Pawel Derkacz
151 00 5994
Security Engineering

Assignment 4

Protocol 1:

Flaws:

An attacker can take on many different IP addresses, and send his choice of vote to the server, bloating the results in his favor. Attacker doesn't even need to seek out previous packets; he can make his own, since all that is used is a public key.

Fix:

I assume that the server is already logging the IPs that sent in their votes, preventing them from sending further votes. If not, (in the event of a public computer needed for voting) have a unique identifier, such as a nonce, as part of the encryption along with sender information (replay protection). That said, the communication model needs to change, since the attacker can still form valid packets with his own information, due to only the public key being used. Therefore, some kind of additional private key, or a third-party, must be used to be able to ensure true security.

Protocol 2:

Flaws:

The only identifier used is the encryption key that is supposed to only be shared between two people, which happens to be a human-created password - meaning it could be easily brute forced by attackers (since the nonce is only +1 on the way back, could try to send information to B and see what returns, using the results to find the password). Should the password/key be found, the attacker can spoof as A and communicate with B.

Fix:

Make sure that both parties know that a communication attempt was made: have nonces and identifiers from both parties in the encryption. However, does not prevent the attacker gaining information by just prodding B with packets, so a better key should be used to make it harder to break through.