Network Security Essentials

Applications and Standards
Third Edition

William Stallings

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Chapter 2

Symmetric Encryption and Message Confidentiality

By: Bhargavi H Goswami
Assistant Professor
Sunshine Group of Institutions
Rajkot, Gujarat, India.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Topic Outline

1. Symmetric Encryption Principles
2. Symmetric Block Encryption Algorithms
3. Stream Ciphers and RC4
4. Cipher Block Modes of Operation
5. Location of Encryption Devices
6. Key Distribution

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Symmetric Encryption Principles

• An encryption scheme has five ingredients:
  - Plaintext
  - Encryption algorithm
  - Secret key
  - Ciphertext
  - Decryption algorithm

• Security depends on the secrecy of the key, not the secrecy of the algorithm
Figure 2.1 Simplified Model of Symmetric Encryption
Cryptography

- Classified along three independent dimensions
  - The type of operations used for transforming plaintext to ciphertext
    - Substitution
    - Transposition
  - The number of keys used
    - Symmetric (single key)
    - Asymmetric (two-keys, or public-key encryption)
  - The way in which the plaintext is processed
    - Block cipher
    - Stream cipher
Table 2.1 Types of Attacks on Encrypted Messages

Email: bhargavigoswami@gmail.com, Mob: 9426669020

<table>
<thead>
<tr>
<th>Type of Attack</th>
<th>Known to Cryptanalyst</th>
</tr>
</thead>
</table>
| Ciphertext only| • Encryption algorithm  
                   • Ciphertext to be decoded                                                        |
| Known plaintext| • Encryption algorithm  
                   • Ciphertext to be decoded  
                   • One or more plaintext-ciphertext pairs formed with the secret key             |
| Chosen plaintext| • Encryption algorithm  
                   • Ciphertext to be decoded  
                   • Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key |
| Chosen ciphertext| • Encryption algorithm  
                   • Ciphertext to be decoded  
                   • Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key |
| Chosen text    | • Encryption algorithm  
                   • Ciphertext to be decoded  
                   • Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key  
                   • Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key |
### Table 2.2 Average Time Required for Exhaustive Key Search

<table>
<thead>
<tr>
<th>Key Size (bits)</th>
<th>Number of Alternative Keys</th>
<th>Time required at 1 encryption/μs</th>
<th>Time required at 10^6 encryptions/μs</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>$2^{32} = 4.3 \times 10^9$</td>
<td>$2^{31} \text{ μs} = 35.8$ minutes</td>
<td>2.15 milliseconds</td>
</tr>
<tr>
<td>56</td>
<td>$2^{56} = 7.2 \times 10^{16}$</td>
<td>$2^{55} \text{ μs} = 1142$ years</td>
<td>10.01 hours</td>
</tr>
<tr>
<td>128</td>
<td>$2^{128} = 3.4 \times 10^{38}$</td>
<td>$2^{127} \text{ μs} = 5.4 \times 10^{24}$ years</td>
<td>5.4 $\times 10^{18}$ years</td>
</tr>
<tr>
<td>168</td>
<td>$2^{168} = 3.7 \times 10^{50}$</td>
<td>$2^{167} \text{ μs} = 5.9 \times 10^{36}$ years</td>
<td>5.9 $\times 10^{30}$ years</td>
</tr>
<tr>
<td>26 characters (permutation)</td>
<td>$26! = 4 \times 10^{26}$</td>
<td>$2 \times 10^{26} \text{ μs} = 6.4 \times 10^{12}$ years</td>
<td>6.4 $\times 10^6$ years</td>
</tr>
</tbody>
</table>
Feistel Cipher Structure

• Virtually all conventional block encryption algorithms, including DES, have a structure first described by Horst Feistel of IBM in 1973.

• The realization of a Feistel Network depends on the choice of the following parameters and design features (see next slide):

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Feistel Cipher Structure

• **Block size**: larger block sizes mean greater security
• **Key Size**: larger key size means greater security
• **Number of rounds**: multiple rounds offer increasing security
• **Subkey generation algorithm**: greater complexity will lead to greater difficulty of cryptanalysis
• **Round function**: greater complexity means greater resistance of cryptanalysis
• **Fast software encryption/decryption**: the speed of execution of the algorithm becomes a concern
• **Ease of analysis**

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Figure 2.2 Classical Feistel Network
Symmetric Encryption Algorithms

• Processes the plaintext input in fixed-size blocks and produces a block of ciphertext of equal size for each plaintext block.

• Focuses on three symmetric block ciphers:
  1. Data Encryption Standard (DES)
  2. Triple DES (3DES)
  3. Advanced Encryption Standard (AES)

Email:bhargavigoswami@gmail.com, Mob: 9426669020
1. Data Encryption Standard (DES)

- The most widely used encryption scheme.
- NIST (National Institute of Standards & Technology), as FIPS PUB 46 (Federal Information Processing Standards).
- The algorithm is referred to the Data Encryption Algorithm (DEA).
- DES is a block cipher.
- Minor variation of Feistel.
- It has 16 rounds of processing.
- The plaintext is processed in 64-bit blocks.
- The key is 56 bits in length, which is divided into 16 subkeys, each one is used for each round.

Decryption:
- Use ciphertext as input to DES.
- Use subkeys $K_i$ in reverse order till $K$ is reached. i.e $K_{16}$ to $K_2, K_1$.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
• With key length of 56 bits, there are $2^{56} = 7.2 \times 10^{16}$ possible keys.
• Here, brute-force attack appears impractical. Why?
• Trying each key per microsecond would take more than 10 thousand years to break cipher.
• So? DES is secure?
• Yes or No?
• DES finally and definitively proved insecure in July 1998, when the Electronic Frontier Foundation (EFF98) announced that it had broken a DES encryption using a special-purpose “DES Cracker” machine that was built for less than $250,000.
• The attack took less than 3 days to break cipher.
• Problem?
• Solution?
• So a 128-bit key is guaranteed to result in an algorithm that is unbreakable by brute force or EFF98.
• 128-bit key would take over $10^{18}$ years to break the code using the EFF cracker.
• This is shown in next figure.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Figure 2.3 Time to Break a Code (assuming $10^6$ decryptions/μs)

Email:bhargavigoswami@gmail.com, Mob: 9426669020
2. Triple DES

- Triple DES (3DES) was first standardized for use in financial applications in ANSI standard X9.17 in 1985.
- 3DES was incorporated as part of the Data Encryption Standard in 1999, with the publication of FIPS PUB 46-3.
- Guidelines for 3DES:
  - 3DES is the FIPS approved symmetric encryption algorithm.
  - The original DES, which uses a single 56-bit key should support 3DES.
  - Government organizations with legacy DES systems are encouraged to transition to 3DES.
  - 3DES and the Advanced Encryption Standard (AES) will coexist as FIPS-approved algorithms, allowing for a gradual transition to AES.
Figure 2.4 Triple DES

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Encryption:

Decryption:

- $C = \text{ciphertext}$
- $P = \text{plaintext}$
- $E[K, X] = \text{encryption of } X \text{ using } K$
- $D[K, Y] = \text{decryption of } Y \text{ using } K$

Use three keys and three executions of the DES algorithm.

- Encrypt-Decrypt-Encrypt.
- Effective key length of 168 bits.
- FIPS 46-3 also allows for the use of two keys, with $K1 = K3$; this provides for a key length of 112 bits.
- Advantage: with 168bit key length brute force attacks are effectively impossible.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
• **Advantage/ Attraction of People:**
  - 168 bit key length which overcomes the vulnerability of brute force attack.
  - Same algorithm procedure as DES.
  - Very resistant to cryptanalysis.

• **Disadvantage:**
  - Algorithm is relatively sluggish in software, does not produce efficient software codes.
  - Slower due to 3 times more rounds than DES.
  - Both DES and 3DES use 64 bit block size, larger block size is desirable for efficiency and security.

• **Solution?**

• **AES.**

Email: bhargavigoswami@gmail.com, Mob: 9426669020
3. Advanced Encryption Standard (AES)


- Requirements:
  - Security better or equal to 3DES,
  - Improved efficiency.
  - Must be a symmetric block cipher with a block length of 128 bits and support for key lengths of 128, 192, and 256 bits.

- Evaluation criteria:
  - Include security,
  - Computational efficiency,
  - Memory requirements,
  - Hardware and software suitability, and
  - Flexibility.
• In first round, 15 proposals were accepted, out of which in 2\textsuperscript{nd} round 5 algorithms were shortlisted and out of them Rijndael was proposed as AES developed by cryptographers from Belgium, i.e Dr. Joan Daemen and Dr. Vincent Rijmen.

• AES uses block length of 128bits and a key length that can be 128, 192 or 256 bits.

• It is not like Feistel Structure.
• Next figure shows overall structure of AES.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Figure 2.5 AES Encryption and Decryption

(a) Encryption
(b) Decryption
AES Procedural Operations:

- Single 128-bit block appearing as square matrix of bytes is copied into 'State' array which is modified at each stage of E and D.
- After the final stage, 'State' is copied to an output matrix (that's y only 3 stages).
- Key Expansion: 128-bit key is depicted as a square matrix of bytes. This key is then expanded into an array of key schedule words: each word is four bytes and the total key schedule is 44 words upto (4*44=176) 176-bits for the 128-bit key. Ordering of bytes is in column order.
- Lets study each step in each round...

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Figure 2.6 AES Encryption Round


3. Mix columns: Substitution that alters each byte in column as a function of all of bytes in column.

4. Add Round Keys: Simple bitwise XOR of current block with a portion of expanded key.

   • For both encryption and decryption, the cipher begins with an
     - Add Round Key stage, followed by
     - Nine rounds that each includes all four stages,
     - Followed by a tenth round of three stages.

   • Only the Add Round Key stage makes use of the key. For this reason, the cipher begins and ends with an Add Round Key stage.

   • Advantage: Each stage is easily reversible.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Other Symmetric Block Ciphers

- Just for your knowledge:
- International Data Encryption Algorithm (IDEA)
  - 128-bit key
  - Used in PGP
- Blowfish
  - Easy to implement
  - High execution speed
  - Run in less than 5KB of memory

Email:bhargavigoswami@gmail.com, Mob: 9426669020
Other Symmetric Block Ciphers

- **RC5**
  - Suitable for hardware and software
  - Fast, simple
  - Adaptable to processors of different word lengths
  - Variable number of rounds
  - Variable-length key
  - Low memory requirement
  - High security
  - Data-dependent rotations

- **Cast-128**
  - Key size from 40 to 128 bits
  - The round function differs from round to round

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Block Cipher v/s Stream Cipher

• A block cipher processes the input one block of elements at a time, producing an output block for each input block.

• Popular Block Ciphers: DES, 3DES, AES.

• Application: File Transfer, Email, Database.

• A stream cipher processes the input elements continuously, producing output one element at a time.


• Application: Data communication channels, web/browser link.
Stream Ciphers and RC4

• Topics:
  1. Stream Cipher Structure
  2. RC4

1. Stream Cipher Structure
   - Encrypts plaintext one byte at a time.
   - Similar to one time pad.
   - Then what is the difference?
     • One time pad uses genuine random number stream.
     • Stream cipher uses a pseudorandom number stream.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Figure 2.7 Stream Cipher Diagram

Email: bhargavigoswami@gmail.com, Mob: 9426669020
• Encryption:
  11001100 plaintext XORed with 01101100 key stream resulting to 10100000 cipher text

• Decryption:
  10100000 ciphertext XORed with 01101100 key stream resulting to 11001100 plaintext

Email: bhargavigoswami@gmail.com, Mob: 9426669020
• A key is input to a pseudorandom bit generator that produces a stream of 8-bit numbers that are apparently random.

• The output of the generator, called a key-stream, is combined one byte at a time with the plaintext stream using the bitwise exclusive-OR (XOR).

• A pseudorandom stream is one that is unpredictable without knowledge of the input key and which has an apparently random character.

• Advantage over block ciphers:
  - With pseudorandom number generator, a stream cipher can be as secure as block cipher of comparable key length.
  - Stream ciphers are always faster than block cipher.
  - Use far less code than block cipher.

• Disadvantage under block ciphers:
  - Block ciphers can re-use keys but not stream ciphers.

• Application:
  - Data communication channels, web/browser link.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Some design considerations for stream cipher =>

a) The encryption sequence should have a large period. A pseudorandom number generator uses a function that produces a deterministic stream of bits that eventually repeats. The longer the period of repeat, the more difficult it will be to do cryptanalysis.

b) For example, there should be an approximately equal number of 1s and 0s. More random-appearing the keystream is, the more randomized the ciphertext is, making cryptanalysis more difficult.

c) To guard against brute-force attacks, the key needs to be sufficiently long.
2. RC4

- RC4 is a stream cipher designed in 1987 by Ron Rivest for RSA Security.
- It is a variable key-size stream cipher with byte-oriented operations.
- Uses: a random permutation.
- Used with:
  - Secure Sockets Layer/Transport Layer Security (SSL/TLS) standards that have been defined for communication between Web browsers and servers.
  - Used in the Wired Equivalent Privacy (WEP) protocol and the newer WiFi Protected Access (WPA) protocol that are part of the IEEE802.11 wireless LAN standard.
- RC4 was kept as a trade secret by RSA Security.
- But in September 1994, the RC4 algorithm was posted on the Internet on the Cypherpunk’s remailers list.
- Algorithm is simple and quite easy to explain

Email: bhargavigoswami@gmail.com, Mob: 9426669020
1. Initialize $S$: Variable length key of from 1 to 256 bytes (8 to 2048 bits) is used to initialize a 256-byte state vector $S$, with elements $S[0]$, $S[1]$, . . . , $S[255]$.

2. $k$ is generated from $S$ by selecting one of the 255 entries in a systematic fashion.

3. Then the entries in $S$ are once again permuted.

4. A temporary vector $T$, is also created.

5. Set vector $T$: If the length of the key $K$ is 256 bytes, then $K$ is transferred to $T$. Otherwise, for a key of length $\text{keylen}$ bytes, the first $\text{keylen}$ elements of $T$ are copied from $K$, and then $K$ is repeated as many times as necessary to fill out $T$. These preliminary operations can be summarized as:

   ```
   /* Initialization */
   for i=0 to 255 do
       S[i] = i;
       T[i] = K[i mod keylen];
   ```

   Email: bhargavigoswami@gmail.com, Mob: 9426669020
Figure 2.8  RC4
6. Use T to produce the initial permutation of S and then perform swapping.

/* Initial Permutation of S */
j = 0;
for i = 0 to 255 do
    j = (j + S[i] + T[i]) mod 256;
    Swap (S[i], S[j]);

7. Stream generation: cycling through all the elements of S[i] and, for each S[i], swapping S[i] with another byte in S.

/* Stream Generation */
i, j = 0;
while (true)
    i = (i + 1) mod 256;
    j = (j + S[i]) mod 256;
    Swap (S[i], S[j]);
    t = (S[i] + S[j]) mod 256;
    k = S[t];

- Encryption: XOR the value $k$ with the next byte of plaintext.
- Decryption: XOR the value $k$ with the next byte of ciphertext.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Given table shows the execution speed of RC4 in comparison with other methods. See last line, it performs well in Pentium II Processor.

Table 2.3  Speed Comparisons of Symmetric Ciphers on a Pentium II

<table>
<thead>
<tr>
<th>Cipher</th>
<th>Key Length</th>
<th>Speed (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DES</td>
<td>56</td>
<td>9</td>
</tr>
<tr>
<td>3DES</td>
<td>168</td>
<td>3</td>
</tr>
<tr>
<td>RC2</td>
<td>variable</td>
<td>0.9</td>
</tr>
<tr>
<td>RC4</td>
<td>variable</td>
<td>45</td>
</tr>
</tbody>
</table>

Email: bhargavigoswami@gmail.com, Mob: 9426669020
• **Advantage:**
  - Vulnerable to attacks after many tries.

• **Dis-advantage:**
  - WEP Protocol needs confidentiality that it is secure from attacks, but RC4 could not give it. Reason? The way in which the key is generated is not secure and errorfree.

• **Conclusion:**
  - Difficulty in designing a secure system that involves both cryptographic functions and protocols specific.
Cipher Block Modes of Operation:

- Symmetric block cipher processes 1 block at a time.
- In DES and 3DES, block size is same, i.e 64 bits.
- Now, ECB (Electronic Code Book) mode where block is encrypted using same key.
- Codebook can be used as for given key there is unique cipher-text for each 64bit block of plaintext.
- Then, for repetitive text appearing in plaintext would create repetitive cipher-text.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
FON: Electronic Code Book Mode

- The plaintext of a file encrypted as 16 DES blocks.

<table>
<thead>
<tr>
<th>Name</th>
<th>Position</th>
<th>Bonus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adams, Leslie</td>
<td>Clerk</td>
<td>$10</td>
</tr>
<tr>
<td>Black, Robin</td>
<td>Boss</td>
<td>$5,000,000</td>
</tr>
<tr>
<td>Collins, Kim</td>
<td>Manager</td>
<td>$1,000,000</td>
</tr>
<tr>
<td>Davis, Bobbie</td>
<td>Janitor</td>
<td>$5</td>
</tr>
</tbody>
</table>

Email: bhargavigoswami@gmail.com, Mob: 9426669020
FON: Cipher Block Chaining Mode

- Cipher block chaining. (a) Encryption. (b) Decryption.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
**Cipher Block Chaining Mode (CBC)**

- The input to the encryption algorithm is the XOR of the current plaintext block and the preceding cipher-text block. See next figure.
- Advantage: Repeating pattern of 64-bits are not exposed.
- Left side, E and D and its relationship.
- Right side, IV function must be known by sender and receiver and also protected same as key.

\[
C_i = E_k [C_{i-1} \oplus P_i]
\]

\[
D_k [C_i] = D_k [E_k (C_{i-1} \oplus P_i)]
\]

\[
D_k [C_i] = (C_{i-1} \oplus P_i)
\]

\[
C_{i-1} \oplus D_k [C_i] = C_{i-1} \oplus C_{i-1} \oplus P_i = P_i
\]

Email: bhargavigoswami@gmail.com, Mob: 9426669020
(a) Encryption

(b) Decryption

Figure 2.9 Cipher Block Chaining (CBC) Mode
• Now if i wanna use stream cipher instead of block, what to do?
• **Cipher Feedback Mode (CFB).**
• Input 64 bit shift register with Initial Vector (IV).
• Significant s bits are XORed with first unit of $P_1$ producing $C_1$.
• Then, shift register is shifted s bits for further processing.
• This process continues until all plaintext units have been encrypted.
• Decryption: the received ciphertext unit is XORed with the output of the encryption function to produce the plaintext unit.

$$C_1 = P_1 \oplus S_s[E(K, IV)]$$

$$P_1 = C_1 \oplus S_s[E(K, IV)]$$

Email: bhargavigoswami@gmail.com, Mob: 9426669020
FON: Cipher Feedback Mode

- (a) Encryption. (c) Decryption.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Figure 2.10 $s$-bit Cipher Feedback (CFB) Mode
Location of Encryption Devices

- We need to decide what to encrypt and where the encryption gear should be located. Alternatives:
  - Link Encryption
    - A lot of encryption devices
    - High level of security
    - Disadvantage: Decrypt each packet at every switch
  - End-to-End Encryption
    - The source encrypts and the receiver decrypts
    - Message is transmitted unaltered across the n/w.
    - Secure Data against attacks on network links and switches.
    - Disadvantage: Router cannot read Destination address, so can’t route the packet on optimum path.
- High Security
  - Both link and end-to-end encryption are needed (see Figure 2.11)
Figure 2.11 Encryption Across a Packet-Switching Network
Key Distribution

Key distribution can be achieved in following ways:
1. A key could be selected by A and physically delivered to B.
2. A third party could select the key and physically deliver it to A and B.
3. If A and B have previously used a key, one party could transmit the new key to the other, encrypted using the old key.
4. If A and B each have an encrypted connection to a third party C, C could deliver a key on the encrypted links to A and B.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
Two kind of keys are identified by schema:

1. **Session key**
   - Data encrypted with a one-time session key; At the conclusion of the session the key is destroyed.

2. **Permanent key**
   - Used between entities for the purpose of distributing session keys.

Elements of Key distribution:

1. **Key Distribution Center (KDC):**
   - Determine which systems are allowed to communicate.
   - Once permission granted provides one time session key to concerned parties communicating.

2. **Security Service Module (SSM):**
   - Consist of functionality of protocol layer.
   - Performs End to End Encryption.
   - Obtain session keys on behalf of users.

See Figure.

Email: bhargavigoswami@gmail.com, Mob: 9426669020
1. Host sends packet requesting connection.
2. Security service buffers packet; asks KDC for session key.
3. KDC distributes session key to both hosts.

Figure 2.12 Automatic Key Distribution for Connection-Oriented Protocol
2nd chapter is over...

- Thank You....
- Assignment Submission?????? Date?
- Test?????? Date?

- 😊😊😊😊😊