Week 8: PKI + Networks

Hash Functions: Integrity of communication.

Alice
Bob
Public network

Question 1: Let’s assume Alice and Bob are communicating over a public network, and they are playing a game where:

a. They both choose an integer number
b. Then, they each disclose this number to each other
c. If the sum of the two numbers is even Alice wins
d. If the sum of the two numbers is odd Bob wins .

If we don’t use hash functions, and there are no eavesdroppers on the network, who is guaranteed to win?

A. Alice
B. Bob
C. Whoever goes first wins
D. Whoever goes last wins

Question 2: Now, let’s have Alice and Bob play the same game but applying hash functions to their messages. This time the game proceeds as follows:

a. Alice and Bob each choose an integer number
b. Alice sends a hash of her number
c. Bob receives Alice’s hash and sends his number
d. Alice then sends her number over.

Assuming there are no eavesdroppers on the network, who is guaranteed to win?

A. Neither, its 50% chance
B. Alice
C. Bob
D. Whoever goes first
E. Whoever goes last

Question 3: We know that hash functions provide the following properties: collision resistance, and one-way or preimage resistance. Assume we play the same game as Question 2:

A. Which property protects Alice if Bob tries to cheat (i.e., change his response mid-way)? (collision resistance or pre-image resistance)
B. Which property protects Bob if Alice tries to cheat (i.e., change her response mid-way)? (collision resistance or pre-image resistance)

Question 4: Circle all possible applications of hash functions

A. Integrity verification of files
B. Time stamping files
C. Password Authentication
D. Proving Identity

Launching Attacks on Hash functions.

Question 1: We know that a hash computes a one-way function from an input string m to an output hash(m). If our input string is 4 bits long, and our hash has a fixed length, can you launch an attack that defeats pre-image resistance?

A. Yes
B. No

Question 2: We know that a hash computes a one-way function from an input string m to an output hash(m). If our input string is 60 bits long, and our hash has a fixed length, is it theoretically possible to launch an attack that defeats collision resistance?

A. Yes
B. No

Question 3: Alice comes up with a new scheme that she claims provides secure authentication. She first hashes her password, and sends it over the network and implements a hash checking function that compares the hashed password with the hash on file. Is this system more secure than sending passwords over the network using symmetric key encryption?

A. Yes
B. No

Digital Certificates

We’ve seen powerful techniques for securing communication such that the only information we must carefully protect regards “keys” of various sorts. Given the success of cryptography in general, arguably the biggest challenge remaining for its effective use concerns exactly those keys, and how to *manage* them. For instance, how does Alice find out Bob’s public key? Does it matter?

We’ve seen one instance of key management in lab and class: the Chain-of-Trust model. Let’s explore other models, and see the advantages and disadvantages of each.

Question 1: Using a Trusted Directory Service: In this approach to key management, some central organization maintains an association between the name of each participant and their public key. Suppose everyone trusts Dirk the Director to maintain this association. Then any time Alice wants to communicate with someone, say Bob, she can contact Dirk to ask him for Bob’s public key.

Issues with the Trusted Directory Service

A. Trust
B. Scalability
C. Reliability
D. Being Online
E. Others… (there are many more!)

Question 2: Chain of Trust Model: In this approach we have a Certificate Authority (CA) that issues certificates to Alice and Bob. In order to verify the CA, we check the mapping from the CA to it’s public key from the CA higher up in the chain. We continue to do so until we reach the root CA. Every browser in the world ships with a list of trusted CAs. Firefox currently ships with about 88 trusted CAs preconfigured in the browser. The browser manufacturers have decided that, whether you like it or not, those CAs are trusted. Is there an advantage to having many CAs configured?

A. Yes, because you get a choice depending on who you trust
B. No, it looks like the web browser will accept *any* certificate issued by *any* of these 88 CAs.

Question 3: Web of Trust Model: This approach was pioneered by PGP, a software package for email encryption. The idea is to democratize the process of public key verification so that it does not rely upon any single central trusted authority. In this approach, each person can issue certificates for their friends, colleagues, and others whom they know. Suppose Alice wants to contact Doug, but she doesn't know Doug. In the simplest case, if she can find someone she knows and trusts who has issued Doug a certificate, then she has a certificate for Doug, and everything is easy.

If that doesn't work, things get more interesting. Suppose Alice knows and trusts Bob, who has issued a certificate to Carol, who has in turn issued a certificate to Doug. In this case, PGP will use this certificate chain to identify Doug's public key.

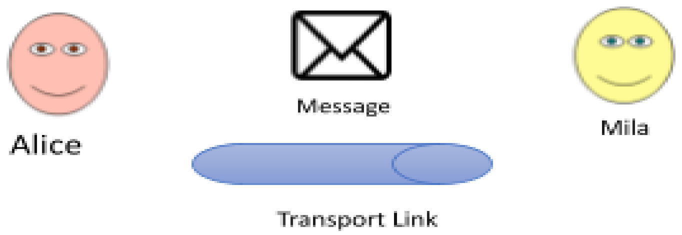
In the latter scenario can we say that Alice has securely obtained a copy of Doug's public key?

A. Yes
B. No

Network Security

Question 1: Protocol Structure

Alice and Mila are Swatties starting out their semester and are roommates. Alice wants to give Mila a reminder to get milk.



1. **Structure of the message:**
   - Construct the message that Alice posts to Mila. (Example on class slides)
   - Other than the sender/receiver information, what other metadata about the message
     might you add?

2. **Organizing a drop-off point.**
    - Who chooses the drop-off point?

3. **Write a protocol to write a note /post—it to your housemate. A protocol defines the message format and the transfer procedure.**

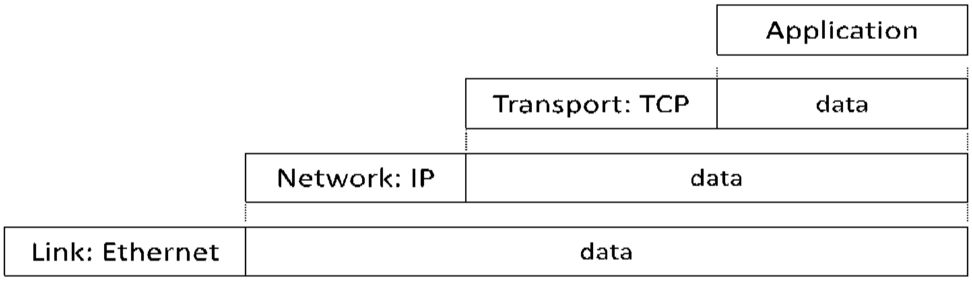
Alice moves to Chicago and Mila to Seattle for summer internships. Alice wants to send Mila a birthday card.

1. **Construct the message and header portions. Have these changed from the previous scenario?**

2. **<u>List the message format and transfer procedure of the “mail sending protocol” that Alice uses.</u>**
    a. **<u>Who chooses the drop-off point?</u>**
    b. **<u>Is this the only protocol used by Alice to deliver mail to Mila?</u>**

3. **<u>Message transportation and delivery. Whose job is it to:</u>**
    a. **<u>choose the carrier?</u>**
    b. **<u>plan the route?</u>**
    c. **<u>deliver the message?</u>**
    d. **<u>ensure the message is not lost?</u>**

Question 2: Message Encapsulation



The Internet has a layered architecture, in order to divide up the responsibilities of transmitting a packet from the source to the destination - similar to postal mail. The application layer is at the top of the hierarchy, and this is where the actual payload or data is constructed. Layering helps separate out the functions, and provides a nice abstraction to the layers above and below about the services it provides.

<u>Message transportation and delivery: In our mail analogy, whose responsibility should it be to carry out the following tasks? Do each of these tasks represent different “services” at different layers?</u>

1. Choice of carrier (USPS vs FedEx)
2. Route planning
3. Transport vehicles
4. Delivery acknowledgement

Networks have many concerns, such as reliability, error checking, and data ordering. Who/what should be responsible for addressing them? (Why?) Discuss which of these options you think is most suitable.

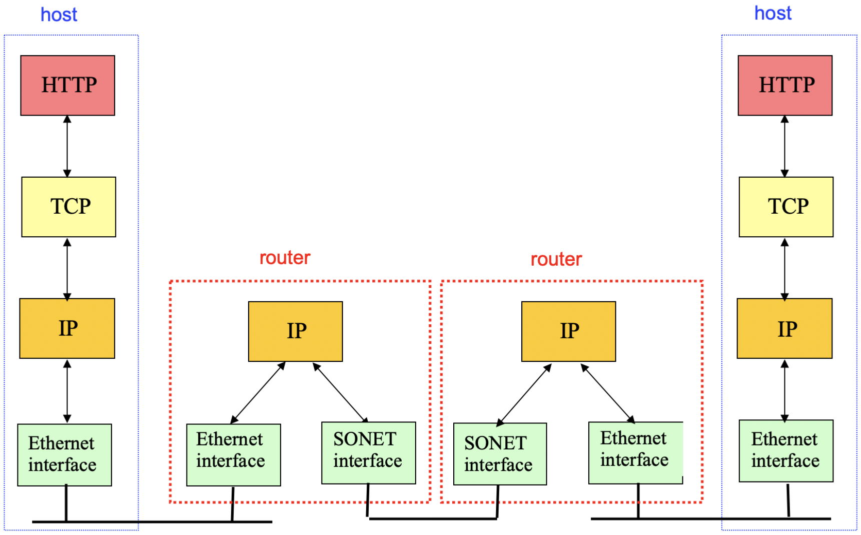
A. The network should take care of these for us.
B. The communicating hosts should handle these.
C. Some other entity should solve these problems.

Given the layered architecture above, discuss the following statement: layering and separation of functions is..

A. Great! It has a nice clean design and we can easily swap any protocol we want at any layer.
B. Not really... there are some glaring disadvantages to it.

Consider the figure shown below. Which layers do routers participate in? (Getting data from host to host.)

A. All of Them
B. Transport through Physical
C. Network, Link and Physical
D. Link and Physical


host
HTTP
TCP
IP
Ethernet interface
router
IP
Ethernet interface
SONET interface
router
IP
SONET interface
Ethernet interface
host
HTTP
TCP
IP
Ethernet interface