**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 16 June 2020 | **Name:** | Anupama J S |
| **Course:** | Introduction to Cyber Security | **USN:** | 4AL16EC005 |
| **Topic:** | * What is cyber security and what is the motivation behind it? * Secure system design and security goals * Threats | **Semester & Section:** | 8th sem “A”section |
| **Github Repository:** | AnupamaJS |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **WHAT IS CYBER SECURITY AND WHAT IS THE MOTIVATION BEHIND IT?**  **C:\Users\User\Pictures\Screenshots\Screenshot (238).png**  Cyber security is the practice of defending computers, servers, mobile devices, electronic systems, networks, and data from malicious attacks. It's also known as information technology security or electronic information security. The term applies in a variety of contexts, from business to mobile computing, and can be divided into a few common categories.  ·         **Network security** is the practice of securing a computer network from intruders, whether targeted attackers or opportunistic malware.  ·         **Application security** focuses on keeping software and devices free of threats. A compromised application could provide access to the data its designed to protect. Successful security begins in the design stage, well before a program or device is deployed.  ·         **Information security** protects the integrity and privacy of data, both in storage and in transit.  ·         **Operational security** includes the processes and decisions for handling and protecting data assets. The permissions users have when accessing a network and the procedures that determine how and where data may be stored or shared all fall under this umbrella.  ·         **Disaster recovery and business continuity** define how an organization responds to a cyber-security incident or any other event that causes the loss of operations or data. Disaster recovery policies dictate how the organization restores its operations and information to return to the same operating capacity as before the event. Business continuity is the plan the organization falls back on while trying to operate without certain resources.  ·         **End-user education** addresses the most unpredictable cyber-security factor: people. Anyone can accidentally introduce a virus to an otherwise secure system by failing to follow good security practices. Teaching users to delete suspicious email attachments, not plug in unidentified USB drives, and various other important lessons is vital for the security of any organization. The most breached industries All businesses are at risk for a cyber-attack, but there are some industries that are more at risk than others for hacking. What makes these industries more vulnerable is the type of data that’s at risk of being stolen, including financial, health, and personal information. Healthcare 24% of all breaches occur in the healthcare industry, 79% of which is medical and personal data such as social security numbers, name and home addresses, information about income, and contact information. Stolen medical information can be used by hackers to gain unauthorized entry to some medical programs or to obtain prescription drugs for personal use or to sell for profit.  56% of the threat is internal, coming from within healthcare organizations, with another 34% being human error as employees leak information to other organizations without being aware of the infraction. Food Services and Accommodation Cyber-attacks in the food services and accommodation industries make up 15% of all breaches. These businesses are at high risk because they consistently collect information from their customers such as credit card numbers, name and address, and contact information. This stolen data can be used for identity theft and to gain access to financial accounts.  99% of the threat is external, with payment information accounting for 93% of the stolen data. Of equal concern is that in the food and accommodation industries,[96% of breaches aren’t discovered for a few months,](https://www.researchgate.net/publication/324455350_2018_Verizon_Data_Breach_Investigations_Report) at which time hackers have already used the information they stole. Public administration Close behind food and accommodation is public administration at 14% of industry breaches. With personal information amounting to 41% of the compromised data. Government data is at high risk for breaches due to a lack of cybersecurity funding, with[57% of government agencies hacked in 2018](https://dtr-gov.thalesesecurity.com/).  Both personal information and confidential government records are highly sought-after by cybercriminals at they can sell this data to foreign entities. Hackers who want to make a political statement are also a threat to public sector information.  **SECURE SYSTEM DESIGN AND SECURITY GOALS**  **C:\Users\User\Pictures\Screenshots\Screenshot (242).png** Security design goals We start by identifying the basic aspects that compose the security of an e-money system.  PROTOCOL SECURITY. By this we mean liveness and safety guarantees, namely, that the protocols achieve their goals and that every participant gets its information, and is secure in the sense that the other parties which are considered adversaries do not compromise or spoil the system. This aspect is the main focus of this paper.  INTERNAL SECURITY. The security of the internal operation system of the issuer of electronic currency, its capability to withstand insider attacks and abuses. The internal network architecture, operation policies, employment of tamper-proof hardware as well as dual control measures and access-control and physical access limitations should be reviewed. The internal security architecture has to be combined with issues such as availability, reliability, load balancing and back-up requirements.  NETWORK SECURITY. The security of the network (e.g., Internet) of users and the issuer, to prevent attacks not via the protocol but rather through ``break-ins;'' these attacks exploit the lack of proper protection into the system and software holes. Careful design of the interface to the external network (firewall protection) is required. Both the internal and the network systems have to be evaluated under ``Global Security Testing,'' which includes penetration attempts and security assessment of design and implementation.  USER SECURITY. Security of the user's assets. The user must obviously protect his electronic currency, and the software and procedures supplied to the user have to provide for protection at a proper level (e.g., beyond password-only protection), but at the same time be user-friendly.  In this paper, we deal specifically with the protocol aspects and their security. In this presentation we do not cover all the protocols, but what we cover seems to capture the basic needs of the system.  For simplicity, nor do we deal with the temporal aspects of maintaining the system, such as long-term key management and cryptographic policies.  **THREATS**  **C:\Users\User\Pictures\Screenshots\Screenshot (243).png** What are the latest cyber threats that individuals and organizations need to guard against? Here are some of the most recent cyber threats that the U.K., U.S., and Australian governments have reported on. Dridex malware In December 2019, the U.S. Department of Justice (DoJ) charged the leader of an organized cyber-criminal group for their part in a global [Dridex malware attack](https://www.ncsc.gov.uk/news/uk-and-us-investigation-into-harmful-international-cyber-campaigns). This malicious campaign affected the public, government, infrastructure and business worldwide.  Dridex is a financial trojan with a range of capabilities. Affecting victims since 2014, it infects computers though phishing emails or existing malware. Capable of stealing passwords, banking details and personal data which can be used in fraudulent transactions, it has caused massive financial losses amounting to hundreds of millions.  In response to the Dridex attacks, the U.K.’s National Cyber Security Centre advises the public to “ensure devices are patched, anti-virus is turned on and up to date and files are backed up”. Romance scams In February 2020, the FBI warned U.S. citizens to be aware of confidence fraud that cybercriminals commit using dating sites, chat rooms and apps. Perpetrators take advantage of people seeking new partners, duping victims into giving away personal data.  The [FBI reports](https://www.fbi.gov/contact-us/field-offices/albuquerque/news/press-releases/fbi-media-alert-avoid-becoming-a-victim-of-romance-scams) that romance cyber threats affected 114 victims in New Mexico in 2019, with financial losses amounting to $1.6 million. Emotet malware In late 2019, [The Australian Cyber Security Centre](https://www.cyber.gov.au/news/national-cyber-security-committee-urges-vigilance-two-concerning-cyber-security-threats-are-wild) warned national organizations about a widespread global cyber threat from Emotet malware.  [Emotet](https://techxplore.com/news/2020-02-sophisticated-emotet-malware-loader-unsophisticated.html) is a sophisticated trojan that can steal data and also load other malware. Emotet thrives on unsophisticated password: a reminder of the importance of creating a secure password to guard against cyber threats. |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 16 June 2020 | **Name:** | Anupama J S |
| **Course:** | Java Tutorial for Complete Beginners | **USN:** | 4AL16EC005 |
| **Topic:** | The java collections framework  1. ArrayList: Arrays the Easy Way  2. Linked Lists  3. HashMap: Retrieving Objects via a Key  4. Sorted Maps  5. Sets  6. Using Custom Objects in Sets and as Keys in Maps  7. Sorting Lists | **Semester & Section:** | 8th sem “A”section |
| **Github Repository:** | AnupamaJS |  |  |
| **AFTERNOON SESSION DETAILS** | | | |
| C:\Users\User\Pictures\Screenshots\Screenshot (234).pngC:\Users\User\Pictures\Screenshots\Screenshot (235).pngArray of ArrayList in Java We often come across 2D arrays where most of the part in the array is empty. Since space is a huge problem, we try different things to reduce the space. One such solution is to use [jagged array](https://www.geeksforgeeks.org/jagged-array-in-java/) when we know the length of each row in the array, but the problem arises when we do not specifically know the length of each of the rows. Here we use [ArrayList](https://www.geeksforgeeks.org/arraylist-in-java/) since the length is unknown. Following is a Java program to demonstrate the above concept. LinkedList in Java Linked List are linear data structures where the elements are not stored in contiguous locations and every element is a separate object with a data part and address part. The elements are linked using pointers and addresses. Each element is known as a node. Due to the dynamicity and ease of insertions and deletions, they are preferred over the arrays. It also has few disadvantages like the nodes cannot be accessed directly instead we need to start from the head and follow through the link to reach to a node we wish to access. To store the elements in a linked list we use a doubly linked list which provides a linear data structure and also used to inherit an abstract class and implement list and deque interfaces.  In Java, LinkedList class implements the [list interface](https://www.geeksforgeeks.org/list-interface-java-examples/). The LinkedList class also consists of various constructors and methods like other java collections.  **Constructors for Java LinkedList:**   1. LinkedList(): Used to create an empty linked list. 2. LinkedList(Collection C): Used to create a ordered list which contains all the elements of a specified collection, as returned by the collection’s iterator.  SortedMap Interface in Java with Examples SortedMap is an interface in [collection framework](https://www.geeksforgeeks.org/collections-in-java-2/). This interface extends [Map interface](https://www.geeksforgeeks.org/map-interface-java-examples/) and provides a total ordering of its elements (elements can be traversed in sorted order of keys). Exampled class that implements this interface is [TreeMap](https://www.geeksforgeeks.org/hashmap-treemap-java/).  sortedmap  The main characteristic of a SortedMap is that, it orders the keys by their natural ordering, or by a specified comparator. So consider using a [TreeMap](https://www.geeksforgeeks.org/hashmap-treemap-java/) when you want a map that satisfies the following criteria: Collections.sort() in Java with Examples **java.util.Collections.sort()** method is present in java.util.Collections class. It is used to sort the elements present in the specified [list](https://www.geeksforgeeks.org/list-interface-java-examples/) of Collection in ascending order. It works similar to [java.util.Arrays.sort()](https://www.geeksforgeeks.org/arrays-sort-in-java-with-examples/) method but it is better then as it can sort the elements of Array as well as linked list, queue and many more present in it.  public static void sort(List myList)  myList : A List type object we want to sort.  This method doesn't return anything Set in Java  * Set is an interface which extends Collection. It is an unordered collection of objects in which duplicate values cannot be stored. * Basically, Set is implemented by HashSet, LinkedHashSet or TreeSet (sorted representation). * Set has various methods to add, remove clear, size, etc to enhance the usage of this interface   filter\_none  edit  play\_arrow  brightness\_4   |  | | --- | | // Java code for adding elements in Set  import java.util.\*;  public class Set\_example  {      public static void main(String[] args)      {          // Set deonstration using HashSet          Set<String> hash\_Set = new HashSet<String>();          hash\_Set.add("Geeks");          hash\_Set.add("For");          hash\_Set.add("Geeks");          hash\_Set.add("Example");          hash\_Set.add("Set");          System.out.print("Set output without the duplicates");            System.out.println(hash\_Set);            // Set deonstration using TreeSet          System.out.print("Sorted Set after passing into TreeSet");          Set<String> tree\_Set = new TreeSet<String>(hash\_Set);          System.out.println(tree\_Set);      }  } |  Java Stream interface Java Stream interface provides two methods for sorting the list: sorted() method Stream interface provides a sorted() method to sort a list. It is defined in Stream interface which is present in **java.util package**. It returns a stream sorted according to the natural order. If the elements are not comparable, it throws java.lang.ClassCastException. The signature of the method is:   1. Stream<T> sorted()   **Parameter**  T: It is a type of stream element. Java Stream.sorted(Comparator comparator) It also returns a stream sorted according to the provided comparator. It is stable for an ordered stream. The signature of the method is:   1. Stream<T> sorted(Comparator<? Super T> comparator)   **Parameters**   * T is the type of stream element. * comparator to be used to compare elements   **Example**  In the following example, we have used the following methods:   * In Java 8, stream() is an API used to process collections of objects. * The collect() method is used to receive elements from a stream and stored them in a collection. * The toList() return the collector which collects all the input elements into a list, in encounter order. | | | |