RGW Security threats

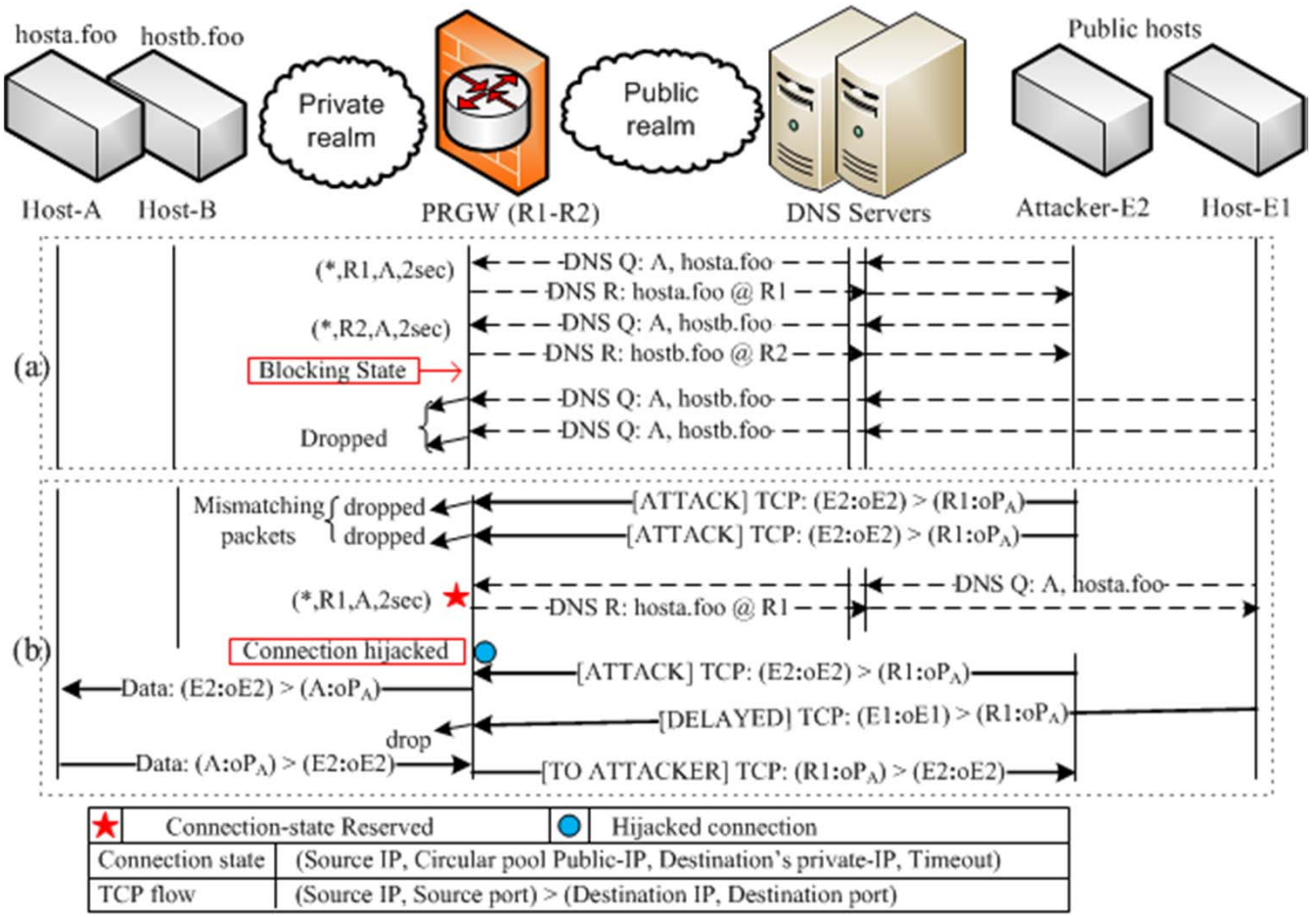
- RGW acts as NAT , for outbound connections.
- It employs a Circular Pool of addresses to provide Internet hosts with connectivity to the private-hosts.
- By dynamically assigning an address from the circular pool, RGW prevents the hosts in the private realm from direct exposure to the Internet, compared to static-NATs.

- However, Circular Pool can be vulnerable to inherent Internet weaknesses: *DNS Floods* and *Connection hijacking* from Internet hosts.

RGW Security threats

DNS Flood: result in depleting the circular pool, resulting in DoS to legitimate users

Connection Hijack: SYN flood from attacker claims the state reserved by a legitimate host, causing DoS to the host.



Security principles

- Flow acceptance must be limited to verifiable sources to tackle source address spoofing and prevent exhaustion of the Circular Pool address space.
- UDP flow initiations shall be admitted after the connection is signaled through a secure channel e.g. SIP(S).
- Under the network stress, access shall be granted based on the source reputation.
- Security shall not require changes to the end-hosts or protocols, in order to ease the deployment.

Security against DNS Floods

- *Rate-limiting*: Simultaneous DNS queries from a DNS server or to a private-host are rate limited.
- *DNS Server classification*: DNS servers are classified into white/grey/blacklists. Whitelisting is agreed with clients based on service level agreements (SLA) and whitelist servers are pre-configured in RGW. By default, the rest of name servers are classified *grey*.
- *Circular pool address allocation*: limits the name queries from grey servers to a portion of the circular pool address space, ensuring that under network stress white (reputed) servers can access to CPOOL resources. A server exceeding its SLA is blacklisted for 'T_O' by CPool.

DNS Security Heuristics

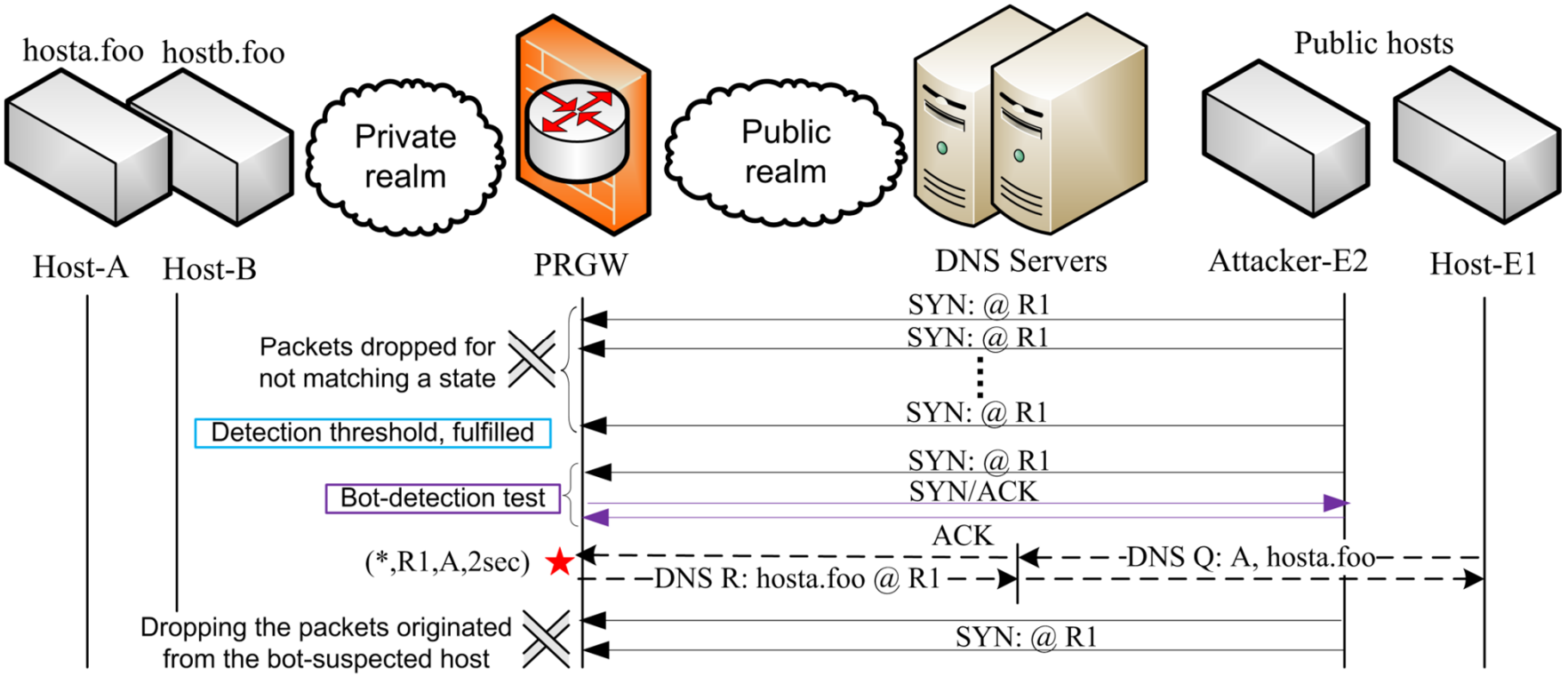
- *Active Monitoring:* The circular pool monitors the arrival rate and success or failure rates of the name queries. This allows dynamic re-classification of the servers that carry attack/flood traffic towards RGW.
- To prevent circular pool from direct exposure to DNS floods, we developed a *DNS-Relay* front-end that forwards the domain queries towards RGW.
- This attempts at security abstraction comparable with DNS reverse proxies, and delegates security against DNS floods to a dedicated entity, i.e. blacklisting an aggressive sender.

Bot-Filtering Algorithm

- In contrast to the networking elements that simply drop the packets mismatching to a flow or a connection state.
- The arrival of the first packet (TCP SYN) to a state that does not exist is monitored for detecting connection-hijack attempts.
- Once a threshold number of such packets from a sender are dropped in time T_0, RGW responds to the next mismatching packet with SYN-ACK bearing a cookie in the Sequence field.
- A subsequent ACK bearing the sent cookie establishes the sender as non-spoofed (bot). Following which, RGW blacklists the sender for 'T_0' and prevents claiming connection state.

- The detection threshold shall be dynamically adjusted to prevent an abuse of this mechanism.

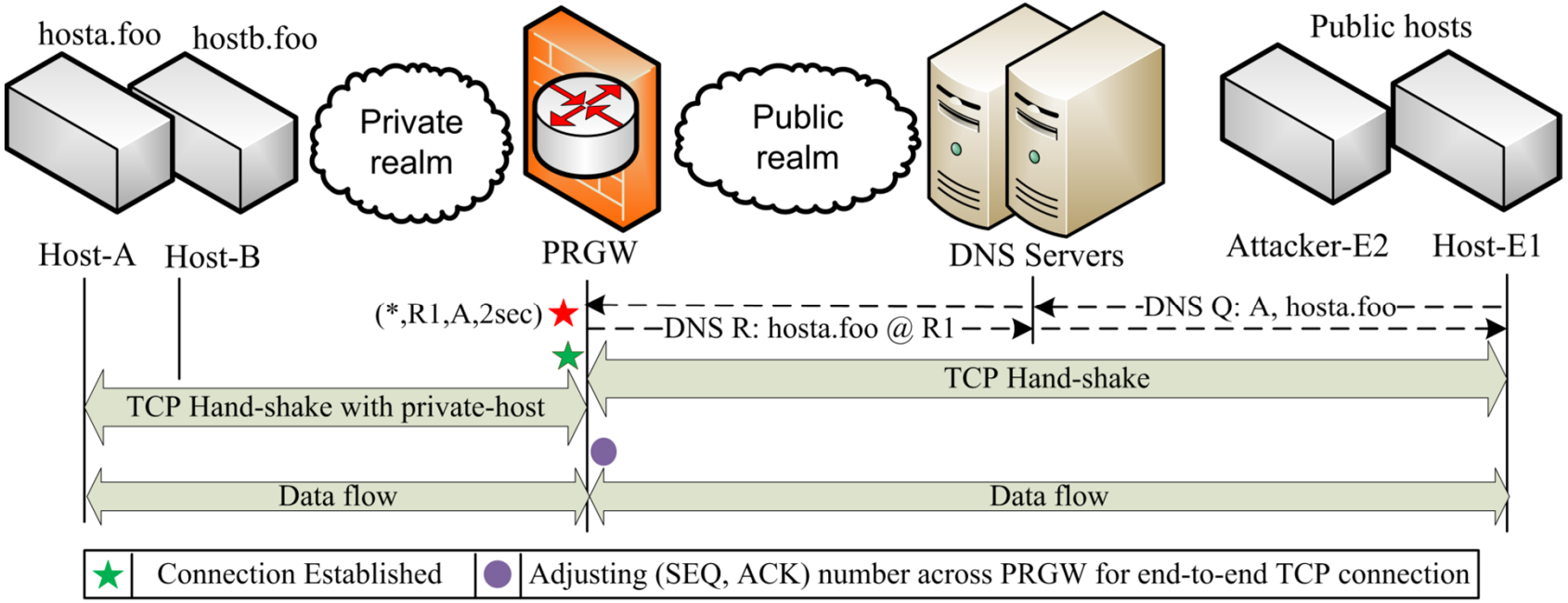
Bot-Filtering Method



Security against Connection Hijacks

- *Bot-Filtering* algorithm attempts security against connection hijacking possibility from the Internet bots.
- *TCP-Splice* prevents connections in circular pool against hijack attempts from spoofed sources.
- For Internet originated incoming connections, the RGW may challenge the sender of the TCP SYN with a cookie encoded in Initial Sequence Number (ISN) of SYN-ACK. Following the success of TCP-handshake, the sender is ascertained as non-spoofed and the connection is accepted. The following data packets between source and destination are forwarded using TCP-splicing principles.
- The sender with dubious past (or history of misbehavior) will fail to claim the connection, due to bot-filtering algorithm.

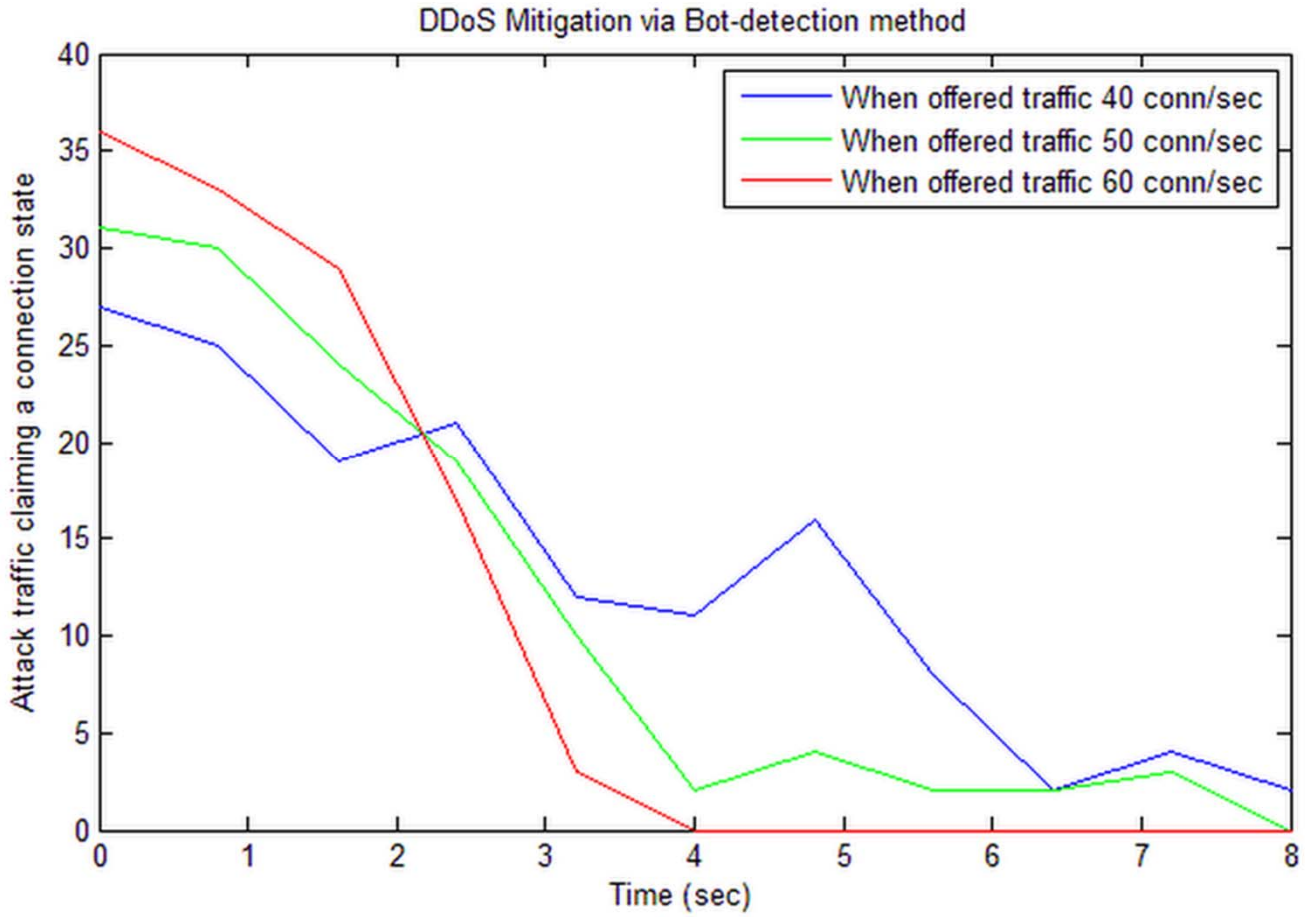
TCP-Splice


hosta.foo
hostb.foo
Private realm
Public realm
Public hosts
Host-A
Host-B
PRGW
DNS Servers
Attacker-E2
Host-E1
(*,R1,A,2sec)
DNS R: hosta.foo @ R1
DNS Q: A, hosta.foo
TCP Hand-shake
TCP Hand-shake with private-host
Data flow
Data flow
Connection Established
Adjusting (SEQ, ACK) number across PRGW for end-to-end TCP connection

Results

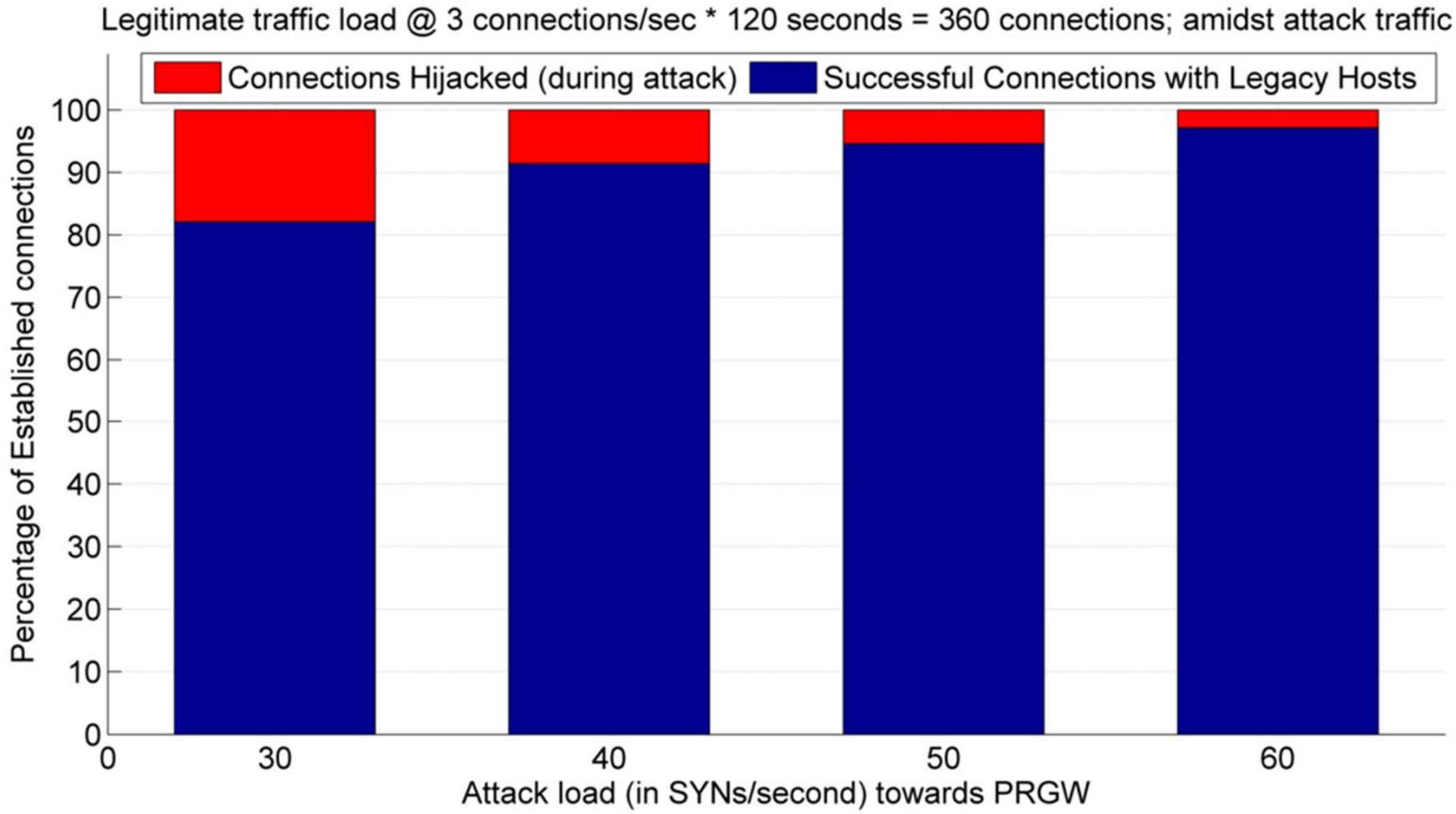
PRGW enabled network
2 Internet hosts:
1 Legitimate host, 1 Attack host

Attack load (in SYNs/second)
Non-spoofed addresses: 8



- The attack load (from non-spoofed) bots dampens with time, as a virtue of Bot-filtering
- Filtering the attack sources enables legitimate hosts to claim their connections

Results



RGW prototype serving a private-realm receives a legitimate connection load of 3 connections/second. Meanwhile, eight non-spoofed (bot) addresses offer a combine attack load (in SYNs/sec)

Thank you!

Questions?