draft-touch-tcp-antispoof

Joe Touch
Postel Center Director
Research Assoc. Prof. CS & EE
USC/ISI

Purpose of this ID

- Outline the ‘current’ problem
- Make taxonomy of solutions
- Make recommendations (?)
- Hint at underlying opportunity (?)
  - May or may not be right for this problem

- ->> BCP?

TCP RST Vulnerability

- Aug. 1998
  - Already Standards-Track

Quick summary of 2385

- TCP checks seqno's
  - Exact on SYNs
  - Within window elsewhere
- Increased window size = big opportunities
  - RSTs in the window kill the connection
- Sol'n is authentication
  - Preshared secret + MD5 TCP option
  - Currently missing keying

What needs to be known?

- IP addresses
  - Dest is announced for servers
  - Some pairwise associations are known
- Ports
  - Dest is (usually) fixed by protocol
  - Src can be predicted or guessed
- Whether segment is "*in the window*"
  - Increases as BW increases

What Changed?

- Bandwidth * delay product
  - Vulnerability = f(BW^2)
  - Higher BW*delay = larger window
  - Higher BW = more attack RSTs can be sent
- Long-lived fragility
  - Persistent BGP connections
  - Well-known endpoints, port
  - BGP interpreting dropped TCPs

Vulnerability as BW^2

BW	BW*del (MB)	RSTs needed	Time
10 Gbps	125	35	1 us
1 Gbps	12.5	344	110 us
100 Mbps	1.25	3,436	10 ms
10 Mbps	0.125	34,360	1 sec
1 Mbps	0.0125	343,598	2 min
100 Kbps	0.00125	3,435,974	3 hours

Proposed Solutions

- Explicit protection
  - TCP/MD5
  - IPsec/IKE
- Obfuscation
  - Window Attenuation
    - RST Attenuation, Timestamps
  - Larger number space
    - Cookies / ISN / conn. IDs, Port randomization

Issues

- Transport vs. net vs. applic.
  - Where to protect attacks on identity?
  - Need to modify all transport protocols
  - Per connection, not per endpoint pair
- Complexity
  - Configuration effort, CA hierarchies
- Performance
  - Throughput impact, CPU load impacts

High-Perf. Anon. Security
(TBP in SAAG Thurs.)

- *What*: To maintain anon. associations
  - Public servers (non a-priori clients)
  - Deliberately anonymous clients.
- *How*: Pairwise keys w/o shared secrets
  - Diffie-Hellman only
- *How fast*: Variety of modes
  - Cookie
  - First-block (header only)
  - Full