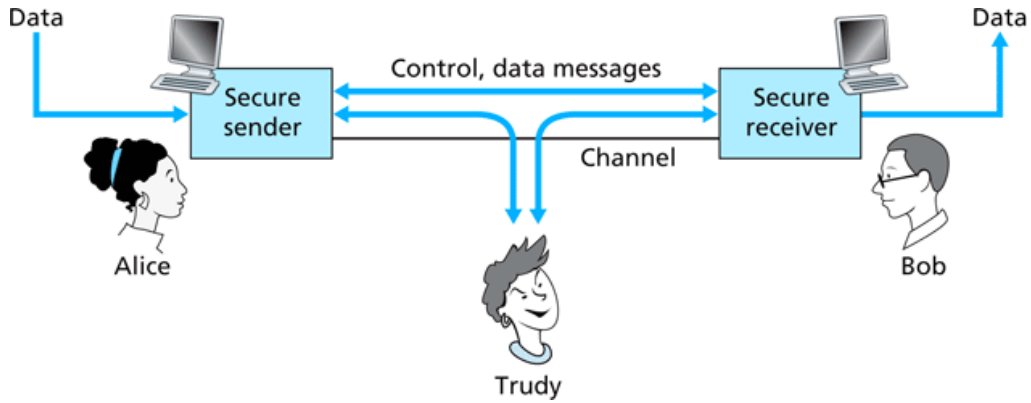
Network Security

Daniel Zappala

CS 360 Internet Programming
Brigham Young University

Network Security

Attacks



- intruder can overhear, modify, insert, or delete messages
- packet sniffing
  - overhear packets sent on the link
  - particularly useful on wireless links
- IP spoofing
  - nothing prevents a host from sending a packet with any IP address
- man-in-the-middle
  - insert a malicious node into the conversation between two hosts
  - can sniff, inject, modify, or delete packets

Denial of Service

- denial of service attack: render a computer unusable by legitimate users
  - vulnerability attack: send crafted messages to stop a service or crash a host
  - bandwidth flooding: send so many packets that the network at a server gets clogged
  - connection flooding: establish a large number of TCP connections at a server
- DDoS: distributed DoS, much harder to detect and defend against

Critical Infrastructure is Vulnerable

- DNS: bandwidth flooding attack
  - flood the DNS root servers with pings
  - carried out Oct 21, 2002 using a botnet
  - many root servers screened out the traffic
  - caching eliminates much of the danger
- other possible DNS attacks
  - flood TLD servers with queries
  - send bogus DNS replies
  - DNS poisoning: send bogus replies to a DNS server
  - send a lot of queries to a server using a spoofed source IP address (reflection attack)

Your Servers are Vulernable

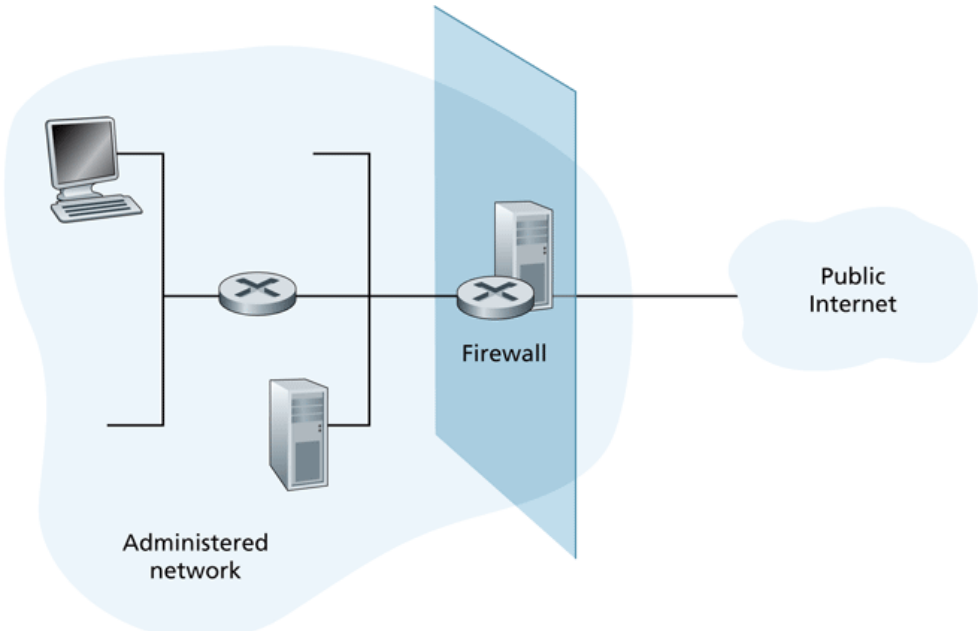
- port scanning: determine which ports are open on a host
- check open ports in case a server with a known security flaw is running
  - e.g. Microsoft SQL Server on port 1434 vulnerable to buffer overflow, exploited by the Slammer worm in 2003-2004
- many port scanners available, e.g. nmap

How Did the Internet Get This way?

- *The Design Philosophy of the DARPA Internet Protocols*, David Clark, Proceedings of ACM SIGCOMM 1988, pp. 106–114.
  - primary goal: interoperability among existing networks
  - secondary goals: fault tolerance, multiple transport protocols, minimum assumptions about network capabilities
  - additional goals: distributed management, cost effective, low effort for host attachment, accountability
- no mention of security: assumed that network participants were trustworthy

Firewalls and Intrusion Detection

Firewalls



- provide a gateway where traffic is checked before entering or exiting an organization
- only authorized traffic is allowed to pass

Packet Filter

action	source address	dest address	protocol	source port	dest port	flag bit
allow	222.22/16	outside of 222.22/16	TCP	> 1023	80	any
allow	outside of 222.22/16	222.22/16	TCP	80	> 1023	ACK
allow	222.22/16	outside of 222.22/16	UDP	> 1023	53	—
allow	outside of 222.22/16	222.22/16	UDP	53	> 1023	—
deny	all	all	all	all	all	all

- based on source and destination addresses, protocol type, source and destination ports, TCP flags, ICMP message type
- first two rules allow internal users to surf the web
- second two rules allow DNS traffic to enter and leave the network

Stateful Packet Filter

- traditional packet filter: examine each packet individually
- stateful packet filter: track TCP connections
  - ensures that packets allowed by the filter must be part of an active connection
  - prevents an attacker from injecting malformed packets that happen to meet a filter rule

Intrusion Detection Systems

- protect a network from attacks
  - a general Intrusion Detection System examines packet contents for attack signatures and generates appropriate alerts
  - an Intrusion Prevention System will also filter out the suspicious traffic
- can detect network mapping, port scans, TCP stack scans, DoS bandwidth flooding attacks, worms, viruses, OS vulnerability attacks, application vulnerability attacks

IDS Types

- signature-based system
  - maintains a database of attack signatures, including standard filter fields and strings found in the packet payload
  - crafted by people who investigate attacks, after they have been observed on the Internet
  - must compare every incoming packet to the list of signatures – requires very high-speed processing
  - snort: open source, comes with a large signature database that is constantly maintained
- anomaly-based system
  - observes traffic and examines patterns
  - anomalies, such as a burst of ICMP traffic or a large number of incoming or outgoing connections trigger a response
  - very challenging to distinguish between normal traffic and unusual traffic

Security Properties

- confidentiality
  - only the sender and receiver should be able to understand the contents of the message
  - may also want more general confidentiality – obscure the fact that you are talking with someone and the pattern
- integrity
  - ensure that communication is not altered in transit
- authentication
  - confirm the identity of the other party
- operational security/availability
  - ensure that services are not disrupted

Confidentiality

Confidentiality

- commonly use three characters:
  - Alice and Bob: want to be able to communicate securely with each other
  - Trudy: would like to attack using man-in-the-middle techniques
- use cryptography to achieve confidentiality
  - messages sent over a public channel
  - plaintext: the message Alice wants to share with Bob
  - ciphertext: the encrypted form of the plaintext
  - K_A a key used by Alice to encrypt or decrypt

Cryptography

- symmetric key cryptography
  - Alice and Bob share a secret key
  - encrypt and decrypt messages with the same key
- public key cryptography
  - Alice and Bob each assigned a public key and a private key
  - encrypt a message in the other's public key
  - private key decrypts the message

Symmetric Key Encryption

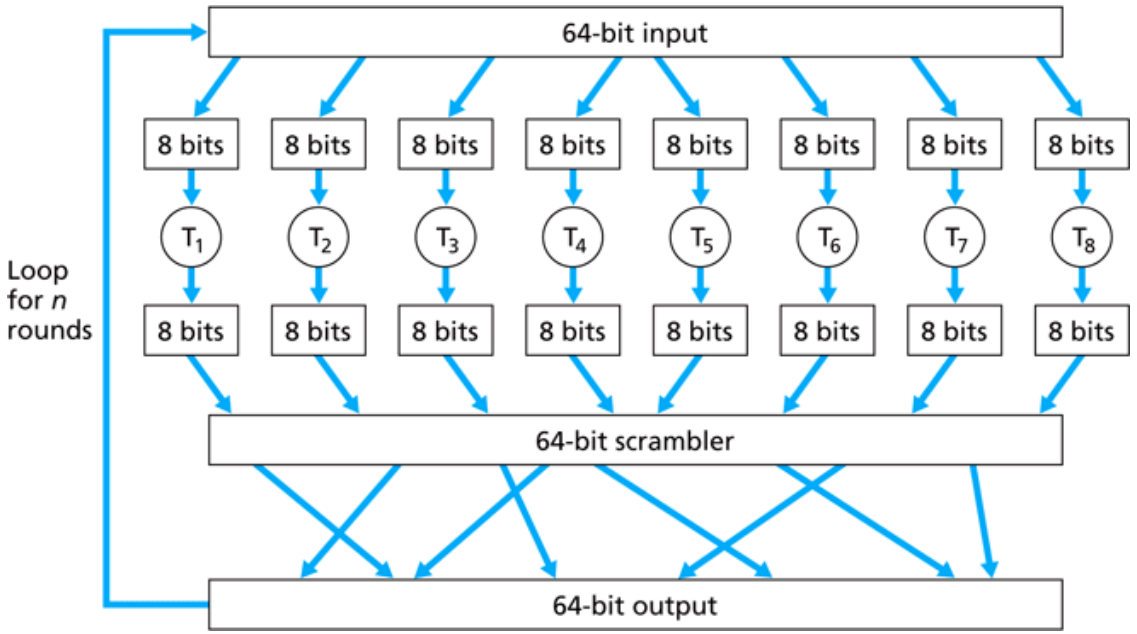
Block Cipher

- used in PGP, SSL, IPsec
- divide message into blocks of k bits
- map each block of plaintext to ciphertext
  - plaintext: 010110001111
  - ciphertext: 101000111001
- 2^3 possible inputs, $8! = 40,320$ permutations
- typically use blocks of 64 bits or larger

input	output	input	output
000	110	100	011
001	111	101	010
010	101	110	000
011	100	111	001

Implementation of a Block Cipher

- keeping a full table of 2^{64} mappings is infeasible
- instead use a function to simulate randomly permuted tables
  - break 64-bit blocks into 8-bit blocks
  - process by an 8-bit table and reassemble
  - scramble the order of the bits
  - loop for n rounds to make each input bit affect most of the output bits

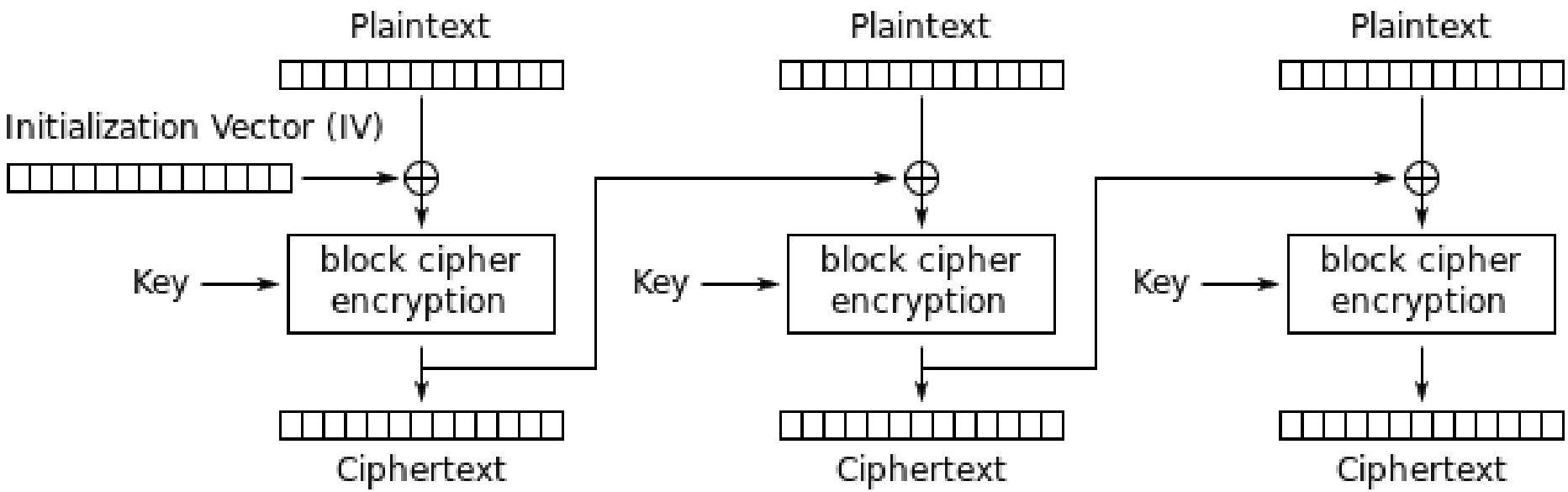


Block Cipher Details

- Advanced Encryption Standard (AES): 128-bit blocks, 128-, 192-, or 256-bit key
  - key length determines table mappings and permutations
- brute-force attacks
  - cycle through all keys: 2^n possible keys for a key length n
  - old DES standard (64-bit blocks, 56-bit key) cracked in 6.4 days using $10,000 of hardware, March 2007
  - a system that can crack DES in one second would take 149 trillion years to crack AES

Cipher Block Chaining

- encrypt a series of blocks



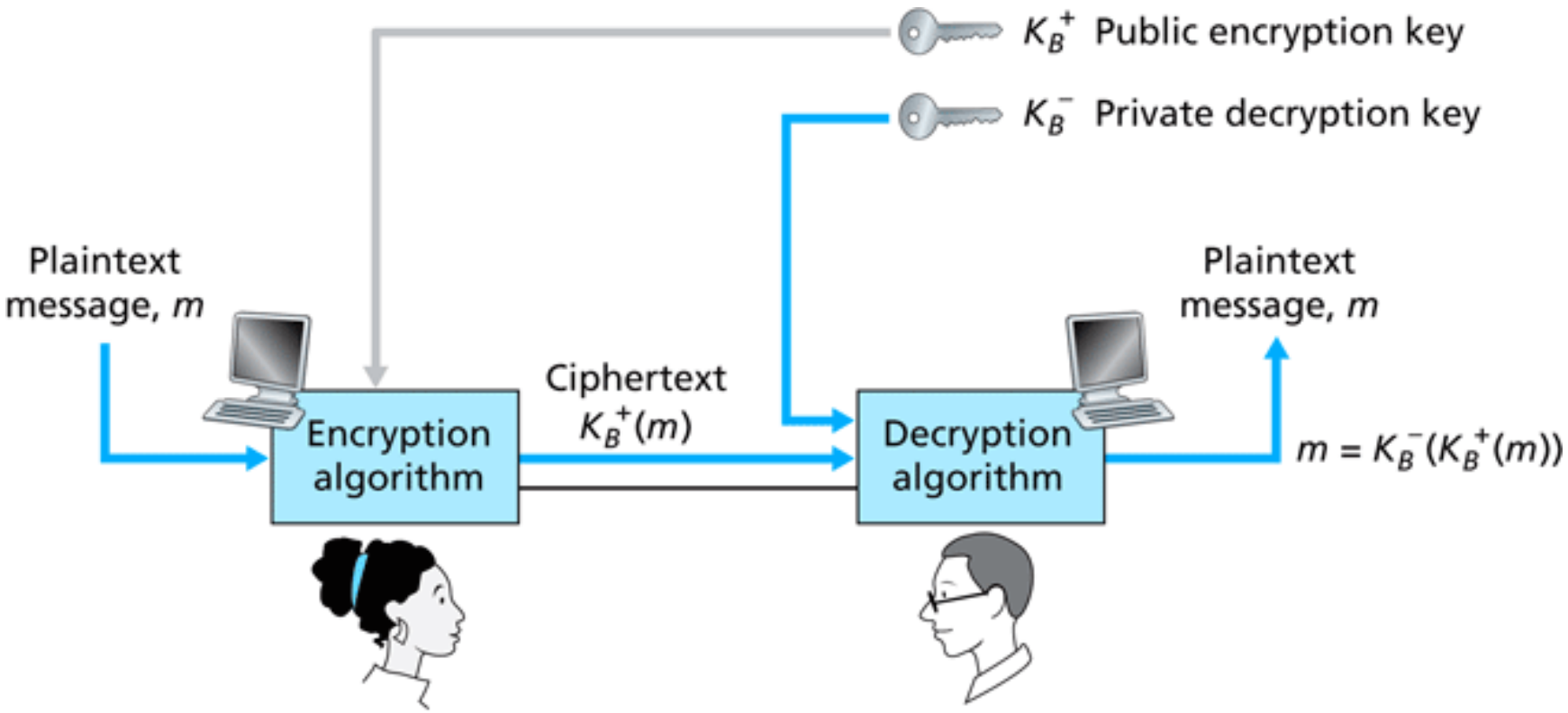
Cipher Block Chaining (CBC) mode encryption

Public Key Cryptography

Public Key Encryption

- symmetric key encryption requires two parties to share a secret
  - must somehow share the secret
  - meet in person, talk on phone
- public-key encryption
  - communicate securely without sharing a private key
  - can also be used for authentication and digital signatures

Example



1. Alice fetches Bob's public key, K_B^+
2. Alice encrypts and sends her message, $K_B^+(m)$, using a well-known encryption technique
3. Bob decrypts with private key, $K_B^-(K_B^+(m))$

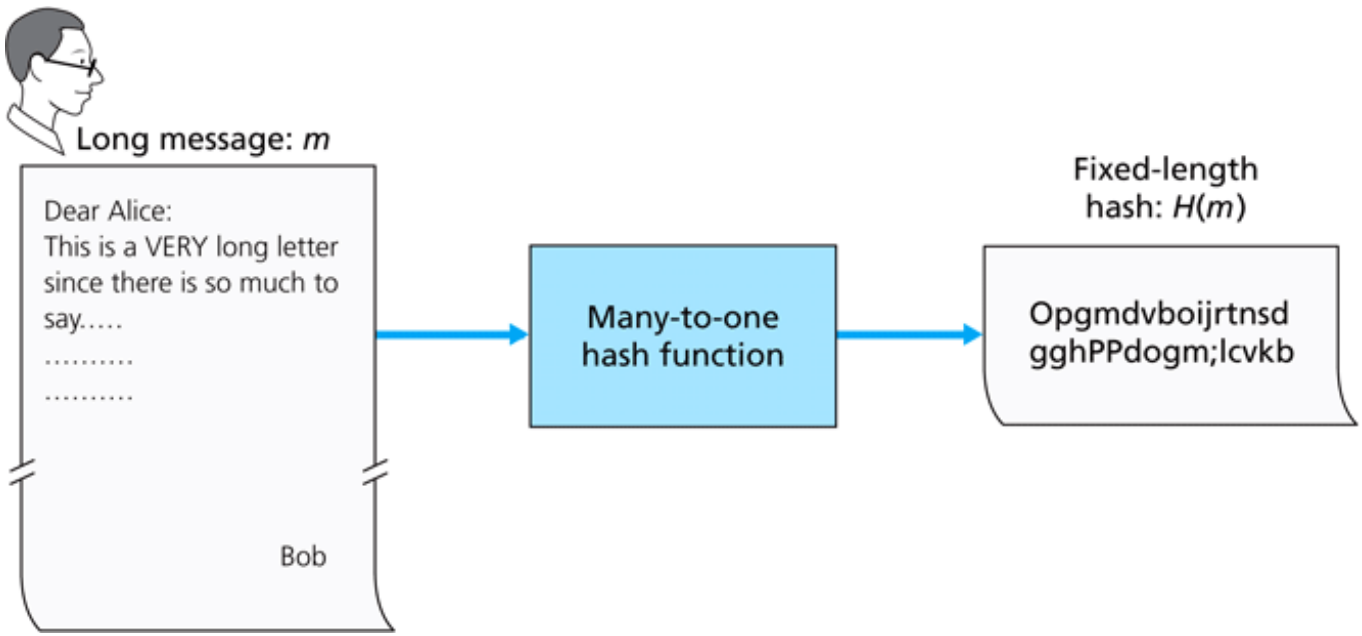
Issues

- chosen-plaintext attack
  - choose some text, encrypt with Bob's public key, try to learn the private key
  - must choose keys so that this is hard
- key generation
  - RSA, DSA, Diffie-Hellman
  - security often based on the fact that there are no known algorithms for quickly factoring a number n into two primes
- expensive
  - relatively expensive compared to symmetric key encryption (several orders of magnitude slower)
  - generally use public key encryption to exchange a symmetric key

Integrity

Cryptographic Hash Function

- takes an input, m, and computes a fixed-size string (hash)
- hash function chosen so that it is computationally infeasible to:
  - reverse the hash and recreate the original message
  - find two messages that hash to the same value



Example

```
SHA224("The quick brown fox jumps over the lazy dog")
0x 730e109bd7a8a32b1cb9d9a09aa2325d2430587ddbc0c38bad911525
SHA224("The quick brown fox jumps over the lazy dog.")
0x 619cba8e8e05826e9b8c519c0a5c68f4fb653e8a3d8aa04bb2c8cd4c
```

Choosing a Hash Function

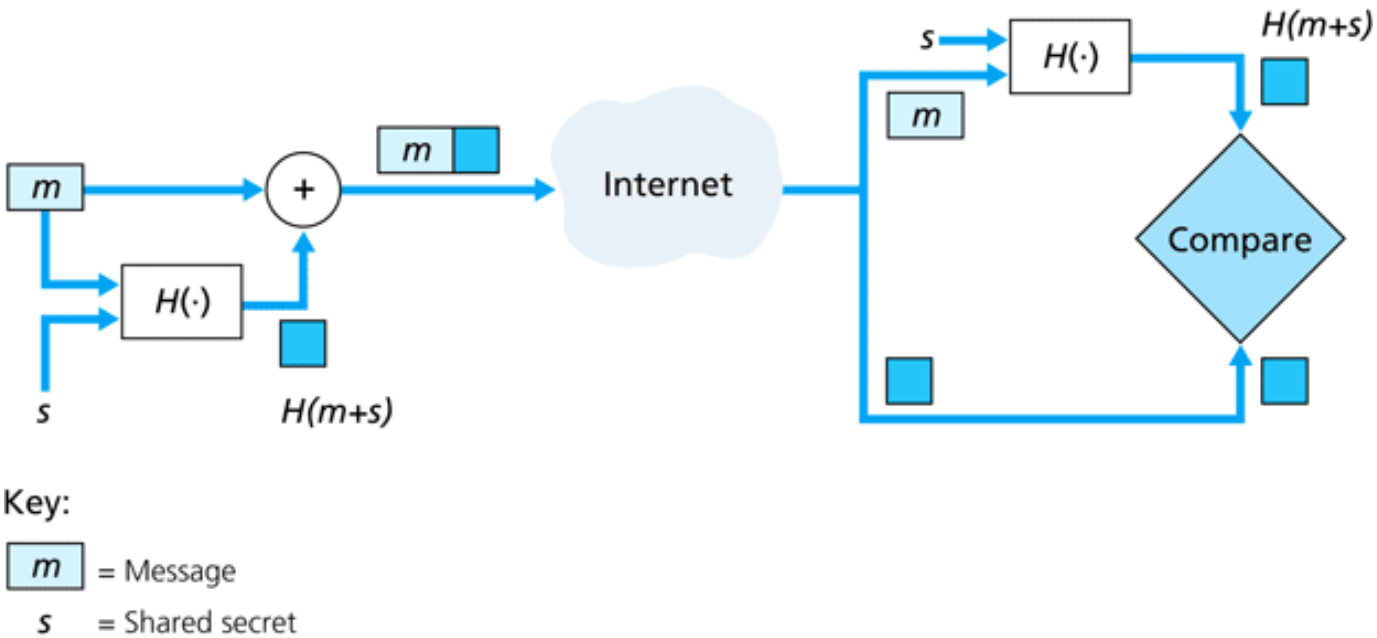
- MD5: 128 bit hash, security is severely compromised ▸ Wikipedia
- SHA-1: 160-bit hash, more secure but recently discovered weakness
  - most efficient attack in 2012 cost $2.77M$ to break a single hash value
- SHA-2: 224, 256, 384, or 512 bits
  - similar to SHA-1 but attacks not yet extended to SHA-2
- SHA-3: 224, 256, 384, or 512 bits
  - completely different algorithm
  - in the process of being standardized by NIST
  - ▸ Wikipedia

Data Integrity

- easy to provide data integrity without authentication
  1. Alice creates hash $H(m)$
  2. Alice sends $(m, H(m))$ to Bob
  3. Bob receives (m, h) and checks if $H(m) = h$
- Bob can't be sure the message came from Alice
- useful anyway
  - checking that you downloaded an unmodified version of a file
  - assumes that the MD5 hasn't been modified

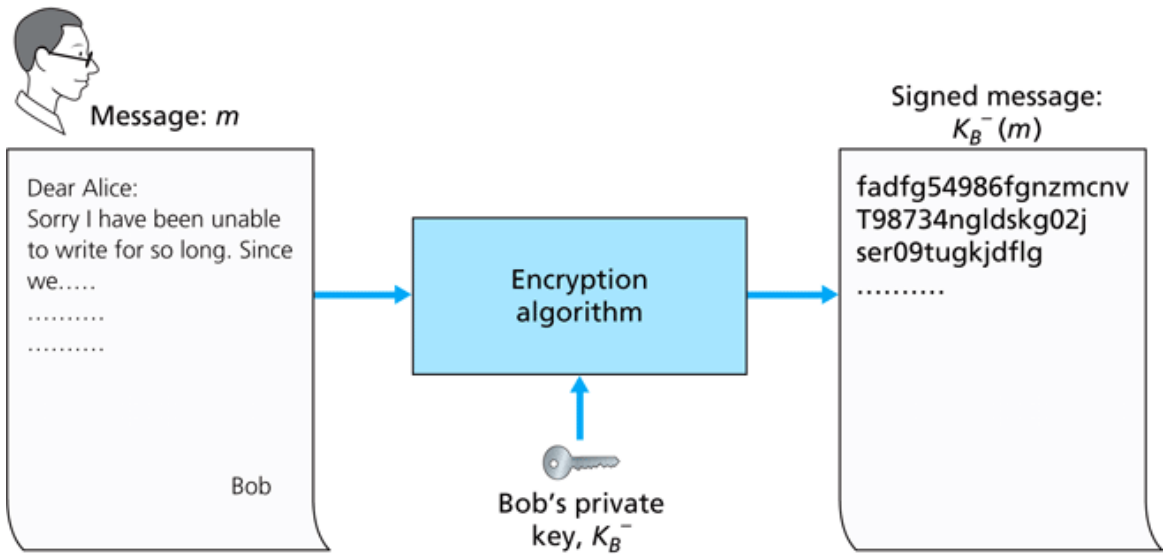
Data Integrity and Authentication

- Alice and Bob share a secret, the authentication key
  1. Alice calculates $H(s + H(s + m))$, the *message authentication code*
  2. Alice sends Bob $(m, HMAC)$
  3. Bob receives $(m, HMAC)$ and checks if $HMAC = H(s + H(s + m))$
- anyone who shares the key can generate an authenticated message



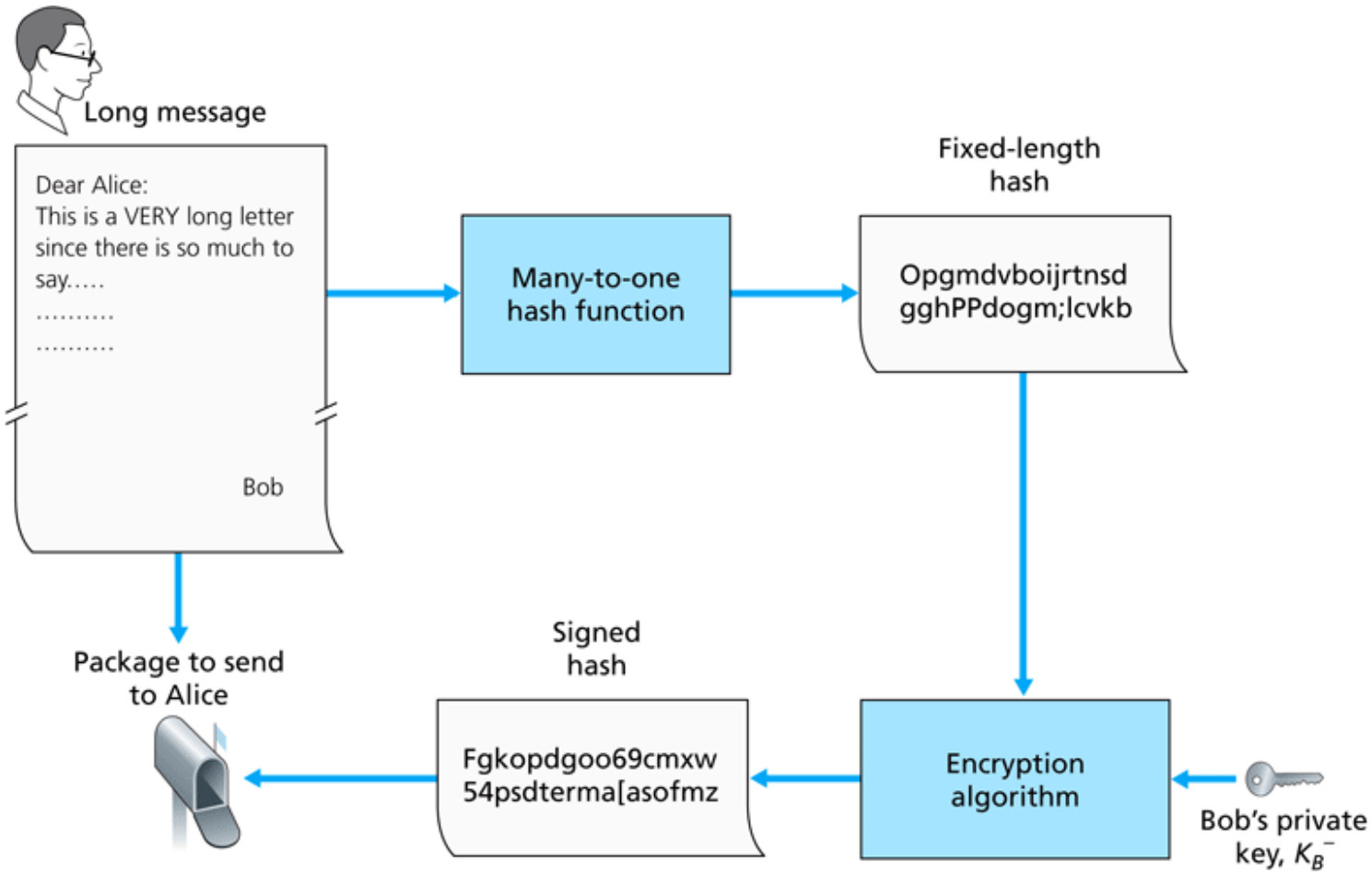
Digital Signatures

- want to be able to verify the owner or creator of a document, or signify agreement with the document's content
- properties
  - verifiability: can prove it was signed by a person
  - non-repudiation: can prove that only that person could have signed it
  - integrity: signature fails if document modified
- $K_B^+(K_B^-(m)) = m$

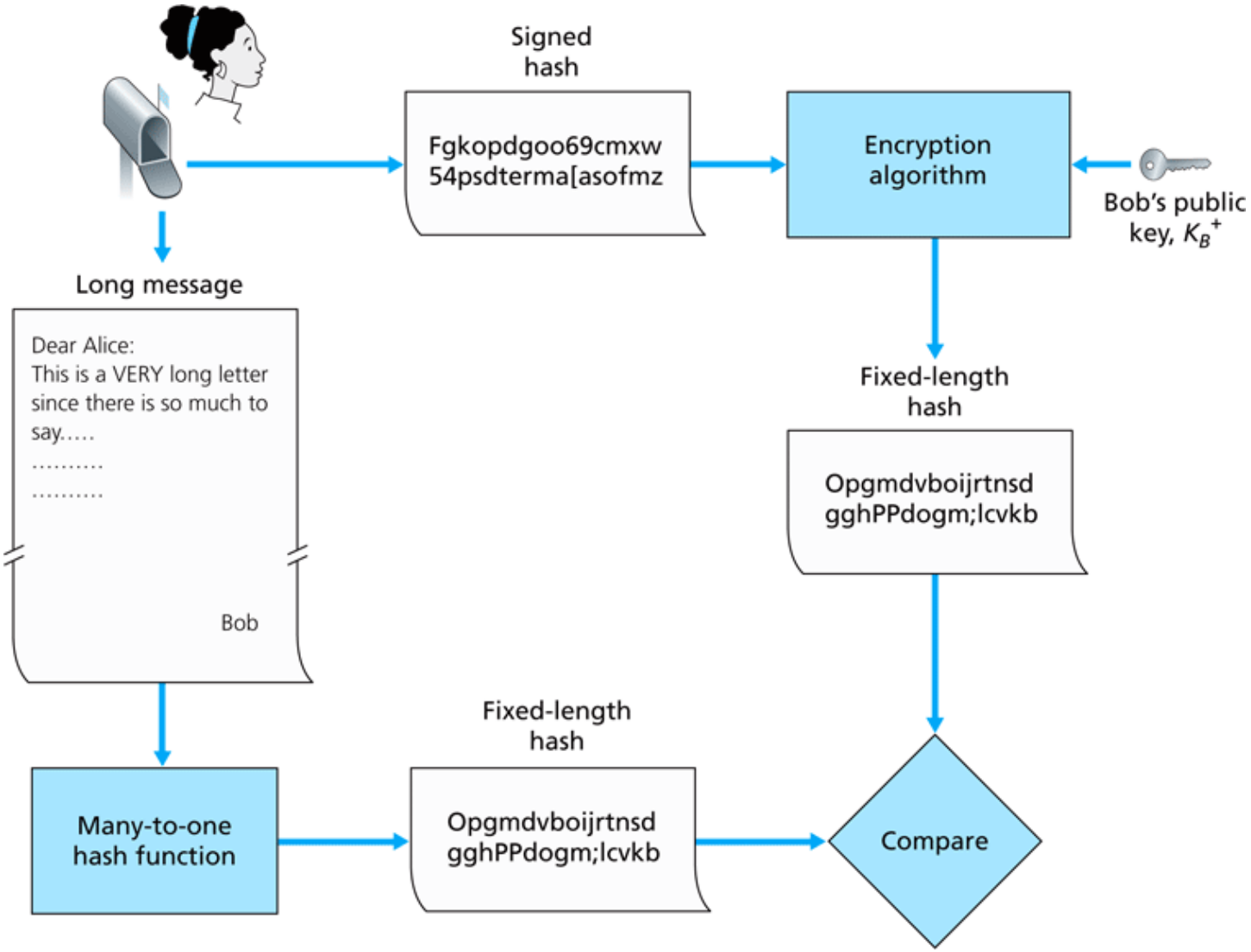


Simplifying Computation

- typically sign a hash of the message instead of the full message (more efficient)



Verifying a Signature



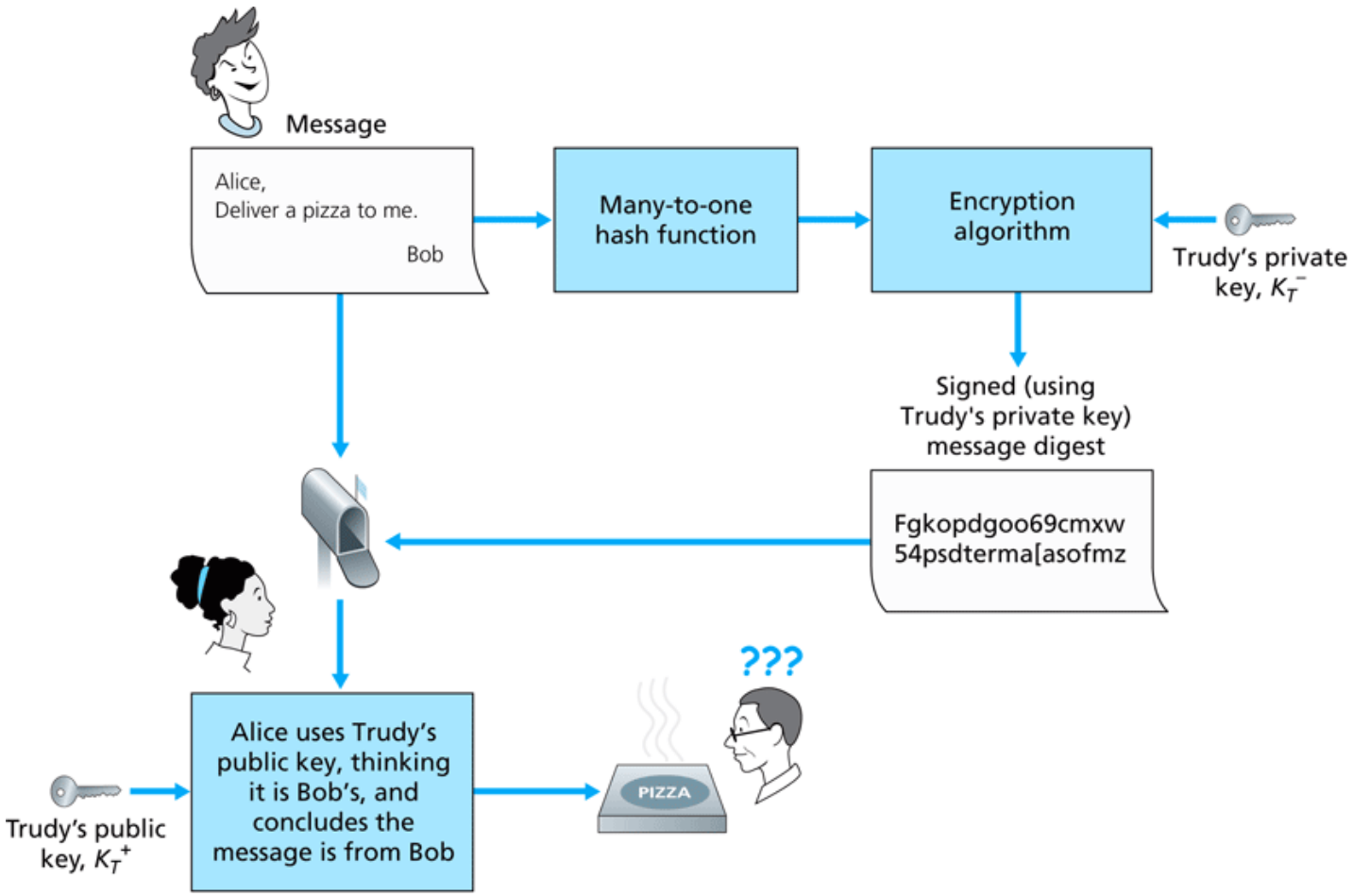
Authentication

Authentication

- prove your identity to someone over the network
  - message authentication verifies only that the message came from a particular person
  - subject to a replay attack
- authenticate first, then exchange messages
- Alice asks Bob to encrypt a message in his private key
  - if it decrypts properly, Alice knows she must be talking to Bob
  - Alice must be sure she really has Bob's public key

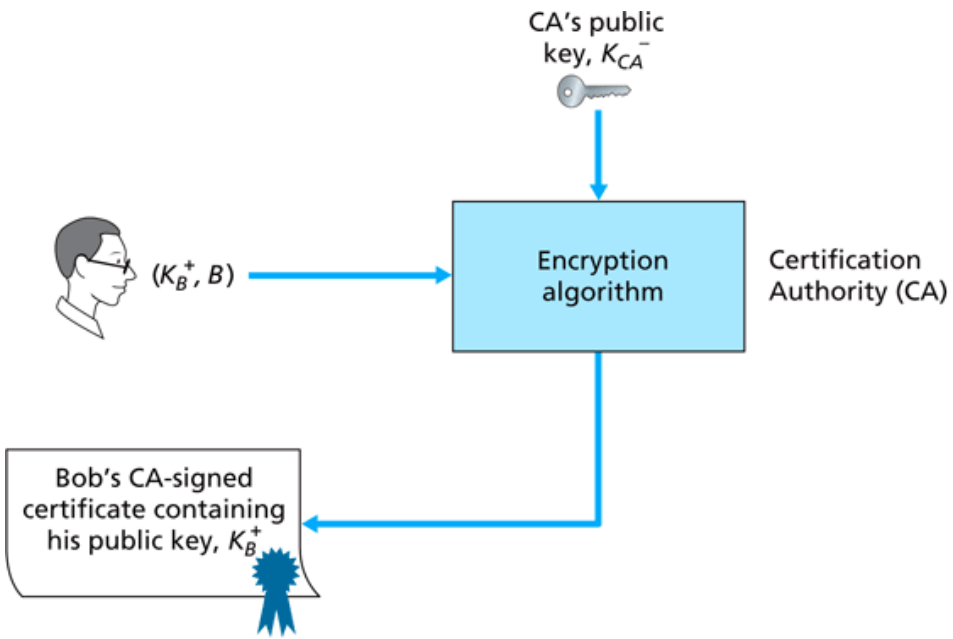
Public Key Exchange

- must be sure to get the public key associated with a given person or organization



Public Key Infrastructure

- typically performed by a Certificate Authority (CA)
  - verifies that an entity (person, computer) is who it says it is - verification procedure is left to the CA
  - creates a certificate that contains the public key and a unique identifier for the entity (e.g. email address, IP address)
  - signs the certificate
- distribute public key of CAs in browser or operating system



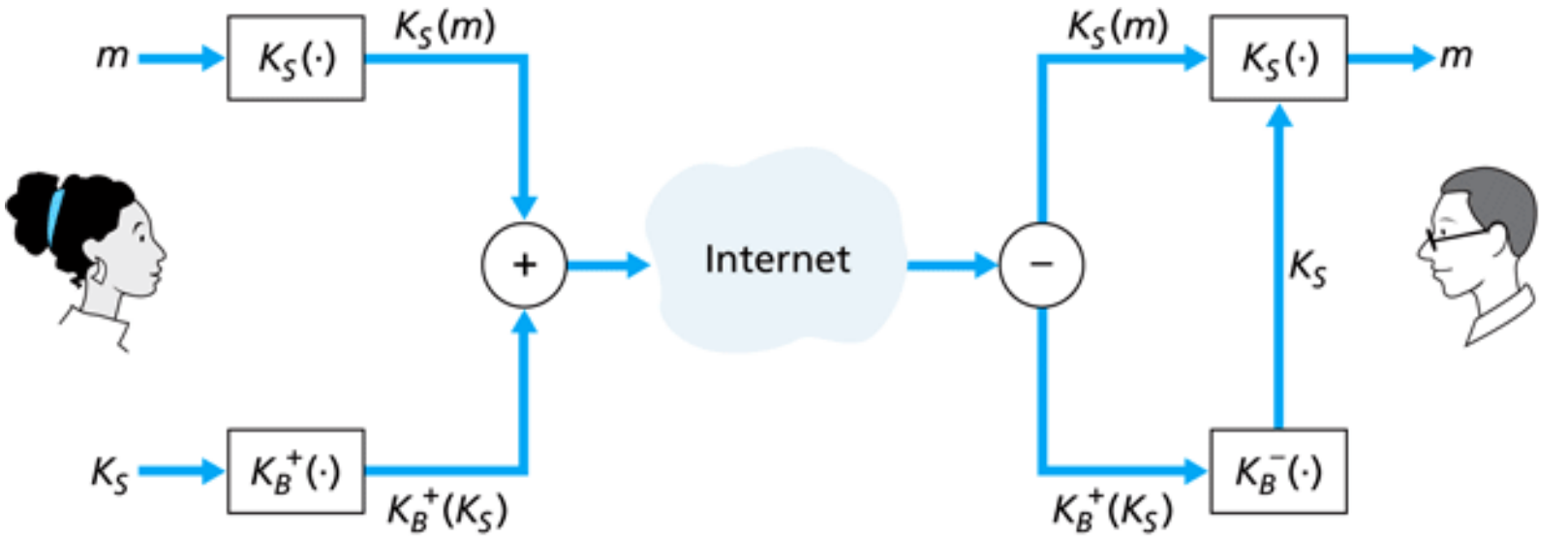
Applications

PGP

PGP: Secure Email

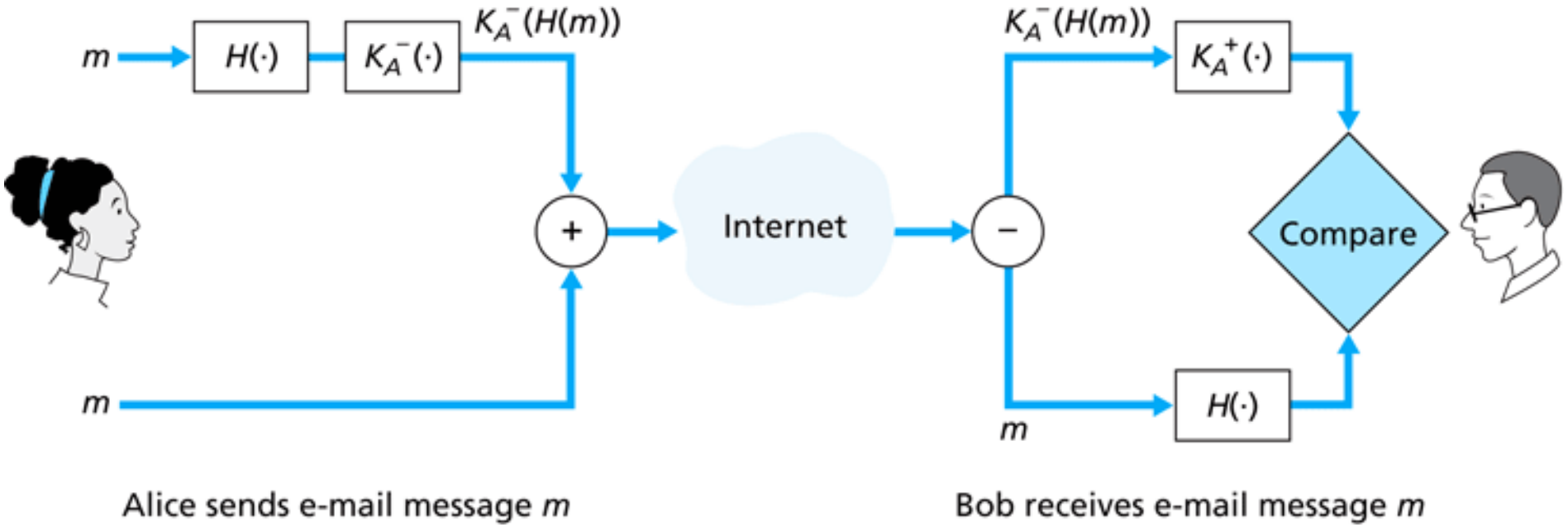
- goals
  - confidentiality
  - message integrity
  - sender authentication
  - receiver authentication

Email Confidentiality



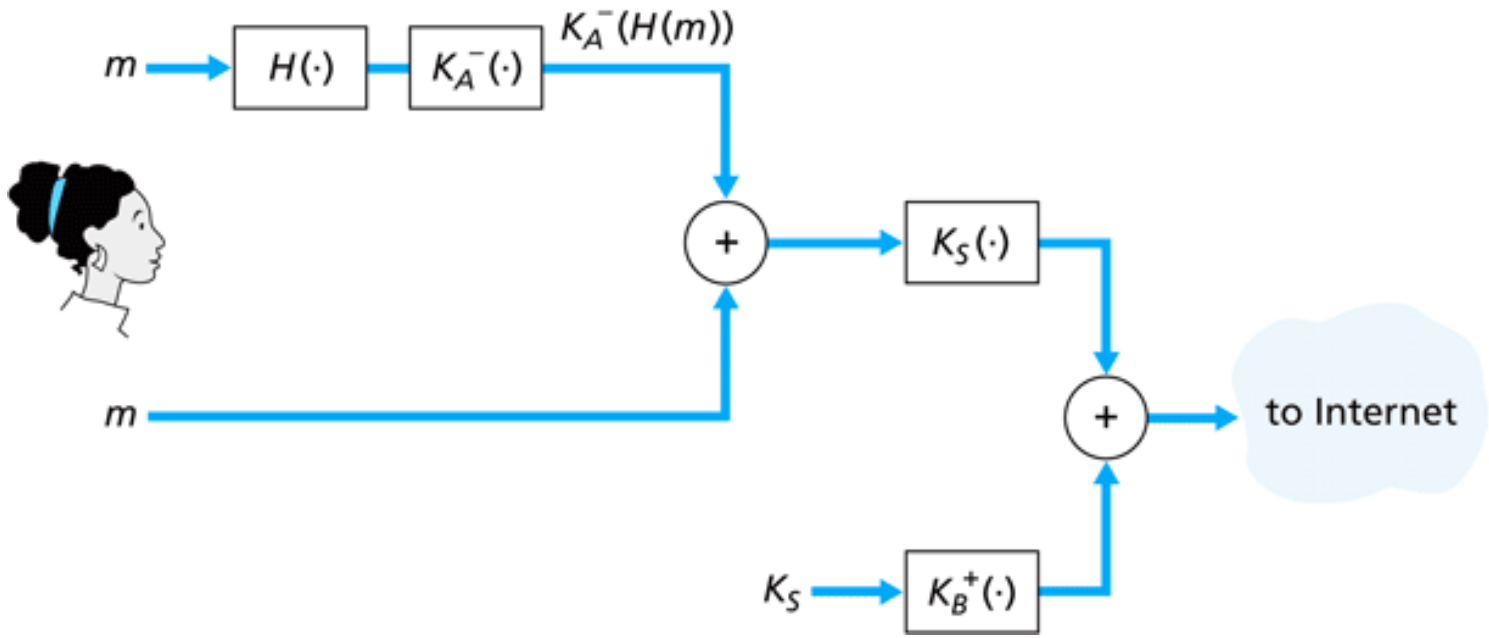
- Alice
  - encrypts message with a symmetric key
  - encrypts session key with Bob's public key
  - sends Bob the encrypted message and session key
- Bob
  - decrypts session key using private key
  - uses session key to decrypt message

Email Integrity and Sender Authentication



- Alice
  - creates a message digest with a hash function
  - signs the digest with her private key
  - sends unencrypted message and digest to Bob
- Bob
  - checks digest using Alice's public key, hash of message
  - reads the message

Confidentiality, Integrity, Sender Authentication



- Alice sends signed digest and message, encrypted with shared symmetric key, plus shared key encrypted in Bob's public key
- Bob reverses the process

PGP

- design is basically the same as previous figure
- can use different hash functions, encryption algorithms
- simplifies creation of public and private key pairs, signed by a web of trust

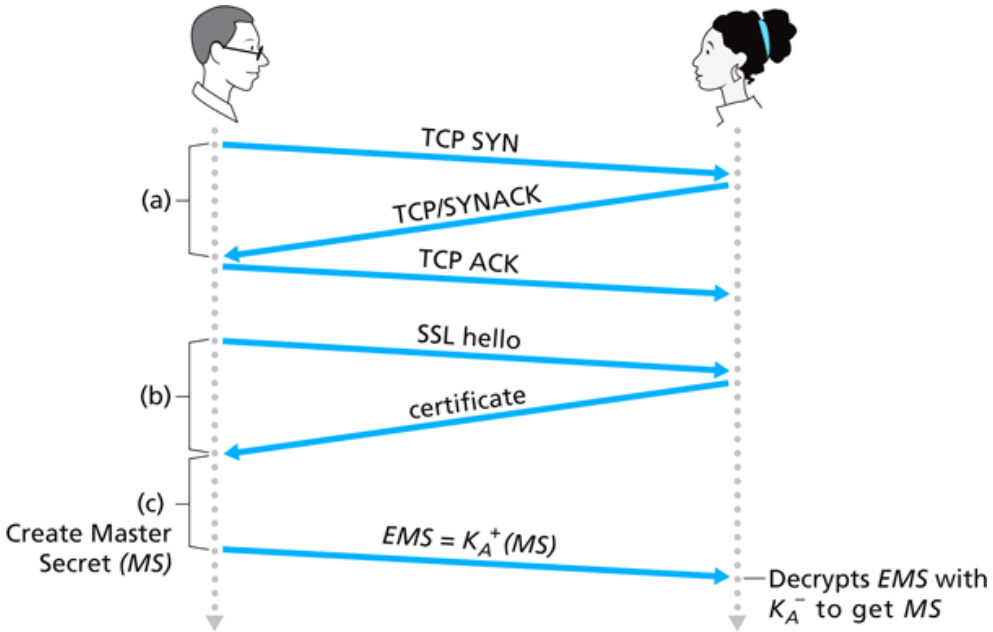
```
-----BEGIN PGP SIGNED MESSAGE-----
Hash:  SHA1
Bob:
Can I see you tonight?
Passionately yours, Alice
-----BEGIN PGP SIGNATURE-----
Version: PGP for Personal Privacy 5.0
Charset:  noconv
yhHJRHhGJGhgg/12EpJ+lo8gE4vB3mqJhFEvZP9t6n7G6m5Gw2
-----END PGP SIGNATURE-----
-----BEGIN PGP MESSAGE-----
Version: PGP for Personal Privacy 5.0
u2R4d+/jKmn8Bc5+hgDsqAewsDfrGdszX68liKm5F6Gc4sDfcXyt
RfdS10juHgbcfDssWe7/K=lKhnMikLo0+1/BvcX4t==Ujk9PbcD4
Thdf2awQfgHbnmKlok8iy6gThlp
-----END PGP MESSAGE
```

TLS

TLS

- provides confidentiality, data integrity, authentication for TCP connections
  - used to secure nearly all e-commerce sites, signified by https
- goals
  - confidentiality - protect credit card information, order privacy
  - data integrity - ensure order is not modified
  - server authentication - ensure user is shopping at the right site
- provides an interface between the application and TCP

TLS Basics



- establish connection, get signed public key from Alice, then send Alice a master secret
- use master secret to generate
  - E_B: encryption key for data from Bob to Alice
  - M_B: MAC key for data from Bob to Alice
  - E_A: encryption key for data from Alice to Bob
  - M_A: MAC key for data from Alice to Bob