University of Houston

Foundations of Security

COSC 6347

Midterm Review

Author
K.M. Hourani

Based on Notes By
Dr. Aron Laszka

October 5, 2020

1 Introduction to Security

1.1 Objectives

	Term	Definition
CIA	Confidentiality	not available to unauthorized entities
	Integrity	cannot be altered by unauthorized entities
	Availability	available to authorized entities
	Non-repudiation Accountability	actions can be provably traced back to an entity
	Privacy	individuals have control over information related to them

1.2 Challenges

Weakest link – principle that the defender needs to find and fix all vulnerabilities, but attacker needs to find only a single vulnerability

Security is a process, not a product – attackers continuously looking for new vulnerabilities, so systems must be regularly updated and continuously monitored. Tension between security and

- usability
- functionality
- efficiency
- time-to-market
- development cost

Value of security often only perceived when there is a security failure

Can be measured by

- checking compliance
- pentesting

2 Introduction to Cryptography

2.1 Attacker Modeling Principles

Security is defined with respect to an attacker model – what the attacker

- can do
- knows
- wants to achieve

Generally better to overestimate the attacker's capabilities, knowledge, and determination.

Safe to assume attacker knows

- algorithms
- system design
- implementation
- configuaration

but the attacker cannot know *truly* random values.

2.2 Security by Obscurity

Security by obscurity – providing security by keeping the design or implementation of a system secret

Generally rejected by security experts, researchers, standard bodies, i.e., everyone.

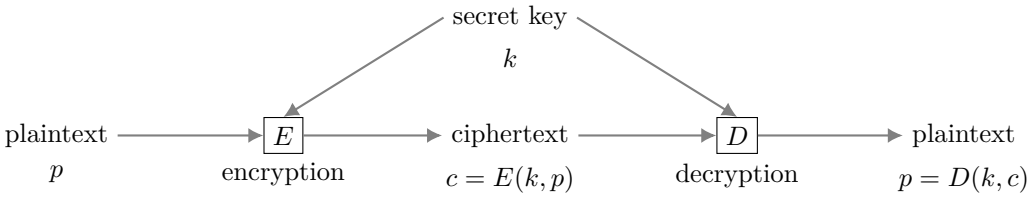
Obscurity can slow down, but not stop, an attack:

- if we thought of something, attacker might also
- attacker might try attack for many possible design/implementation choices

Can create false sense of security.

2.3 Symmetric-Key Ciphers

Sender and receiver share a secret key k



Types of attacks:

Acronym	Attack	Description
COA	ciphertext only	only the algorithms used and the ciphertext are known
KPA	known plaintext	one or more plaintext-cipher pairs is known
CCA	chosen ciphertext	one or more *chosen* plaintext-cipher pairs is known
CPA	chosen plaintext	can obtain the ciphertext for any plaintext
CTA	chosen text	both chosen ciphertext and chosen plaintext
	brute-force	every possible key is tried
	cryptanalytic	relies on the nature of the algorithm/characteristics of the plaintext

2.4 Kerckhoffs's Principle

Kerckhoffs's Principle – a cryptographic system should be secure, even if all of its details, except for the key, are publicly known. Rejection of security by obscurity

3 Stream Ciphers

3.1 Perfect Security

Perfect security – attacker gains no information about the plaintext from observing the ciphertext, formally,

$$\mathbb{P}(P = p) = \mathbb{P}(P = p \mid E(K, P) = c)$$

i.e., that the plaintext and ciphertext are independent

One-time pad – perfect security in which a single-use encryption key at least as long as the plaintext is chosen randomly and used to encrypt only a single message

3.2 Semantic Security

Semantic security – attacker advantage for any efficiently computable guess is negligible over random guessing Many-time pad: reusing the one-time key for multiple plaintext. Attacker can recover $p_1 \oplus p_2$:

$$\begin{aligned} c_1 \oplus c_2 &= (p_1 \oplus k) \oplus (p_2 \oplus k) \\ &= (p_1 \oplus p_2) \oplus (k \oplus k) \\ &= p_1 \oplus p_2 \end{aligned}$$

and if attacker knows p_1, can recover p_2:

$$\begin{aligned} p_1 \oplus (c_1 \oplus c_2) &= p_1 \oplus (p_1 \oplus p_2) \\ &= (p_1 \oplus p_1) \oplus p_2 \\ &= p_2 \end{aligned}$$

3.3 General Model of Stream Ciphers

Make one-time pad practical by securely extending the key.

Pseudorandom Number Generator

pseudorandom number generator (PRNG) – takes fixed-length seed and generates a sequence of bits using a deterministic algorithm

Requirements:

- performance – generates key as long as plaintext, so must be computationally efficient
- security – generated sequence must be indistinguishable from true randomness
  - cryptanalytic attack
    - uniform distribution – 0s and 1s occur with approximately same frequency
    - independence – no subsequence can be inferred from another, disjoint subsequence
  - brute-force attack
    - n bit key has 2^n possible values – attacker can try all
    - key must be sufficiently long – in 2014, NIST recommends 112-bits
    - as computers become faster, key length must be increased

How Stream Cipher Works

stream cipher – takes fixed-length seed and uses a PRNG to produce sequence of bits as long as the plaintext then encrypts with XOR

Use PRNG to generate the sequence up to the length of the plaintext, then to

encrypt — XOR plaintext with key
decrypt — XOR ciphertext with key

3.4 Key-Reuse Problem

If attacker learns $p_1 \oplus p_2$, $p_2 \oplus p_3$, $p_1 \oplus p_3$, ..., they can recover other plaintexts. Solutions:

- one continuous sequence that allows seeking to any position in the key
- nonce – number used once
  - xor key with nonce for each plaintext to produce different key

3.5 RC4

Old WiFi and Web Security standard

RC4 Advantages

- variable key length (from 8 to 2048 bits)
- very simple, uses byte-oriented operations:
  - only 8 to 16 machine operations required per output byte

Applications

- Wifi: WEP and WPA
  - broken in 2001, deprecated in 2004
- Web Security (HTTPS): SSL and TLS
  - broken in 2013, deprecated in 2015

RC4 has been retired.

3.6 Salsa20/ChaCha20

State of the Art Stream Cipher Salsa20 (and more secure, more efficient variant ChaCha20)

Key length is 128 or 256 bits.

Advantages

- fast software implementation (simple 32-bit operations)
- can seek to any position in output sequence
- 64-bit nonce part of algorithm to prevent key-reuse

currently, no attacks better than brute-force attack known.

Algorithm

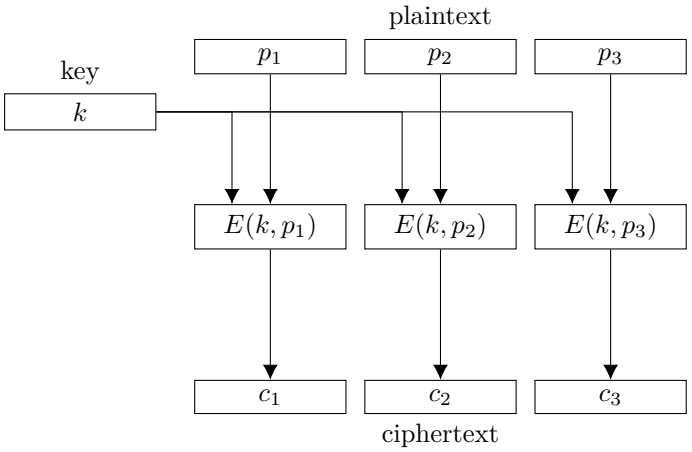
- Output in blocks of 16×32 bits
- internal state: 16×32 bits
  - initialized using key, nonce, and seek position
- State updated with XOR, 32-bit addition mod 2^{32}, and rotating 32 bit values
- Performs 20 rounds of XOR-add-rotate, each of which updates all values in state
- State added to original state to obtain output

4 Block Ciphers

Unlike stream ciphers, block ciphers have different encryption and decryption operations. A block cipher encrypts plaintext in fixed-length blocks

4.1 Design Considerations

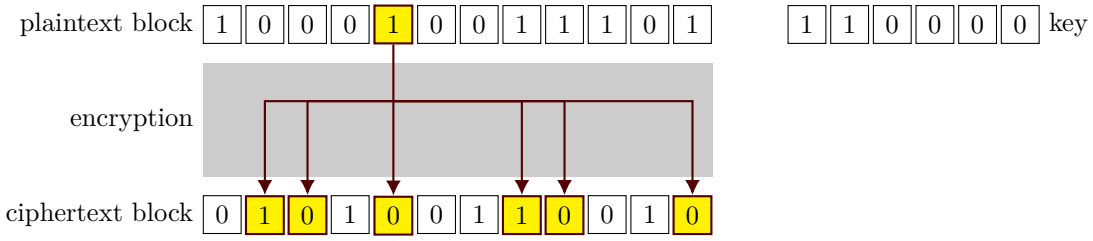
- Key Size
  - number of possible k-bit keys is 2^k
  - k must be sufficiently large to prevent brute-force attacks
- Block Size
  - too short $\rightarrow$ does not hide patterns in plaintext
    - e.g. $n = 8$ bits is 1 character
    - same as substitution cipher
  - too long – impractical, wasteful
- encryption must be invertible
  - different input blocks must be transformed into different output blocks
  - can be viewed as a permutation on all n-bit blocks
  - $(2^n)!$ possible permutations



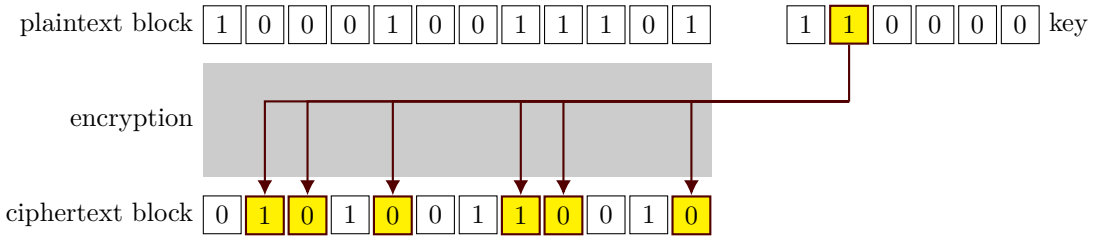
4.2 Secure Block Cipher

An n-bit block cipher is secure (for a computationally bounded attacker) if it is indistinguishable from a random permutation of n-bit blocks.

diffusion – each plaintext bit should affect the value of many ciphertext bits



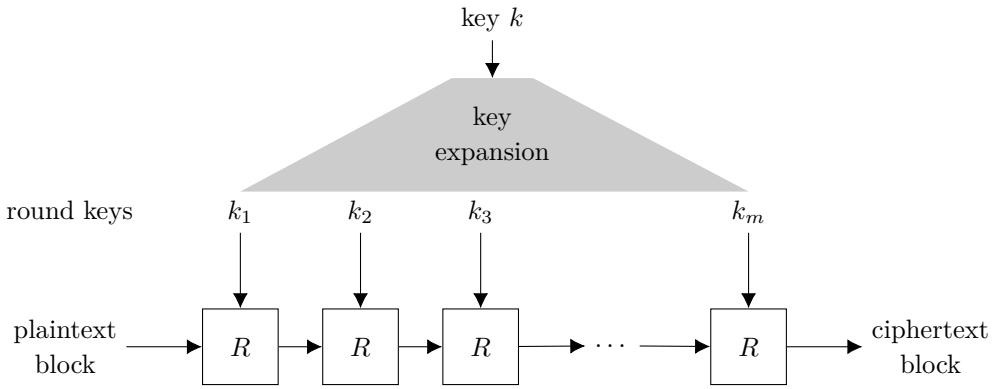
confusion – each bit of the ciphertext should depend on many bits of the key



4.3 Iterated Block Ciphers

Hard to design a single invertible function that satisfies diffusion and confusion. Use a round function

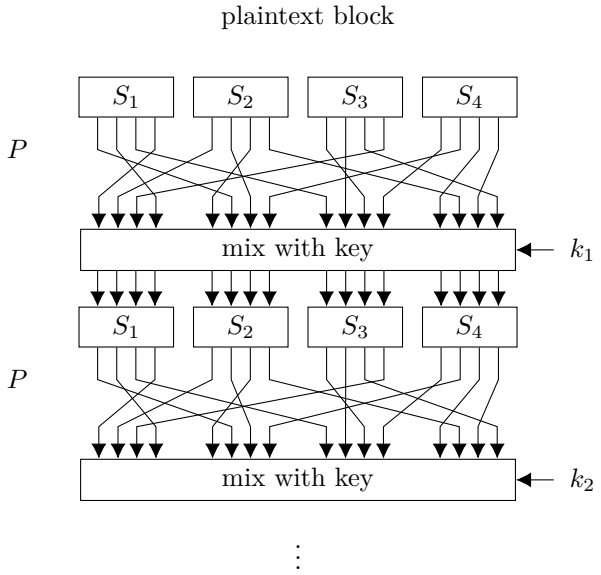
- R – round function
  - relatively weak transformation that introduces diffusion and confusion
  - by iterating, builds strong block cipher



4.4 Substitution-Permutation Ciphers

Common subtype of iterated block cipher, each round R consists of

- Substitution S
  - substitutes small block with another small block
  - ideally, changing one input bit changes half of output bits
- Permutation P
  - permutation of all bits



4.5 DES

Data Encryption Standard (DES)

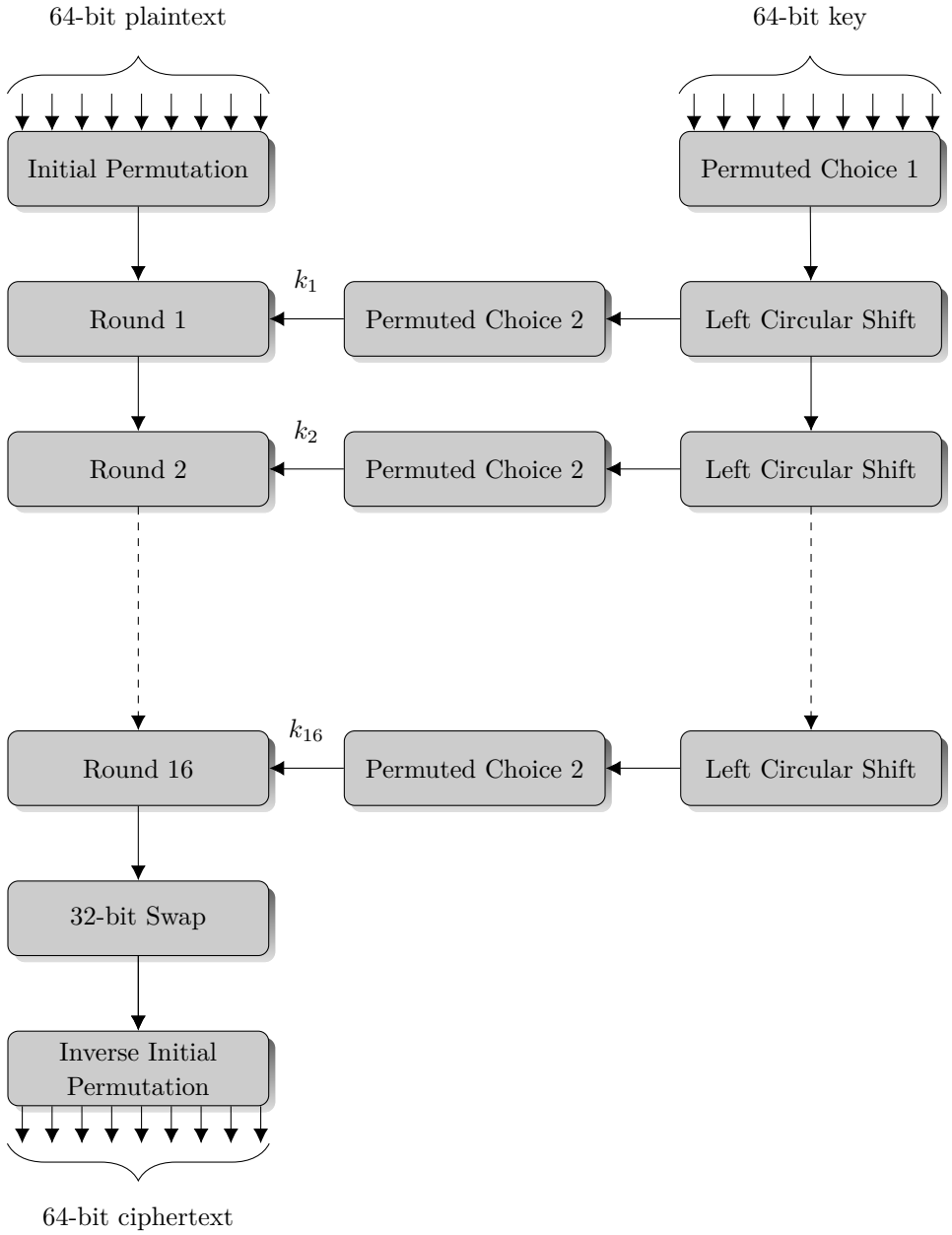
- block size – 64 bits
- key size – 56 bits
  - 56 bit random
  - 8 bit parity check
- iterated substitution cipher of 16 rounds
- initial permutation
  - no cryptographic significance
  - facilities loading blocks in and out of 8-bit hardware
- key permutation
  - discards parity bits
  - no cryptographic significance

Advantages

- relatively secure against cryptanalytic attacks – best attack in 2^{43} steps
- thoroughly studied and widely supported

Disadvantages

Vulnerable to brute-force attacks – 56-bit key $\rightarrow 2^{56}$ possible keys.



4.6 Feistel Network

Consists of encryption and decrpytion round

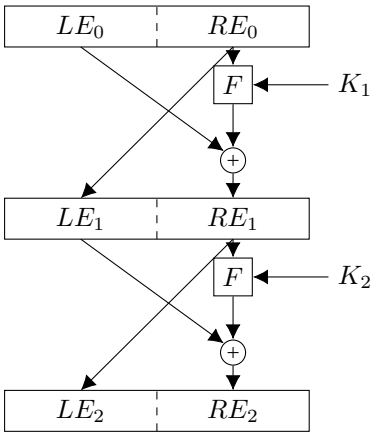
- Encryption round
  - input – block from previous round (or plaintext)
  - divide input in half – L_i and R_i
  - derive round key k_i from secret key (different each round)
  - output

$$L_{i+1} = R_i$$
$$R_{i+1} = L_i \oplus F(k_i, R_i)$$

- Decryption round

– we can invert encryption without inverting F

$$\begin{aligned} R_i &= L_{i+1} \\ L_i &= R_{i+1} \oplus F(k_i, L_{i+1}) \\ &= R_{i+1} \oplus F(k_i, R_i) \end{aligned}$$



4.7 AES

Advanced Encryption Standard (AES)

- Substitution-permutation
  - but **not** a Feistel network
- each round must be invertible for decrpytion
- key expansion and schedule – generates different round key each round
- number of rounds n depends on key size k

k	n
128	10
192	12
256	14

4.8 AES Round

- input
  - 128-bit state from previous round (or plaintext) as 4×4 byte matrix
  - 128-but round key from key schedule
- output – 128-bit state
- each round consists of multiple steps
  - AddRoundKey – xor round key to state
  - 128-but round key from key schedule
  - substitution and permutation
    - SubBytes
    - ShiftRows
    - MixColumns

SubBytes

- Each byte is replaced using an 8-bit substitution box (S-box)
  - defined using mathematical operations: multiplicative inverse over a finite field + affine transformation
- designed to resist cryptanalysis
  - minimize correlation to linear functions

– minimize difference propagation

ShiftRows

- Cyclically shifts 2nd, 3rd, and 4th rows left

row	shift
2nd	1
3rd	2
4th	3

- ensures the 4 bytes of each column are spread to 4 different columns → provides diffusion
  - without this step each input byte would only affect a single column

MixColumns

- Each column is multiplied by a fixed matrix
  - invertible linear transformation
- good mixing among bytes of each column → provides diffusion
  - in conjunction with ShiftRows, ensures each output bit depends on every input bit after a few rounds

4.9 AES Decryption

- each step is invertible
  - InvertMatrixColumns – multiply by matrix inverse
  - InvertShiftRows – shift rows cyclically to right
  - InvertSubBytes – invert affine transformation and multiplicative inverse
  - InvertAddRoundKey – xor round key to state
- Round keys are used in reverse order

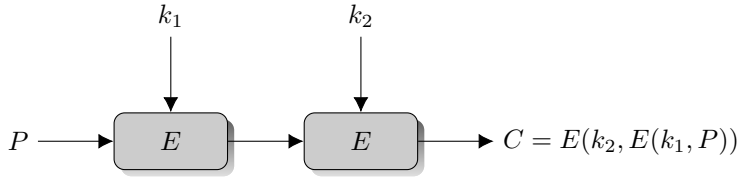
4.10 AES Performance and Security

- Operations on bytes and 32-bit words
  - most operations can be precomputed
- Supported by hardware – AES instruction set for CPUs
- very secure – best known attack takes 2^{126} steps, only 4x faster than brute-force attack

4.11 Multiple Encryption

Use same encryption algorithm multiple times, each time with a different key

2DES

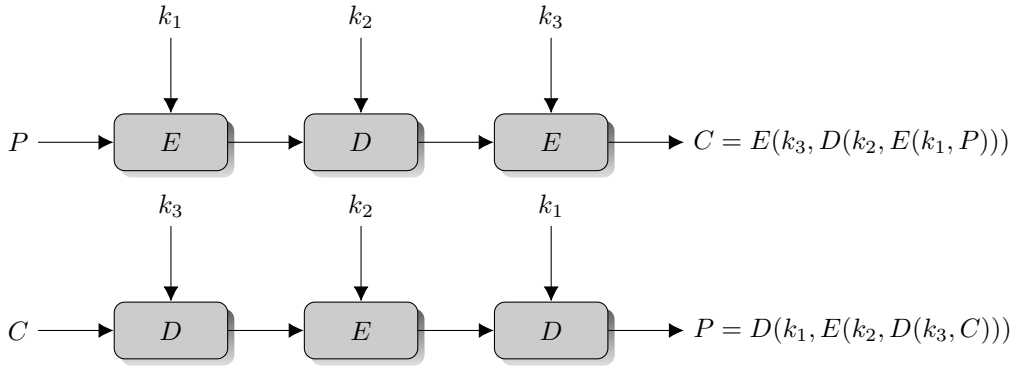


4.11.1 Meet in the Middle

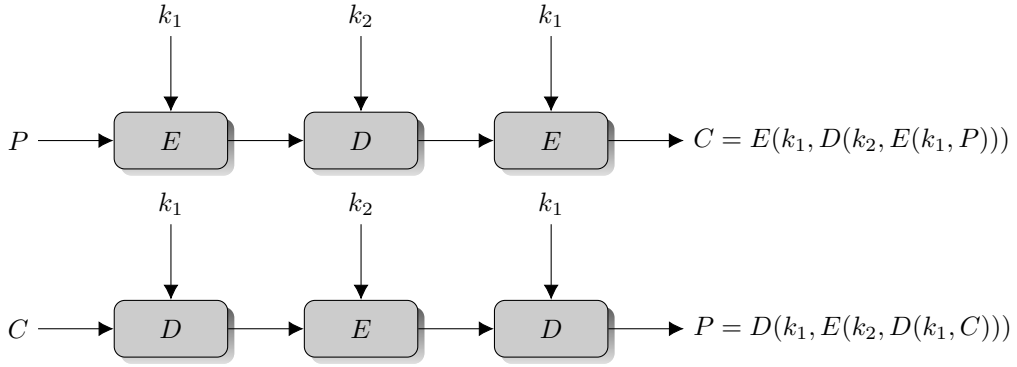
meet-in-the-middle attack – trade time for storage

- brute-force attack requires 2^{112} steps
- store $\sqrt{2^{112}} = 2^{56}$ values, $\approx 2^{56}$ steps
- generally, storing $2^{56-m} \rightarrow \approx 2^{56+m}$ steps

3DES Using 3 keys instead of 2. Naive implementation suffers same vulnerability to meet-in-the-middle attack as 2DES. Instead use EDE – Encryption-Decryption-Encryption



Above has 3 keys, but vulnerable to more sophisted MITM attack – effectively only 112-bit security. Taking $k_1 = k_3$ provides 80-bits of effective security.

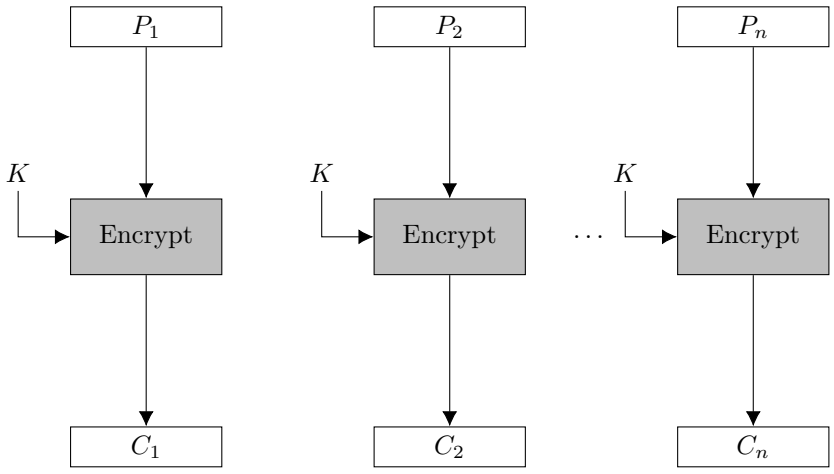


5 Block Cipher Modes of Operation

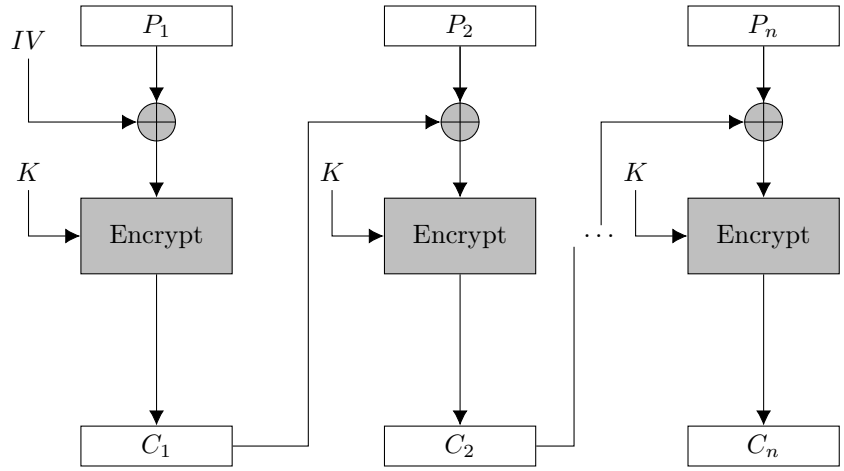
Orientation	Mode		Use
Block	ECB	Electronic Code Book	single block
	CBC	Cipher Block Chaining	commonly used
Stream	OFB	Output Feedback	no random access
	CFB	Cipher Feedback	self-synchronized stream cipher
	CTR	Counter Mode	very efficient, very commonly used

Mode	Advantages	Disadvantages
ECB	blocks can be encrypted/decrypted in parallel	identical plaintext → identical ciphertext
		attacker can rearrange or remove blocks from ciphertext
CBC	hides patterns in the plaintext	blocks cannot be encrypted in parallel
	blocks can be decrypted in parallel	attacker might be able to rearrange or remove blocks from ciphertext
		IV needs integrity protection
		attacker might be able to tamper with bits of the plaintext
OFB	bit errors do not propagate	blocks cannot be encrypted or decrypted in parallel
	pre-computation is possible	attacker can tamper with the bits of the plaintext
CFB	blocks can be decrypted in parallel	blocks cannot be encrypted in parallel
	self-synchronizing stream cipher	attacker might be able to tamper with the bits of the plaintext
		attacker might be able to rearrange or remove blocks
CTR	blocks can be encrypted and decrypted in parallel	attacker can tamper with bits of the plaintext
	bit errors do not propagate	
	pre-computation is possible	

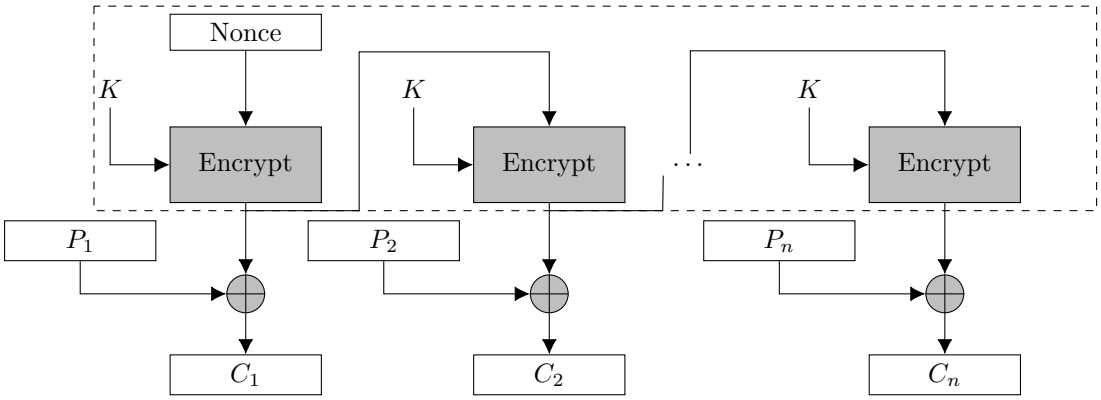
ECB



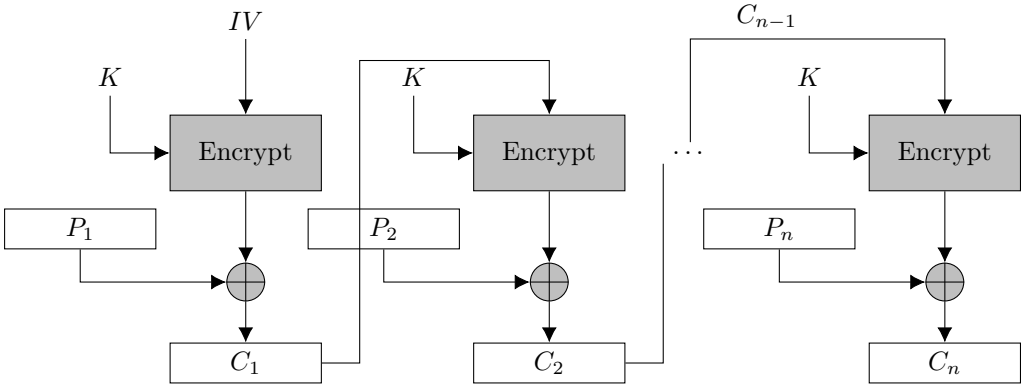
CBC



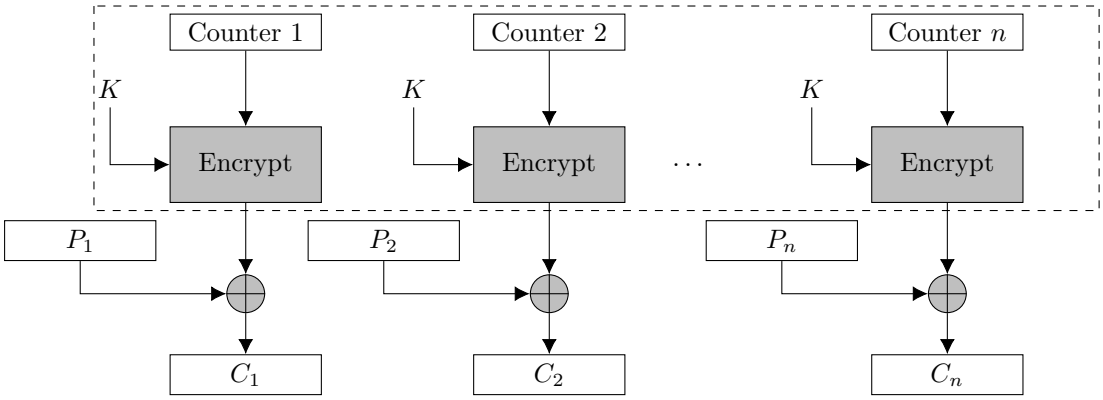
OFB



CFB



CTR



6 Public-Key Cryptography

public-key cryptography, also called asymmetric-key cryptography

Use a pair of keys, one public, one private. Solves

- public-key encryption → key exchange
- digital signature → non-repudiation

6.1 Public-Key Encryption

- everyone knows public key → sender can encrypt
- received knows private key → receiver can decrypt
- attacker does not know private key → cannot decrypt
- public key can be published

3 algorithms:

- key generation $G \to (PU, PR)$
- encryption – $E(PU, M) \to C$
  - takes public key PU and plaintext M and outputs ciphertext C
- encryption – $D(PR, C) \to P$
  - takes private key PR and ciphertext C and outputs plaintext P

Unlike symmetric-key, requires largest keys and is much slower

	Symmetric	Asymmetric
Typical Design	series of subs. and perms.	hard mathematical problems
Key	completely random	special structure, expensive to generate
Rec. Key Size	128 - 256 bits	2048 - 15360 bits
Performance	fast	slow

6.2 RSA

Choose two large primes p and q, set $n = pq$ and choose e such that $\gcd(e, \varphi(n)) = 1$. Set $d = e^{-1} \bmod \varphi(n)$, then

$$PU = (e, n)$$
$$PR = (d, n)$$

Encrypt plaintext M with $C = M^e \bmod n$ and decrypt with $M = C^d \bmod n$.

6.3 Security of RSA

Security comes from difficulty of determining $C^{1/e} \bmod n$ efficiently. Best known algorithm is to factor $n = pq$ and compute $e^{-1} \bmod \varphi(n)$. Integer factorization is assumed to be hard but this is unproved.

Very slow encryption, so commonly used to encrypt a secret key for use with symmetric-key encryption. Comparable symmetric key security (number of bits):

Symmetric	RSA
80	1024
128	3072
256	15360

6.4 ElGamal Encryption

Choose a large prime q, primitive root α of q, and $X \in \{1, 2, \ldots, q-1\}$. Set $Y = \alpha^x \bmod q$ and

$$PU = (q, \alpha, U)$$
$$PR = (q, \alpha, X)$$

Encryption:
Choose random $k \in \{0, 1, \ldots, q-2\}$ and set $K = y^k \bmod q$. Then return (C_1, C_2) where

$$C_1 = \alpha^k \bmod q$$
$$C_2 = KM \bmod q$$

Decryption:
Set $K = C_1^X \bmod q$, then $M = C_2 K^{-1} \bmod q$.

6.5 Security of ElGamal

Security comes from difficulty of discrete logarithm, widely believed to be computationally hard.

Recover X requires computing discrete-$\log_\alpha$ of Y mod$q$.

Recover k requires computing discrete-$\log_\alpha$ of C_1 mod$q$.

6.6 Elliptic Curve Cryptography

Elliptic Curve is a set of points (x, y) such that

$$y^2 = x^3 + ax + b$$

Binary operation $P + Q$: draw line through P and Q, find where it intersects the curve, call this R. Then $P + Q = -R$. Combined with a point at infinity (the identity), forms an abelian (commutative) group. Can naturally define $kP = \underbrace{P + P + \cdots + P}_{k-\text{times}}$ and so can implement ElGamal in a straightforward fashion.

160-bit ECC is comparable in security to 1024-bit RSA key.

7 Hash Functions

A hash function H maps a variable-length input to a fixed-length hash value. It must be

- efficient – computing $H(M)$ is easy
- one-way – finding an input for which the output is a given hash-value is hard
- collision-resistant – finding two inputs for which the hash-values are the same is hard
- pseudorandom

7.1 Security Requirements

Requirement	Definition
preimage resistance one-way property	given a hash value h, it is computationally infeasible to find an input y such that $H(y) = h$
second preimage resistance weak collision resistance	given input x, it is computationally infeasible to find y such that $x \neq y$ but $H(x) = H(y)$
collision resistance strong collision resistance	computationally infeasible to find any pair of inputs (x, y) such that $x \neq y$ but $H(x) = H(y)$; implies weak collision resistance

7.2 Brute-Force Attacks

Try random inputs until a collision is found. If output is m bits, then probability of success for a single try is 2^{-m}. Expected number of tries until success is 2^m.

Collision resistance attacks are a consequence of the birthday paradox:

Theorem 7.1 ▶ The Birthday Paradox

The probability that a collision occurs given a random map from $\{1, 2, \ldots, n\}$ to $\{1, 2, \ldots, m\}$ is

$$1 - \prod_{i=1}^{n-1} 1 - \frac{i}{m}$$

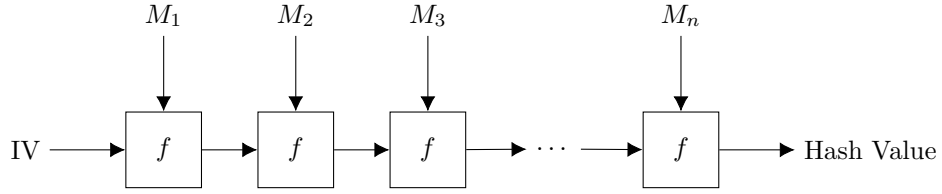
Since $1 - x \approx e^{-x}$, we have

$$1 - \prod_{i=1}^{n-1} 1 - \frac{i}{m} \approx 1 - e^{-\frac{n^2}{2m}}$$

so when $n = \Omega(\sqrt{m})$, a collision is likely.

7.3 Iterative Hash Function

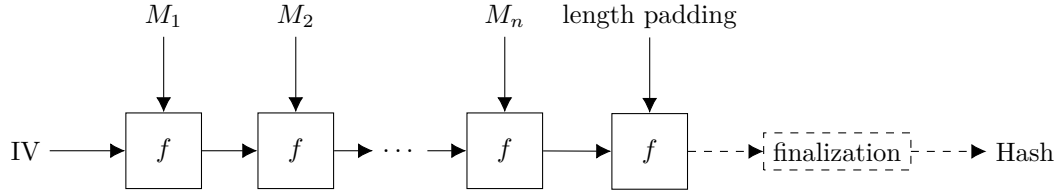
Divide input M into fixed-length blocks M_1, M_2, …, M_n. Then



- IV: initialization vector
- f: compression function
  - one-way and collision resistant
  - takes two fixed-length inputs, produces one fixed-length output

7.4 Merkle-Damgård Construction

Method to build cryptographic hash frunctions from compression functions.



length padding (Merkle-Damgård Strengthening) includes the length of the input as well as a fixed pattern

if f is collision resistant and proper length padding is used, hash function is collision resistant

7.5 CBC Hash Functions

Algorithm	Properties	Security
MD5	based on Merkle-Damgård compression func 4 rounds, 16 ops each 512-bit block 128-bit hash	not collision resistant cryptanalysis can break in 2^{18} steps less than 1 sec on average computer
SHA-1	160-bit hash	collision in 2^{65} steps
SHA-2	family: SHA-224, 256, 384, 512 output 224, 256, 384, 512 bit hash	similar to SHA-1, some weaknesses no practical attacks yet
SHA-3	sponge construction arbitrary output length	

Algorithm	Security
MD5	not secure at all
SHA-1	not secure
SHA-2	mostly secure
SHA-3	secure

8 Message Authentication

Term	Definition
content modification	content of a message is changed
sequence modification	sequence of messages is changed, including potentially deletion
timing modification	messages are delayed or repeated
masquerade	messages from a fraudulent source are inserted

8.1 Message Authentication Code

takes a secret key k and arbitrary-length input M and produces tag T

$\text{MAC}(k, M)$

- can be efficiently computed given k and M
- cannot be efficiently computed given only M
- looks like a pseudorandom function to attacker
- does not need to be invertible

proves authenticity and integrity

8.1.1 Brute-Force Attacks

Brute-force forging a tag depends on key length

- too short → high probability arbitrary tag matching modified message
  - n-bit tag, probability is 2^{-n}
- too long → consumes bandwidth

8.1.2 Key Search Attacks

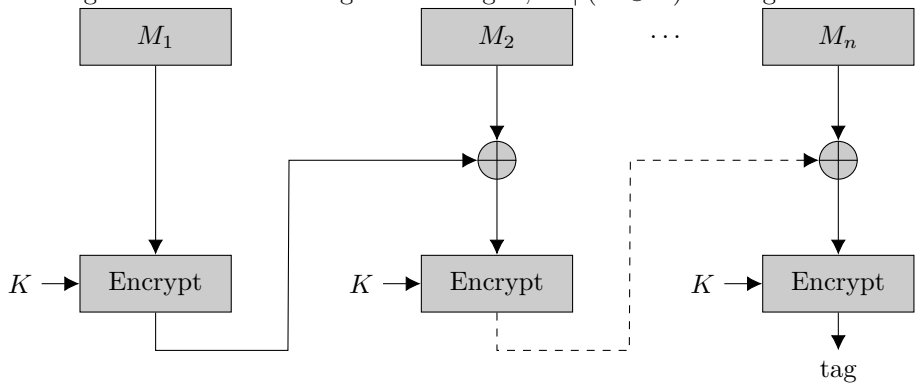
If key has length k, finding the right key takes 2^{k-1} steps on average

8.2 Block Cipher MAC

CBC MAC

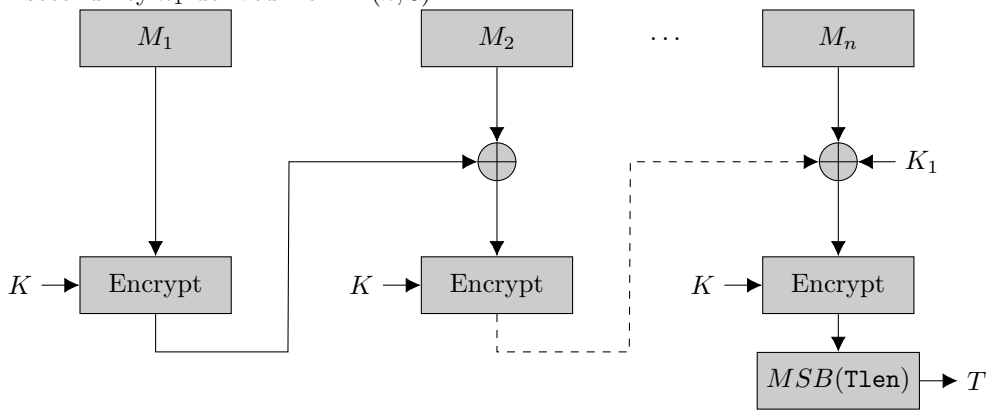
- use different key for CBC encryption and CBC-MAC.
- not secure for variable length messages

– given one-block message X and tag T, $X \mid (X \oplus T)$ has tag T



Cipher-based MAC (CMAC)

- thwarts forgery for variable-length messages
- second key k_1 derived from $E(k, 0)$



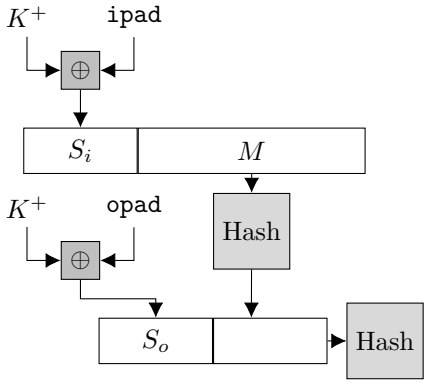
8.3 Hash MAC (HMAC)

- provably secure if hash is pseudorandom
- works with any hash
  - more efficient with iterative hash
- widely used

Structure

- b — block size of hash
- `IV` initial value of hash function
- inputs
  - $Y_0, Y_1, \ldots, Y_{L-1}$ — message
  - K^+ — 0-padded key
  - `ipad` — 00110110 repeated
  - `opad` — 01011100 repeated
- output HMAC(K, M)

$$H(k^+ \oplus \texttt{opad} \mid H(k^+ \oplus \texttt{ipad} \mid M))$$



Precomputation

- precompute

$$f\big(\texttt{IV}, k^+ \oplus \texttt{ipad}\big)$$
$$f\big(\texttt{IV}, k^+ \oplus \texttt{opad}\big)$$

- for L input blocks, requires only $L + 1$ compressions

8.4 Authenticated Encryption

encryption system that provides both confidentiality and integrity

- motivation
  - widely-used cryptographic primitives are (almost always) secure
  - secure encryption + secure authentication $\not\Rightarrow$ secure combination
  - some security protocols have used cryptographic primitives in an insecure way
- approaches
  - authentication then encryption (e.g. SSL/TLS)
  - encryption then authentication (e.g. IPSec)
  - independently encrypt and authenticat (e.g. SSH)

8.4.1 Counter with CBC-MAC (CCM)

- Encryption — based on CTR block cipher mode
- Authentication — based on CBC-MAC message authentication
- Combination — authenticate, then encrypt
  - compute CBC-MAC of message, a nonce, and associated data
  - encrypt message and authentication tag in CTR mode

8.4.2 Galois/Counter Mode (GCM)

- Encryption — based on CTR block cipher mode
- Authentication $\text{GHASH}_H(X)$
  - inputs — hash key H, message blocks X_1, X_2, …, X_m (128-bit blocks)
  - outputs — $(X_1 \cdot H^m) \oplus (X_2 \cdot H^{m-1}) \oplus \ldots \oplus (X_{m-1} \cdot H^2) \oplus (X_m \cdot H)$
    - $\cdot$ — special multiplication for 128-bit numbers
    - H^m, H^{m-1}, …, H^2 can be precomputed in parallel
- Combination — encrypt, then authenticate
  - authentication includes message length and associated data
- very efficient and parallelizable (widely used)

9 Digital Signatures

Motivation

- Message authentication does not protect sender/receiver from each other
  - receiver can forge a message and claim it is from sender
  - sender can deny sending a message and claim it was forged by receiver
- digital signature ≈ authentication + non-repudiation
  - integrity and authenticity protection; non-repudiation
  - similar to traditional signature – signee cannot deny signing document
  - in many countries, digital signatures have legal significance

Like public-key cryptography

- signee knows private key → can sign
- verifier knows public key → can verify
  - public key published so anyone can verify
- attacker (forger) does not know private key → cannot sign

Algorithms

- key generation $G()$
  - randomized algo outputs (PU, PR)
- signature $\textsc{Sign}(PR, M)$
  - takes PR and message M and outputs signature S
- verification $\textsc{Verify}(PU, M, S)$
  - takes PU, message M, and S, and outputs accept/reject

Public-Key Encryption

- key generation $G()$
  - randomized algo outputs (PU, PR)
- signature $\textsc{Decryption}(PR, C)$
  - takes PR and ciphertext C and outputs plaintext M
- verification $\textsc{Encryption}(PU, M, S)$
  - takes PU and plaintext M and outputs ciphertext C

9.1 Hash-then-Sign

Attacker can forge signature for random messages

- pick aribtrary X and use as signature
  - signature of $E(PU, X)$ is X

Sign cryptographic hash of message. Advantages

- compatibility — most public-key encryption algos take fixed-length input
- efficiency — signature shorter and faster to compute
- security – prevents existential forgery (attacker cannot compute forged message for arbitrary signature using only public-key)

9.2 RSA, DSA, ECDSA

RSA

- Very widely used with SHA-256 (and other versions of SHA)
- Standard: PKCS #1
  - RSASSA-PKCS1-v1_5
  - RSASSA-PSS
    - PSS = Probabilistic Signature Scheme
      - adds randmized padding (salt) to message
    - provably secure (if RSA is secure)

DSA

- Digital Signature Algorithm
  - designed for signature, cannot be used for encryption

- efficient variant of the ElGamal signature scheme (much smaller signatures, modular arithmetic operations with lower moduli)

ECDSA

- Elliptic Curve Digital Signature algorithms
  - based on elliptic curve cryptography
  - shorter keys and increased efficiency

10 Key Distribution

10.1 Key Freshness

- Secret keys may become insecure when used for a long time
- key freshness requirement — renew secret key frequently

10.2 Secret-Key Hierarchy

- session key
  - renewed frequently
  - used to encrypt and authenticate data
- master key
  - renewed infrequently
  - used to distribute session keys

10.3 Approaches

- decentralized
  - master key required for each pair
    - then any pair can agree on session keys easily
  - may work for small local networks
  - does not scale well – requires $\binom{n}{2} = (n)(n-1)/2 = \mathcal{O}(n^2)$ master keys
- key distribution center (KDC)
  - acts as trusted third party
  - each party shares their secret master key with KDC
  - scales well – requires n master keys

10.4 Kerberos

Extended Needham-Schroeder Protocol — protocol that aims to establish a session key between two parties on a network

uses nonces

Kerberos Network Authentication Protocol

- nodes can communicate over a non-secure network and to prove their identities to each other
- similar to Extended Needham-Schroeder Protocol but uses timestamp instead of nonce

11 Public-Key Distribution

Difficult to distribute keys manually through secure channel.

Using public-key cryptography → parties do not need a shared secret key

11.1 Diffie-Hellman Key Exchange

- Security depends on hardness of discrete logarithms
- ElGamal based on Diffie-Hellman
- Secure against eavesdropping (passive attack)
- Elliptic Curve D-H (ECDH) — same principle, more efficient

11.2 Station-to-Station Protocol

- Assume A knows PU_B and B knows PU_A
- sign messages with public keys
- provides key and entity authentication
- secure against man-in-the-middle attacks

11.3 Distribution of Public Keys

- Public Announcement
  - announce publicly on broadcast channel
  - weakness — anyone can forge announcement
    - impersonated entity can detect and inform
    - forger can impersonate entity until detection
- Public-Key Authority
  - everyone knows public key of authority
  - authority maintains directory of public keys
  - weakness
    - authority is single point of failure
    - authority must be online

11.4 Digital Certificates

- enable participants to securely exchange keys without contacting authority
- certificate = $\underbrace{\text{owner's name, public key, timestamp, } \ldots}_{\text{signed by certificate authority}}$
- requirements
  1. any participant can read certificate to determine name/public key of owner
  2. only certificate authority can create certificates
  3. any participant can verify certificate originated from authority
  4. any participant can verify certificate is recent

X.509 Certificate — ITU-T standard for public-key certificates and related functions

11.5 Certificate Authorities in Practice

- OS and browsers come with list of trusted certificate authorities
  - CAs trusted by developers
  - user can add or remove CAs to list
- types of CAs
  - commercial — charges fee for issuing certificate
  - governmental
  - private non-profit (e.g. CAcert)

Use certificate chains to establish trust when two entities do not share a common CA.

11.6 Certificate Revocation

- revocation necessary if
  - private key is compromised

        * attacker can use their certificate to impersonate the entity
    - owner no longer certified
    - authority's certificate is compromised
- each CA maintains a Certificate Revocation List (CRL)
    - signed and published by the CA
    - when checking validity of certificate, first check if it is in CRL
    - for efficiency, clients cache list → revoked cert may be accepted until cache expires

Glossary

accountability actions can be provably traced back to an entity

Advanced Encryption Standard encryption standard consisting of invertible rounds in which a different key is generated each round

affine cipher cipher $E(x) = (ax + b) \bmod m$

asymmetric-key cryptography see public-key cryptography

attacker model what the attacker can do, what they know, and what they want to achieve

authenticated encryption encryption system that provides both confidentiality and integrity

authenticity information comes from verified and trusted sources (e.g., user authentication)

availability information and system functionality is available to authorized entities

birthday attack for an m-bit hash, trying $\sqrt{2^m} = 2^{m/2}$ inputs until a collision is found

birthday paradox probability that two people share a birthday in a group of N is

$$\prod_{i=1}^{N-1} \frac{365 - i}{365} \approx e^{-\frac{N^2}{730}}$$

which is approximately 0.5 at $N = 23$. More generally, if we sample N values from a set of M elements, a collision is likely if $N > \sqrt{M}$

block cipher a cipher in which plaintext is encrypted in fixed-size blocks and decryption is a different operation than encryption

block cipher mode of operation technique for enhancing the effect of a cryptographic algorithm or adapting the algorithm for an application

brute-force attack attack model in which every possible key is tried on a given ciphertext until the original plaintext is recovered

caesar cipher a cipher in which each letter of the plaintext is replaced by a letter some fixed number of positions down the alphabet

CBC-MAC MAC based on Cipher Block Chaining (CBC) mode of operation; uses different keys for CBC encryption and CBC-MAC auth; not secure for variable-length messages

cerificate chain chain of certificate authoritys used to verify a public-key certificate should two entities not share a common certificate authority

certificate authority entity that issues public-key certificates

ChaCha20 more secure and efficient variant of Salsa20

chosen ciphertext attack attack model in which one or more *chosen* plaintext-cipher pairs is known

chosen plaintext attack attack model in which the attacker can obtain the ciphertext for any plaintext

chosen text attack attack model in which the attacker can obtain the ciphertext for any plaintext *and* one or more chosen plaintext-cipher pairs is known

Cipher Block Chaining Block-oriented Block Cipher Mode that allows for parallel decryption (but not encryption) of blocks and hides patterns in the plaintext. Possible for attacker to rearrange or remove blocks from the ciphertext or tamper with bits of the plaintext and IV must have integrity protection. Application — general-purpose block-oriented transmission

Cipher Feedback Stream-oriented Block Cipher Mode with self-synchronizing stream cipher in which blocks can be decrypted (but not encrypted) in parallel. Attacker can potentially tamper with bits of the plaintext or rearrange or remove blocks. Application — general-purpose stream-oriented transmission

cipher-based MAC MAC that thwarts forgery for variable-length messages

ciphertext only attack attack model in which only the algorithms used and the ciphertext are known

collision resistance see strong collision resistance

computersecurity protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability, and confidentiality of information system resources (includes hardware, software, firmware, information/data, and telecommunications)

confidentiality information is not available to unauthorized entities

confusion each bit of the ciphertext should depend on many bits of the key

content modification communication channel attack in which the content of a message is changed

Counter Mode Stream-oriented Block Cipher Mode in which blocks can be encrypted and decrypted in parallel, pre-computation is possible, and bit errors do not propagate. Attacker can potentially tamper with the bits of the plaintext. Application — general-purpose transmission

counter with CBC-MAC encryption based on Counter Mode (CTR); authentication based on CBC-MAC; authenticate, then encrypt

cryptanalytic attack attack model in which the attacker relies on the nature of the algorithm and knowledge of the general characteristics of the plaintext

cryptographic hash function pseudorandom, efficient, collision-resistant and one-way function that maps a variable-length input to a fixed-length hash value

Data Encryption Standard federally approved encryption standard which uses an iterated substitution-permutation cipher of 16 rounds with 64-bit block size and 56-bit key size

data integrity information cannot be modified in an unauthorized and undetected way

decentralized secret-key system in which each pair of communication parties shares a secret master key; easy to set up but does not scale well, as it requires $\binom{n}{2} = (n)(n-1)/2$ keys

denial of service attack against availability

Diffie-Hellman Key Exchange the first public-key algorithm; security is based on the difficulty of computing discrete logarithms

diffusion each plaintext bit should affect the value of many ciphertext bits

digital signature a mathematical scheme to provide both message authenticity and non-repudiation

digital signature algorithm NIST algorithm designed for signature; cannot be used for encryption; efficient variant of ElGamal Encryption with much smaller signatures and modular arithmetic operations with smaller moduli

digital signature standard Federal Information Processing Standard (FIPS) 186, introducted in 1993; latest version includes RSA, DSA, elliptic-curve signatures

Double DES DES utilizing 2 56-bits. Effective security is only 80-bits

Electronic Code Book Block-oriented Block Cipher Mode that allows for parallel encryption and decryption of blocks, but in which identical plaintext blocks result in identical ciphertext blocks and allows for attackers to rearrange or remove blocks from ciphertext. Application — secure transmission of a single block

ElGamal Encryption encryption scheme which depends on the difficulty of computing discrete logarithms

Elliptic Curve Cryptography approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields

elliptic curve digital signature algorithm signature algorithm based on elliptic curve cryptography with shorter keys and increased efficiency

Extended Needham-Schroeder Protocol protocol that aims to establish a session key between two parties on a network

Galois/Counter Mode encryption based on Counter Mode (CTR); authentication – $\text{GHASH}_H(X)$ takes hash key H and 128-bit message blocks $X = X_1, X_2, \ldots, X_m$ and outputs

$$(X_1 \cdot H^m) \oplus (X_2 \cdot H^{m-1}) \oplus \cdots \oplus (X_{m-1} \cdot H^2) \oplus (X_m \cdot H)$$

hash-based MAC MAC that uses a hash function; provably secure if the hash is pseudorandom; more efficient with iterative hash function; used in IPSsec and SSL/TLS protocols

hash-then-sign sign a cryptographic hash of the message; compatibile with most public-key encryption algorithms, efficient, and prevents existential forgery

integrity information and system functionality cannot be altered by unauthorized entities

iterative hash function hash in which the input is divided into fixed-length blocks and each block is hashed

Kerberos Network Authentication Protocol protocol based on the Extended Needham-Schroeder Protocol that allows communication over non-secure network; uses timestamps instead of nonces

Kerckhoffs's Principle a cryptographic system should be secure, even if all of its details, except for the key, are publicly known

key distribution center a centralized key distributor, acting as a trusted third party, shares a secret master key with each communication party; scales well, requiring only n keys, but must trust third party

key-reuse problem security flaw in which attacker can decipher plaintext given multiple ciphertexts encrypted with the same key

known plaintext attack attack model in which one or more plaintext-cipher pairs is known

man-in-the-middle attack attack in which the attacker secretly relays and possibly alters communications between parties who believe they are directly communicating with each other

masquerade communication channel attack in which messages from a fraudulent source are inserted

master key a key that is renewed infrequently, used to distribute session keys

MD5 Message-Digest Algorithm hash function based on Merkle-Damgård Construction consisting of four rounds, each consisting of 16 operations, with 512-bit block length and 128-bit hash length; not collision resistant – can be broken in $2^{18} = 262144$ steps, less than a second on an average computer

meet-in-the-middle attack a known plaintext attack in which attacker stores intermediate values from encryptions and decryptions to reduce the time necessary to brute-force the decryption keys, effectively trading off time for storage

Merkle-Damgård Construction general method for building a cryptographic hash function from a collision-resistant, one-way compression function; collision-resistant with sufficiently long padding

message authentication code takes a secret key K and an arbitrary-length input M and produces tag T

non-cryptographic hash function computationally inexpensive but generally insecure hash function that can be used for error detection and error correction

non-repudiation see accountability

nonce number used once

one-time pad perfect security in which a single-use encryption key at least as long as the plaintext is chosen randomly and used to encrypt only a single message

one-way property hash with property that, given a hash value h, it is computationally infeasible to find an input y such that $H(y) = h$

Output Feedback Stream-oriented Block Cipher Mode in which bit errors do not propagate and pre-computation is possible, but blocks cannot be encrypted or decrypted in parallel and attacker can tamper with bits of the plaintext. Application — stream-oriented transmission over noisy channel

perfect security attacker gains no information about the plaintext from observing the ciphertext, formally,

$$\mathbb{P}(P = p) = \mathbb{P}(P = p \mid E(K, P) = c)$$

i.e., that the plaintext and ciphertext are independent

PKCS #1 an RSA signature published by RSA Laboratories republished as RFC 3447; older standard of RSASSA-PKCS1-v1_5

preimage resistance see one-way property

privacy assures that individuals have control or influence over information related to them

pseudorandom number generator takes fixed-length seed and generates a sequence of bits using a deterministic algorithm

public-key certificate electronic document used to prove the ownership of a public key

public-key cryptography using a pair of keys – one private and one public

public-key cryptography key distribution one communication party needs the public key of the other

RC4 (Rivest Cipher 4) stream cipher with variable key length and which uses byte-oriented operations. No longer in use

RSA Cryptosystem encryption scheme which depends on the difficulty of factoring large numbers

RSA signature apply RSA encryption to the hash of the message and send both the encrypted message and hash — receiver decrypts message with public key and applies same hash function and verifies the the hash is the same; commonly used with SHA-256

RSASSA-PSS probabilistic signature scheme (PSS) form of RSA encryption which adds a salt; provably secure assuming RSA is secure

Salsa20 fixed-length key stream cipher that uses 32-bit operations and which can seek to any position in output sequence. 64-bit nonce is part of the algorithm to mitigate key-reuse problem

salt randomized padding added to a message

second preimage resistance see weak collision resistance

secure block cipher a block cipher that is indistinguishable from a random permutation of the blocks (for a computationally bounded attacker)

security by obscurity providing security by keeping the design or implementation of a system secret

security through minority providing security by using software products that are not widely adopted

semantic security attacker advantage for any efficiently computable guess is negligible over random guessing

sequence modification communication channel attack in which the sequence of messages is changed, including potential deletion of messages

session key a frequently renewed key used to encrypt and authenticate data

SHA-1 160-bit hash with Merkle-Damgård Construction; collision can be found in 2^{65} steps

SHA-2 family of functions: SHA-224, 256, 384, and 512, producing 224, 256, 384, and 512-bit ouputs, respectively; same underlying structure and operations as SHA-1; some weaknesses have been found

SHA-3 hash function to replace SHA-2, uses Sponge Construction; output length can be arbitrary

Sponge Construction hash function that can take an input stream of arbitrary length and return an output stream of any desired length; data is "absorbed" into the sponge and the result is "squeezed" out

Station-to-Station Protocol cryptographic key agreement scheme based on Diffie-Hellman Key Exchange; provides key and entity authentication and security against man-in-the-middle attacks

stream cipher takes fixed-length seed and uses a PRNG to produce sequence of bits as long as the plaintext then encrypts with XOR

strong collision resistance hash with property that it is computationally infeasible to find any pair of inputs (x, y) such that $x \neq y$ but $H(x) = H(y)$; implies weak collision resistance

substitution cipher permutation over the alphabet

system integrity system performs its intended function

timing modification communication channel attack in which messages are delayed or repeated

Triple DES DES utilizing 3 56-bit keys. Effective security is only 112-bits

weak collision resistance hash with property that, given input x, it is computationally infeasible to find y such that $x \neq y$ but $H(x) = H(y)$

weakest link principle that the defender needs to find and fix all vulnerabilities, but attacker needs to find only a single vulnerability

X.509 Certificate ITU-T standard for public-key certificates and related functions

Acronyms

2DES Double DES

3DES Triple DES

AES Advanced Encryption Standard

CBC Cipher Block Chaining

CCA chosen ciphertext attack

CCM counter with CBC-MAC

CFB Cipher Feedback

CIA confidentiality, integrity, and availability

CMAC cipher-based MAC

COA ciphertext only attack

CPA chosen plaintext attack

CTA chosen text attack

CTR Counter Mode

DES Data Encryption Standard

DoS denial of service

DSA digital signature algorithm

ECB Electronic Code Book

ECDSA elliptic curve digital signature algorithm

FIPS Federal Information Processing Standard

HMAC hash-based MAC

KDC key distribution center

KPA known plaintext attack

MAC message authentication code

MD5 MD5 Message-Digest Algorithm

MITM man-in-the-middle attack; sometimes also meet-in-the-middle attack

OFB Output Feedback

PRNG pseudorandom number generator

PSS probabilistic signature scheme

RSA Rivest, Shamir, Adleman RSA Cryptosystem

SHA Secure Hash Algorithm SHA-1, SHA-2, SHA-3