|  |  |
| --- | --- |
| Names | IDs |
| Nour Amr | 20196059 |
| Ahmed Mohamed | 20196008 |
| Mazen Ayman | 20196069 |
| Hosni Khaled | 20196017 |
| Mohamed Khaled | 20196042 |

# MCQ Write the letter of the most correct answer

Part A: Lecture 1 - Cyber security Introduction:

1. From cybersecurity domain:

|  |  |
| --- | --- |
| 1. Network security | 1. Cloud security |
| 1. Application security | 1. All of the above |

1. The act of securing computers, programs data or networks from potential threats like undeliberate or deliberate data change, unauthorized access, or even destruction

|  |  |
| --- | --- |
| 1. Cyber security | 1. Software security |
| 1. Software quality | 1. Software maintenance |

1. From cyber security concepts – threat, vulnerability & risk

|  |  |
| --- | --- |
| 1. Risk = Threat + Vulnerability | 1. Risk = Threat - Vulnerability |
| 1. Risk = Threat x Vulnerability | 1. Risk = Threat / Vulnerability |

1. From cyber security concepts – CIA Triad, information should be reality available for the authorized users

|  |  |
| --- | --- |
| 1. Confidentiality | 1. Integrity |
| 1. Availability | 1. All of the above |

1. In the CIA Triad, which one of the following is not involved?

|  |  |
| --- | --- |
| 1. Authenticity | 1. Confidentiality |
| 1. Availability | 1. Integrity |

1. From cyber security concepts – zero trust

|  |  |
| --- | --- |
| 1. Assume breach | 1. Never trust, always verify |
| 1. Implement least privilege | 1. All of the above |

1. From cyber security concepts – AAA, Which of the following not from the AAA

|  |  |
| --- | --- |
| 1. Authentication | 1. Authorization |
| 1. Accounting | 1. Availability |

Part 2: Lecture2 Secure SDLC:

1. ……… is a process that helps developers create software in a secure way from the initial planning stages to the final testing and release

|  |  |
| --- | --- |
| 1. Secure SDLC | 1. Test SDLC |
| 1. Integrate SDLC | 1. Maintain SDLC |

1. What are the key security risks within the application

|  |  |
| --- | --- |
| 1. Type of information application is processing | 1. Functionality |
| 1. Use case modelling | 1. All of the above |

1. From secure SDLC – Security architecture which tier that is not commonly used in industry

|  |  |
| --- | --- |
| 1. 1-tier | 1. 2-tier |
| 1. 3 tier | 1. None of the above |

1. From secure SDLC – Security architecture which tier has the least security architecture

|  |  |
| --- | --- |
| 1. 1-tier | 1. 2-tier |
| 1. 3 tier | 1. None of the above |

1. From secure SDLC – Security architecture which tier that is commonly used in desktop application

|  |  |
| --- | --- |
| 1. 1-Tier | 1. 2-Tier |
| 1. 3-Tier | 1. None of the above |

1. From secure SDLC – Security architecture which tier similar to basic client-server model

|  |  |
| --- | --- |
| 1. 1-Tier | 1. 2-Tier |
| 1. 3-Tier | 1. None of the above |

1. From secure SDLC – Security architecture the tier that has main security issue which is clients are accessing database directly

|  |  |
| --- | --- |
| 1. 1-Tier | 1. 2-Tier |
| 1. 3-Tier | 1. None of the above |

1. From secure SDLC – Security architecture which tier that prevent direct interaction of the client with the server by reducing access to unauthorized data

|  |  |
| --- | --- |
| 1. 1-Tier | 1. 2-Tier |
| 1. 3-Tier | 1. None of the above |

1. In which phase in secure SDLC we select technology to use (webserver , OS type, language type)

|  |  |
| --- | --- |
| 1. Security requirements | 1. Security architecture |
| 1. Security development | 1. Security design |

1. In which phase in secure SDLC we execute test plans created during design phase

|  |  |
| --- | --- |
| 1. Security requirements | 1. Security architecture |
| 1. Security testing | 1. Security design |

1. Which of the following considered in security deployment

|  |  |
| --- | --- |
| 1. Run security configuration review for application | 1. Periodic assessments |
| 1. Monitor your application through audit trails (Logs) | 1. All of the above |

1. Which of the following considered point/points of weakness in application assessments – SAST

|  |  |
| --- | --- |
| 1. Can not identify subjective or business logic related issues | 1. Extremely slow in adopting new versions of programming languages |
| 1. Requires less effort than dynamic analysis when dealing with tool results | 1. A and B |

1. Which of the following considered point/points of strength in application assessments – SAST

|  |  |
| --- | --- |
| 1. Cover all of application vulnerability testing | 1. Can be run in parallel with development to reduce overhead at the end of the life cycle |
| 1. Can be leveraged into checking for more sophisticated attacks by doing additional manual | 1. Slow in identify coding flaws |

1. Which of the following considered point/points of weakness in application assessments – DAST

|  |  |
| --- | --- |
| 1. Can not identify subjective or business logic related issues | 1. Extremely slow in adopting new versions of programming languages |
| 1. Depend heavily on qualifications of tester | 1. Requires more effort than dynamic analysis when dealing with tool results |

1. Which of the following considered point/points of strength in application assessments – DAST

|  |  |
| --- | --- |
| 1. Cover all of application vulnerability testing | 1. Can be leveraged into checking for more sophisticated attacks by doing additional manual |
| 1. Quick in identify coding flaws | 1. A and B |

1. Mis-configuration of the application vulnerabilities

|  |  |
| --- | --- |
| 1. Path traversal | 1. Web server misconfiguration HTTP verb tamper |
| 1. Weak password policy | 1. All of the above |

1. Data exposure of the application vulnerabilities

|  |  |
| --- | --- |
| 1. SSL not used | 1. Store data in clear text |
| 1. A and B | 1. Outdated PHP version |

1. From general application security guidelines: -

|  |  |
| --- | --- |
| 1. Run SAST only | 1. Run DAST only |
| 1. Run SAST and DAST | 1. None of the above |

Part 3: Lec3 Secure coding session:

1. How can a session be defined?

|  |  |
| --- | --- |
| 1. Run SAST only | 1. Run DAST only |
| 1. Run SAST and DAST | 1. None of the above |

1. Session hijacking refers to:

|  |  |
| --- | --- |
| 1. Exploiting a user's valid session | 1. Creating a new session for the user |
| 1. Terminating a user's session | 1. Encrypting a user's session data |

1. Which of the following is a recommended practice for avoiding phishing attacks?

|  |  |
| --- | --- |
| 1. Click on all links in an email to verify their authenticity | 1. Only click on links in an email that you have verified to have been sent from a legitimate sender |
| 1. Open email attachments without verifying the sender | 1. Disable email spam filters to receive all messages |

1. Which of the following is a type of Cross-Site Scripting (XSS) attack?

|  |  |
| --- | --- |
| 1. Server-Side Scripting (SSS) | 1. Distributed Denial of Service (DDoS) |
| 1. SQL Injection | 1. Reflected XSS |

1. Reflected XSS is a type of XSS attack that:

|  |  |
| --- | --- |
| 1. Injects client-side scripts into web pages viewed by other users | 1. Stores malicious scripts on the server for later execution |
| 1. Redirects users to a different website without their consent | 1. Exploits vulnerabilities in database management systems |

1. What is the most common protection mechanism against CSRF exploits?

|  |  |
| --- | --- |
| 1. Captcha verification | 1. Token-based protection |
| 1. Session management | 1. User authentication |

Part 4: Lecture1 intro and operators:

1. Symmetric ciphers are cryptographic techniques in which:

|  |  |
| --- | --- |
| 1. Encryption and decryption processes use different keys | 1. Encryption process uses a different key, but decryption process uses the same key |
| 1. Encryption and decryption processes use the same key | 1. Encryption and decryption processes do not require any keys |

1. Which of the following is one of the five ingredients of symmetric ciphers?

|  |  |
| --- | --- |
| 1. Digital Signature | 1. Public Key |
| 1. Plaintext | 1. Hash Function |

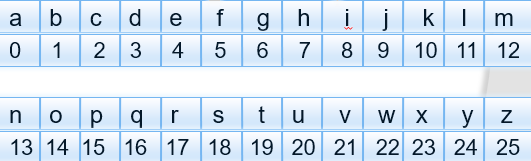
1. Which of the following techniques is used by symmetric ciphers to transform plaintext into ciphertext?

|  |  |
| --- | --- |
| 1. Hashing | 1. Substitution and/or transposition |
| 1. Compression | 1. Public key encryption |

1. Which of the following operators is NOT commonly used in encryption and decryption?

|  |  |
| --- | --- |
| 1. Mod(%) | 1. Square root operator (√) |
| 1. And(&) | 1. XOR(^) |

Part 4: Lecture 2 - Substitution Ceaser and Vernam:

1. Which of the following is the encrypted form of the word "hello" using the Caesar cipher with a key value of k = 3?

|  |  |
| --- | --- |
| 1. khoor | 1. hfnos |
| 1. kjppr | 1. gnkkq |

1. Which of the following is the decrypted form of the word "pxixe" using the Caesar cipher with a key value of k = 3 and the given cipher alphabet?

|  |  |
| --- | --- |
| 1. salsa | 1. salty |
| 1. saber | 1. salah |

1. The Vernam cipher is a substitution technique that operates on:

|  |  |
| --- | --- |
| 1. Letters | 1. Binary data (bits) |
| 1. symbols | 1. Numbers |

1. The Vernam cipher implements which operation on each bit of the plaintext?

|  |  |
| --- | --- |
| 1. Addition (+) | 1. Subtraction (-) |
| 1. Exclusive OR (XOR) (^) | 1. Numbers |

1. In the Vernam cipher, what must be the relationship between the key length and the plaintext length?

|  |  |
| --- | --- |
| 1. Key length must be greater than the plaintext length. | 1. Key length must be equal to the plaintext length. |
| 1. Key length must be less than the plaintext length. | 1. There is no relationship between the key length and the plaintext length. |

Part 5: Lecture 3 - Transposition & Rotor Machine:

1. From this figure using Rail fence algorithm the cipher text will be: -

|  |  |
| --- | --- |
| 1. gmiodonnorg | 1. gromoginodn |
| 1. gormoingnod | 1. gnrnoonigomd |

1. From this figure using Rail fence algorithm the cipher text will be: -

|  |  |
| --- | --- |
| 1. gomgoornniod | 1. gornogmodgoin |
| 1. gomornogidogn | 1. gomrigodonn |

1. From this figure using Rail fence algorithm, by decrypt the ciphertext, Then plaintext will be:

|  |  |
| --- | --- |
| 1. All conquers love. | 1. Conquers love all |
| 1. Love conquers all. | 1. all of the above |

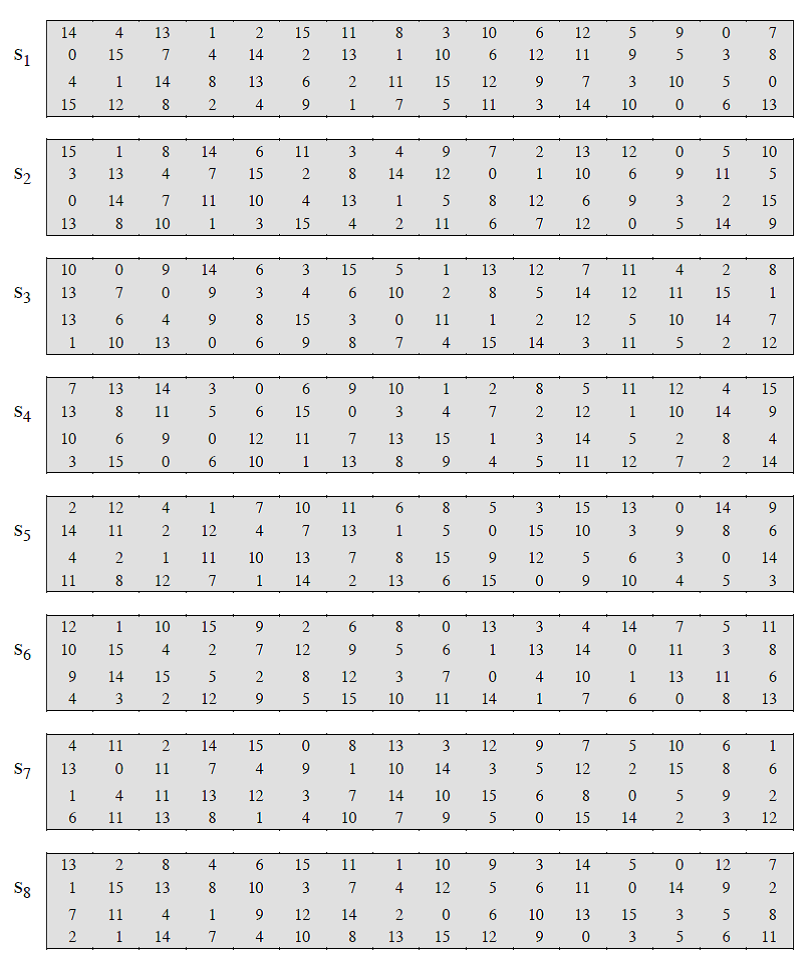
1. From this figure, using columnar Transposition algorithm, key= 4321 then the cipher text will be: -

|  |  |
| --- | --- |
| 1. irg aoy mmp fet | 1. fet mmp aoy irg |
| 1. aoy mmp fet irg | 1. mmp aoy fet irg |

1. The principle of multiple stages of encryption was primarily applied in a class of systems known as:

|  |  |
| --- | --- |
| 1. Caesar ciphers | 1. Rail Fence ciphers |
| 1. Rotor machines | 1. Vernam ciphers |

Part 5: Lecture 6 - DES Algorithm:

1. from the following table if the input =110111

|  |  |
| --- | --- |
| 1. 7 | 1. 14 |
| 1. 10 | 1. 3 |

1. When using a stream cipher, the two users share a symmetric key that is used for:

|  |  |
| --- | --- |
| 1. Encryption only | 1. Decryption only |
| 1. A and B | 1. Key generation |

1. In a block cipher, encryption and decryption are performed on:

|  |  |
| --- | --- |
| 1. Individual bits | 1. Individual bytes |
| 1. Individual characters | 1. Blocks of plaintext |

Part 6: Lecture 4&5 - Software engineering and secure code:

1. The main types of injection attacks that your application may be vulnerable to are:

|  |  |
| --- | --- |
| 1. XSS (Cross-Site Scripting) and CSRF (Cross-Site Request Forgery) | 1. XSS (Cross-Site Scripting) and SQL injection |
| 1. SQL injection and CSRF (Cross-Site Request Forgery) | 1. SQL injection and LDAP injection |

1. Which of the following statements is true regarding SQL injection attacks?

|  |  |
| --- | --- |
| 1. A successful SQL injection attack can lead to unauthorized access to sensitive data. | 1. SQL injection attacks are typically harmless and do not result in any significant consequences. |
| 1. SQL injection attacks only target server infrastructure and do not affect user data. | 1. SQL injection attacks are not a common threat and rarely result in data breaches. |

1. Which of the following options correctly represents the three main types of XSS attacks?

|  |  |
| --- | --- |
| 1. Reflected XSS, Stored XSS, Server-side XSS | 1. Reflected XSS, Stored XSS, DOM-based XSS |
| 1. Server-side XSS, Stored XSS, DOM-based XSS | 1. Reflected XSS, Client-side XSS, Stored XSS |

Part 7: Lecture 1,2,3 &4 – general questions:

1. ......... determines how easy it is to interface the system with another system.

|  |  |
| --- | --- |
| 1. Reusability | 1. Reliability |
| 1. Usability | 1. Interoperability |

1. ......... tests that the system does not offer opportunities to breach security.

|  |  |
| --- | --- |
| 1. initialization testing | 1. Security testing |
| 1. stress testing | 1. performance testing |

1. Complexity could be measure by two approaches:

|  |  |
| --- | --- |
| 1. Lines-of-code metric (LOC) | 1. McCabe’s cyclomatic-complexity metric |
| 1. A and B | 1. None of the above |

1. Which technique should be used for testing to avoid any problem?

|  |  |
| --- | --- |
| 1. Black box testing | 1. White box testing |
| 1. A and B | 1. None of the above |

1. ......... is essentially a contract: it defines all interfaces and the pre- and postconditions.

|  |  |
| --- | --- |
| 1. Service | 1. Provider |
| 1. Consumer | 1. registry |

1. ......... is the software entity which calls a service provider to request a service.

|  |  |
| --- | --- |
| 1. Service | 1. Provider |
| 1. Consumer | 1. registry |

1. ......... is the software entity that implements the service; it accepts and executes requests from consumers.

|  |  |
| --- | --- |
| 1. Service | 1. Provider |
| 1. Consumer | 1. registry |

1. .........is a software entity, which allows the lookup of services, service providers and their location – in other words it allows the service to be found.

|  |  |
| --- | --- |
| 1. Service | 1. Provider |
| 1. Consumer | 1. registry |

1. The root causes of most security vulnerabilities are .........

|  |  |
| --- | --- |
| 1. Poor Software Design and Engineering | 1. Few Requirements |
| 1. Low Testing | 1. Poor requirements |