DNS based DDoS mitigation solution

Applying UDP & Source Code

Sukhun Yang

Seoul National University

August 4, 2023

Contents

Table of Contents

Step 1

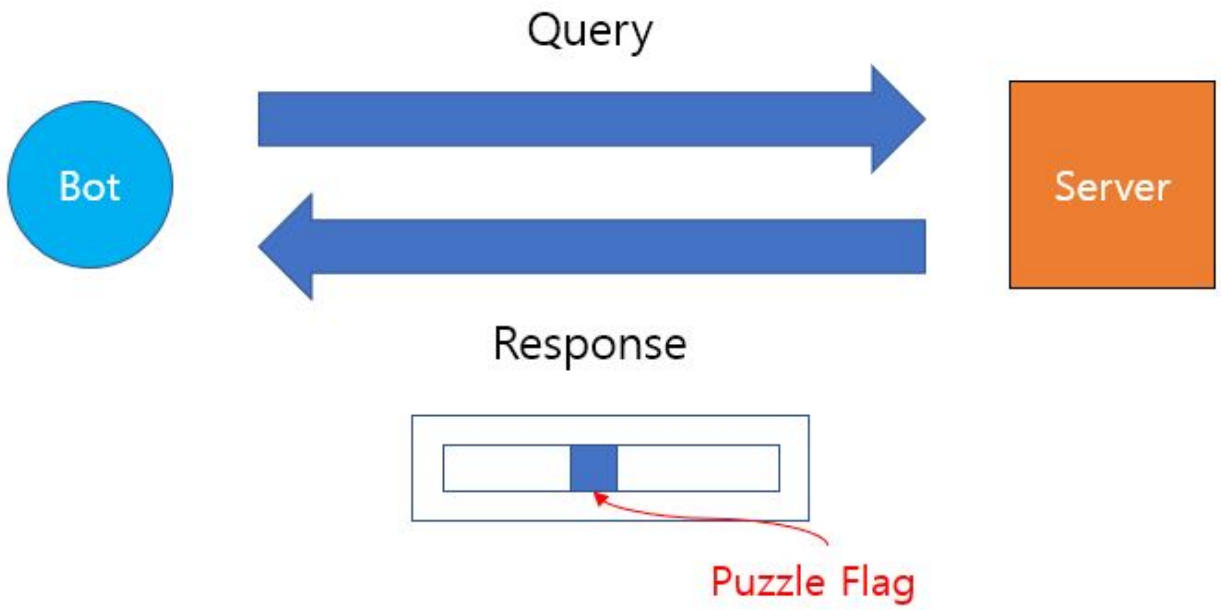

Figure 1: Step 1

Step 1. Target server detects DDoS attack and sets puzzle flag

Step 2

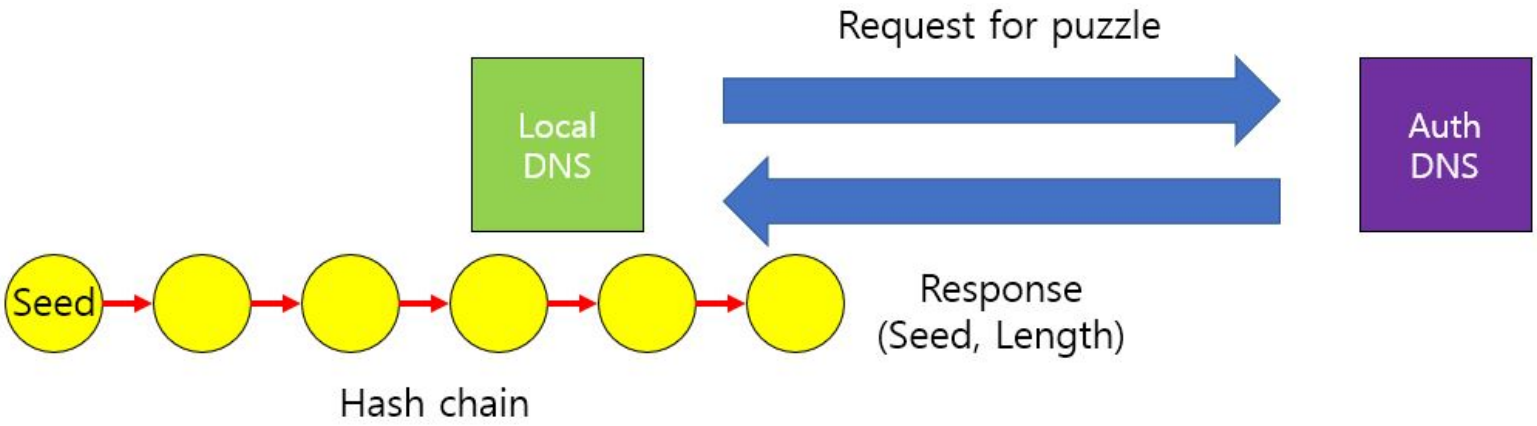

Figure 2: Step 2

Step 2. Local DNS server get seed and length from authoritative DNS server and generate

Step 3

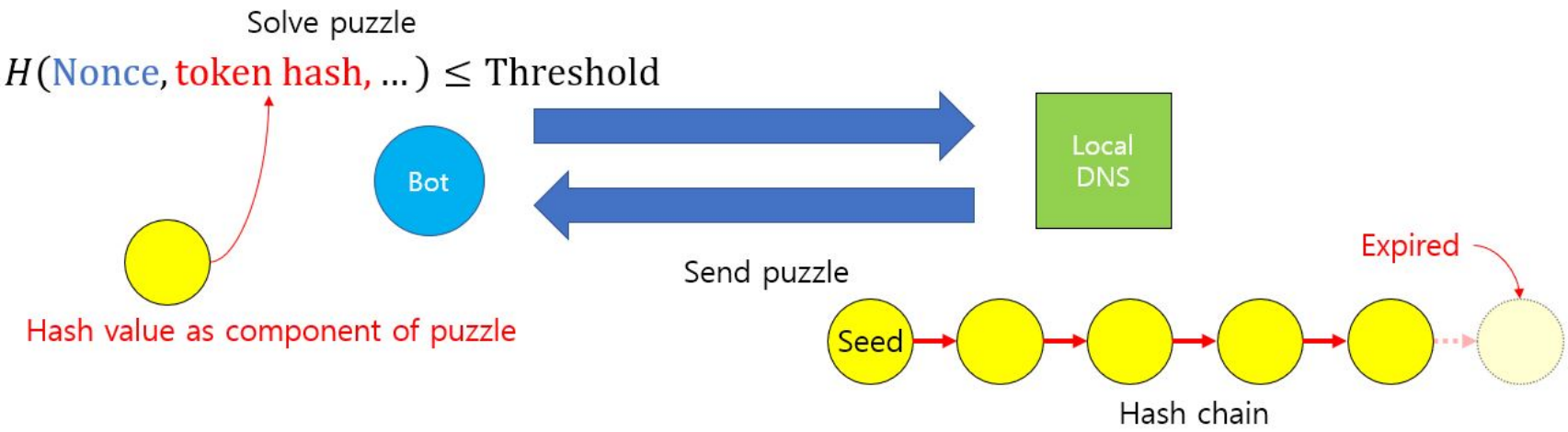

Figure 3: Step 3

Step 3. Local DNS server give seed value to user when user request information

Step 4

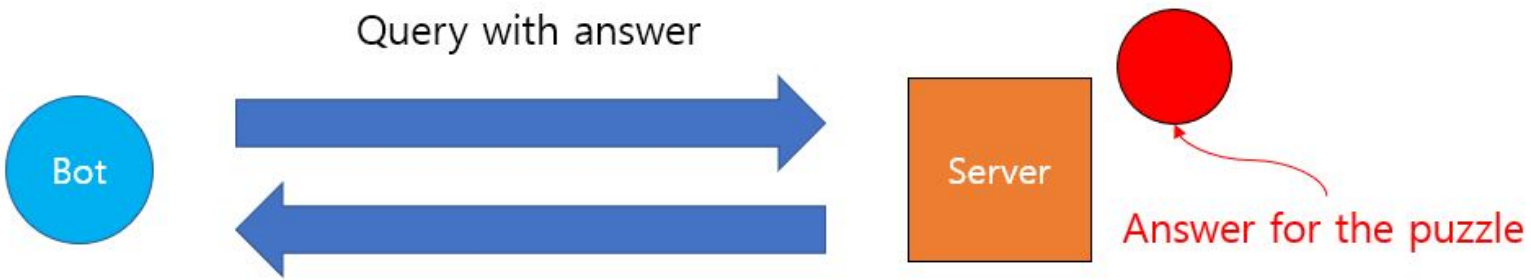

Figure 4: Step 4

Step 4. User solve the puzzle and send answer to target server

Step 5

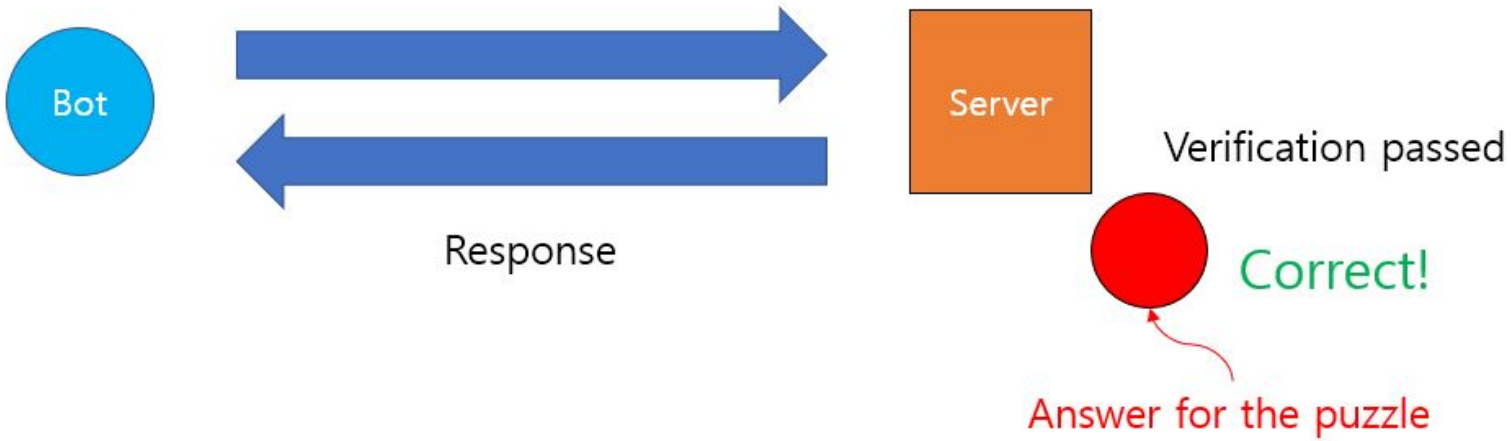

Figure 5: Step 5

Step 5. Query processing after checking whether the Puzzle and Solution are valid in the target server

Logical diagram

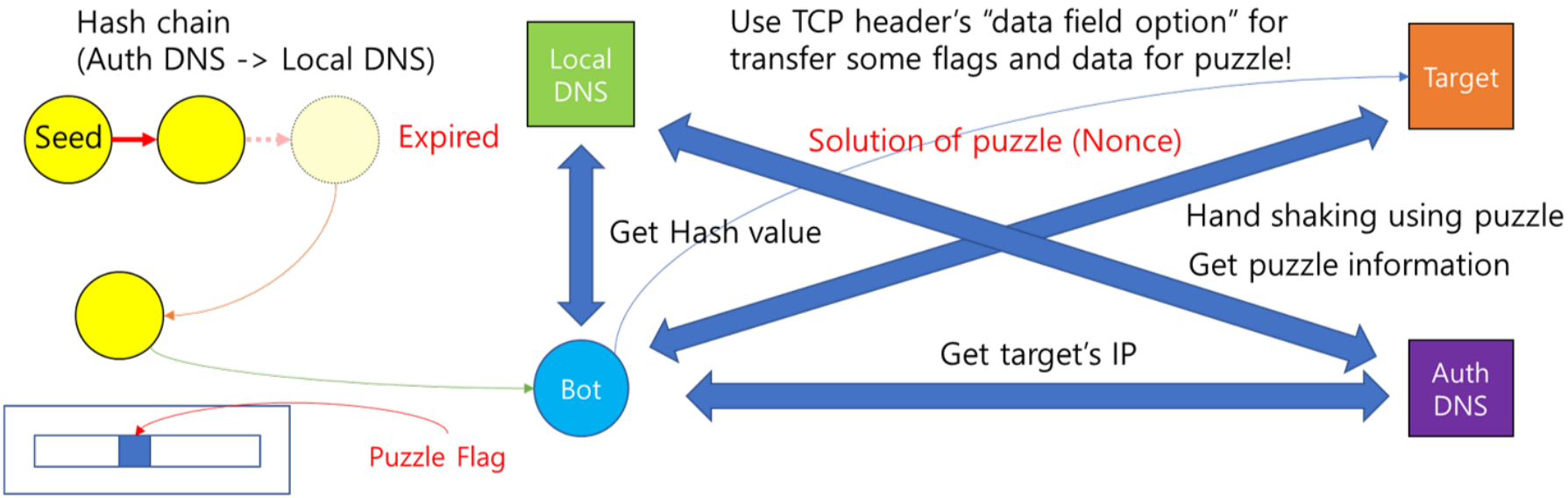

Figure 6: Logical diagram

TCP handshaking

Slightly modified for transmission and verification of puzzle data in the TCP handshaking process.
In the figure on the right, the black line is the existing TCP handshaking process, and the red line is the newly added process.

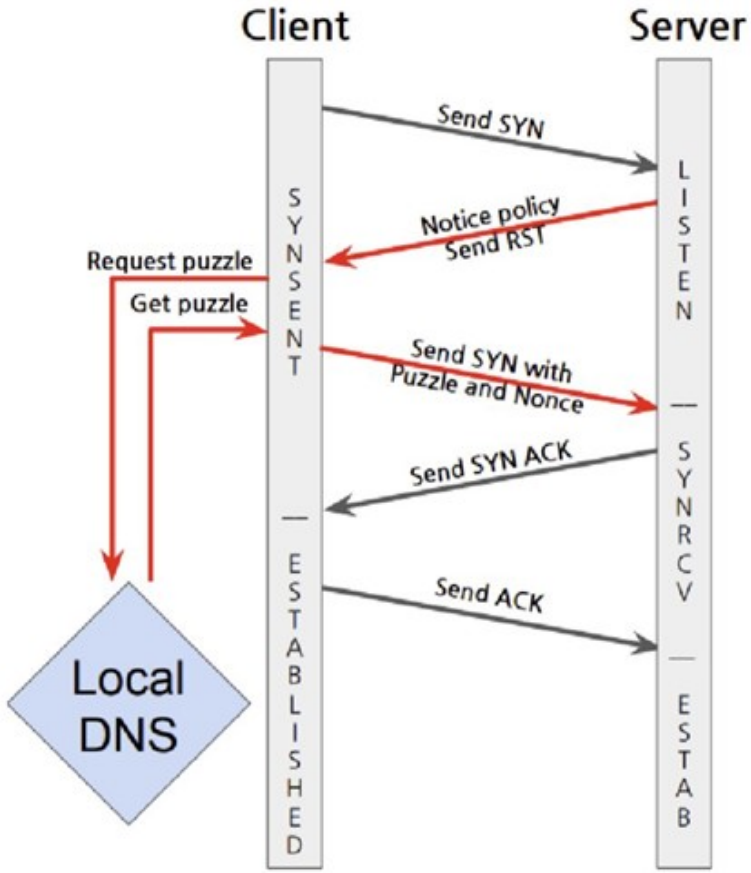

Figure 7: TCP modified handshaking

Implementation issues

1. Hooking using module
   ⇒ Unable to access tcp handshaking due to security regulations
2. Modify many functions and structures directly at linux kernel
   ⇒ Basically, it is call-by-value, so puzzle options are omited
3. In the case of TCP communication, this communication goes up to the IP layer, and rather, the values of IP header and buffer are used for actual communication.

New step 1

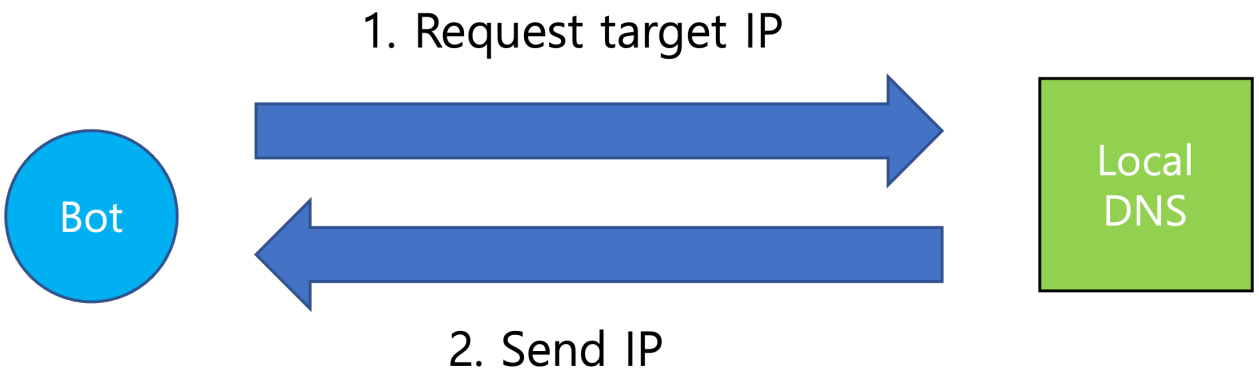

Figure 8: New step 1

New step 2

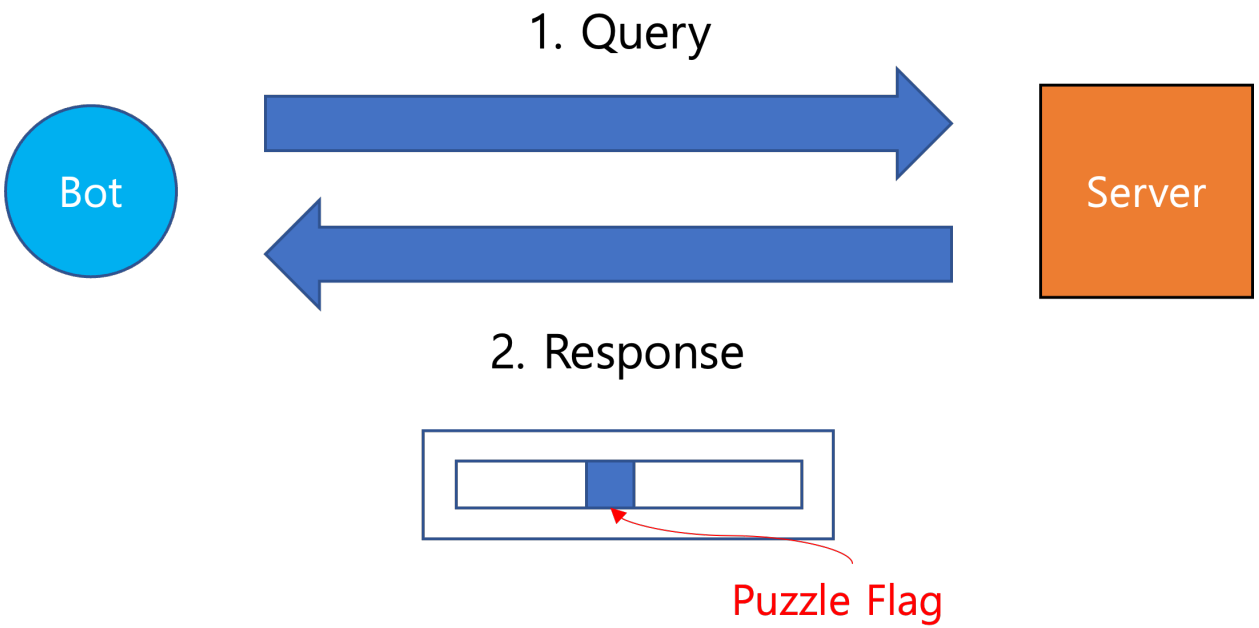

Figure 9: New step 2

New step 3

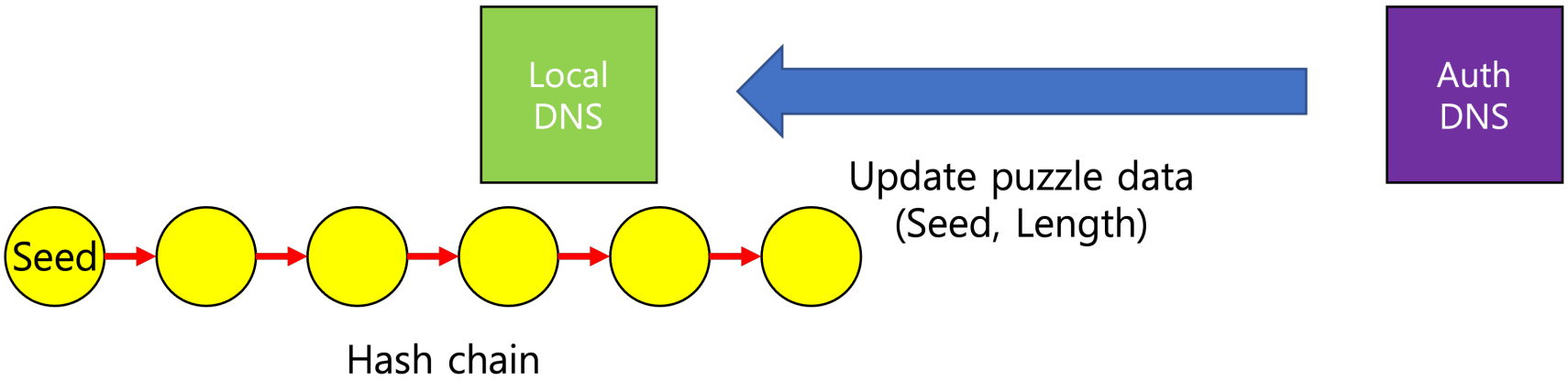

Figure 10: New step 3

New step 4

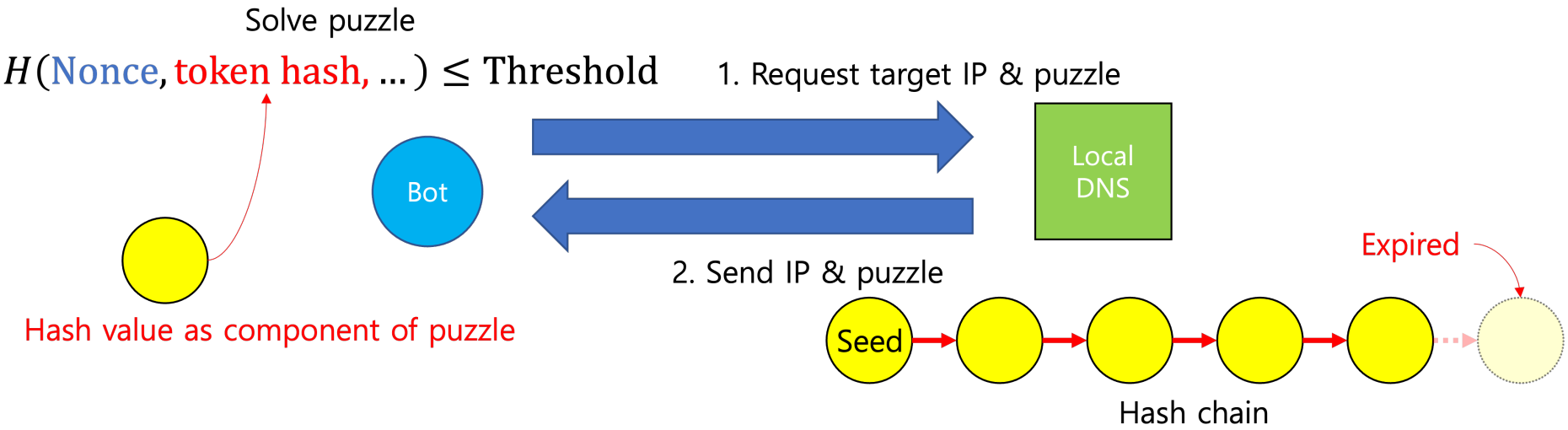

Figure 11: New step 4

New step 5

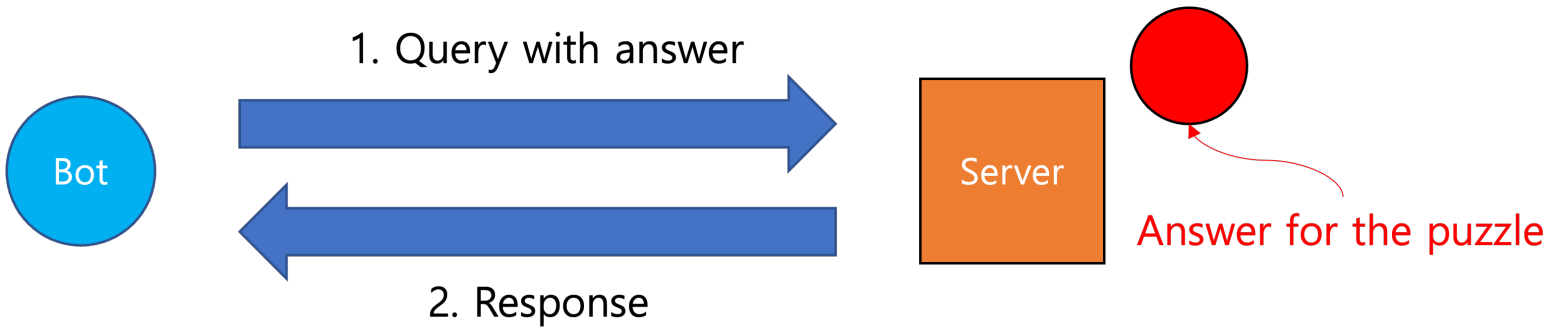

Figure 12: New step 5

New logical diagram

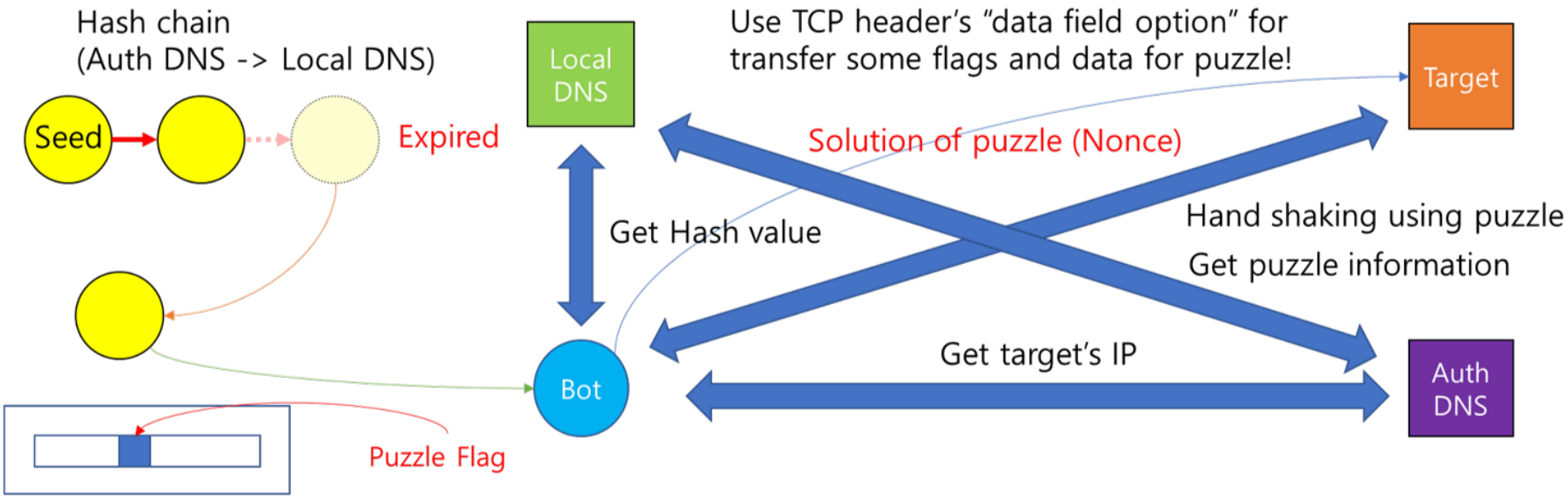

Figure 13: New logical diagram

Pros & Cons

Pros

1. TCP proceeds after UDP terminates
2. All processes are expected to be available in the kernel
3. Can be processed in one communication

Cons

1. Updating each local DNS consumes resources
2. Realistic...?

Also, conversely, what if local DNS asks auth DNS for a puzzle every time?

Table of Contents

Presentation video

CID2 Midterm Presentation Video

https://www.youtube.com/watch?v=ia6vt7F9ONk

Explanation of code

Table 1: Modified kernel part

net/ipv4/tcp_ipv4.c	• Change the order of function calls • tcp_v4_send_reset(): Add puzzle information
net/ipv4/ip_output.c	• ip_send_unicast_replay(): Add puzzle information

Explanation of code

Table 2: Modified kernel part

net/ipv4/tcp_input.c	• TCP header parsing • tcp_rcv_state_process(): Check puzzle & nonce • tcp_rcv_synsent_state_process(): Add reset signal • tcp_synsent_state_process(): Update puzzle info. and re-send

Explanation of code

Table 3: Modified kernel part

net/ipv4/tcp_output.c	• __tcp_transmit_skb(): Add puzzle info.s • Update TCP header writing
net/puzzle.c	• Add system calls • Insert puzzle hash function

Source code

Modified Linux Kernel Source

https://github.com/minjun0305/linux_for_practice

Demo Video

https://www.youtube.com/watch?v=dnjCd09DRFc

Source code

Currently Working

https://github.com/Sagit25/DNS-based-DDoS-mitigation