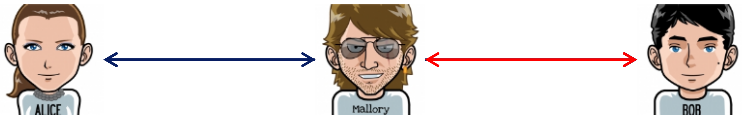
Chapter 9: Network Security

Outline

- Goals and threats
- Cryptography
- Network security mechanisms

Goals and Threats



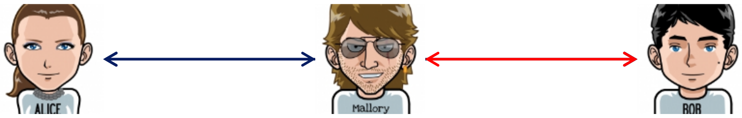
Goals

- Confidentiality
  - Only intended receiver can decode message
- Authentication
  - Sender and receiver can confirm each other's identity
- Message integrity
  - Sender and receiver can ensure message not altered without detection
- Access and availability
  - Sender and receiver can communicate

Threats to communications

- Eavesdropping
  - Message interception
  - Information leakage
- Impersonation
  - Spoof source address
- Hijacking
  - Replace sender or receiver in ongoing connection
- Message insertion
  - Spurious messages delivered
  - Valid message replayed
- Denial of service
  - Prevent communication (service from being used)

Goals and Threats

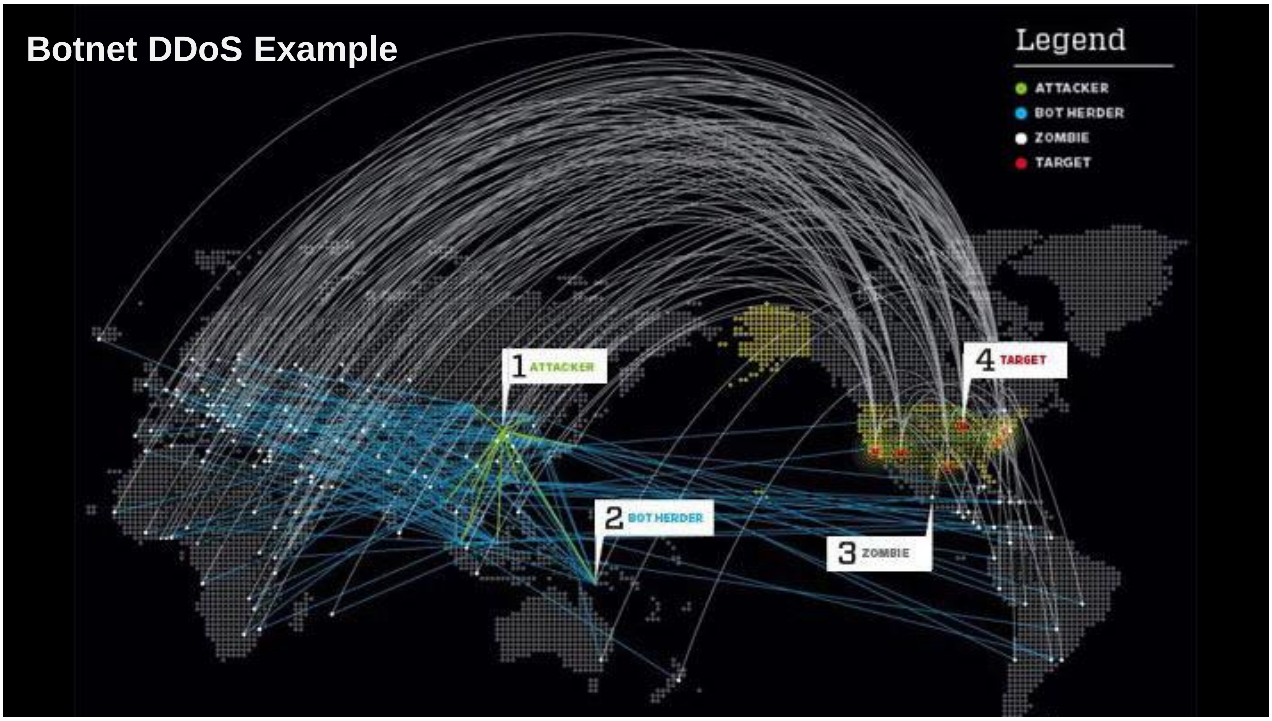


Goals

- Confidentiality
  - Only intended receiver can decode message
- Authentication
  - Sender and receiver can confirm each other's identity
- Message integrity
  - Sender and receiver can ensure message not altered without detection
- Access and availability
  - Sender and receiver can communicate

Threats to end-hosts

- Malware
  - Virus: self-replicating infection *by* user intervention
  - Worm: self-replicating infection *without* user intervention
- Spyware
  - Records and reports keystrokes, private information
- Botnets
  - A collection of malware infected hosts controlled by a *bot master*
  - Used for **spam** and DDoS attacks



Botnet DDoS Example

454

Policy vs. Enforcement

MONTANA STATE UNIVERSITY

- Security policy for an online transaction
  - Authenticate vendor to buyer
  - Communicate credit card number to vendor securely
  - Authenticate identity of buyer
  - Deliver goods upon payment
- Policy enforcement
  - Authentication
  - Message security
  - Message integrity
  - Non-repudiation
  - Privacy

Mountains & Minds

455

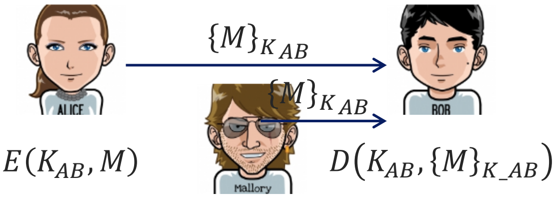
455

Terminology

K_A	Alice's secret key
K_B	Bob's secret key
K_{AB}	Secret key shared between Alice and Bob
K_{Apriv}	Alice's private key (known only to Alice)
K_{Apub}	Alice's public key (published by Alice for all to read)
$\{M\}_K$	Message M encrypted with key K
$[M]_K$	Message M signed with key K

Cryptography

Symmetric Cryptography



$\{M\}_{K_{AB}}$

$\{M\}_{K_{AB}}$

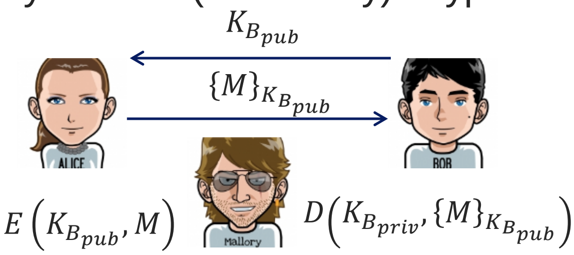
$E(K_{AB}, M)$

$D\left(K_{AB}, \{M\}_{K\_AB}\right)$

- Techniques
  - Confusion – reordering function
  - Diffusion – redundancy
- Key size > brute force attack

- How to distribute K_{AB} securely?
- How to prevent ***replay attack***?

Asymmetric (Public key) Crypto



$K_{B_{pub}}$

$\{M\}_{K_{B_{pub}}}$

$E\left(K_{B_{pub}}, M\right)$

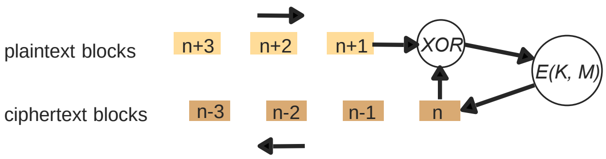
$D\left(K_{B_{priv}}, \{M\}_{K_{B_{pub}}}\right)$

- Depends on *trap-door* functions
- 100 to 1000 times slower than symmetric
- Used to:
  - Exchange shared keys
  - Sign messages

Ciphers

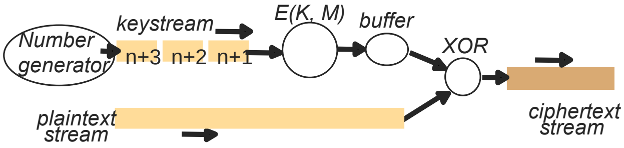
MONTANA STATE UNIVERSITY

Block ciphers



- Crypto algorithms encrypt small blocks of data ~ 64-bit
- Blocks transmitted as soon as encrypted
- Cipher Block Chaining (CBC)
  - XOR with previous block to prevent statistical attack
  - Random first block sent in clear text

Stream ciphers

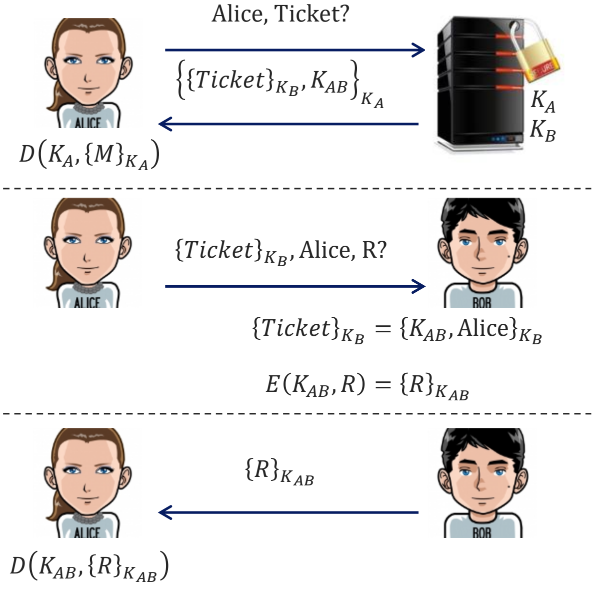


- Streaming data has variable datarate
  - Don't want to wait for block size
  - Don't want to pad to block size
- Generate keystream, encrypt, and store in buffer
- Mix buffer with streaming data

Authentication

MONTANA STATE UNIVERSITY

- Needham-Schroeder
  - Alice wants to access resource R held by Bob
  - Alice needs to authenticate to Bob
  - Alice gets secure *ticket* (also called a *challenge*) from server
  - Alice sends ticket to Bob with her request
  - Bob examines ticket
  - Bob sends resource, for example WiFi key, to Alice



Alice, Ticket?

$\left\{\{Ticket\}_{K_B}, K_{AB}\right\}_{K_A}$

K_A
K_B

$D\left(K_A, \{M\}_{K_A}\right)$

$\{Ticket\}_{K_B}$, Alice, R?

$\{Ticket\}_{K_B} = \{K_{AB}, \text{Alice}\}_{K_B}$

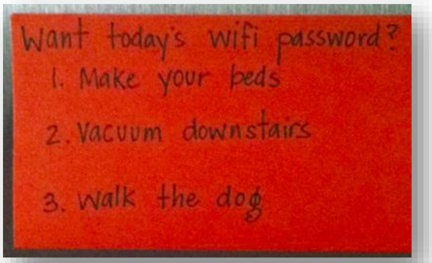
$E(K_{AB}, R) = \{R\}_{K_{AB}}$

$\{R\}_{K_{AB}}$

$D\left(K_{AB}, \{R\}_{K_{AB}}\right)$

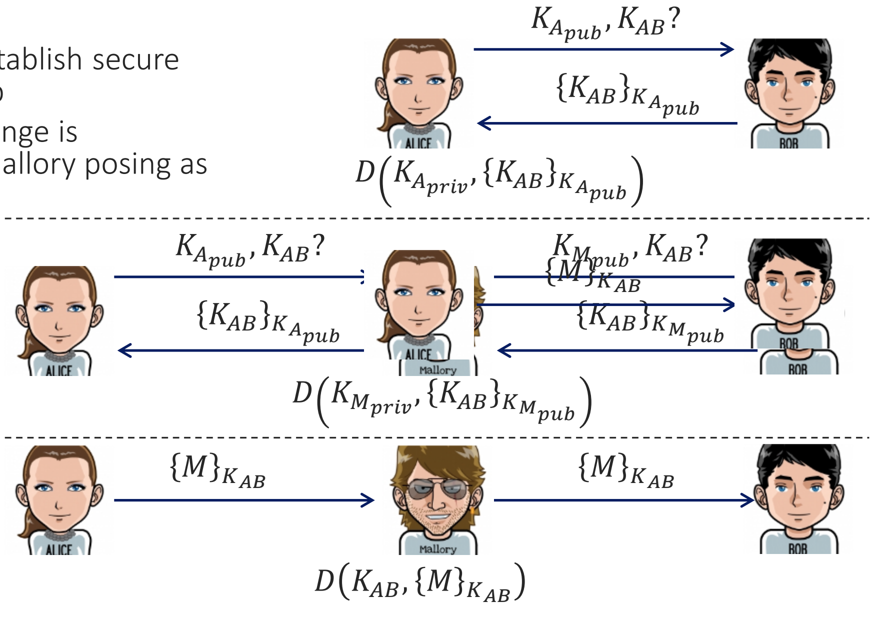
WiFi Encryption

- Wired Equivalent Privacy (WEP)
  - Key 10 or 26 hexadecimal digits
  - Uses a stream cipher with same key for all packets
- Wi-Fi Protected Access (WPA)
  - Temporal Key Integrity Protocol (TKIP)
    - Dynamic key for each packet
  - WPA-Personal – WPA-PSK (pre-shared key)
    - 8 to 63 printable ASCII characters
    - 256 bit key is calculated by applying the PBKDF2 key derivation function to the passphrase, using the SSID as the salt and 4096 iterations of HMAC-SHA1
  - WPA-Enterprise
    - Requires an authentication server
    - Extensible Authentication Protocol (EAP) suite used for authentication
  - Wi-Fi Protected Setup (WPS)
    - Simplifies authentication process
    - Current implementation vulnerable to attacks.



Man in the middle attack

- Alice wants to establish secure channel with Bob
- Shared key exchange is intercepted by Mallory posing as Bob



$K_{A_{pub}}, K_{AB}?$

$\{K_{AB}\}_{K_{A_{pub}}}$

$D\left(K_{A_{priv}}, \{K_{AB}\}_{K_{A_{pub}}}\right)$

$K_{A_{pub}}, K_{AB}?$

$\{K_{AB}\}_{K_{A_{pub}}}$

$K_{M_{pub}}, K_{AB}?$

$\{M\}_{K_{AB}}$

$\{K_{AB}\}_{K_{M_{pub}}}$

$D\left(K_{M_{priv}}, \{K_{AB}\}_{K_{M_{pub}}}\right)$

$\{M\}_{K_{AB}}$

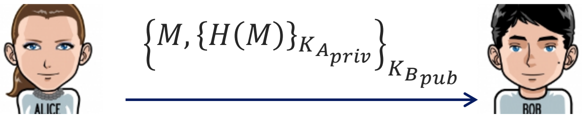
$\{M\}_{K_{AB}}$

$D\left(K_{AB}, \{M\}_{K_{AB}}\right)$

Digital Signatures

- Bob needs to know if message from Alice
- Alice signs the document
  - Creates secure digest $H(M)$ using secure hash function, i.e., $P\big(H(M) = H(M')\big) \cong 0$
  - Signs digest with private key
- Bob can authenticate the signature using Alice's public key

1.Request	Get balance
2.Name	Alice
3.Account	6262626
4.Signature	H(field 2 + field 3)



$$\left\{M, \{H(M)\}_{K_{A_{priv}}}\right\}_{K_{B_{pub}}}$$

$$D\left(K_{B_{priv}}, \left\{M, \{H(M)\}_{K_{A_{priv}}}\right\}_{K_{B_{pub}}}\right)$$

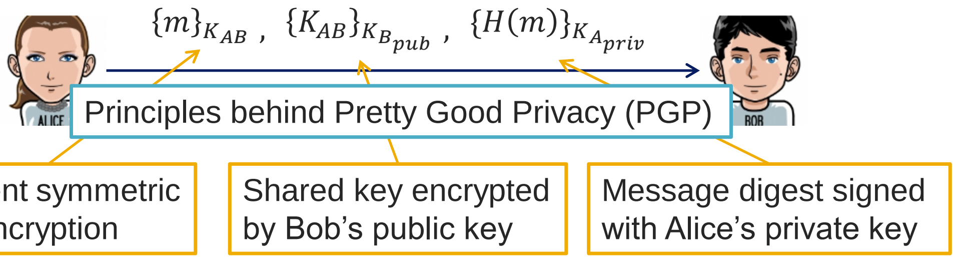
$$D\left(K_{A_{pub}}, \{H(M)\}_{K_{A_{priv}}}\right)$$

$$H(M) =?\, H(\text{ decrypted } M\ )$$

Why sign digest rather than the whole message?

Secure email

- Suppose Alice wants to securely communicate with Bob. Design an *efficient* communication mechanism that provides *confidentiality*, *message integrity*, and *sender authentication*.



$\{m\}_{K_{AB}}$, $\{K_{AB}\}_{K_{B_{pub}}}$, $\{H(m)\}_{K_{A_{priv}}}$

Principles behind Pretty Good Privacy (PGP)

Efficient symmetric key encryption

Shared key encrypted by Bob's public key

Message digest signed with Alice's private key

Is your email secure?

end-to-end
End-To-End