**Week 3: Encryption, VPNs & Security Protocols**

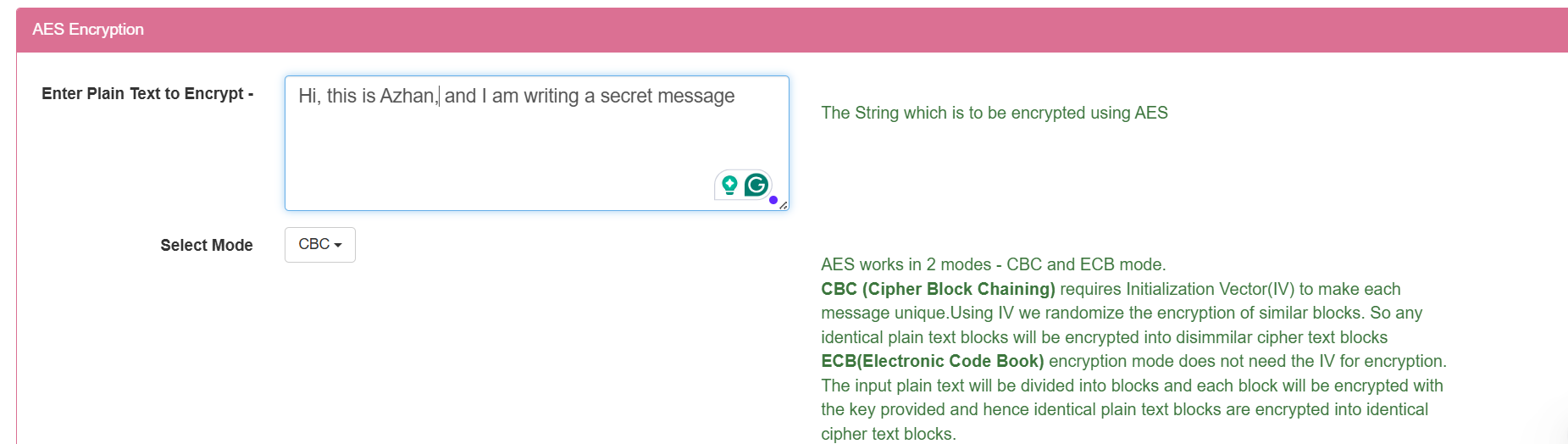
**Objective:** Study encryption standards and secure communication protocols.

**Task#05: Create and decode a basic encrypted message using online tools for AES and RSA.**

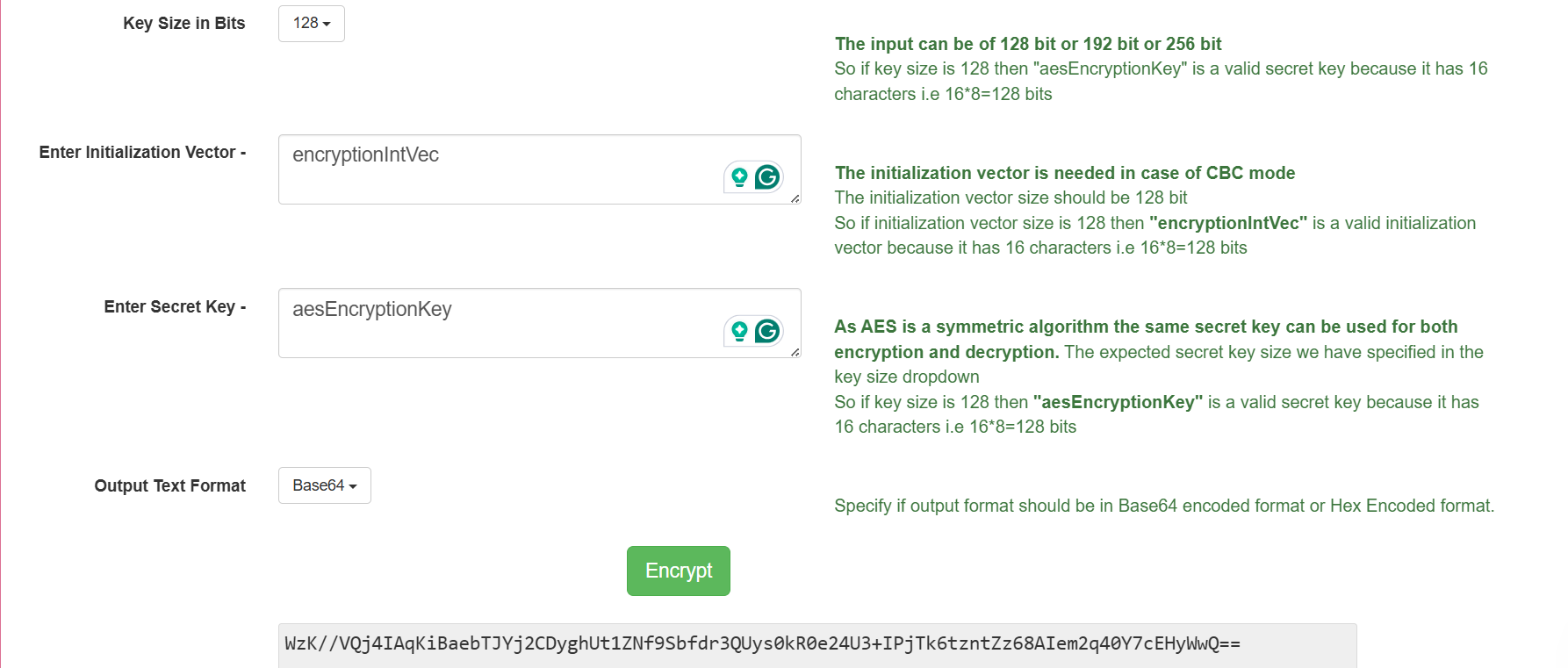
**Solution:**

**For AES Encryption:**

The first image displays an online **AES Encryption tool**, showing the plaintext message "Hi, this is Azhan and I am writing a secret message" using the **CBC (Cipher Block Chaining)** mode.

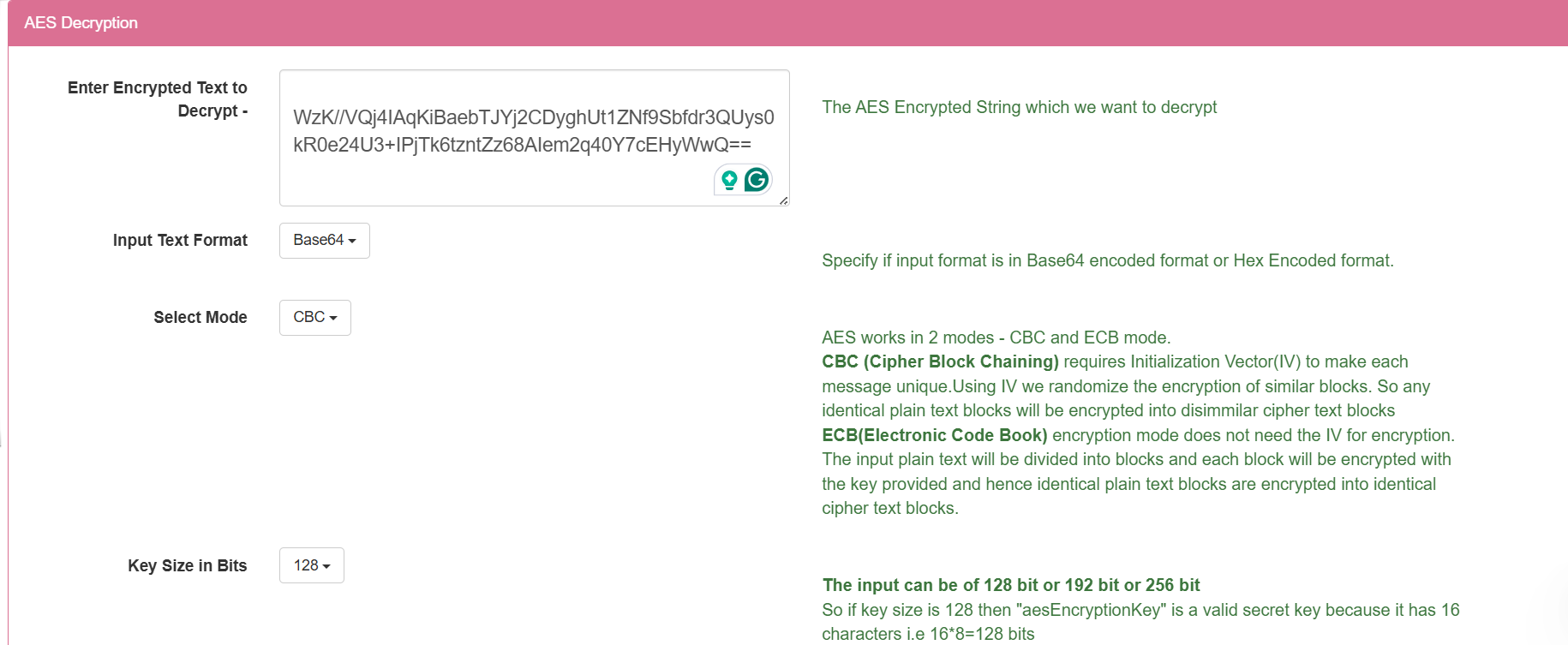


The Second image shows the successful completion of an **AES encryption process**, where a message has been encrypted using a 128-bit key in CBC mode, with "aesEncryptionKey" and "encryptionIntVec" as the secret key and initialization vector, respectively. The resulting ciphertext is displayed in Base64 format. In the end, the encrypted text has been provided.

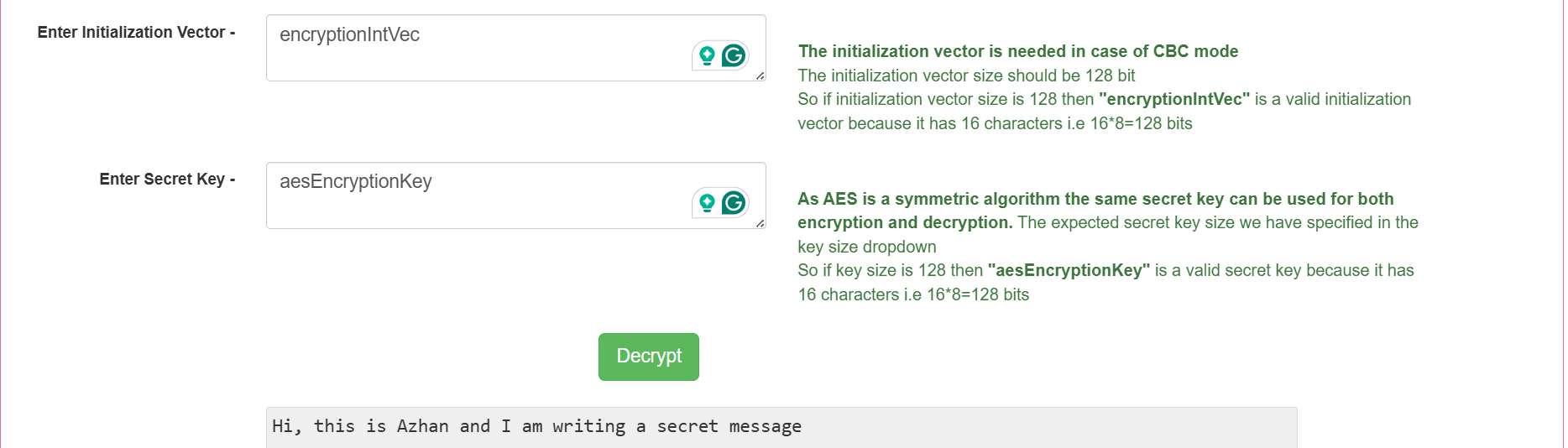


**For AES Decryption:**

The first image displays an online **AES Decryption tool**, where encrypted text (ciphertext) has been input in Base64 format, and the decryption process is configured to use **CBC (Cipher Block Chaining)** mode with a **128-bit key size**. This indicates preparation to decrypt a previously encrypted message.

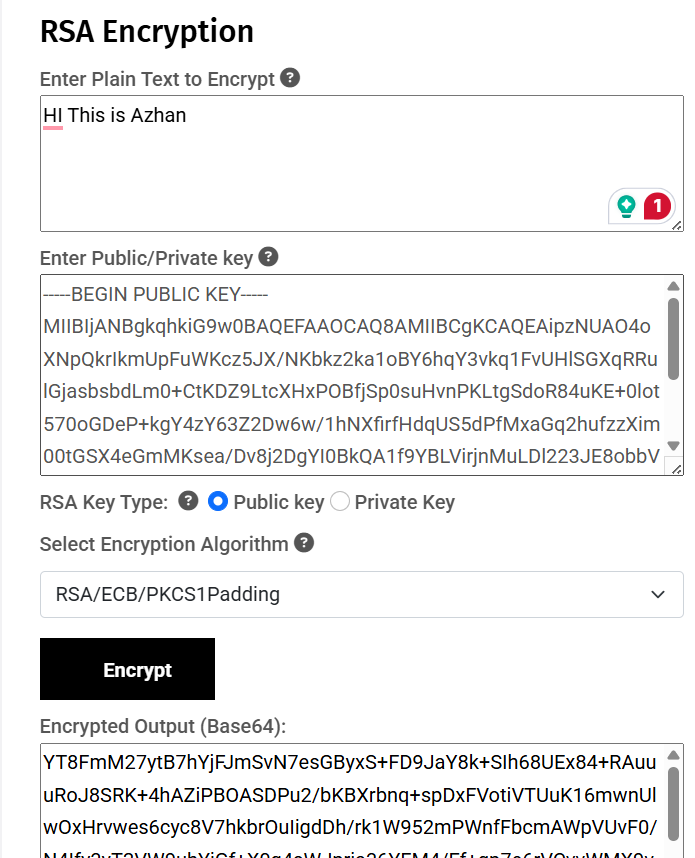


The second image demonstrates the **successful decryption of an AES-encrypted message**. It shows the online tool utilizing the correct Initialization Vector (encryptionIntVec) and Secret Key (aesEncryptionKey) to decrypt the ciphertext, revealing the original plaintext: "Hi, this is Azhan and I am writing a secret message."



**For RSA Encryption:**

I've popped in a simple message, "Hi! This is Azhan," into the "Plain Text to Encrypt" box. Then, I've loaded up a **Public Key** into the "Enter Public/Private key" section. You can tell it's the Public Key because the "RSA Key Type" radio button is clearly set to "Public Key". The tool is set to use the "RSA/ECB/PKCS1Padding" algorithm for the encryption. Below, in the "Encrypted Output (Base64)" area, there's a long string of jumbled characters. That's my "Hi! This is Azhan" message, but now it's all scrambled up and secure, shown in Base64 format. It's pretty cool to see how that public key just transformed my simple text into something unreadable!



**For RSA Decryption:**

So, I've moved on to the decryption part for my RSA message! In this screenshot, you can see I've taken that long, jumbled encrypted text (the Base64 output from before) and pasted it into the 'Enter Encrypted Text' box. Below that, I've put in my secret **Private Key** – you can tell it's the private one because that radio button is selected. The tool's using the same 'RSA/ECB/PKCS1Padding' algorithm as before. And then, boom! After hitting 'Decrypt,' the magic happened, and my original message, 'Hi This is Azhan,' popped right out in the 'Decrypted Output' section!

