Assumptions

- Host-pair keying.

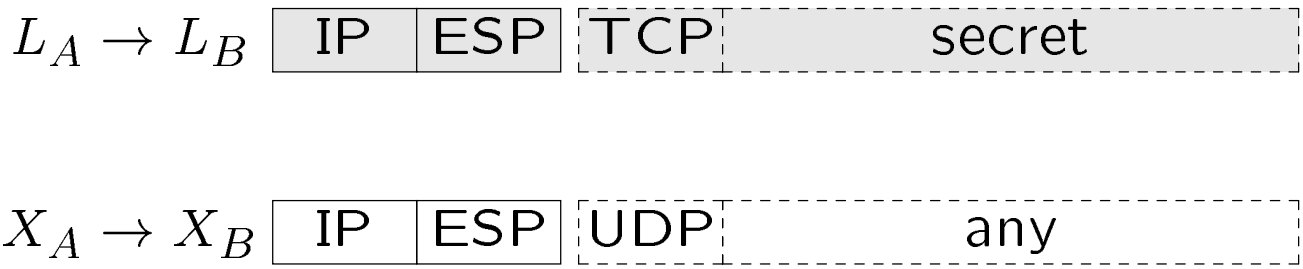
- Both hosts are multiuser machines; the attacker has logins on both machines. (It seems likely that one can omit one login with judicious use of the UDP echo server. It also seems probable that the attack works on router-to-router encryption, if a common key is used.)

- The attacker also has a machine that can both eavesdrop and inject messages.

- ESP used; no AH on insertion attack.

- Attack launched within SKIP transient key lifetime.

Reading Someone's Data



Inject from your PC

Injecting Text

$L_A \rightarrow L_B$

$X_A \rightarrow X_B$

Inject from your PC

$L_A \rightarrow L_B$

Countering AH?

- Assume that authentication takes place after fragmentation.
- Fragmentation data cannot be checksummed.
- Send a large packet so your insertion data is at the start of a fragment.
- The kernel will sign that for you; add a new header with no fragment indicators but the same payload, and reinject.

Conclusion

- The danger comes from a combination of shared keys and no mandatory integrity checking.

- Even per-user won't solve the problem entirely, because of router-to-router encryption and shared services such as NFS.

** We must add a mandatory integrity check to confidentiality mode. This in turn renders useless the combination of ESP and AH.