R2 "Off-Path TCP Exploits: Global Rate Limit Considered Dangerous"

Target system

OS whose TCP stack implements RFC 5961 faithfully with global late limit in sending challenge acks

As of today, Linux 3.6+ and <4.7 has the vulnerability

TCP Communication parties with the vulnerable OS (at least one of them) can be exploited to session termination and TCP session hijacking.

Vulnerability

A side-channel vulnerability which leaks if any challenge ack is invoked at the host.

The linux kernel has a global system variable to limit the number of challenge acks generated per second; existence of sent challenge ack in a time window can be inferred by elaborative generation of challenge acks to the limit from the off-path attacker's session.

Exploitation

The existence of ongoing connection can be inferred by exploiting the side-channel vulnerability with the SYN receiving scheme; this attack gives the client port number to the attacker.

The sequence number is inferred by the same vulnerability with the RST receiving scheme; if the sequence number (i.e., RCV.NXT) is known, the connection can be reset by the attacker.

The acknowledgement number is also inferred with Data receiving scheme; if both seq and ack numbers are known, malicious data can be injected to the ongoing TCP session.

Evaluation and experimental method

For completeness of the attack, time synchronization and search schemes of number spaces are devised and evaluated to show that they can be completed in a timely manner.

It is shown that SSH connections to Amazon EC2 servers can be reset with the proposed attack in a minute.

It it also shown that Tor relay nodes can be also reset in a similar time scale and it further implicates unavailability of the service or lack of privacy guarantee.

As to the hijacking attack, it is shown that desynchronization of existing TCP connection and injecting a malicious payload is practically feasible with a short attack time and high probability by mounting the the hijacking attack to the USA today website.

Defense

Like other network configuration variables, the limit variable can be defined per TCP session and also globally. The per-session limit should be orders of magnitude smaller than the global limit.

In TCP receiving schemes, it is not harmful to probabilistically challenging with acks. If challenge acks are not deterministically generated, the attacker cannot exploit the side-channel vulnerability. The side effect of such probabilistic challenge ack would be late connection termination (possibly consuming the TCB in the kernel for a while) or late triggering for a new data if previous acknowledgements are lost.

Finally, eliminating challenge ack is indeed a solution; there is no amplification effect as an acknowledgement packet as small as 64B and the attacker also needs to send at least 64B size of packet to trigger the challenge ack.
IP source validation, e.g., egress filtering and firewalling, can prevent spoofing attack in the first place.

Future work

There are a plenty of system-wide global variables in the modern OSes; we may inspect them for potential side-channel vulnerabilities.
We may devise a network-level defense system against side-channel vulnerability exploits.
For example, continuous guessing a wide range of source port numbers or sequence numbers can be detected in the network-level.