Markle's Puzzle

The approach in Markle's puzzle is to trade communication overhead for computation overhead.

3-9.pdf
939.3 KB

3.8 Keyed Hash

-> In this the file is ~~be~~ hashed with some secret key.

3.9 Markle's Puzzle

Step 1: Generate 128 bit keys.
128 bit keys = $k_1, k_2, k_3, \ldots k_n$.
64 bit keys = $s_1, s_2, s_3 \ldots s_n$ -> secret keys.
sequence # = $t_1, t_2, t_3 \ldots t_n$.

Step 2:
Select symmetric Algo 64 bit key used encrypt secrete key.

~~Es~~ $E_{s_1}(t_1, k_1, h(k_1)) = m_1$
$E_{s_2}(t_2, k_2, h(k_2)) = m_2$
$\vdots$
$E_{s_n}(t_n, k_n, h(k_n)) = m_n$

Step 3: Alice will dump $M_1 \ldots M_n$ to Bob.
Step 4: Bob randomly pick a message M_i.
Step 5: Bob uses brute force to open M_i using all keys.
Step 6: Bob opens message and choose seq t_i and sends to Alice.
Now both sides will know k_i.
(IMG 1)

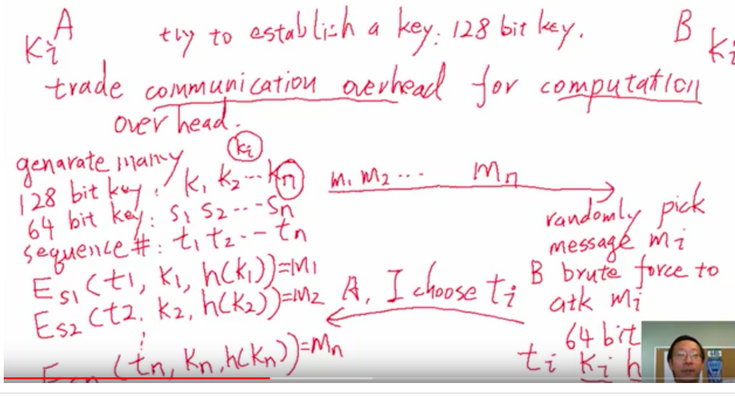
-> If the attacker wants to open ~~the~~ match the message he has to try for n times since there are n messages & 64 bit key
∴ bad guy has to crack $n \cdot 2^{64}$
bob has to crack ~ 2^{64}

-> If Alice can generate 2^{36} messages
Attacker computation over head = $2^{36} \cdot 2^{64}$
$= 2^{100}$

∴ If Bob need 2^{36} seconds to crack, Bad guys have 2^{100} seconds.

Block 5 - Random Numbers.

Dual Encryption :-



Dual Encryption