TCP packet authentication on the cheap

Eric Rescorla

IETF 60 TCPM meeting

(with some ideas from Vern Paxson)

Background

- Current attacks depend on forging TCP packets
  - Standard defense is to integrity check packets
- TCP MD5 would do the job
  - But keying is a problem
  - Not specified in the RFC
    - Manual keying is sort of assumed
- IPsec way too heavyweight

Some observations

- We're mostly concerned about blind attacks
  - A solution that only works against off-path attackers is OK
  - If we believe in on-path attackers we've got big problems
- We don't need a perfect solution
  - Attack doesn't have to be impossible
  - ... just a lot harder

Cheap solution 1: Nonce-keyed TCP MD5

- TCP MD5's problem was key management
  - But with off-path attackers it's easier
- Exchange nonces in the TCP SYN handshake
  - In some newly invented option
  - Use a function of the nonces as TCP MD5 key
    - e.g. H(Nonce_A || Nonce_B)
    - Diffie-Hellman is too heavyweight here
- Work factor for off-path attacker: 2^{128}
- Interoperability
  - What do intermediaries do with new options?
  - What about intermediaries that change data stream?

Cheap solution 2: Short nonce-keyed TCP MAC

- TCP MD5 option adds 18 bytes
  - MACs can actually be much shorter
    - 80 bits gives you a 2^{80} work factor
- Create a new option with a shorter MAC
  - Can also use something stronger like HMAC
- Slightly more implementation effort
- Interoperability
  - Mostly the same as with solution 1

Cheap solution 3: auto-keyed TCP MAC

- Do we really need 2^{80} bits of security?
  - Probably not
- What if we get our randomness from the ISNs?
  - Key = H(ISN_1 || ISN_2)
- Need to signal that we're using this
  - Probably easiest to use a new TCP MAC option
    - So might as well truncate
  - Maybe could have automatic detection with TCP MD5 option
- Security depends on ISN randomness
- Interoperability
  - Maybe easier since can be done with no new option
  - Still have a problem with intermediaries that change the data

Cheap solution 4: auto-keyed TCP checksum

- Maybe we don't need a MAC at all
  - Just modify the TCP checksum to include the ISNs
- Security not as good
  - Only 2^{16} bits
  - Maybe this is good enough
- Potentially more interoperable
  - Updating checksum works
  - Recomputing does not
  - Intermediaries which check the checksum are a disaster

Bottom line

- Best choice depends on intermediary behavior
  - Not well understood
  - … at least by me
- The more we worry about intermediaries the better the earlier (stronger) options look