6.033 Lecture 21
Security II

Americo De Filippo

July 10, 2023

References

[1] Saltzer, Jerome H. and M. Frans Kaashoek. Principles of Computer System Design: An Introduction (2009): **Section: 11.2 - .3**

In the module of last time we had a client that was going to communicate through a network with a server. We have said that we had 3 problems that we were addressing:

- **Authentication**
- **Authorization**
- **Confidentiality**

Last time we have speak about cyptography, what it does is via some kind of encyption we are going to trasmit the messages that may seems meaningless and only with the key you can reverse the process and have the real message. We talked about private and public key protocols [1].

1 Authentication

When we authenitcate a user we want to achieve:

1. Who is requesting?
2. Message sent == message recv

1.1 mechanical or techincal

How we are going to authenticate a user? We can do this via One time pad, this is an XOR tequinique in order to encypt the bit stream, the problem with this is that lucifer can manipolate the message in some random way and perhaps change the message. So what we want is a checksum dependent on a key operation.

[1] perhaps read a paper on this is.

1.2 Key distribution problem

If Alice cannot phisicaly meet with Bob, she could send a message to Bob telling him her private key, but at this point we have the same problem of knowing that was in fact Alice to send the message. Instead we are going to use certificates, the ideas is that we are going to use a third in the communication that is both known to both of them (Charles). Alice at this point is going to send to Bob the public key the same as before, at this point Bob is going to ask Charles if that is the correct public key of Alice.

2 Secure Communication Channel or Confidentiality

First we are going to authenticate the user with the previus tequinique, then we are going to use public key to authenticate and use a shared key protocol. There are 3 proprieties that we want for our protocol.

- **Freshness**: Recent history message, timestamps.
- **Appropriate**: I'm actually the intendent recv of this message.
- **Forward secrecy**: We should be able to change the key and the protocl would still work.