

# INFORMATION SECURITY & PRIVACY

## TELECOM 2810 – PROJECT

### AIM:

To demonstrate Message digest and various crypto techniques

### TOOLS USED:

We are using Eclipse IDE for running java programs and the current version used is Eclipse Neon.

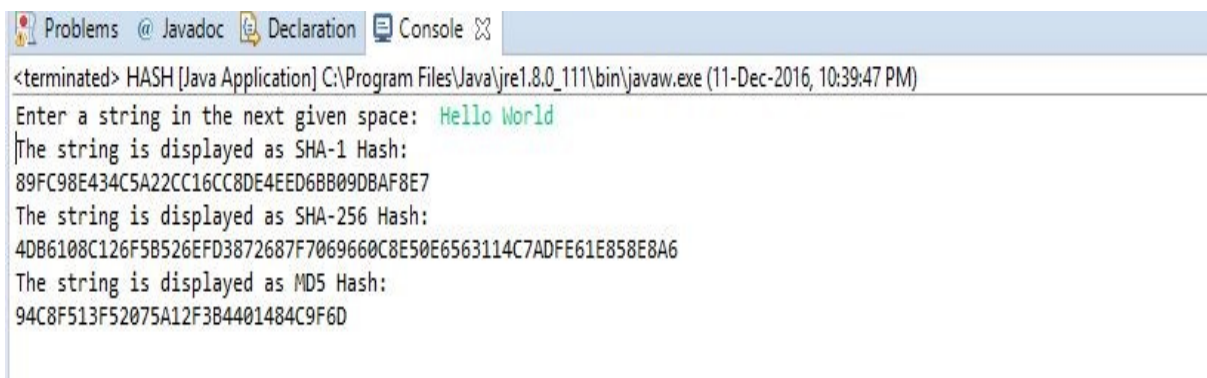
### MESSAGE DIGEST:

Message Digest is a hashing function used in cryptography. Here MD5 and SHA scheme is demonstrated. Hashing is generally used to preserve integrity of a data.

### PROCEDURE:

- 1) Create a new project called “HASHING”.
- 2) Create a new class called as “hash”.
- 3) Type the program and execute “hash.java”.

### OUTPUT:



```
<terminated> HASH [Java Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (11-Dec-2016, 10:39:47 PM)
Enter a string in the next given space: Hello World
The string is displayed as SHA-1 Hash:
89FC98E434C5A22CC16CC8DE4EED6BB09DBAF8E7
The string is displayed as SHA-256 Hash:
4DB6108C126F5B526EFD3872687F7069660C8E50E6563114C7ADFE61E858E8A6
The string is displayed as MD5 Hash:
94C8F513F52075A12F3B4401484C9F6D
```

## **VARIOUS CRYPTO TECHNIQUES:**

Crypto techniques like Authentication, Signature, Encryption, Public Key System and X.509 certificates are demonstrated here.

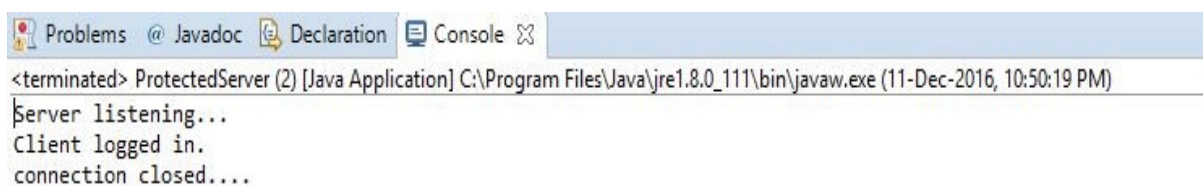
### **AUTHENTICATION:**

It involves implementing double strength password using message digest for login purpose. Here three classes are used, namely, ProtectedClient, ProtectedServer and Protection.

#### PROCEDURE:

- 1) Create a new project called “Authentication”.
- 2) Create a new class called as “ProtectedClient”.
- 3) Create a new class called as “ProtectedServer”.
- 4) Create a new class called as “Protection”.
- 5) Type the program and execute “ProtectedServer.java”.
- 6) Type the program and execute “ProtectedClient.java”.

#### OUTPUT:



```
<terminated> ProtectedServer (2) [Java Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (11-Dec-2016, 10:50:19 PM)
Server listening...
Client logged in.
connection closed....
```

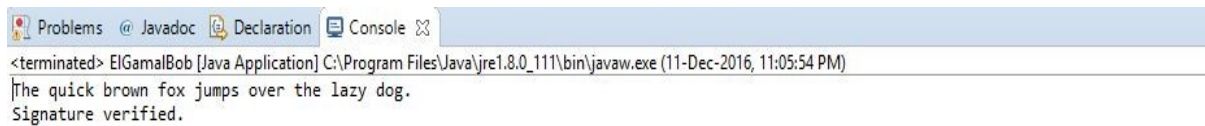
### **SIGNATURE:**

Elgamal signature forms basis on discrete algorithms. It allows a third party to confirm the authenticity of message over an insecure channel. Nowadays Elgamal has become archaic.

### PROCEDURE:

- 1) Create a new project called “Signature”.
- 2) Create a new class called as “ElGamalAlice”.
- 3) Create a new class called as “ElGamalBob”.
- 4) Type the program and execute “ElGamalAlice.java”.
- 5) Type the program and execute “ElGamalBob.java”.

### OUTPUT:



### **ENCRYPTION:**

CipherClient and CipherServer are the programs used here. The client generates a DES key and stores it. A object is encrypted using that key and it is sent to the server. Once the object is received, Server tries to decrypt it using the same key which was used by Client.

### PROCEDURE:

- 1) Create a new project called “Encryption”.
- 2) Create a new class called as “CipherServer”.
- 3) Create a new class called as “CipherClient”.
- 4) Type the program and execute “CipherServer.java”.
- 5) Type the program and execute “CipherClient.java”.

### OUTPUT:



## **PUBLIC KEY SYSTEM:**

RSA Public Key System is demonstrated here. We are considering that Alice and Bob are communicating and their messages need to be either confidential or integrity/authentication or both.

### PROCEDURE:

- 1) Create a new project called as “RSAPublicKeyAlice”.
- 2) Create a new project called as “RSAPublicKeyBob”.
- 3) Type the program and execute “RSAPublicKeyAlice.java”.
- 4) Type the program and execute “RSAPublicKeyBob.java”.

### OUTPUT:



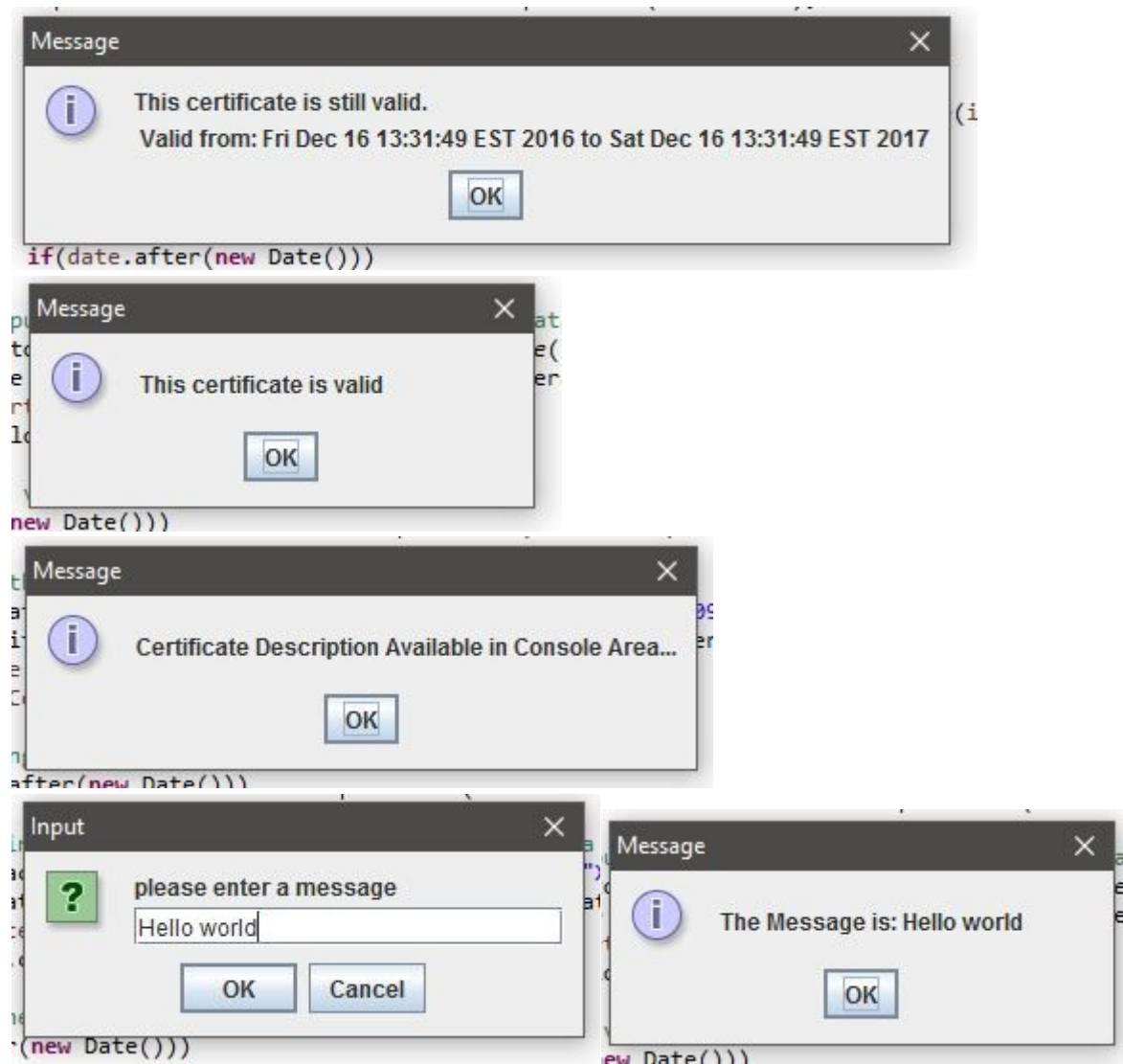
## **X.509 CERTIFICATE:**

X.509 certificate is used to identify whether the Key in a certificate belongs to that user or entity within that certificate. It contains information about version, serial number, algorithm, issuer distinguished name, validity period and extensions.

## PROCEDURE:

- 1) Create a new project called “X509certificates”.
- 2) Create a new class called as “ClientX509”.
- 3) Create a new class called as “ServerX509”.
- 4) Type the program and execute “ClientX509.java”.
- 5) Type the program and execute “ServerX509.java”.

## OUTPUT:



Version: V3

Subject: CN=Arun Kumar, OU=U Pitt, O=Ischool, L=Pittsburgh, ST=PA, C=US  
Signature Algorithm: SHA256withRSA, OID = 1.2.840.113549.1.1.11

Key: Sun RSA public key, 2048 bits

modulus:  
2037438406128444150373331191228252950124366012147610016051292252599663797840543961  
1683813862744942944855532547736925493763654737109405956883691920264893792442764758  
7856832970898086338475442472641646186221275873939068602043335847736331567652688148  
0941941916591502065234225951838110734112879625669427748385279703179814272382628327  
1286052396033196301888580989392627808592955968627083299786246154262570946954659680  
4131243270295784935848153109857526755766507343664528056620682084417868116169275632  
5610910193134549026821464995056794982623904101062008467822571390276978253366448965  
6811286661940215073954564951528527689287911

public exponent: 65537

Validity: [From: Fri Dec 16 13:31:49 EST 2016,  
To: Sat Dec 16 13:31:49 EST 2017]

Issuer: CN=Arun Kumar, OU=U Pitt, O=Ischool, L=Pittsburgh, ST=PA, C=US

SerialNumber: [ 48c3bcc0]

Certificate Extensions: 1

[1]: ObjectId: 2.5.29.14 Criticality=false

SubjectKeyIdentifier [

KeyIdentifier [

0000: 45 BE E2 3F 41 B2 5E 23 61 0B 93 12 36 8A D6 61 E...?A.^#a...6...a  
0010: 2C 6B 8B 12 ,k..

]

]

]

Algorithm: [SHA256withRSA]

Signature:

0000: 3A 5F 96 20 19 52 FA EF C5 B5 56 F5 6D 58 76 27 :\_. .R....V.mXv'  
0010: 7B F1 6F E4 94 92 4B 8F 75 B9 D4 FC E4 F4 40 91 ..o...K.u.....@.  
0020: 6D D0 A0 1A 00 D3 86 EC 3A B5 AC 6E 66 70 8D F1 m.....:..nfp..  
0030: DB 65 0B 88 E8 A6 AF C3 E6 D8 DB BD B5 C5 78 22 .e.....x"  
0040: 39 DA BA 57 C2 E9 17 2C B4 62 C1 80 45 E2 1A 2D 9..W...,.b..E.-  
0050: FA 17 A0 32 A4 E9 89 45 44 9D 3F 3C DD 1B 0F D9 ...2...ED.?<....  
0060: B0 B3 EF 5F 3F 81 CA FB 5F 00 9F EA DE 19 7D BD ...\_?...\_.....  
0070: 6F A7 F7 E6 26 12 33 70 FE 7B E4 18 93 A7 DA CD o...&.3p.....  
0080: 1F 59 F7 82 C5 C8 E8 EA 30 53 0C 87 A5 64 87 9A .Y.....0S...d..  
0090: 9D F0 D8 23 DD 01 04 51 8E 6B 51 81 33 E7 5B 00 ...#...Q.kQ.3.[.  
00A0: 68 D1 3D 5B 70 D4 45 1E AF FB AA E0 8C 2C 83 ED h.=[p.E.....,..  
00B0: 9D E0 4F 9F 4D 76 44 36 FF 51 B2 20 75 59 08 82 ..O.MvD6.Q. uY..  
00C0: 94 83 3C 2D 05 1E E5 96 60 65 78 46 F7 BD 9C 7D ..<-.....`exF....  
00D0: 43 6E 61 74 C3 77 3A B0 16 AC AF B7 1F AF 72 8E Cnat.w:.....r..  
00E0: 20 FF 3B DA 52 CA 7D 65 36 78 D2 44 24 FE 8F 8B .;.R...e6x.D\$...  
00F0: E9 38 83 AB B5 9C 5B F3 DD 97 93 1B 3E CA 0D FD .8....[.....>...

]