Introduction to Cryptography

Developed for the Azera Group

By: Joseph D. Fournier B.Sc.E.E., M.Sc.E.E

Definition: Cryptography

- **Cryptography**
  - In a narrow sense
    - Mangling information into apparent unintelligibility
    - Allowing a secret method of un-mangling
  - In a broader sense
    - Mathematical techniques related to information security
    - About secure communication in the presence of adversaries
- **Cryptanalysis**
  - The study of methods for obtaining the meaning of encrypted information without accessing the secret information
- **Cryptology**
  - Cryptography + cryptanalysis

Purpose of Cryptography

- Confidentiality
- Authentication
- Integrity
- Nonrepudiation
- Access Control
- Availability

Encryption Basics

Encryption algorithms use two basic principles

- Substitution: each element of plaintext is mapped into another element
- Transposition: elements in the plaintext are rearranged

Categories of Encryption

- Symmetric: sender and receiver use the same key (aka single-key, and secret-key)
- Asymmetric: sender and receiver use different keys (aka two-key, and public-key)

Processing Encryption

- Block cipher: processes the input a block of elements at a time (typically 64-bits)
- Stream cipher: processes the input continuously producing an element at a time

Viability

- No encryption scheme is full proof!
- Two requirements are needed to make encryption viable:
  - The cost of breaking exceeds the value of the encrypted information
  - The time required to break the cipher exceeds the useful lifetime of the information

Cryptanalysis

Definition: attempting to break a cryptography algorithm

- Brute force: exhaustively searching the entire key space
- Dictionary: using well known words to guess the key

Exhaustive Key Search

- It is difficult to estimate the amount of effort required to cryptanalyze ciphertext successfully (more on this later)
- The strength of an algorithm is typically based on key size
- Usually only 50% of key space has to be searched for success

Exhaustive Key Search (cont.)

Key Size	Number of Alt. Keys	1 encryption/µs	10^6 encryptions/ µs
32	$2^{32} = 4.3\text{x}10^9$	2^{31} µs = 35.8 min	2.15 ms
56	$2^{56} = 7.2\text{x}10^{16}$	2^{55} µs = 1142 yrs	10.01 hrs
128	$2^{128} = 3.4\text{x}10^{38}$	2^{127} µs = 5.4 x 10^{24} yrs	5.4 x 10^{18} yrs
26 chars.	$26! = 4\text{x}10^{26}$	2 x 10^{26} µs = 6.4 x 10^{12} yrs	6.4 x 10^6 yrs

Diffusion and Confusion

The process of thwarting cryptanalysis based upon statistical analysis

- Terms were introduced by Claude Shannon in 1945 (1949).
- Diffusion: statistical structure of the plaintext is dissipated into long-range statistics
- Confusion: relationship between the statistics of the ciphertext and the value of the encryption key is as complex as possible

Popular Forms of Encryption

- Hash functions
- Block ciphers
- Secret Key
- Public Key

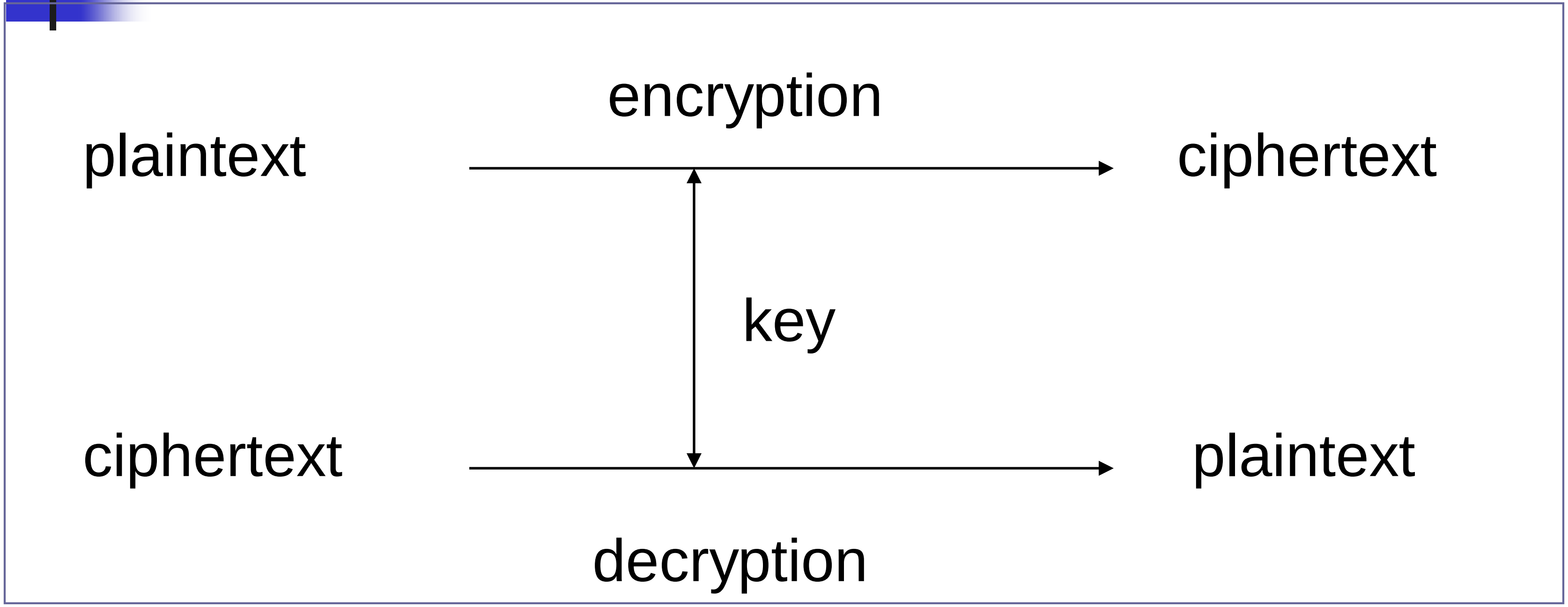
Hash Functions

- Accepts an arbitrary sized input and produces a fixed size output
- Provides error detection
- One-way: for any give code h, it is computationally infeasible to find x such that H(x) = h
- Strong collision resistance: given a block x it is computationally infeasible to find x ≠ y with H(y) = H(x)
- Strong collision resistance: computationally infeasible to find any pair (x,y) such that H(x) = H(y)
- It's easy to generate a code given a message, but virtually impossible to generate a message given a code
- Examples: MD4, MD5, SHA-1, RIPEMD-160, Crypt3

Block Cipher

- Operates on a fixed number of elements at a time
- All most all block ciphers are based upon a structure created by Feistel, called the Feistel Cipher
- Feistel Cipher is composed of multiple iterations of substitutions, and permutations
- Feistel's Cipher is a practical application of Shannon's work
- Examples: DES, 3DES, AES, Blowfish, Twofish

Secret Key Encryption



encryption

plaintext

ciphertext

key

ciphertext

plaintext

decryption

- Using a single key for encryption/decryption.
- The plaintext and the ciphertext having the same size.
- Also called *symmetric* key cryptography

Secret Key Encryption

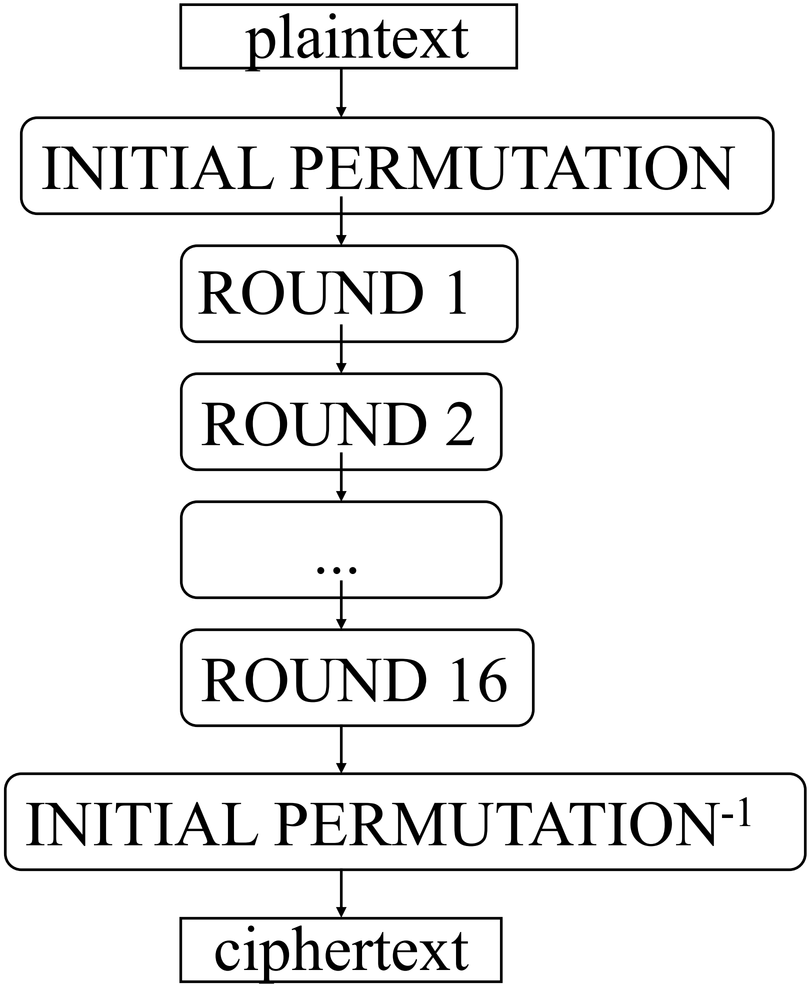
- DES (Data Encryption Standard)
- 3DES (Triple DES)
- IDEA (International Data Encryption Algorithm)
- AES (Advanced Encryption Standard)

Data Encryption Standard (DES)

- Most widely used encryption standard
- Developed by IBM in the late 1960’s as part of a research project on computer cryptography
- A revised edition was developed for the NSA
- The key size of 128-bits was reduced to 56-bits

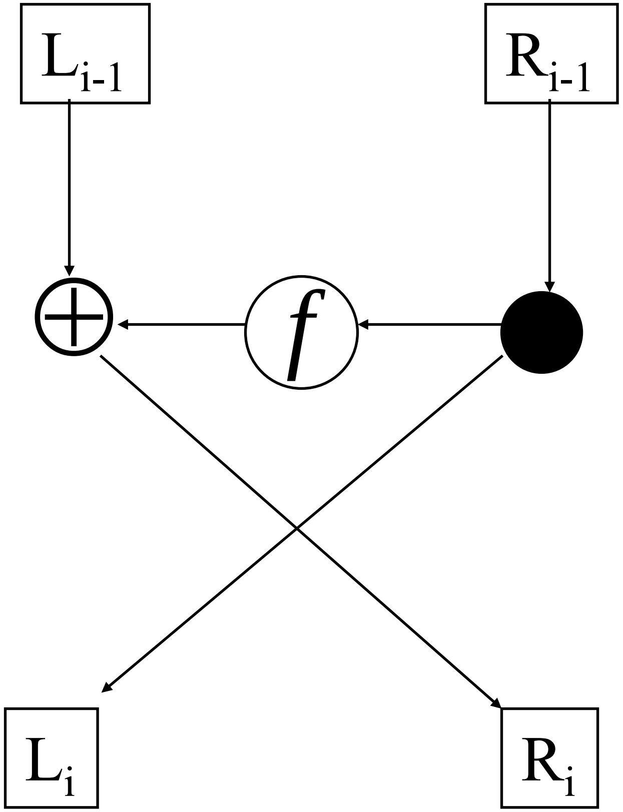
Data Encryption Standard (DES)

- Block cipher: 64 bits at a time
- Initial permutation rearranges 64 bits (no cryptographic effect)
- Encoding is in 16 rounds



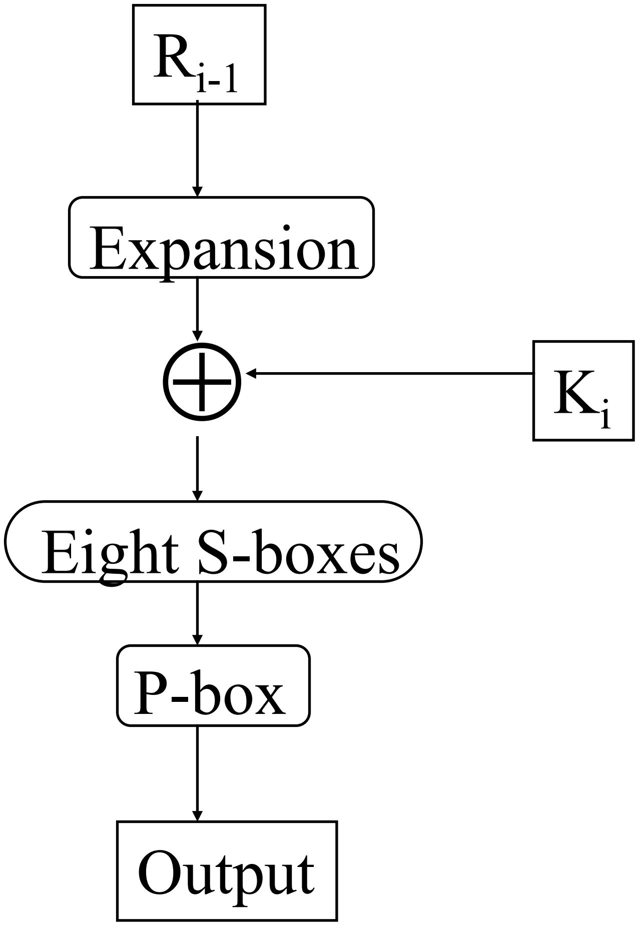
Data Encryption Standard (DES)

- 64 bits divided into left, right halves
- Right half goes through function f, mixed with key
- Right half added to left half
- Halves swapped (except in last round)



Data Encryption Standard (DES)

- Expand right side from 32 to 48 bits (some get reused)
- Add 48 bits of key (chosen by schedule)
- S-boxes: each set of 6 bits reduced to 4
- P-box permutes 32 bits



Data Encryption Standard (DES)

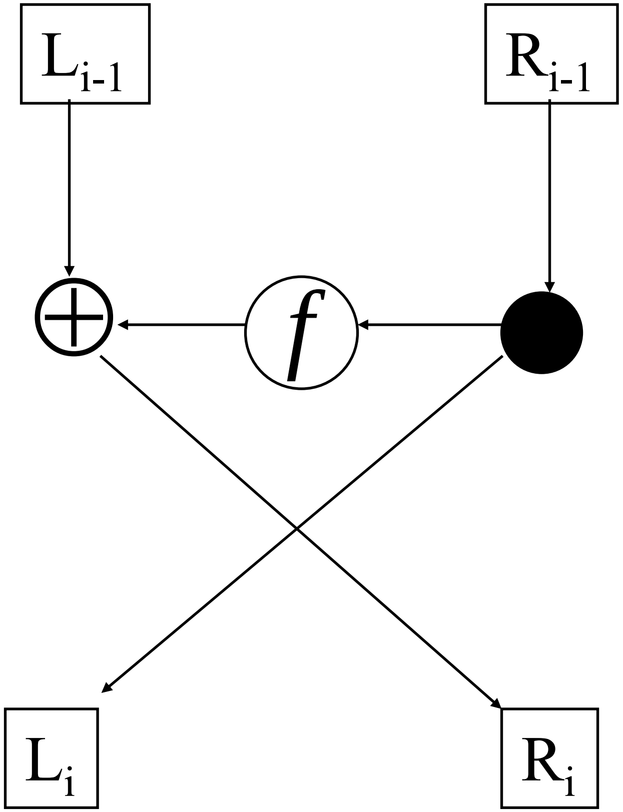
- Equations for round i:

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f\left(R_{i-1}\right)$$

- In other words:

$$R_{i-1} = L_i$$
$$L_{i-1} = R_i \oplus f\left(L_i\right)$$

- So decryption is the same as encryption
- Last round, no swap: really is the same



Triple DES Encryption

- $C = DES_{k3}(DES_{k2}(DES_{k1}(P)))$.
- Data block size: 64-bit
- Key size: 168-bit key; effective key size: 112 (due to man-in-the-middle attack)
- Encryption is slower than DES
- Securer than DES

Triple DES Encryption

- Run DES three times:
  - ECB mode: $C_i = E_{K_3}\left(D_{K_2}\left(E_{K_1}\left(P_i\right)\right)\right)$
- If $K_2 = K_3$, this is DES
  - Backwards compatibility
- Known not to be just DES with K_4 (1992)
- Has 112 bits of security, not 3 × 56 = 168

Triple DES Encryption

- Double-DES: $C_i = E_B(E_A(P_i))$
- Given P_1, C_1: Note that $D_B(C_1) = E_A(P_1)$
- Make a list of every $E_K(P_1)$.
- Try each L: if $D_L(C_1) = E_K(P_1)$, then maybe K = A, L = B. (2^{48} L's might work.)
- Test with P_2, C_2: if it checks, it was probably right.
- Time roughly 2^{56}. Memory very large.

IDEA Encryption

- Authors: Lai & Massey, 1991
- Data block size: 64-bit
- Key size: 128-bit
- Encryption is slower than DES
- Security
  - Nobody has yet published results on how to break it
- Having patent protection

IDEA Encryption

- DES algorithm has been a popular secret key encryption algorithm and is used in many commercial and financial applications. However, its key size is too small by current standards and its entire 56 bit key space can be searched in approximately 22 hours
- IDEA is a block cipher designed by Xuejia Lai and James L. Massey in 1991
- It is a minor revision of an earlier cipher, PES (Proposed Encryption Standard)
- IDEA was originally called IPES (Improved PES) and was developed to replace DES

IDEA Encryption

- IDEA operates with 64-bit plaintext and cipher text blocks and is controlled by a 128-bit key
- Completely avoid substitution boxes and table lookups used in the block ciphers
- The algorithm structure has been chosen such that when different key sub-blocks are used, the encryption process is identical to the decryption process

IDEA Encryption

- The 64-bit plaintext block is partitioned into four 16-bit sub-blocks
- six 16-bit key are generated from the 128-bit key. Since a further four 16-bit key-sub-blocks are required for the subsequent output transformation, a total of 52 (= 8 x 6 + 4) different 16-bit sub-blocks have to be generated from the 128-bit key.

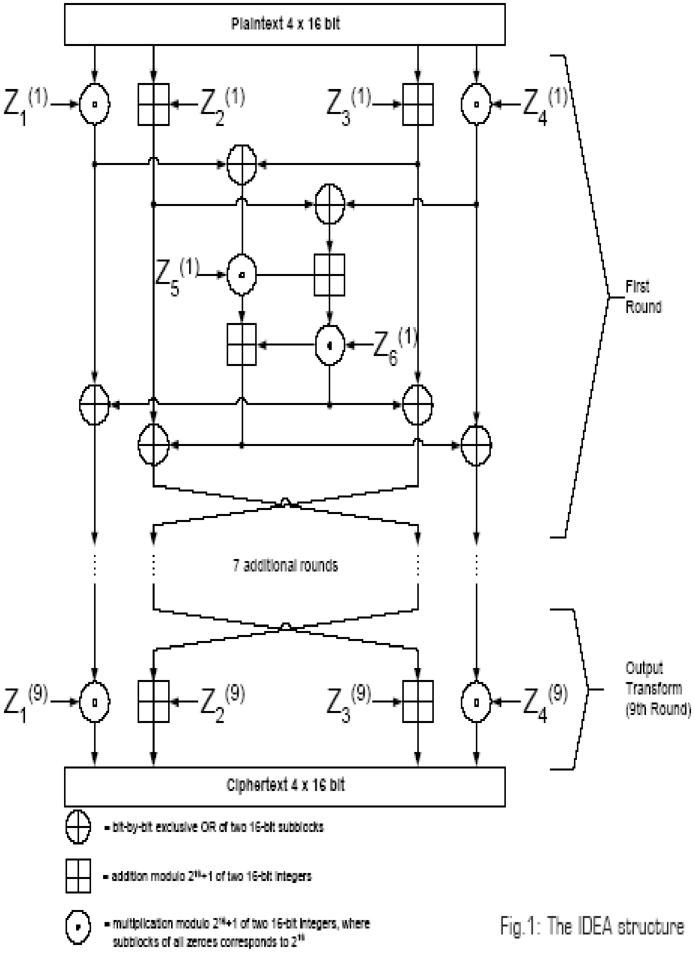

Plaintext 4 x 16 bit
$Z_1^{(1)}$
$Z_2^{(1)}$
$Z_3^{(1)}$
$Z_4^{(1)}$
$Z_5^{(1)}$
$Z_6^{(1)}$
First Round
7 additional rounds
$Z_1^{(9)}$
$Z_2^{(9)}$
$Z_3^{(9)}$
$Z_4^{(9)}$
Output Transform (9th Round)
Ciphertext 4 x 16 bit
= bit-by-bit exclusive OR of two 16-bit subblocks
= addition modulo 2^{16}+1 of two 16-bit integers
= multiplication modulo 2^{16}+1 of two 16-bit integers, where subblocks of all zeroes corresponds to 2^{16}

Fig.1: The IDEA structure

IDEA Encryption

- The computational process used for decryption of the ciphertext is essentially the same as that used for encryption
- The only difference is that each of the 52 16-bit key sub-blocks used for decryption is the inverse of the key sub-block used during encryption
- In addition, the key sub-blocks must be used in the reverse order during decryption in order to reverse the encryption process

AES Encryption

- Authors: Daemen & Rijmen
- Block size:128-bit
- Key size: 128-bit, 192-bit, 256-bit
- Encryption is fast
- Security
  - As of 2005, no successful attacks are recognized.
  - NSA stated it secure enough for non-classified data.

AES Encryption

- DES cracked, Triple-DES slow:
- 1997: AES announced, call for algorithms
- August 1998: 15 candidate algorithms
- October 2000: Rijndael selected
  - Two Belgians: Joan Daemen, Vincent Rijmen
- May 2001: Comment period ended
- Summer 2001: Finalized, certified until ‘06

AES Encryption

- Similar to DES: block cipher (with different modes), but 128-bit blocks
- 128-bit, 192-bit, or 256-bit key
- Mix of permutations, “S-boxes”
- S-boxes based on modular arithmetic with polynomials:
  - Non-linear
  - Easy to analyze, prove attacks fail

Public Key Encryption

- Based on mathematics as opposed to substitution and permutation
- Mostly used for key management and signature applications
- Computationally expensive compared to other encryption algorithms
- Composed of two keys: a key for encryption, and a key for decryption (doesn’t matter which one)
  - Public Key: encryption
  - Private Key: decryption

RSA Algorithm

- Developed by Rivest, Shamir, and Adleman

$$C = M^e \bmod n$$

$$M = C^d \bmod n = (M^e)^d \bmod n$$

Plaintext block: M, Ciphertext block: C

RSA (cont.)

- Both sender and receiver must know the value of *n*
- The sender knows the value of *e*
- Only the receiver knows the value of *d*
  - Public Key: KU = {e,n}
  - Private Key: KR = {d,n}

Key Generation

- Select p,q (both prime)
- Calculate $n = p \times q$
- Calculate $\Phi(n) = (p-1)(q-1)$
- Select integer e: $\gcd(\Phi(n),e) = 1$;
  $1 < e < \Phi(n)$
- Public Key: $KU = \{e,n\}$
- Private Key: $KR = \{d,n\}$

Numbers…Please!

- Using freely available libraries benchmark results were gathered to determine the amount of time it for various encryption algorithms to execute
- SSL handshake performance was benchmarked
- The OpenSSL and Crypto++ libraries were used to obtain the results

Crypto Benchmark

Algorithm	Bytes Processed	Time	MB/s	Crypt/s
CRC-32	134217728	0.703	182.07	N/A
MD5	134217728	0.922	138.83	N/A
SHA-1	67108864	1.078	59.369	N/A
DES	16777216	1.094	14.625	239620
Blowfish	16777216	0.750	21.333	349525
AES (128)	33554432	0.953	33.578	249823
AES (192)	33554432	1.125	28.444	233016
AES (256)	33554432	1.266	25.276	207064

Crypto Benchmark

Operation	Iteration	Total Time	ms/op
RSA 512 Encrypt	8885	1.000	0.11
RSA 512 Decrypt	692	1.000	1.45
RSA 1024 Encrypt	3992	1.000	0.25
RSA 1024 Decrypt	137	1.000	7.30
RSA 512 Sign	689	1.000	1.45
RSA 512 Verify	9830	1.000	0.10
RSA 1024 Sign	135	1.000	7.41
RSA 1024 Verify	4263	1.000	0.23

Kerberos

- Secure
- Reliable
- Transparent
- Scalable

Project Athena

- Started at MIT in 1983 to integrate computers into the curriculum
- Over 6,000 computers had to be integrated
- Other projects came out Athena, including the X windowing system
- Athena – Greek Goddess of wisdom, justice, war, culture, law, and crafts

Kerberos

- Designed to securely manage all of the computers in the Athena project
- Watchdog of Hades
- Usually had three heads, a serpent’s tail, a mane of snakes, and a lion’s claw
- Kerberos supposed to have 3 tasks – authentication, auditing, and accounting, only one was implemented

Access Control

Three approaches to access control:

1. Nothing
2. Require the host to prove its identity but trust the host word's as to who to user is (rsh, rlogin)
3. Require the user to prove his identity for each required service, and server must prove its identity

Kerberos

- Based upon the protocol proposed by Needham and Schroeder
- Only conventional encryption was used
- Kerberos IV makes use of DES
- Kerberos I, II, and III were internal versions

Components of Kerberos

- Administrative Server (KDBM) available in master and slaves
- Authentication Server (Kerberos server)
- Ticket Granting Server (TGS)
- Encryption Library
- Database Library
- User Programs
- Applications

What’s in a Name?

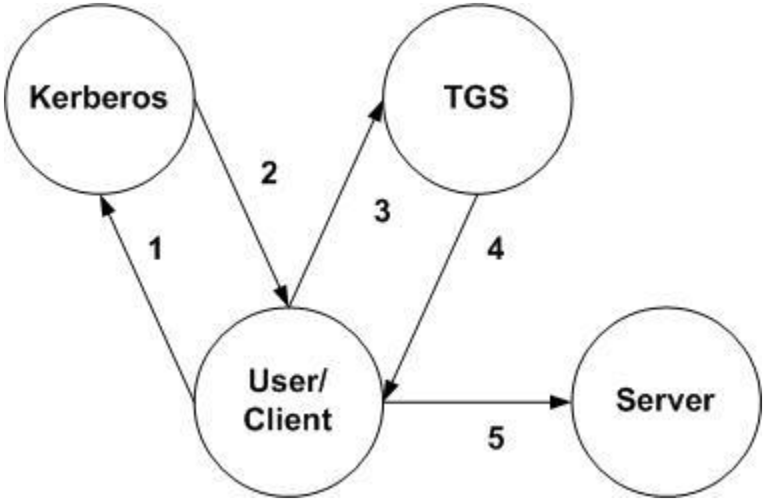
- Consists of a primary name, an instance, and a realm expressed as name.instance@realm
  - Primary name: name of user or service
  - Instance name: can be used to indicate other privileges such as root
  - Realm: name of an administrative entity that maintains authentication data

Logon Process

- User obtains credentials to be used to request access to other service
- User requests authentication for a specific service
- User presents the granted credentials to the end server

Kerberos Authentication Protocol

- Request for TGS ticket
- Ticket for TGS
- Request for Server ticket
- Ticket for Server
- Request for service



Credentials

- Two types of credentials:
  - Ticket: securely passes the identity of the user between the authentication server and the end server
  - Authenticator: contains information that when compared against a ticket proves that the client presenting the ticket was the same one the ticket was issued too

Ticket

$$\{s,c,addr,timestamp,life,K_{s,c}\}K_s$$

- Good for a single server and service
- Ticket contains information such as name of server, IP address of client, timestamp, a lifetime, and a random session key (RSK)
- Ticket is encrypted using the key of the server it is to be used for

Authenticator

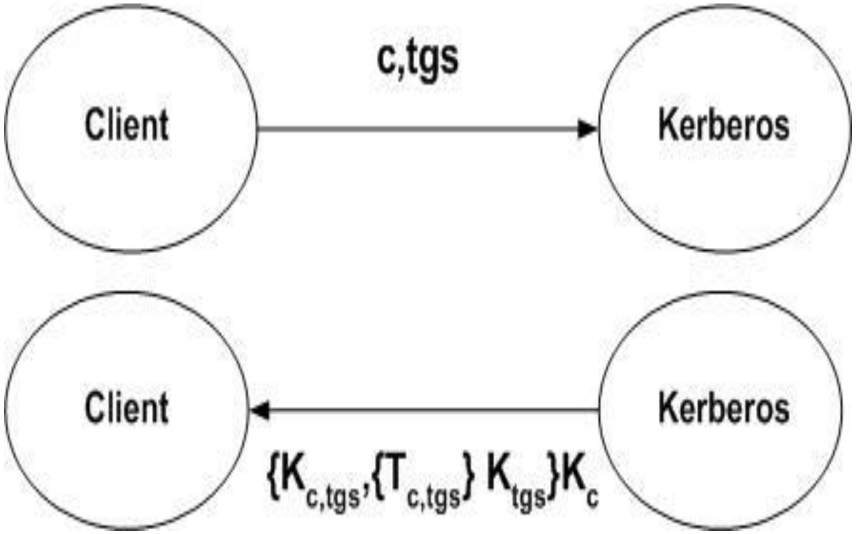
$$\{c, addr, timestamp\}K_{s,c}$$

- Unlike a ticket an authenticator can only be used once
- This is not a issue because the client can build all the authenticators it needs

Logging On

Server

- Client makes request to Kerberos with user name and TGS
- Server verifies it knows the client, and generates a RSK
- Server creates a ticket for the TGS
- Ticket is encrypted in a key known only to the TGS and Kerberos server
- The client’s key (derived from the user’s password) is used to decrypt the message

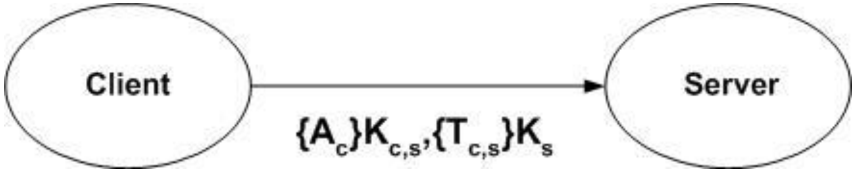


Service

- To gain access to a server, the application builds an authenticator containing the client’s name, IP address, and current time
- Authenticator is encrypted using the session key that was received with the ticket for the server

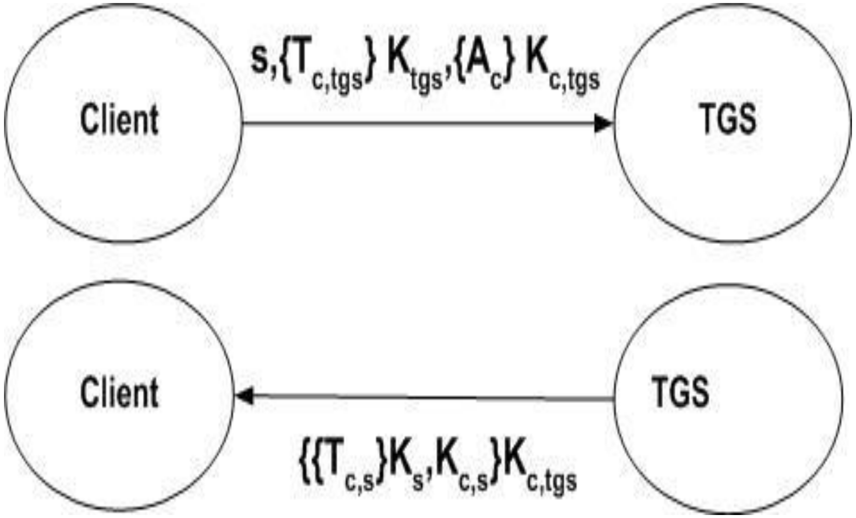
Requesting a Service

- Assume the user already has a ticket for the server
- Authenticator is built
- Client sends the authenticator with the ticket to the server
- Server decrypts ticket, then the authenticator and verifies the client's identity



My First Ticket

- Every time a program wants to make use of a service it doesn’t yet have a ticket for it makes a request to the TGS
- It builds an authenticator and the service that it wants to use


Client
$s,\{T_{c,tgs}\}K_{tgs},\{A_c\}K_{c,tgs}$
TGS
Client
TGS
$\{\{T_{c,s}\}K_s,K_{c,s}\}K_{c,tgs}$

My First Ticket (cont.)

- TGS builds a new RSK to be used between the client and server. It then builds a ticket for the new server containing the client’s name, server’s name, current time, the client’s IP address, and the new session key it generated.

Kerberos Database

- Database is encrypted in master’s key
- Multiple databases can be used for fault tolerance, speed, and efficiency
- Only the master database is allowed to accept changes
- Replication entails the master database dumping its contents every hour and pushing them to the slaves

KDBM Server

- KDBM only accepts requests to add principles or change the password for existing principles
- TGS will not grant tickets for the KDBM, only the authentication service can do this
    - This prevents other people from changing one principal’s password if they leave a machine unattended

Inter-realm Access

- User’s will want to communicate with other realms
- Realms must agree on a key to share for inter-realm access