

# .Net Cryptography

Dr. Guillermo A Francia, III  
Jacksonville State University

## Cryptographic Provider

- An implementation of a cryptographic service consisting of classes.
- These .Net cryptographic classes are located in the `System.Security.Cryptography` namespace.

## Provider Classes

- Symmetric Cryptographic Provider
- Asymmetric Cryptographic Provider
- Hash Provider

## Symmetric Cryptography

- A single key is used to encrypt and decrypt data

## Asymmetric Cryptography

- Uses a pair of keys : a private key and a public key. These keys are mathematically related.

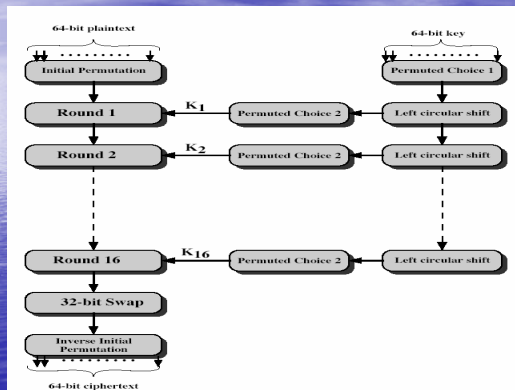
## Hash Function

- Also called one-way hash function or message digest.
- Creates a fixed-length array of bytes, which is random and unique, from a given data of any length.
- Used to digitally sign documents.

## Symmetric Cryptographic Algorithm Classes in .Net

- Data Encryption Standard (DES)
- Ron's Code 2 (RC2). Ron Rivest's variable key-size block cipher.
- Rijndael
- Triple DES

## Data Encryption Standard (DES)



IP: Initial Permutation

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

IP<sup>-1</sup>: Inverse Initial Permutation

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

## Triple DES

- Triple-DES is DES with two 56-bit keys applied:
  - a plaintext is DES encrypted with the first key.
  - the second key is used to DES decrypt the encrypted message
  - the twice-scrambled message is then encrypted again with the first key to yield the final ciphertext.

## Asymmetric Cryptographic Algorithm Classes in .Net

- Digital Signature Algorithm (DSA)
- Rivest-Shamir-Adleman (RSA) Algorithm

## Hash Function (Message Digest) Algorithm Classes in .Net

- Message Digest 5 (MD5)
  - Hash size is 128 bits
- Secure Hash Algorithm 1 (SHA1)
  - Hash size is 160 bits
- SHA256
  - Hash size is 256 bits
- SHA384
- SHA512

## File Encryption

Step 1: Create an instance of a provider

```
Dim desAlg As New DESCryptoServiceProvider
```

Step 2: Create a password derived key and initialization vector (IV)

```
Dim salt As Byte() = _  
    System.Text.Encoding.ASCII.GetBytes("Cs300")
```

```
Dim pdb As New Rfc2898DeriveBytes(pwd, salt)
```

```
Dim key() As Byte = pdb.GetBytes(8)
```

```
Dim IV() As Byte = pdb.GetBytes(8)
```

## File Encryption

### Step 3: Create an instance of a crypto transformer

```
Dim iTransf As System.Security.Cryptography.ICryptoTransform
```

### Step 4: Decide whether to encrypt or decrypt

```
'for encryption  
iTransf = desAlg.CreateEncryptor(key, IV)  
'for decryption  
iTransf = desAlg.CreateDecryptor(key, IV)
```

### Step 5: Create and instance of a cryptosystem

```
Dim cStream As New CryptoStream(fsOutput, iTransform, _  
    CryptoStreamMode.write)
```

## File Encryption

### Step 6: Iterate through the file

```
While (processedBytes < fileLength)  
    'read a 4096 byte data block to encrypt  
    bytesRead = fsInput.Read(bufferByte, 0, bufferLen)  
    'copy byte array into crypto stream  
    cStream.Write(bufferByte, 0, bytesRead)  
    'increment counter  
    processedBytes += bytesRead  
End While
```

### Step 7: Close all streams

```
cStream.Close()  
fsInput.Close()  
fsOutput.Close()
```

## Message Digest

### Step 1: Create an instance of a provider

```
Dim hashProvider As New MD5CryptoServiceProvider
```

### Step 2: Compute the hash value of the input file

```
Dim outDigest As Byte() = hashProvider.ComputeHash(fsInput)
```