Echo Cookie TCP Option

Bob Briscoe

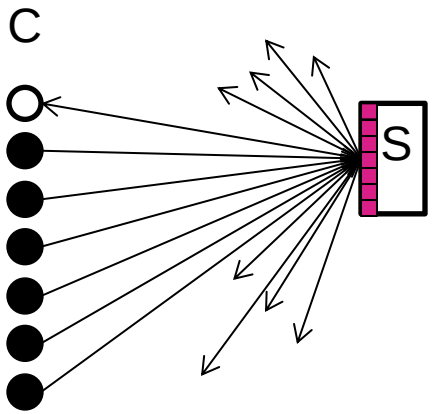
Nov 2014

Bob Briscoe's work is part-funded by the European Community under its Seventh Framework Programme through the Trilogy 2 project (ICT-317756)

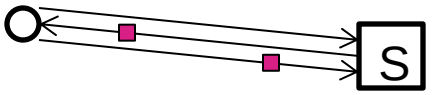
status

- draft-briscoe-tcpm-echo-cookie-00
  - initial individual draft
- arose from SYN-option-space extension work, but orthogonal
- separated out as focused draft
  - all SYN-option-space-extensions need something like this
  - replaces tcpcrypt SYNCOOKIE/ACKCOOKIE suboptions

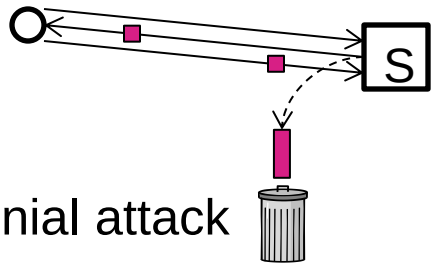
Problem



- SYN flood
  - exhausts TCP server's pending connection state
  - while SYN/ACK checks validity of source address
- SYN cookie,.. and friends
  - store server connection state in flight not in memory
  - still needed (despite some thinking server config is sufficient)
  - but... further problem
- 15 bits of cookie space
  - embedded in 16b initial seq no (ISN) and
    9 lowest significant bits of timestamp (if supported)
  - only enough for degraded max seg size, wnd scale & SACK-ok
  - plus some scope for server to infer other options it negotiated



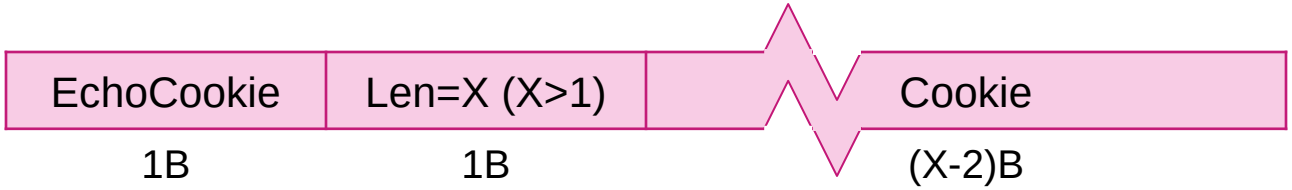
- with more, larger options on SYN: not enough space
  - with SYN-extension: *really* not enough space



- SYN flood becomes either connection state or option denial attack

Echo Cookie TCP Option

- underlying the space problem:
  - SYN cookie limited to fields that all TCP clients echo (ISN, TS)
- solution: a larger cookie jar
  - mandatory to implement with any new TCP option
  - and mandatory with extra SYN option space
  - ie. other options implicitly signal client support for EchoCookie
- the EchoCookie option
  - if host receives a cookie, it MUST reflect it back
  - sender can choose size and contents



- client MAY include 2-octet EchoCookie flag option on SYN
  - e.g. when using options that do not signal implicit support

security considerations (discuss on list pls)

- if client negotiated state using a secured protocol
  - cookie MUST be echoed with at least as strong security
- could be used as a reflection attack?
  - SYN/ACK MUST NOT exceed size of SYN
  - no need to include data in SYN within cookie
    - server not ACKing the data causes a retransmit anyway
    - TFO cookie serves as proof the source address is valid
  - server can/SHOULD rate-limit to repeated and/or unresponsive source IPs?
- server SHOULD only use when under stress?
- mechanism server uses to verify returned cookie?
  - no need to standardise?
- any other new attack vectors?

next steps

- security discussion pls
- applicability:
  - solely SYN/ACK – ACK?
  - solely server-client-server?
  - any segment?

- intended status: proposed std?
- adoption?