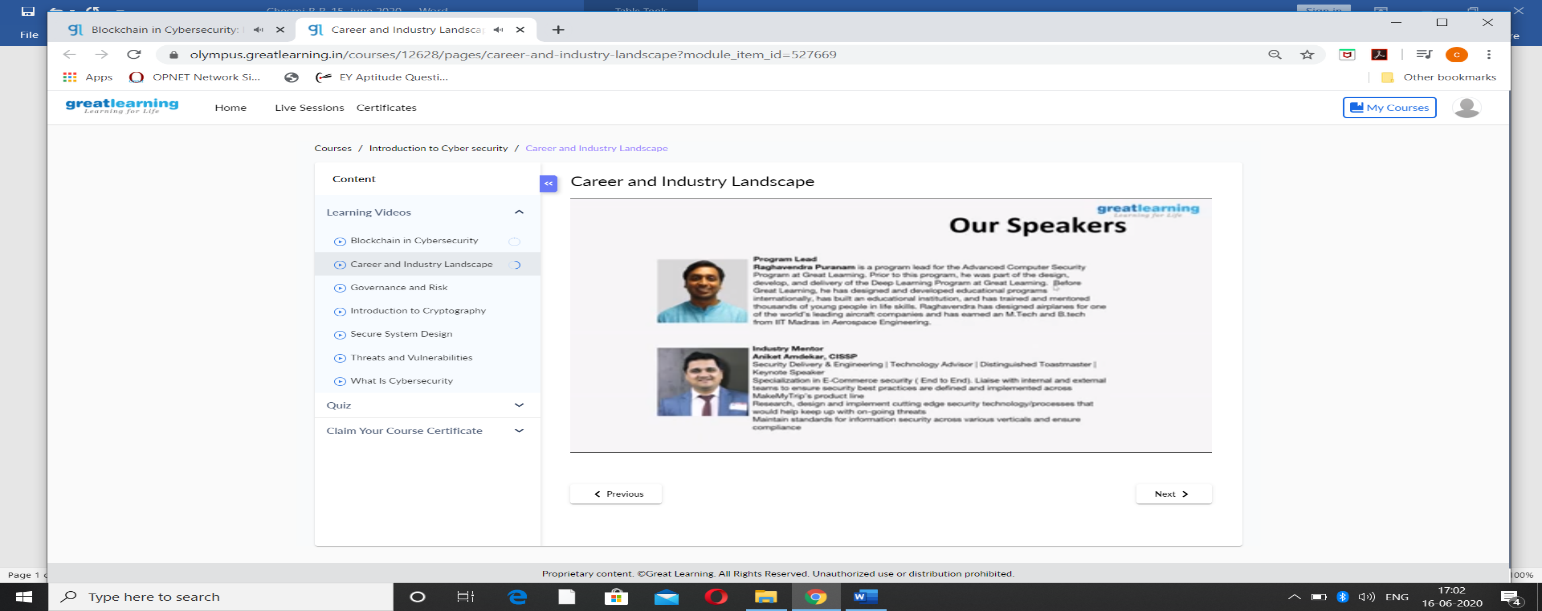
**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **17-06-2020** | **Name:** | **Rashmi KB** |
| **Course:** | **Cyber Security** | **USN:** | **4AL16EC056** |
| **Topic:** | **Blockchain in cyber security** | **Semester & Section:** | **8th sem B sec** |
| **Github Repository:** | **rashmikb** |  |  |

FORENOON SESSION DETAILS



****

**What is Cyber Security?**

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.

**Cryptography** is the study and practice of techniques for secure communication in the presence of third parties called adversaries. It deals with developing and analyzing protocols which prevents malicious third parties from retrieving information being shared between two entities thereby following the various aspects of information security.

Data Confidentiality, Data Integrity, Authentication and Non-repudiation are core principles of modern-day cryptography.

1. **Confidentiality** refers to certain rules and guidelines usually executed under confidentiality agreements which ensure that the information is restricted to certain people or places.
2. **Data integrity** refers to maintaining and making sure that the data stays accurate and consistent over its entire life cycle.
3. **Authentication** is the process of making sure that the piece of data being claimed by the user belongs to it.
4. **Non-repudiation** refers to ability to make sure that a person or a party associated with a contract or a communication cannot deny the authenticity of their signature over their document or the sending of a message.

### **Types of Cryptography**

There are four primary types of cryptography in use today, each with its own unique advantages and disadvantages.

They are called hashing, symmetric cryptography, asymmetric cryptography, and key exchange algorithms.

#### 1. Hashing

Hashing is a type of cryptography that changes a message into an unreadable string of text for the purpose of verifying the message’s contents, not hiding the message itself.

This type of cryptography is most commonly used to protect the transmission of software and large files where the publisher of the files or software offers them for download. The reason for this is that, while it is easy to calculate the hash, it is extremely difficult to find an initial input that will provide an exact match for the desired value.

For example, when you download Windows 10, you download the software which then runs the downloaded file through the same hashing algorithm. It then compares the resulting hash with the one provided by the publisher. If they both match, then the download is completed.

However, if there is even the slightest variation in the downloaded file (either through the corruption of the file or intentional intervention from a third party) it will drastically change the resulting hash, potentially nullifying the download.

Currently, the most common hashing algorithms are [MD5 and SHA-1](https://crypto.stackexchange.com/questions/18612/how-is-sha1-different-from-md5" \t "_blank), however due to these algorithm’s multiple weaknesses, most new applications are transitioning to the [SHA-256](http://www.xorbin.com/tools/sha256-hash-calculator" \t "_blank) algorithm instead of its weaker predecessors.

#### 2. Symmetric Cryptography

Symmetric Cryptography, likely the most traditional form of cryptography, is also the system with which you are probably most familiar.

This type of cryptography uses a single key to encrypt a message and then decrypt that message upon delivery.

Since symmetric cryptography requires that you have a secure channel for delivering the crypto key to the recipient, this type of cryptography is all but useless for transmitting data (after all, if you have a secure way to deliver the key, why not deliver the message in the same manner?).

As such, its primary application is the protection of resting data (e.g. Hard Drives and data bases)

In the Revolutionary War example that I mentioned earlier, Washington’s method for transmitting information between his officers would have relied on a symmetric cryptography system. He and all of his officers would have had to meet in a secure location, share the agreed upon key, and then encrypt and decrypt correspondence using that same key.

Most modern symmetric cryptography relies on a system known as AES or [Advanced Encryption Standards](https://thebestvpn.com/advanced-encryption-standard-aes/).

While the traditional DES models were the industry norm for many years, [DES was publicly attacked](https://www.sans.org/reading-room/whitepapers/vpns/day-des-died-722" \t "_blank) and broken in 1999 causing the National Institute of Standards and Technology to host a selection process for a stronger and more updated model.

After an arduous 5-year competition between 15 different ciphers, including MARS from IBM, RC6 from RSA Security, Serpent, Twofish, and Rijndael, the NIST selected [Rijndael as the winning cipher](http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf" \t "_blank).

It was then standardized across the country, earning the name AES or Advanced Encryption Standards. This cipher is still widely used today and is even implemented by the NSA for the purposes of guarding top secret information.

#### 3. Asymmetric Cryptography

Asymmetric cryptography (as the name suggests) uses two different keys for encryption and decryption, as opposed to the single key used in symmetric cryptography.

The first key is a public key used to encrypt a message, and the second is a private key which is used to decrypt them. The great part about this system is that only the private key can be used to decrypt encrypted messages sent from a public key.

While this type of cryptography is a bit more complicated, you are likely familiar with a number of its practical applications.

It is used when transmitting email files, remotely connecting to servers, and even digitally signing PDF files. Oh, and if you look in your browser and you notice a URL beginning with “https://”, that’s a prime example of asymmetric cryptography keeping your information safe.

#### 4. Key Exchange Algorithms

Although this particular type of cryptography isn’t particularly applicable for individuals outside of the cyber-security realm, I wanted to briefly mention to ensure you have a full understanding of the different cryptographic algorithms.

A key exchange algorithm, like Diffie-Hellman, is used to safely exchange encryption keys with an unknown party.

Unlike other forms of encryption, you are not sharing information during the key exchange. The end goal is to create an encryption key with another party that can later be used with the aforementioned forms of cryptography.

Here’s an example from the [Diffie-Hellman wiki](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange" \t "_blank) to explain exactly how this works.

Let’s say we have two people, Alice and Bob, who agree upon a random starting color. The color is public information and doesn’t need to be kept secret (but it does need to be different each time). Then Alice and Bob each selects a secret color that they do not share with anyone.

Now, Alice and Bob mix the secret color with the starting color, resulting in their new mixtures. They then publicly exchange their mixed colors.