference between logic bombs, viruses, and worms is that a logic bomb is triggered when specific conditions are met. An example of a logic bomb is a program that starts deleting files when a certain user ID is deleted.

A virus is malicious software (malware) that relies upon other application programs to execute and infect a system.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Escalation of Privileges and Rootkits

---

Pregunta #52 de 67

Id. de pregunta: 1105617

Trabaja para una empresa que crea soluciones de software personalizadas para los clientes. Recientemente, un cliente ha solicitado que su empresa proporcione un depósito de garantía de software. ¿Cuál es el propósito de esta solicitud?

✗ **A)** Para asegurarse de que existen las licencias de software adecuadas

✗ **B)** Para proporcionar una cuenta para comprar licencias de software

✗ **C)** Para proporcionar una copia de seguridad de todo el software utilizado por su empresa

✓ **D)** Para proporcionar el código fuente de un proveedor de software en caso de que el proveedor queque no haya salido del negocio

Explanation

The purpose of a software escrow is to provide a software vendor's source code in the event the vendor goes out of business. In a software escrow, a third party is responsible for holding the source code and other applicable materials. The software escrow contract ensures that both the software vendor and customer are protected.

All the other options are invalid.

Objective:
Software Development Security

Sub-Objective:
Assess security impact of acquired software

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Security Impact of Acquired Software

---

Pregunta #53 de 67

Id. de pregunta: 1105587

Una vez completado un proyecto de desarrollo de software, la administración decide reasignar sus recursos físicos, después de asegurarse primero de que no queden datos residuales en el medio. ¿Qué término se utiliza para describir esta práctica?

- ✓ **A)** reutilización de objetos
- ✗ **B)** intercambio dinámico de datos
- ✗ **C)** polimorfismo
- ✗ **D)** metadatos

Explanation

Object reuse refers to the allocation or reallocation of system resources after ensuring that there is no residual data left on the medium. Object reuse implies that all the confidential data is removed from the storage media to avoid disclosure of residual data. If a system allows simultaneous execution of multiple objects for different users, you should ensure that there is no disclosure of residual information. Object reuse implies that all the sensitive data should be removed from the memory location or from storage media before another subject can access the object. Object reuse involves reallocating storage space and ensuring no residual data remains that can be used for malicious purposes.

Metadata provides an insight into obscure data relationships. Metadata is the result of a new correlation between data components based on user instructions. Metadata is extracted from data mining techniques.

Polymorphism refers to an object oriented programming (OOP) concept and implies that different objects can provide different output based on the same input. This is achieved due to the difference in the functional properties of objects where each object performs a specific sub task.

The Dynamic Data Exchange (DDE) mechanism enables direct communication between two applications by using interprocess communications (IPC). Based on the client/server model, DDE allows two programs to directly exchange commands between each other. The source of the data is referred to as the server, and the system accessing the data is referred to as the client.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Object Reuse

---

Pregunta #54 de 67

Id. de pregunta: 1105608

¿Qué instrucción define correctamente el virus multiparte?

- ✓ **A)** Un virus multiparte puede infectar tanto los archivos ejecutables como los sectores de arranque de las unidades de disco duro.
- ✗ **B)** Un virus multiparte puede ocultarse del software antivirus distorsionando su código.
- ✗ **C)** Un virus multiparte se codifica en lenguaje de macros.
- ✗ **D)** Un virus multiparte puede cambiar algunas de sus características mientras se replica.

Explanation

A multipart virus can infect both executable files and boot sectors of hard disk drives. The multipart virus resides in the memory and then infects boot sectors and executable files of the computer system.

Macro viruses are platform independent and are typically used with Microsoft Office products. Macro viruses are programs written in Word Basic, Visual Basic, and VBScript. Macro viruses pose a major threat because the simplicity of the underlying language makes them easy to develop.

A stealth virus hides the changes it makes to system files and boot records, making it difficult to detect its presence. A stealth virus maintains a copy of a file before infecting it and presents the original copy to the monitoring software. Therefore, a stealth virus modifies the actual file and makes it difficult to detect the presence of the virus.

A self-garbing type of virus can hide itself from antivirus software by distorting its own code. When a self-garbling virus spreads, it jumbles and garbles its own code to prevent the antivirus software from detecting its presence. A small part of a virus code later decodes the jumbled part to obtain and subsequently execute the rest of the virus code. The ability of the self-garbling virus to format its own code makes it difficult for an antivirus to detect its presence.

At some point during the patch application process, a file may become infected with a virus. When this is discovered, you will need to recover the file by replacing the existing, infected file with an uninfected backup copy. This may possibly result in an older version of the file being restored that does not have all the patches applied.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Identity and Access Management, Malicious Software

---

Pregunta #55 de 67

Id. de pregunta: 1105598

¿Cuál es la definición de polimorfismo?

- ✗ **A)** la capacidad de suprimir detalles superfluos para que se puedan examinar las propiedades importantes
- ✗ **B)** El proceso de categorización de objetos que serán apropiados para una solución
- ✗ **C)** una representación de un problema del mundo real
- ✓ **D)** Cuando diferentes objetos responden al mismo comando o entrada de maneras diferentes

Explanation

Polymorphism occurs when different objects respond to the same command or input in different ways.

Abstraction is the ability to suppress superfluous details so that the important properties can be examined.

Object-oriented design (OOD) is a representation of a real-world problem.

Object-oriented analysis (OOA) is the process of categorizing objects that will be appropriate for a solution.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Polymorphism

---

Pregunta #56 de 67

Id. de pregunta: 1111819

¿Qué software malicioso se basa en otras aplicaciones para ejecutar e infectar el sistema?

✗ **A)** una bomba lógica
✓ **B)** un virus
✓ **C)** un caballo de Troya
✗ **D)** un gusano

Explanation

Both virus and Trojan horse are correct.

A virus is malicious software (malware) that relies upon other application programs to execute and infect a system. The main criterion for classifying a piece of executable code as a virus is that it spreads itself by means of hosts. The hosts could be any application on the system. A virus infects a system by replicating itself through application hosts. The different types of viruses are as follows:

- Stealth virus: It hides the changes it makes as it replicates.
- Self-garbling virus: It formats its own code to prevent antivirus software from detecting it.
- Polymorphic virus: It can produce multiple operational copies of itself.
- Multipart virus: It can infect system files and boot sectors of a computer system.
- Macro virus: It generally infects the system by attaching itself to MS-Office applications.
- Boot sector virus: It infects the master boot record of the system and is spread via infected floppy disks.
- Compression virus: It decompresses itself upon execution but otherwise resides normally in a system.

A Trojan horse is malware that is disguised as a useful utility, but embeds malicious codes within itself. When the disguised utility is run, the Trojan horse performs malicious activities in the background, such as deleting system files

and planting a backdoor into a system and provides a useful utility at the front end. Trojan horses use covert channels to perform malicious activities.

The standard security best practices for mitigating risks from malicious programs, such as viruses, worms and Trojans, include implementation of antivirus software, use of host-based intrusion detection system, and imposition of limits on the sharing and execution of programs.

A worm does not require the support of application programs to be executed; it is a self-contained program capable of executing and replicating on its own without the help of system applications or resources. Typically, a worm is spread by e-mails, transmission control protocols (TCPs), and disk drives.

A logic bomb malware is similar to a time bomb that is executed at a specific time on a specific date. A logic bomb implies a dormant program that is triggered following a specific action by the user or after a certain interval of time. The primary difference between logic bombs, viruses, and worms is that a logic bomb is triggered when specific conditions are met.

A Trojan horse is malware that is disguised as a useful utility, but embeds malicious codes within itself. When the disguised utility is run, the Trojan horse performs malicious activities in the background, such as deleting system files and planting a backdoor into a system and provides a useful utility at the front end. Trojan horses use covert channels to perform malicious activities.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Identity and Access Management, Malicious Software

---

Pregunta #57 de 67

Id. de pregunta: 1105618

Recientemente, el servidor de archivos de su empresa fue víctima de un ataque de piratas informáticos. Después de investigar el ataque, descubre que se utilizaron varios equipos para implementar el ataque, lo que finalmente provocó que el servidor de archivos se sobrecargase. ¿Qué ataque ocurrió?

✗ **A)** ataque a tierra

✗ **B)** ataque de denegación de servicio (DoS)

✗ **C)** ping de ataque a muerte

✓ **D)** ataque de denegación de servicio distribuido (DDoS)

Explanation

A distributed denial-of-service (DDoS) attack occurred. A DDoS attack is an extension of the denial-of-service (DoS) attack. In DDoS, the attacker uses multiple computers to target a critical server and deny access to the legitimate users.

The primary components of a DDoS attack are the client, the masters or handlers, the slaves, and the target system. The initial phase of the DDoS attack involves using numerous computers and planting backdoors that are controlled by master controllers and referred to as slaves. Handlers are the systems that instruct the slaves to launch an attack against a target host. Slaves are typically systems that have been compromised through backdoors, such as Trojans, and are not aware of their participation in the attack. Masters or handlers are systems on which the attacker has been able to gain administrative access.

The primary problem with DDoS is that it is an issue with the availability of critical resources instead of an issue with confidentiality and integrity. Therefore, it is difficult to address using security technologies, such as SSL and PKI.

Launching a traditional DoS attack might not disrupt a critical server operation. Launching a DDoS attack can bring down the critical server because the server is being overwhelmed with the processing of multiple requests until it ceases to be functional. Stacheldraht, trinoo, and tribal flow network (TFN) are examples of DDoS tools.

A land attack involves sending a spoofed TCP SYN packet with the target host's IP address and an open port as both the source and the destination to the target host on an open port. The land attack causes the system to either freeze or crash because the computer continuously replies to itself.

A ping of death is another type of DoS attack that involves flooding target computers with oversized packets, exceeding the acceptable size during the process of reassembly, and causing the target computer to either freeze or crash. Other denial of service attacks named, smurf and fraggle, deny access to legitimate users by causing a system to either freeze or crash.

A DoS attack is an attack on a computer system or network that causes loss of service to users. The DoS attack floods the target system with unwanted requests. It causes the loss of network connectivity and services by consuming the bandwidth of the target network or overloading the computational resources of the target system. The primary difference between DoS and DDoS is that in DoS, a particular port or service is targeted by a single system and in DDoS, the same process is accomplished by multiple computers.

There are other types of denial of service attacks such as buffer overflows, where a process attempts to store more data in a buffer than amount of memory allocated for it, causing the system to freeze or crash.

Objective:
Software Development Security

Sub-Objective:
Assess security impact of acquired software

References:

CISSP Cert Guide (3rd Edition), Chapter 4: Communication and Network Security, DDoS

Pregunta #58 de 67

Id. de pregunta: 1105602

Ha implementado una nueva red para un cliente. La administración ha solicitado que implemente software antivirus que sea capaz de detectar todo tipo de código malicioso, incluido el malware desconocido.

¿Qué tipo de software antivirus debe implementar?

- ✗ **A)** detección basada en firmas
- ✓ **B)** detección heurística
- ✗ **C)** inmunización
- ✗ **D)** bloqueo de comportamiento

Explanation

You should implement heuristic detection anti-virus software. This type of anti-virus software is capable of detecting all types of malicious code, including unknown malware.

A signature-based detection anti-virus software detects viruses based on the virus signatures located in its database. If the virus signature is not in the database, the virus will not be detected, meaning this type of anti-virus software cannot detect unknown malware.

An immunization antivirus software attaches code to files or applications to make it appear as if the file or application was already infected. An immunizer is virus-specific, meaning that a new immunizer is needed for every virus. This method is considered obsolete.

A behavior-blocking anti-virus software examines what is occurring on a system, looking for suspicious activity. If a potential malware is detected, the software is terminated.

Heuristic detection and behavior blocking are considered proactive. Signature-based detection cannot detect new malware.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Anti-malware Software

Pregunta #59 de 67

Id. de pregunta: 1114018

Debe ver los eventos en los registros de nombres de host. ¿Qué registro en el Visor de eventos debe ver?

- ✓ **A)** DNS
- ✗ **B)** Aplicación
- ✗ **C)** Sistema
- ✗ **D)** Seguridad

Explanation

You should use the DNS log in Event Viewer to view events on host name registrations. You should log DNS entries so that you can watch for unauthorized DNS clients or servers. Without a DNS log, you would be unable to discover how long an entry was being used.

None of the other logs will contain this type of information. The Application log contains events logged by applications. The Security log contains events based on the auditing configuration. Only administrators can configure and view auditing. The System log contains events logged by computer system components.

Auditing deters perpetrators' attempts to bypass the system protection mechanisms, reviews patterns of access to individual objects, and discovers when a user assumes a functionality with privileges greater than his own.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

Troubleshooting DNS, https://technet.microsoft.com/en-us/library/bb962024.aspx

---

Pregunta #60 de 67

Id. de pregunta: 1105601

Su organización tiene varios quioscos de computadoras sin disco que se inician a través de medios ópticos ubicados en el vestíbulo de la oficina. Recientemente, los usuarios informaron que los equipos sin disco han sido infectados con un virus. ¿Qué debe hacer para asegurarse de que se elimina el virus?

- ✓ **A)** Reinicie los equipos sin disco.

✗ **B)** Inicie un programa antivirus en los ordenadores sin disco a través de una unidad flash USB.

✗ **C)** Inicie de forma remota un programa antivirus en los equipos sin disco.

✗ **D)** Reinicie el servidor al que se conectan los equipos sin disco.

Explanation

To ensure that a virus is removed from a diskless computer, you should simply reboot the computer. The virus will only exist in the computer's memory because the computer does not have a hard drive or full operating system.

None of the other options is correct. The easiest way to remove a virus from a diskless computer is to reboot the computer.

Objective:
Software Development Security

Sub-Objective:
Assess the effectiveness of software security

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Anti-malware Software

---

Pregunta #61 de 67

Id. de pregunta: 1105574

¿Qué complemento de explorador web usa Authenticode por motivos de seguridad?

✗ **A)** Interfaz de puerta de enlace común (CGI)

✗ **B)** Secuencias de comandos entre sitios (XSS)

✗ **C)** Java

✓ **D)** ActiveX

Explanation

ActiveX uses Authenticode for security. Authenticode is a certificate technology that allows ActiveX components to be validated by a server. Users need to be careful when confirming the installation of ActiveX components or controls. Automatically accepting an ActiveX component or control creates an opportunity for security breaches.

None of the other options uses Authenticode for security.

Cross-site scripting (XSS) is a type of security vulnerability typically found in Web applications that allows code injection by hackers into the Web pages viewed by other users. It is used to trick a user into visiting a site and having code

execute locally.

Java is a self-contained script that is downloaded from a server to a client and run within a Web browser.

CGI is a scripting method that was used extensively in older Web servers. CGI scripts captured data from users using simple forms.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, ActiveX

4.3 ActiveX and Authenticode, http://safari.informit.com/1565922697/ch04-37804

Pregunta #62 de 67

Id. de pregunta: 1111806

Su organización tiene una base de datos en clúster tolerante a errores que mantiene registros de ventas. ¿Qué técnica transaccional se utiliza en este entorno?

- ✓ **A)** OLTP
- ✗ **B)** almacenamiento de datos
- ✗ **C)** ODBC
- ✗ **D)** OLE DB

Explanation

Online transaction processing (OLTP) is used in this environment. OLTP is a transactional technique used when a fault-tolerant, clustered database exists. OLTP balances transactional requests and distributes them among the different servers based on transaction load. OLTP uses a two-phase commit to ensure that all the databases in the cluster contain the same data.

Object Linking and Embedding Database (OLE DB) is a method of linking data from different databases together.

Open Database Connectivity (ODBC) is an application programming interface (API) that can be configured to allow any application to query databases.

Data warehousing is a technique whereby data from several databases is combined into a large database for retrieval and analysis.

Security requirements are considered a part of software risk analysis during the project initiation phase of the SDLC. The SDLC identifies the relevant threats and vulnerabilities based on the environment in which the product will perform data processing, the sensitivity of the data required, and the countermeasures that should be a part of the product. It is important that the SDLC methodology be adequate to meet the requirements of the business and the users.

The SDLC includes the following phases:

- Plan/Initiate Project
- Gather Requirements
- Design
- Develop
- Test/Validate
- Release/Maintain
- Certify/Accredit
- Change Management and Configuration Management/Replacement

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 5: Asset Security, OLTP ACID Test

---

Pregunta #63 de 67

Id. de pregunta: 1105619

Su empresa decide que debe comprar un nuevo producto de software para ayudar al personal de marketing a administrar sus campañas y recursos de marketing. ¿Durante qué fase del proceso de adquisición de software se implementa realmente el producto?

✓ **A)** Fase de supervisión
✗ **B)** Fase de contratación
✗ **C)** Fase de mantenimiento
✗ **D)** Fase de planificación

Explanation

During the monitoring phase, the product is actually deployed. You should ensure that the supplier completes the contract and that you formally accept the final product.

During the planning phase, the software requirements are documented. You should also create an acquisition strategy during this phase and develop the evaluation criteria.

During the contracting phase, you should issue the request for proposal (RFP), evaluate the proposals, and complete final contract negotiations with the selected seller.

In the maintaining phase, you should maintain the software, including possibly decommissioning the software at some future date.

Objective:
Software Development Security

Sub-Objective:
Assess security impact of acquired software

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Security Impact of Acquired Software

---

Pregunta #64 de 67

Id. de pregunta: 1111808

¿Cuál es la función principal de COCOMO?

- ✗ **A)** análisis de amenazas
- ✓ **B)** estimación de costos
- ✗ **C)** estimación del riesgo
- ✗ **D)** estimación del tiempo

Explanation

The primary function of the Construction Cost Model (COCOMO) is cost estimation. The basic version of COCOMO estimates software development effort and cost as a function of the size of the software product in source instructions.

COCOMO is all about estimating the costs associated with software development. It does not primarily provide time estimation, risk estimation, or threat analysis.

The SDLC includes the following phases:

- Plan/Initiate Project
- Gather Requirements
- Design
- Develop

- Test/Validate
- Release/Maintain
- Certify/Accredit
- Change Management and Configuration Management/Replacement

The system design specification phase of the SDLC focuses on providing details on which kind of security mechanism will be a part of the software product. The system design specification phase also conducts a detailed design review and develops a plan for validation, verification, and testing. The organization developing the application will review the product specifications together with the customer to ensure that the security requirements are clearly stated and understood, and that the planned functionality features are embedded in the product. Involving security analysts at this phase ensures maximum benefit to the organization. It also enables you to understand the security requirements and features of the product and to report existing loopholes.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

Overview of COCOMO, http://www.softstarsystems.com/overview.htm

---

Pregunta #65 de 67

Id. de pregunta: 1105572

¿Qué tipo de canal se utiliza cuando un proceso escribe datos en un disco duro y otro proceso los lee?

✗ **A)** canal de temporización encubierto
✗ **B)** canal de almacenamiento externo
✗ **C)** canal de temporización no coincidente
✓ **D)** canal de almacenamiento encubierto

Explanation

A covert storage channel is used when one process writes data to a hard drive and another process reads it. In a covert storage attack, a higher-level subject writes data to a storage area and a lower-level subject reads it.

A covert timing channel is used when a process transmits data to another process.

An overt channel was developed for communication. Processes should use overt channels, not covert channels. Overt channels are not divided into categories, such as timing or storage channels.

Objective:
Software Development Security

Sub-Objective:
Identify and apply security controls in development environments

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Covert Channel

---

Pregunta #66 de 67

Id. de pregunta: 1114011

Está desarrollando una nueva aplicación de software para un cliente. El cliente está definiendo actualmente los requisitos de la aplicación. ¿Qué proceso se está completando?

- ✓ **A)** prototipado
- ✗ **B)** muestreo
- ✗ **C)** abstracción
- ✗ **D)** interpretación

Explanation

A prototype or a blueprint of the product is developed on the basis of customer requirements. Prototyping is the process of putting together a working model, referred to as a prototype, to test various aspects of a software design, to illustrate ideas or features, and to gather feedback in accordance with customer requirements. A prototype enables the development team and the customer to move in the right direction.

Prototyping can provide significant time and cost savings because it will involve fewer changes later in the development stage. A product is developed in modules. Therefore, prototyping provides scalability. Complex applications can be further subdivided into multiple parts and represented by different prototypes. The software design and development tasks can be assigned to multiple teams.

A sample is a generic term that identifies a portion that is a representative of a whole. Interpreters are used to execute the program codes by translating one command at a time. Abstraction is an object-oriented programming (OOP) concept that refers to hiding unnecessary information to highlight important information or properties for analysis. Abstraction involves focusing on conceptual aspects and properties of an application to understand the information flow. Abstraction involves hiding small, redundant pieces of information to provide a broader picture.

Interpreters are used to execute the program codes by translating one command at a time.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

CISSP Cert Guide (3rd Edition), Chapter 8: Software Development Security, Prototyping

---

Pregunta #67 de 67

Id. de pregunta: 1105564

¿Qué instrucción define correctamente el intercambio dinámico de datos (DDE)?

- ✓ **A)** DDE permite que varias aplicaciones compartan e intercambien el mismo conjunto de datos.
- ✗ **B)** DDE es una interfaz de software que permite la comunicación entre una aplicación y una base de datos.
- ✗ **C)** DDE es una interfaz para vincular información entre varias bases de datos.
- ✗ **D)** DDE es una técnica gráfica que se utiliza para realizar un seguimiento del progreso de un proyecto durante un período de tiempo.

Explanation

The dynamic data exchange (DDE) process enables direct communication between two applications using interprocess communications (IPC). Based on the client/server model, DDE allows two programs to exchange commands between themselves. The source of the data is referred to as the server, and the system accessing the data is referred to as the client. In the client/server model, the server is the data storage resource and is responsible for data backups and protection/maintenance of the database.

None of the other options defines DDE. OLE DB is an interface to link information between various databases. A Gantt chart is a graphical technique that is used to track the progress of a project over a period of time.

Objective:
Software Development Security

Sub-Objective:
Understand and integrate security in the Software Development Life Cycle (SDLC)

References:

About Dynamic Data Exchange, http://msdn.microsoft.com/en-us/library/windows/desktop/ms648774(v=vs.85).aspx