Experiment 6

ELLIPTIC CURVE CRYPTOGRAPHY

3.1 Aim

To implement Elliptical curve cryptography algorithm

6.2 Algorithm

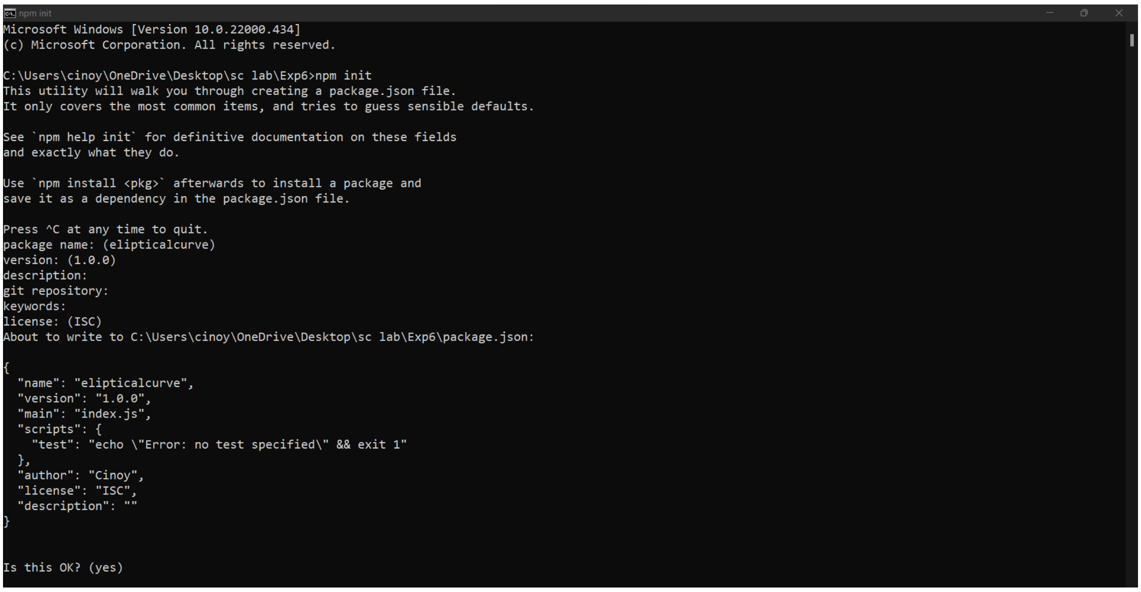
1. START
2. Create a Node.js project to work .
3. Create the ecc.js file.
4. Import the internal crypto module of Node.js .
5. Create an Elliptic Curve Diffie-Hellman (ECDH) key exchange object using a predefined curve, secp256k1 using createECDH function of the crypto module for Alice
6. Generate private and public EC Diffie-Hellman key values, and return the public key.
7. Perform the same step 5 and 6 to generate keys for Bob.
8. Compute the shared secret using otherPublicKey as the other party's public key and return the computed shared secret using the computeSecret function of the crypto module.
9. Log the Secret of Alice and Bob in the console.
10. Convert the Secret of Alice and Bob into Hex.
11. Log the converted Hex string in console
12. STOP

6.3 Getting set up in the NodeJs

Node.js is an open-source and cross-platform runtime environment built on Chrome's V8 JavaScript engine for executing JavaScript code outside of a browser.

Node.js supports a large number of third-party modules that help to perform many different kinds of tasks. Crypto module is one of the third-party modules that help encrypt or decrypt or hash any data ,which we want to secure from outside the world. The main function of this module is to convert the plain text or data to the encrypted format(hashed, CipherText) which is non-readable.

The crypto module provides cryptographic functionality that includes a set of wrappers for OpenSSL's hash, HMAC, cipher, decipher, sign, and verify functions.



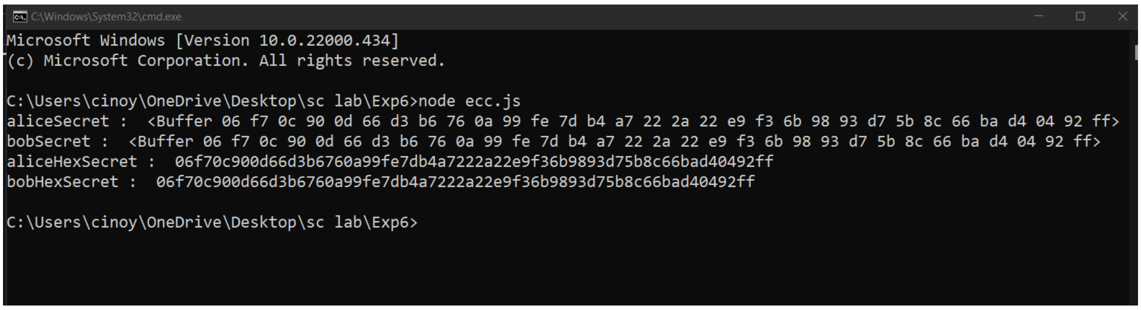
```
Microsoft Windows [Version 10.0.22000.434]
(c) Microsoft Corporation. All rights reserved.

C:\Users\cinoy\OneDrive\Desktop\sc lab\Exp6>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (elipticalcurve)
version: (1.0.0)
description:
git repository:
keywords:
license: (ISC)
About to write to C:\Users\cinoy\OneDrive\Desktop\sc lab\Exp6\package.json:

{
  "name": "elipticalcurve",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Cinoy",
  "license": "ISC",
  "description": ""
}


Is this OK? (yes)
```

Figure 1: Setting up NodeJs project

6.4 Program

```
// import crypto module
const createECDH = require(’crypto’);

// Generate Alice’s keys
const alice = createECDH(’secp256k1’);
const aliceKey = alice.generateKeys();

// Generate Bob’s keys
const bob = createECDH(’secp256k1’);
const bobKey = bob.generateKeys();

// Exchange and generate the secret
const aliceSecret = alice.computeSecret(bobKey);
console.log(”aliceSecret : ”, aliceSecret);

const bobSecret = bob.computeSecret(aliceKey);
console.log(”bobSecret : ”, bobSecret);

// convert secret into hex
let aliceHexSecret = aliceSecret.toString(”hex”);
console.log(”aliceHexSecret : ”, aliceHexSecret);

let bobHexSecret = bobSecret.toString(”hex”);
console.log(”bobHexSecret : ”, bobHexSecret);
```

6.5 Output



```
Microsoft Windows [Version 10.0.22000.434]
(c) Microsoft Corporation. All rights reserved.

C:\Users\cinoy\OneDrive\Desktop\sc lab\Exp6>node ecc.js
aliceSecret :  <Buffer 06 f7 0c 90 0d 66 d3 b6 76 0a 99 fe 7d b4 a7 22 2a 22 e9 f3 6b 98 93 d7 5b 8c 66 ba d4 04 92 ff>
bobSecret :  <Buffer 06 f7 0c 90 0d 66 d3 b6 76 0a 99 fe 7d b4 a7 22 2a 22 e9 f3 6b 98 93 d7 5b 8c 66 ba d4 04 92 ff>
aliceHexSecret :  06f70c900d66d3b6760a99fe7db4a7222a22e9f36b9893d75b8c66bad40492ff
bobHexSecret :  06f70c900d66d3b6760a99fe7db4a7222a22e9f36b9893d75b8c66bad40492ff

C:\Users\cinoy\OneDrive\Desktop\sc lab\Exp6>
```

Figure 2: Elliptical curve cryptography

6.6 Result

The Elliptical curve cryptography was implemented successfully.