Cryptography: Message Authentication
Code
Announcements

- Homework 1 handled out on Sept 1, due on Sept 10
- Will have first quiz on Sept 8
Limitation of Using Hash Functions for Authentication

- Require an authentic channel to transmit the hash of a message
  - anyone can compute the hash value of a message, as the hash function is public
  - not always possible
- How to address this?
  - use more than one hash functions
  - use a key to select which one to use
Hash Family

• A hash family is a four-tuple \((X, Y, K, H)\), where
  – \(X\) is a set of possible messages
  – \(Y\) is a finite set of possible message digests
  – \(K\) is the keyspace
  – For each \(K \in K\), there is a hash function \(h_K \in H\). Each \(h_K : X \to Y\)
• Alternatively, one can think of \(H\) as a function \(K \times X \to Y\)
Message Authentication Code

- A MAC scheme is a hash family, used for message authentication
- \( \text{MAC} = C_K(M) \)
- The sender and the receiver share \( K \)
- The sender sends \( (M, C_k(M)) \)
- The receiver receives \( (X,Y) \) and verifies that \( C_K(X)=Y \), if so, then accepts the message as from the sender
- To be secure, an adversary shouldn’t be able to come up with \( (X’,Y) \) such that \( C_K(X)=Y \).
Example of Insecure Hash Families

- Let $h$ be a one-way hash function
- $H(K,M) = h(K \ || \ M)$, where $\ ||$ denote concatenation
  - Insecure as MAC
  - Given $M$ and $a=h(K \ || \ M)$, can compute $M'=M||\ldots$ and $a'$, such that $h(K||M') = a'$
- $H(K,M) = h(M \ || \ M)$,
  - Also insecure as MAC
HMAC: Constructing MAC from Cryptographic Hash Functions

\[ \text{HMAC}_K[M] = \text{Hash}((K^+ \oplus \text{opad}) \ || \ \text{Hash}((K^+ \oplus \text{ipad})||M))] \]

- \(K^+\) is the key padded (with 0) to \(B\) bytes, the input block size of the hash function
- \(\text{ipad}\) = the byte 0x36 repeated \(B\) times
- \(\text{opad}\) = the byte 0x5C repeated \(B\) times.
HMAC Overview

```
K^+   ipad
   ≈
S_i  Y_0  Y_1  ...  Y_{L-1}

IV  n bits
    Hash

K^+   opad
   ≈
   b bits
H(S_i || M)
   pad to b bits
   S_o

IV  n bits
    Hash

IV  n bits
    HMAC_K(M)
```
HMAC Security

- If used with a secure hash functions (e.g., SHA-256) and according to the specification (key size, and use correct output), no known practical attacks against HMAC
Encryption and Authentication

• Three ways for encryption and authentication
  – Authenticate-then-encrypt (AtE), used in SSL
    • a = MAC(x), C=E(x,a), transmit C
  – Encrypt-then-authenticate (EtA), used in IPSec
    • C=E(x), a=MAC(C), transmit (C,a)
  – Encrypt-and-authenticate (E&A), used in SSH
    • C=E(x), a=MAC(x), transmit (C,a)
• Which way provides secure communications when embedded in a protocol that runs in a real adversarial network setting?
Encryption Alone May Be Insufficient for Privacy

- If an adversary can manipulate a ciphertext such that the observable behavior (such as success or failure of decryption) differs depending on the content of plaintext, then information about plaintext can be leaked.
- To defend against these, should authenticate ciphertext, and only decrypt after making sure ciphertext has not changed.
- Encrypt-then-authenticate (EtA) is secure
  - $C = E(x), a = MAC(C)$, transmit $(C, a)$.
Encryption Alone May Be Insufficient for Privacy: An Artificial Example

- Given a secure stream cipher (or even one-time pad) $E$, consider encryption $E^*$
  - $E^*[x] = E[encode[x]]$
    - $encode[x]$ replaces 0 with 00, and 1 with either 01 or 10.
  - How to decrypt?
  - $E^*[x]$ is secure

- Using $E^*$ may not provide confidentiality in some usage
  - Consider the case an adversary flips the first two bits of $E^*[x]$
  - When the bits are 01 or 10, flipping results in no change after decrypt
  - When the bits are 00, flipping result in decryption failure
  - Learning whether decryption succeeds reveal first bit
AtE and E&A are insecure

- Authenticate-then-encrypt (AtE) is not always secure
  - $a = MAC(x)$, $C = E(x,a)$, transmit $C$
  - As first step is decryption, its success or failure may leak information.
  - AtE, however, can be secure for some encryption schemes, such as CBC or OTP (or stream ciphers)

- Encrypt-and-authenticate (E&A) is not secure
  - $C = E(x)$, $a = MAC(x)$, transmit $(C,a)$
  - MAC has no guarantee for confidentiality
Readings for This Lecture

- Wikipedia
  - Message Authentication Code

- Optional reading
  - Hugo Krawzyck.: The Order of Encryption and Authentication for Protecting Communications
Coming Attractions …

- Operating System Security Basics