draft-xu-intarea-challenge-icmpv4-00
draft-xu-intarea-challenge-icmpv6-00

Enhancing ICMP/ICMPv6 Error Message Authentication Using Challenge-Confirm Mechanism

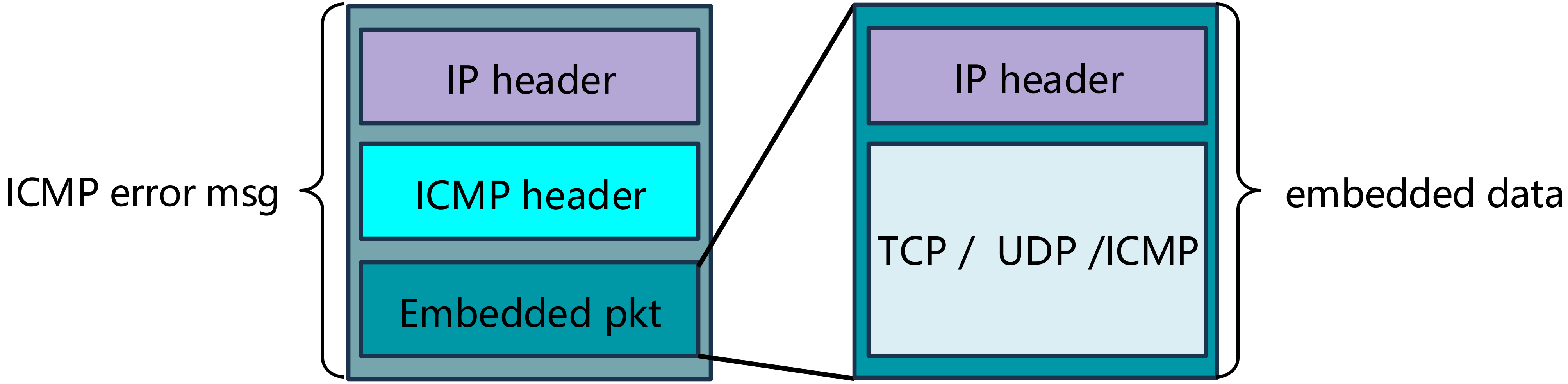
Presenter: Ao Wang, Yuxiang Yang

March 2025

Problem Statement

ICMP Error Message

- ICMP error message
  - ICMP error messages, defined in RFC 792 / 4443, are used to **report network errors**, aiding in network diagnostics and troubleshooting.
- Verification of ICMP error message
  - Verification of ICMP error messages involves verifying the integrity and accuracy of these messages to accurately reflect network issues.



Specifications on ICMP Error Message

- RFC 792 / 1122 specifies:
  - Every ICMP error message **includes the Internet header and at least the first 8 data octets of the datagram** that triggered the error; more than 8 octets MAY be sent; this **header and data MUST be unchanged** from the received datagram.

- RFC 1812 specifies:
  - The ICMP datagram **SHOULD contain as much of the original datagram as possible** without the length of the ICMP datagram exceeding 576 bytes.

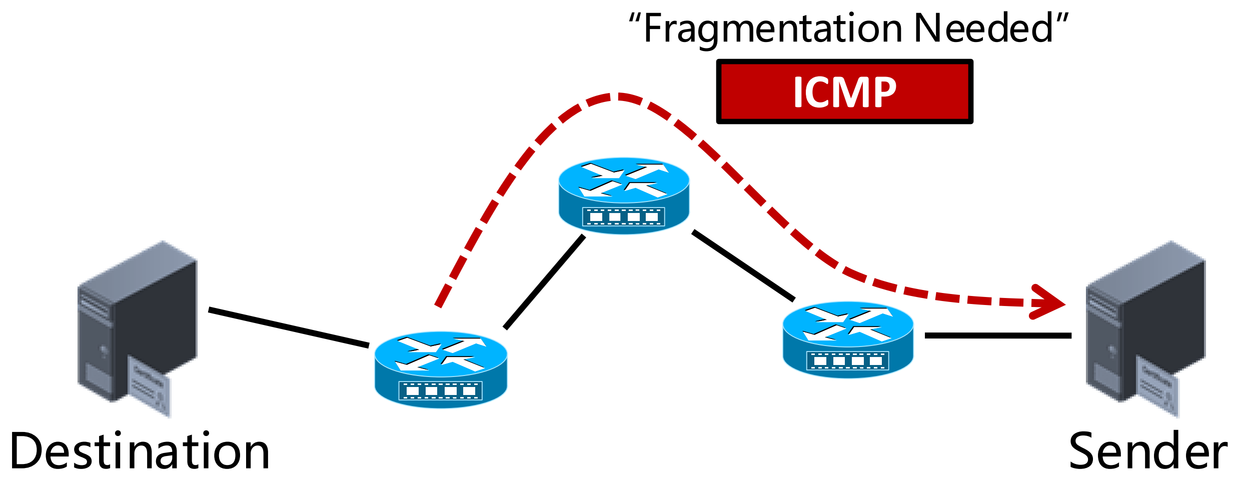
- RFC 4443 specifies:
  - Every ICMPv6 error message (type < 128) **MUST include as much of the IPv6 offending (invoking) packet (the packet that caused the error) as possible** without making the error message packet exceed the minimum IPv6 MTU.

Problem

Current ICMP/ICMPv6 specifications have inherent limitations that allow off-path attackers to forge ICMP error messages.

□ Lack of Source IP Address-Based Verification

◆Certain ICMP/ICMPv6 error messages can originate from **any intermediate router** along the packet path.



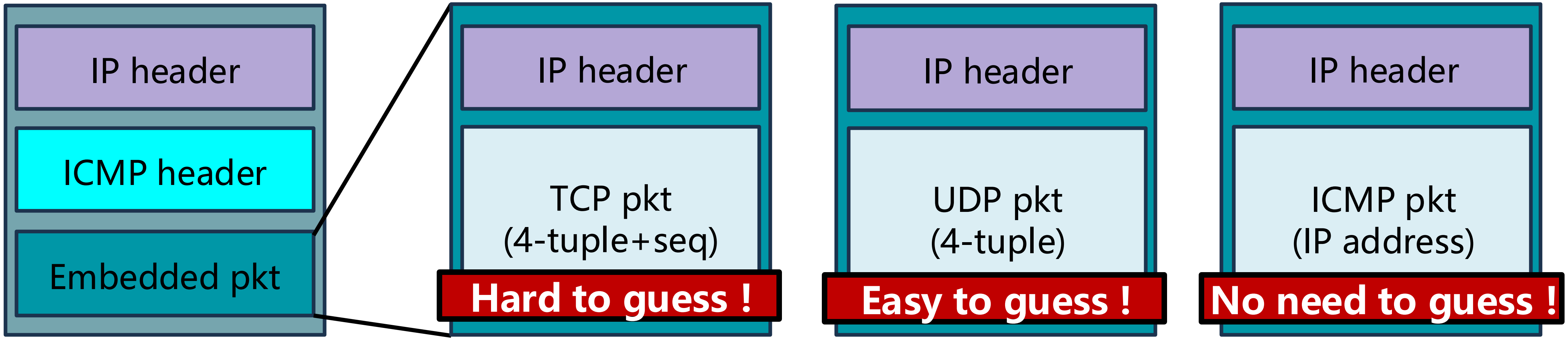
Problem

Current ICMP/ICMPv6 specifications have inherent limitations that allow off-path attackers to forge ICMP error messages.

□ Check on Embedded Packet is Bypassable

◆For stateful embedded packets (e.g., TCP), **hard to bypass**.

◆For stateless embedded packets (e.g., UDP, ICMP), **easy to bypass**.



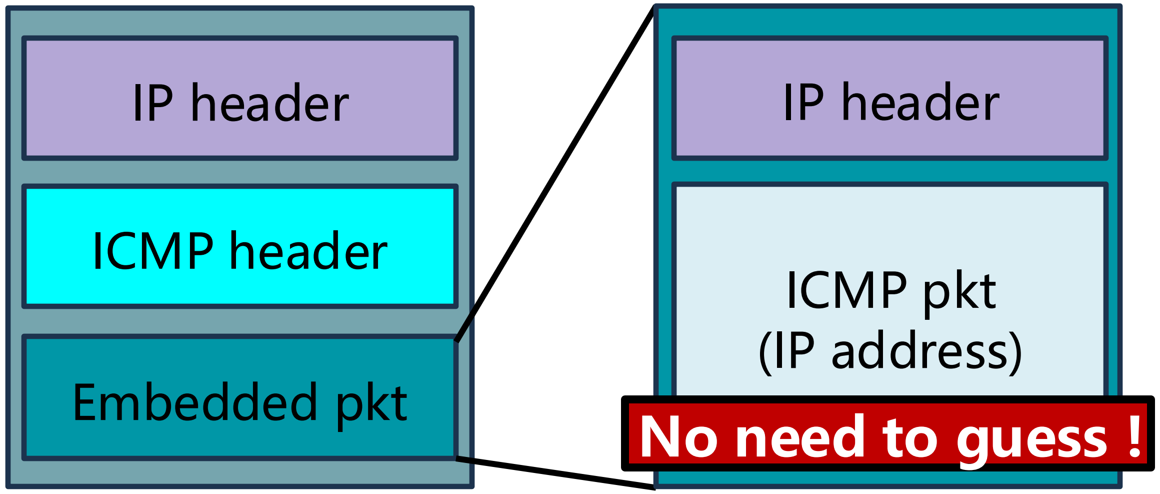
Problem

Current ICMP/ICMPv6 specifications have inherent limitations that allow off-path attackers to forge ICMP error messages.

□ Check on Embedded Packet is Bypassable

◆For stateful embedded packets (e.g., TCP), **hard to bypass**.

◆For stateless embedded packets (e.g., UDP, ICMP), **easy to bypass**.

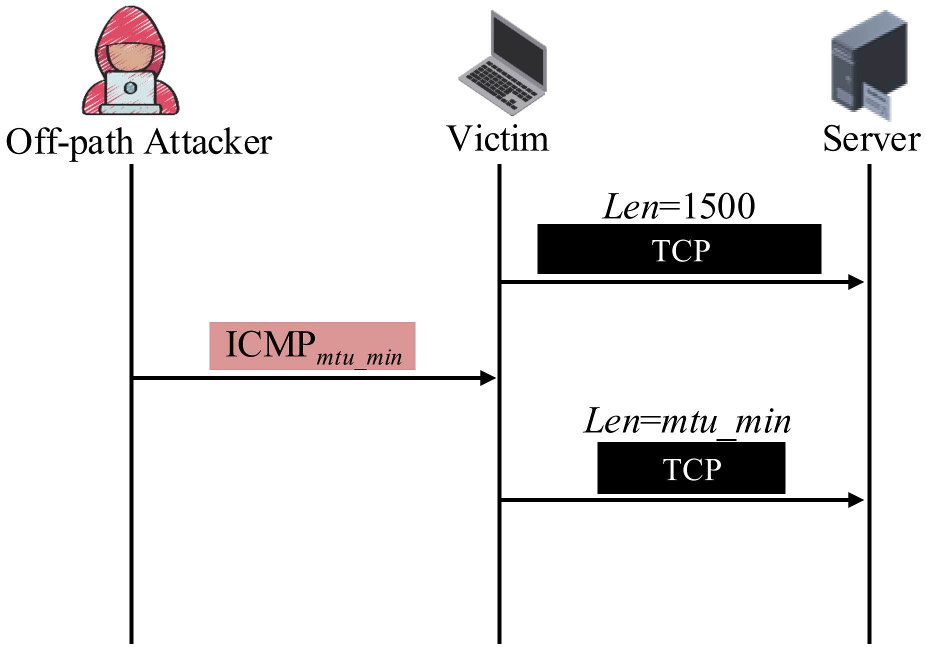


Can impact all upper-layer protocols !

Attack Cases

□ ICMP Fragmentation Attack

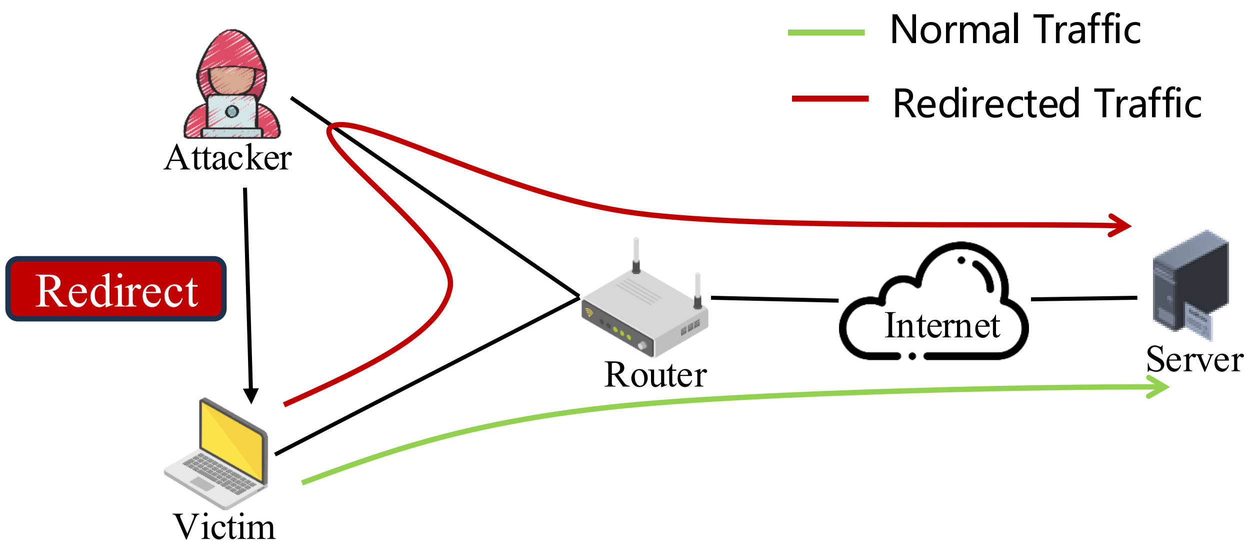
- ◆Forge ICMP "Fragmentation Needed" messages to lower hosts Path MTU.



[1] Feng, X., Li, Q., Sun, K., Fu, C., and K. Xu, "Off-path TCP hijacking attacks via the side channel of downgraded IPID"

□ ICMP Redirect Attack

- ◆Forge ICMP Redirect messages to tamper with a victim's gateway, enabling Man-in-the-Middle (MitM) attack.

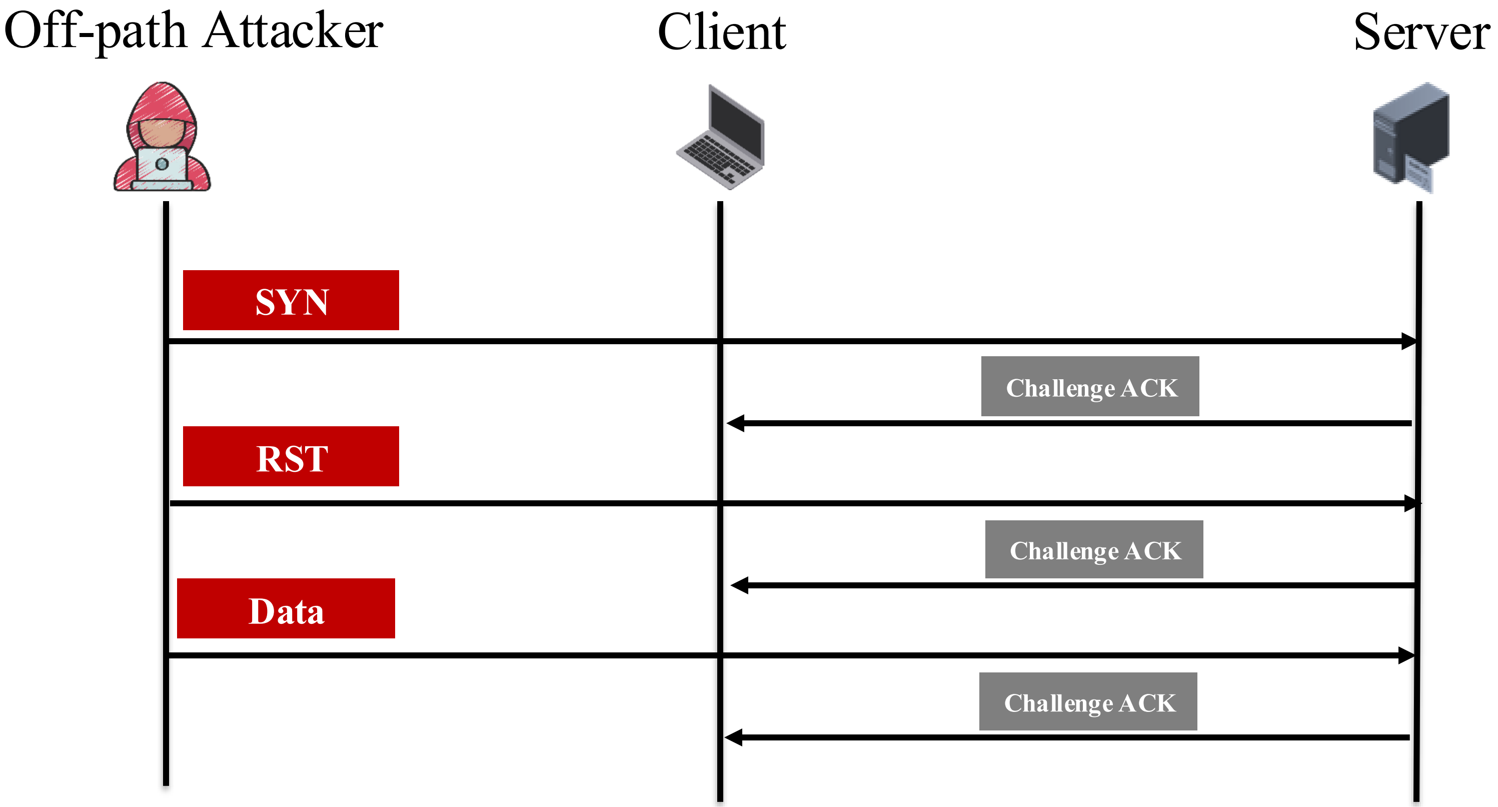


[2] Feng, X., Li, Q., Sun, K., Yang, Y., and K. Xu, "Man-in-the-middle attacks without rogue AP: When WPAs meet ICMP redirects"

Proposed Solution

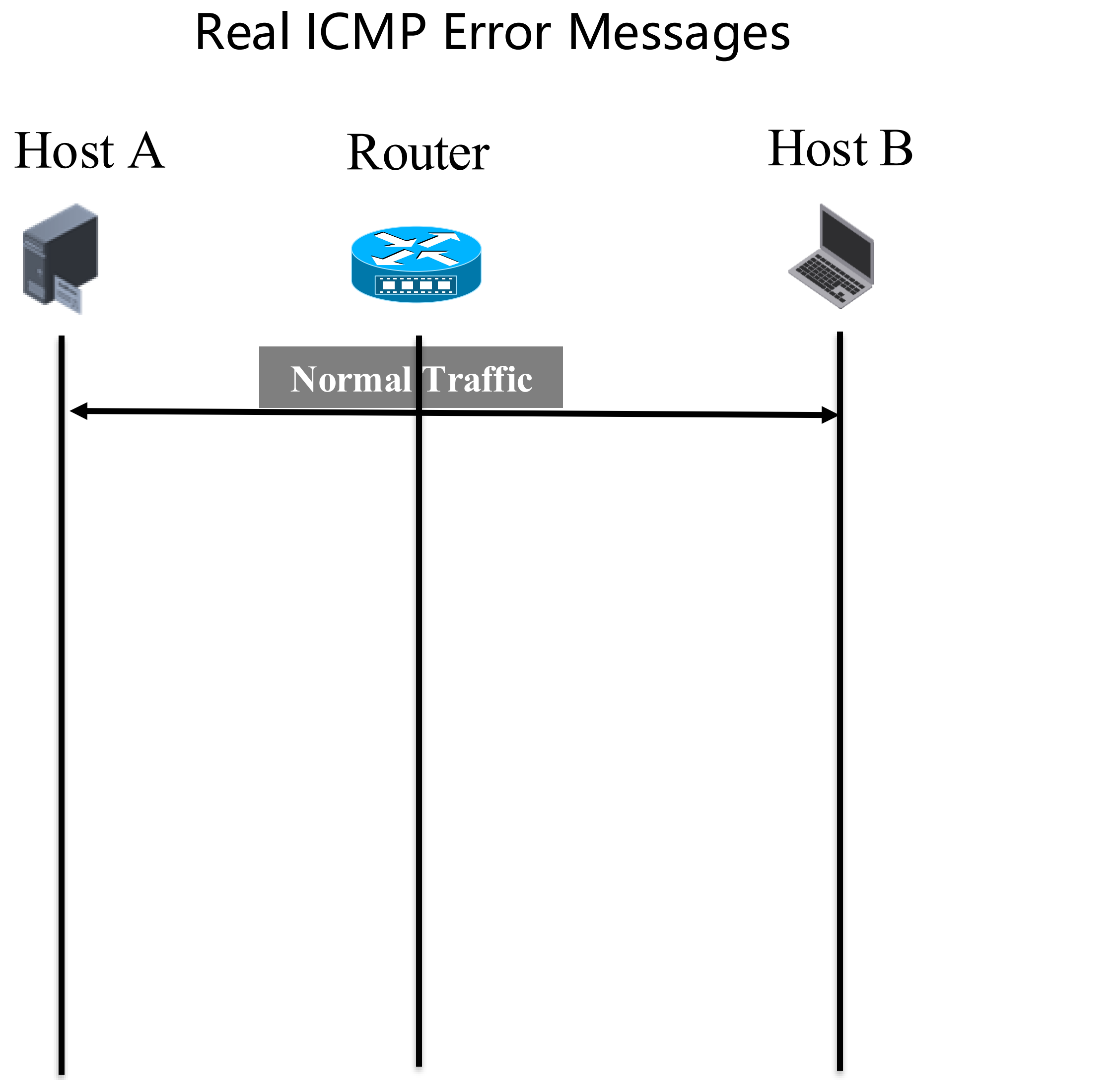
Inspiration: TCP Challenge ACK Mechanism

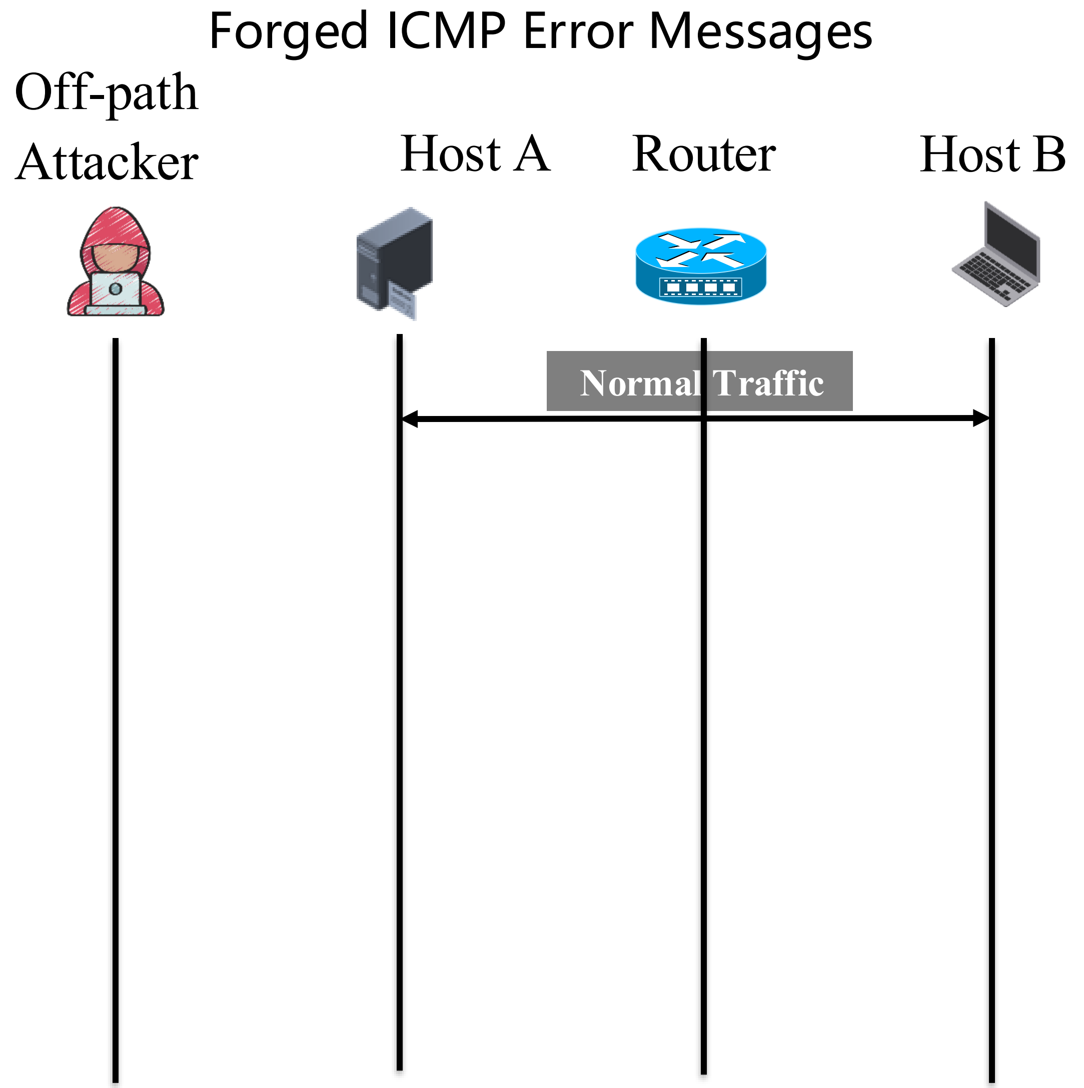
- □ RFC 5961： Proposed to defend against blind in-window TCP attacks by sending **challenge ACK packets** to the peer.



Can we design a similar mechanism by double-checking ICMP error messages to enhance the security of ICMP protocol?

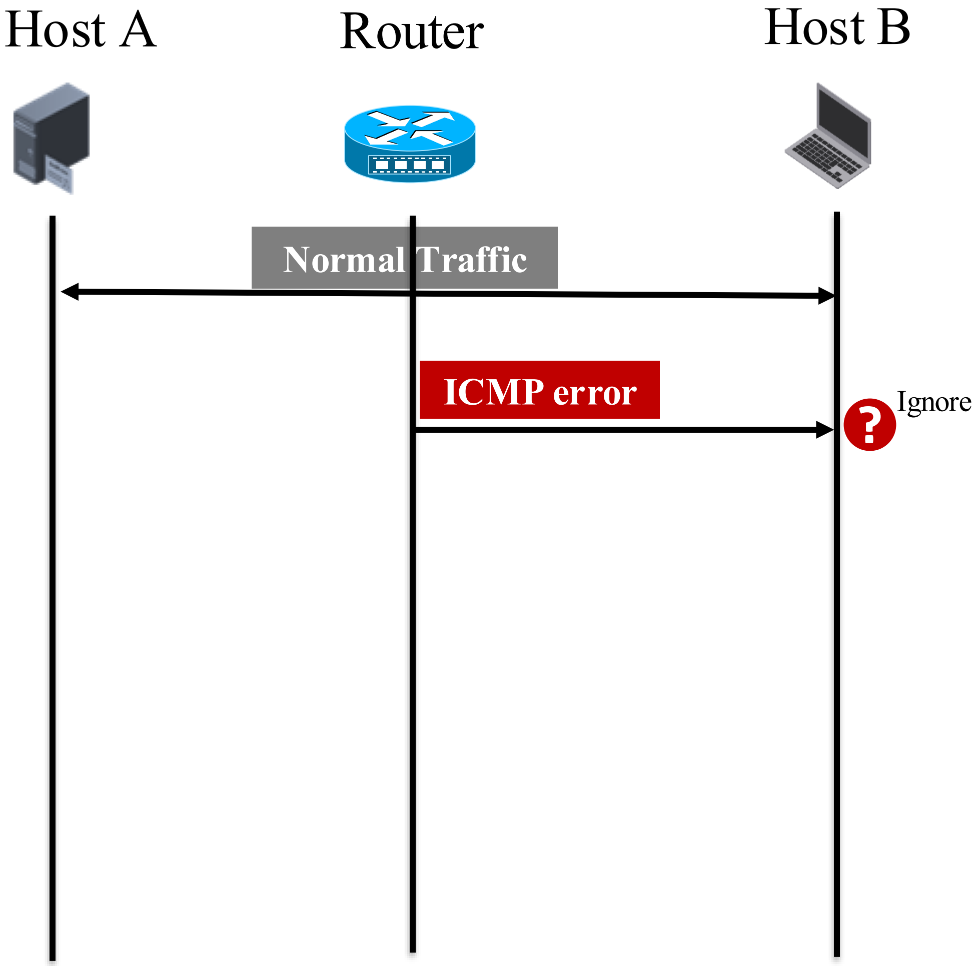
The ICMP Challenge-Confirm Mechanism



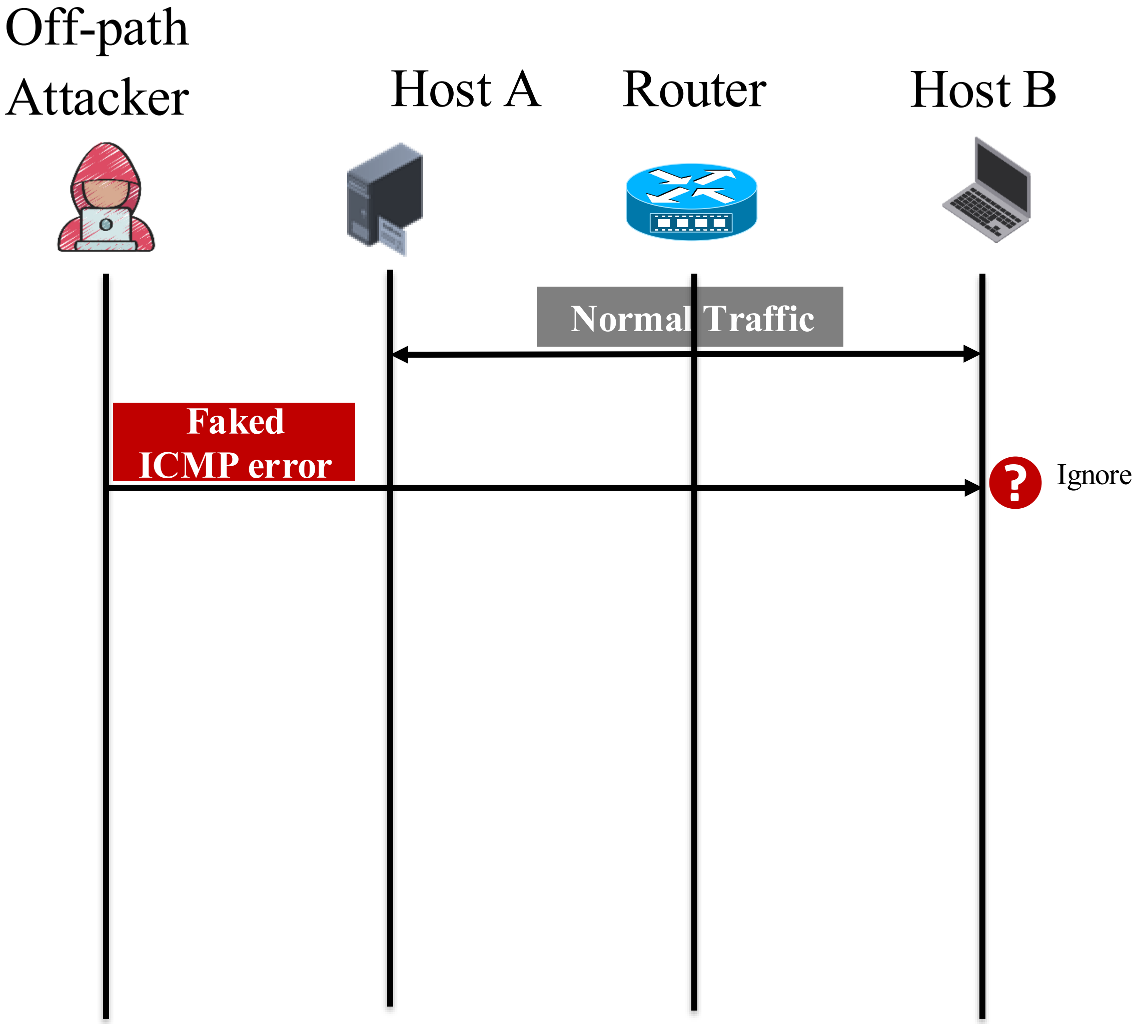


The ICMP Challenge-Confirm Mechanism

Real ICMP Error Messages

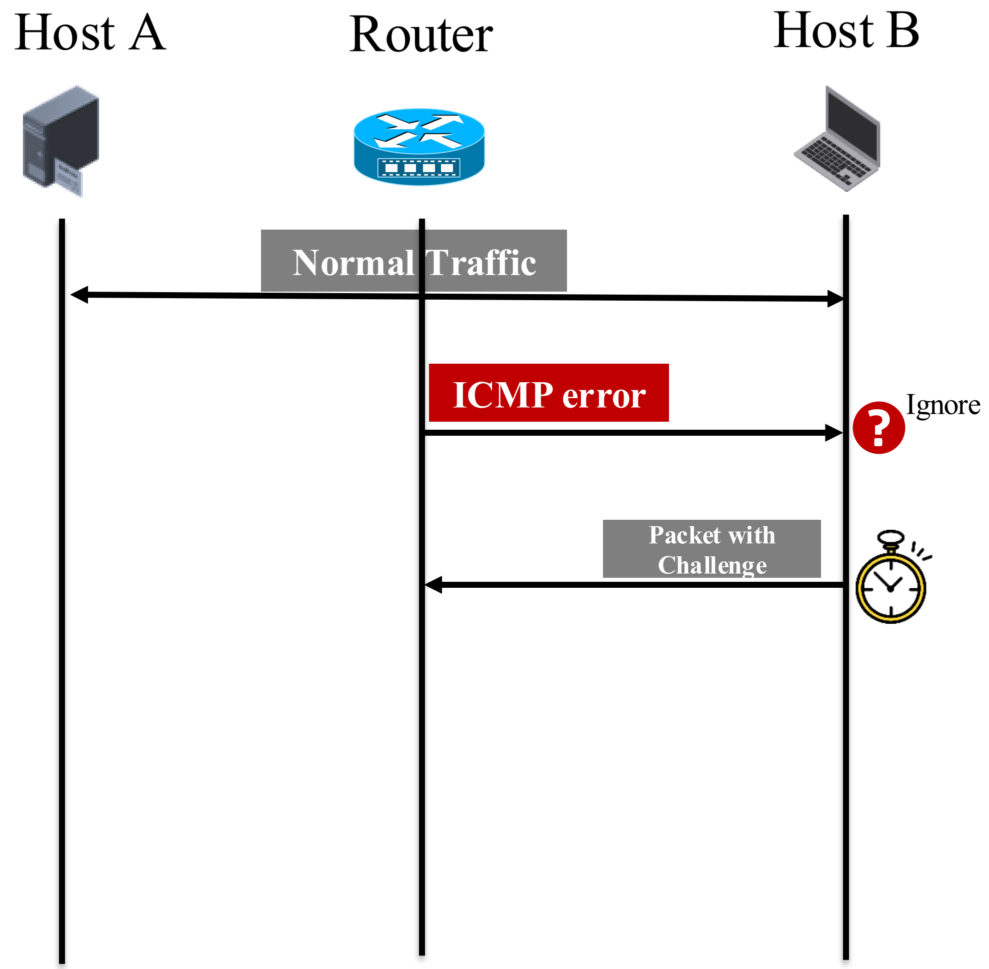


Forged ICMP Error Messages

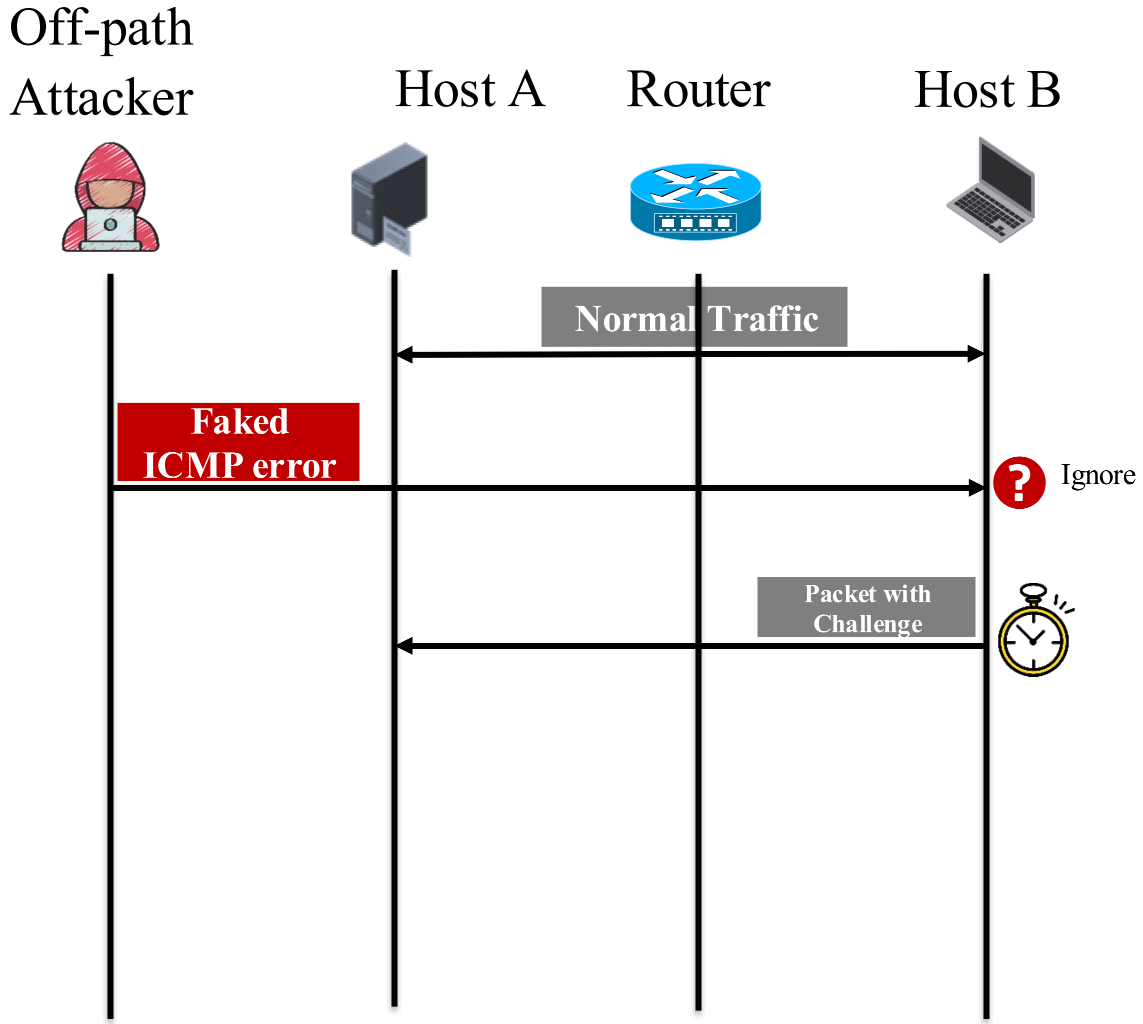


The ICMP Challenge-Confirm Mechanism

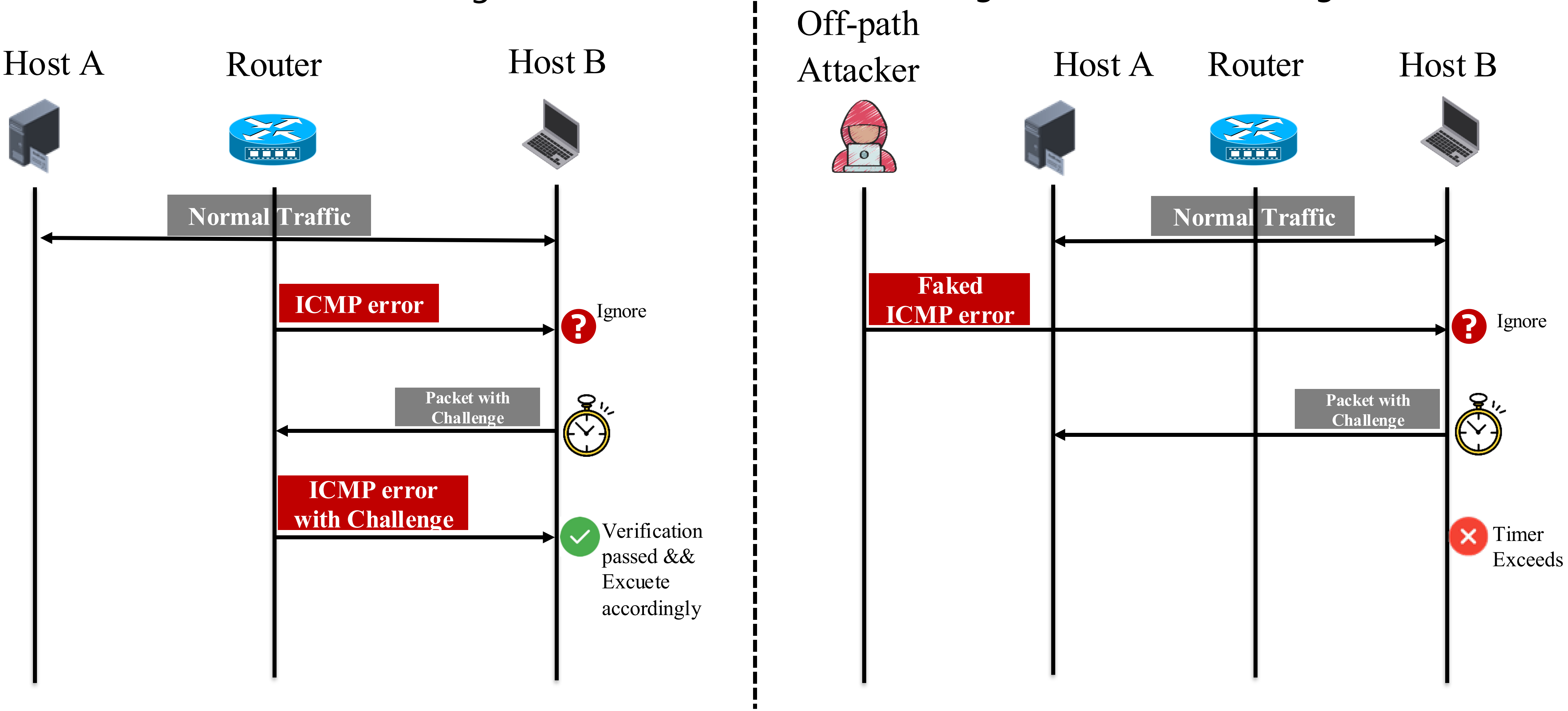
Real ICMP Error Messages



Forged ICMP Error Messages



The ICMP Challenge-Confirm Mechanism



Real ICMP Error Messages

Host A

Router

Host B

Normal Traffic

ICMP error

Ignore

Packet with Challenge

ICMP error with Challenge

Verification passed && Excuete accordingly

Forged ICMP Error Messages

Off-path Attacker

Host A

Router

Host B

Normal Traffic

Faked ICMP error

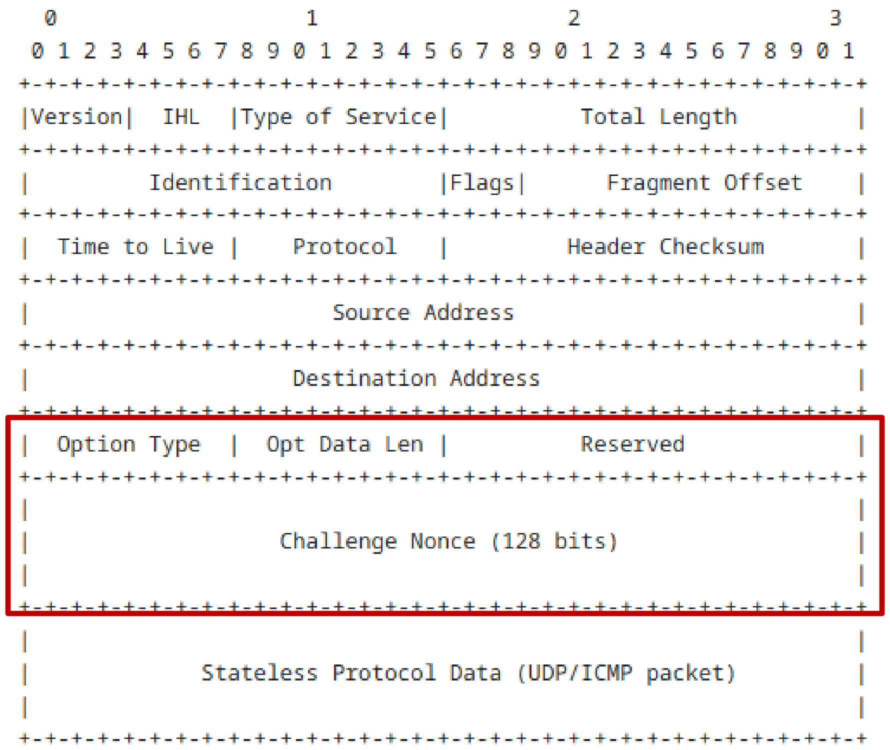
Ignore

Packet with Challenge

Timer Exceeds

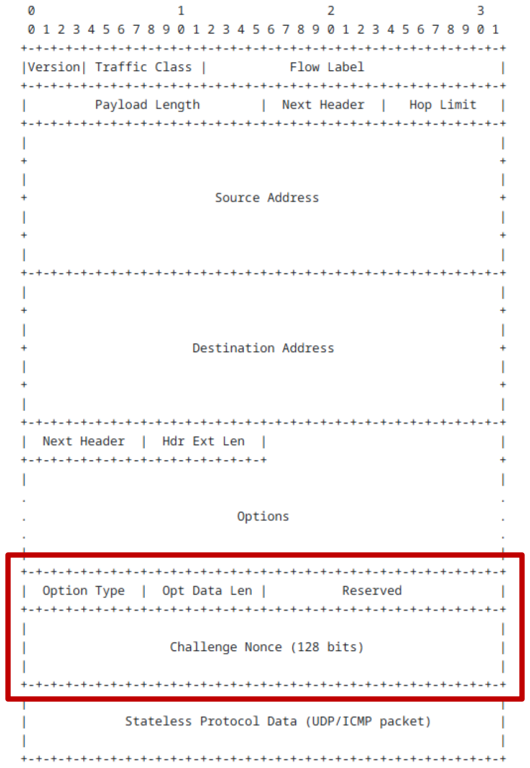
Updated Packet Formats

- The challenge will be carried in IP option as a random number.
  - The middle routers and peers do not have to perform any additional processing on the option.



```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Option Type  |  Opt Data Len |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                    Challenge Nonce (128 bits)                 |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|              Stateless Protocol Data (UDP/ICMP packet)        |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Challenge-Confirm Option in IPv4



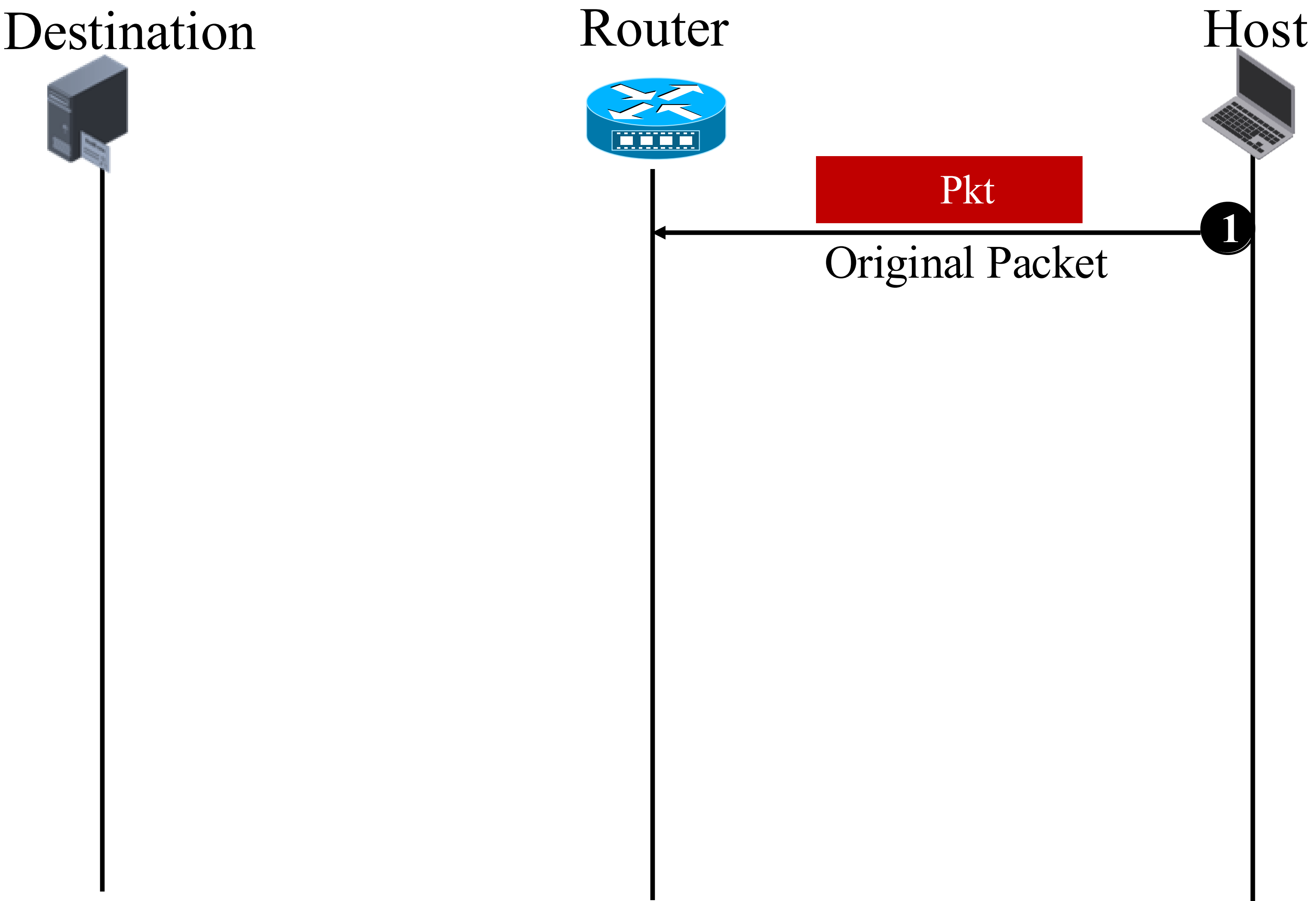
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |  Next Header  |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                         Source Address                        +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                      Destination Address                      +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |  Hdr Ext Len  |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                                                               |
.                                                               .
.                            Options                            .
.                                                               .
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Option Type  |  Opt Data Len |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                    Challenge Nonce (128 bits)                 |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Stateless Protocol Data (UDP/ICMP packet)        |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Challenge-Confirm Option in IPv6

Case Study

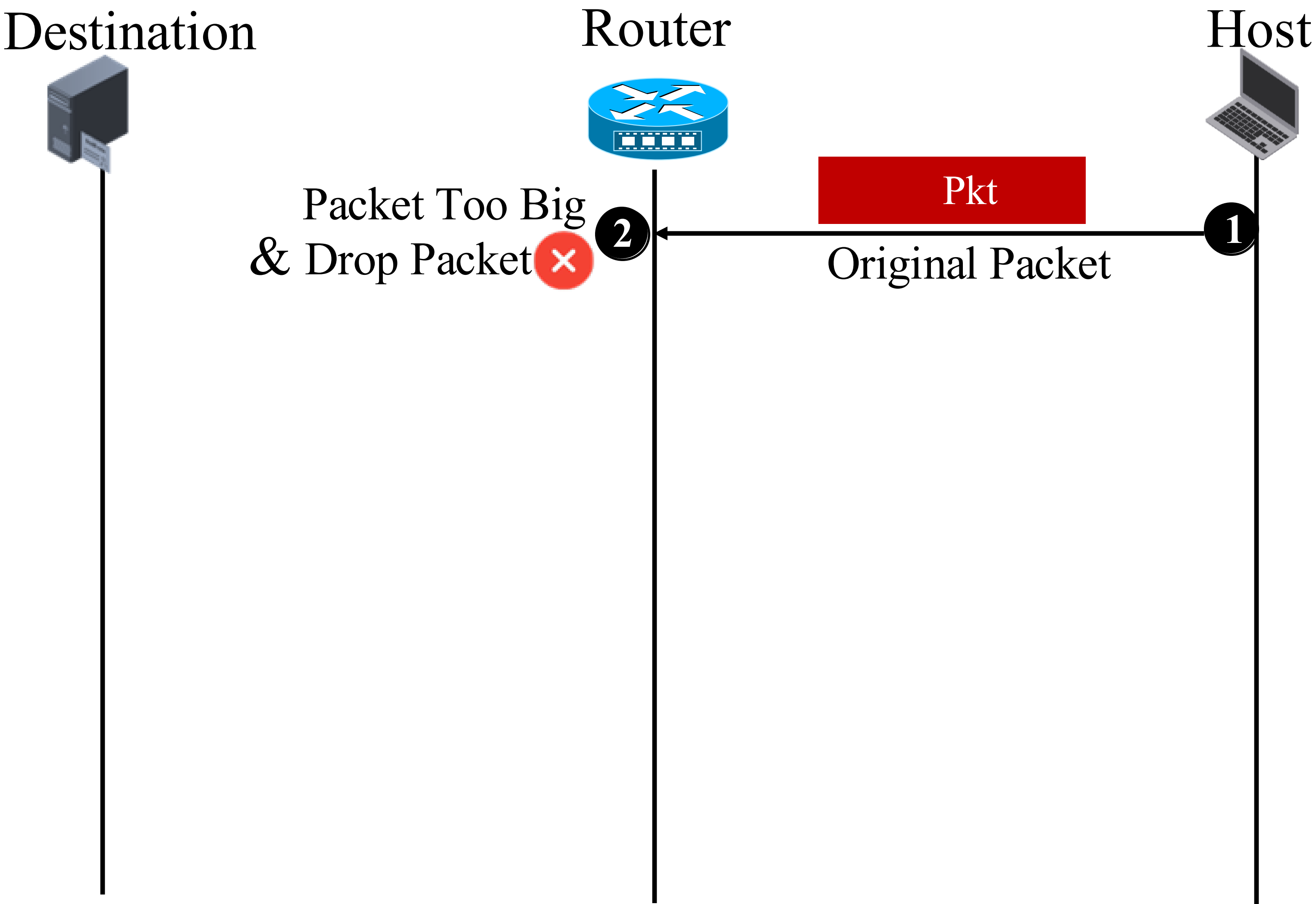
Case Study

- The mechanism when dealing with ICMP Fragmentation Needed messages



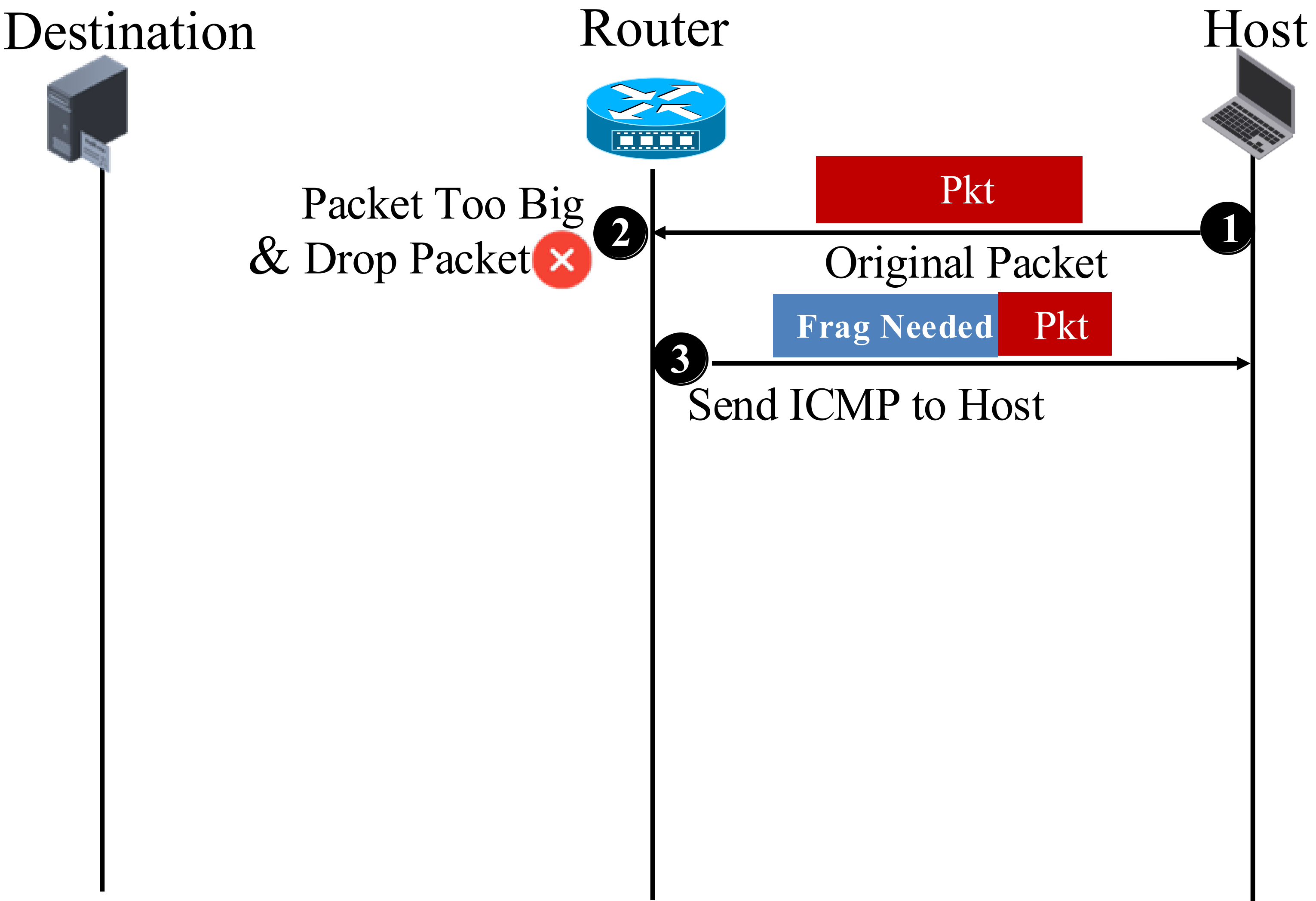
Case Study

- □ The mechanism when dealing with ICMP Fragmentation Needed messages



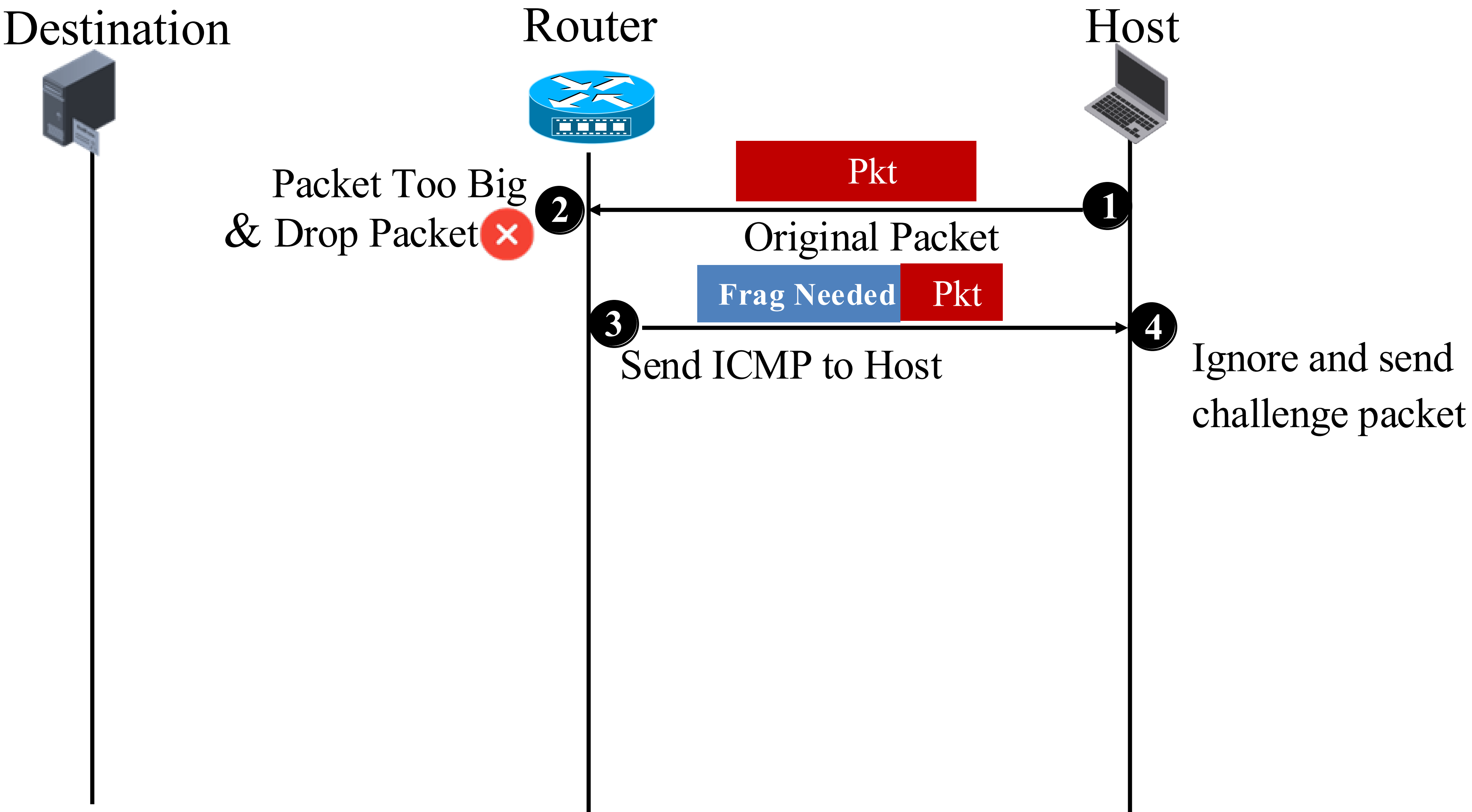
Case Study

□ The mechanism when dealing with ICMP Fragmentation Needed messages



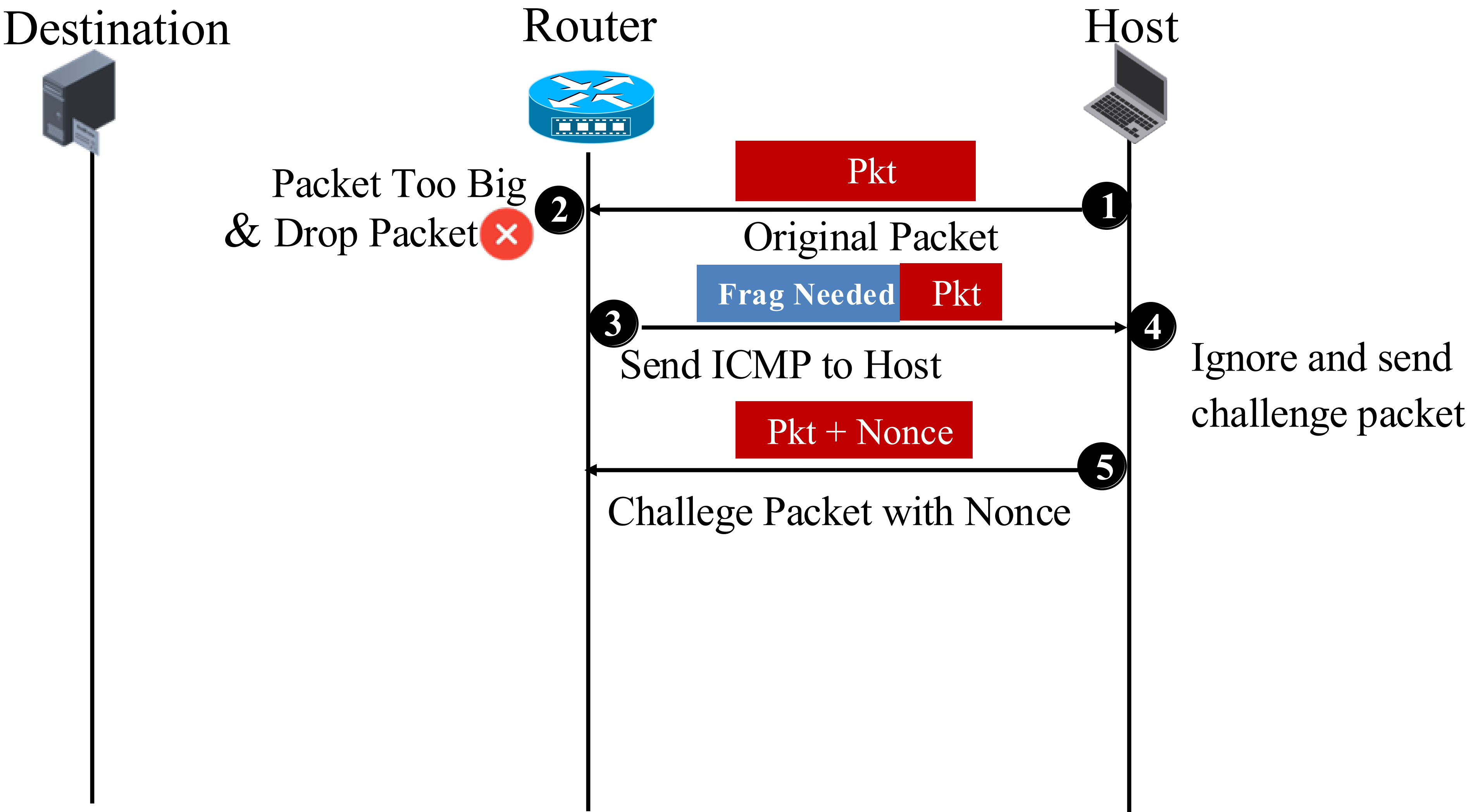
Case Study

- The mechanism when dealing with ICMP Fragmentation Needed messages



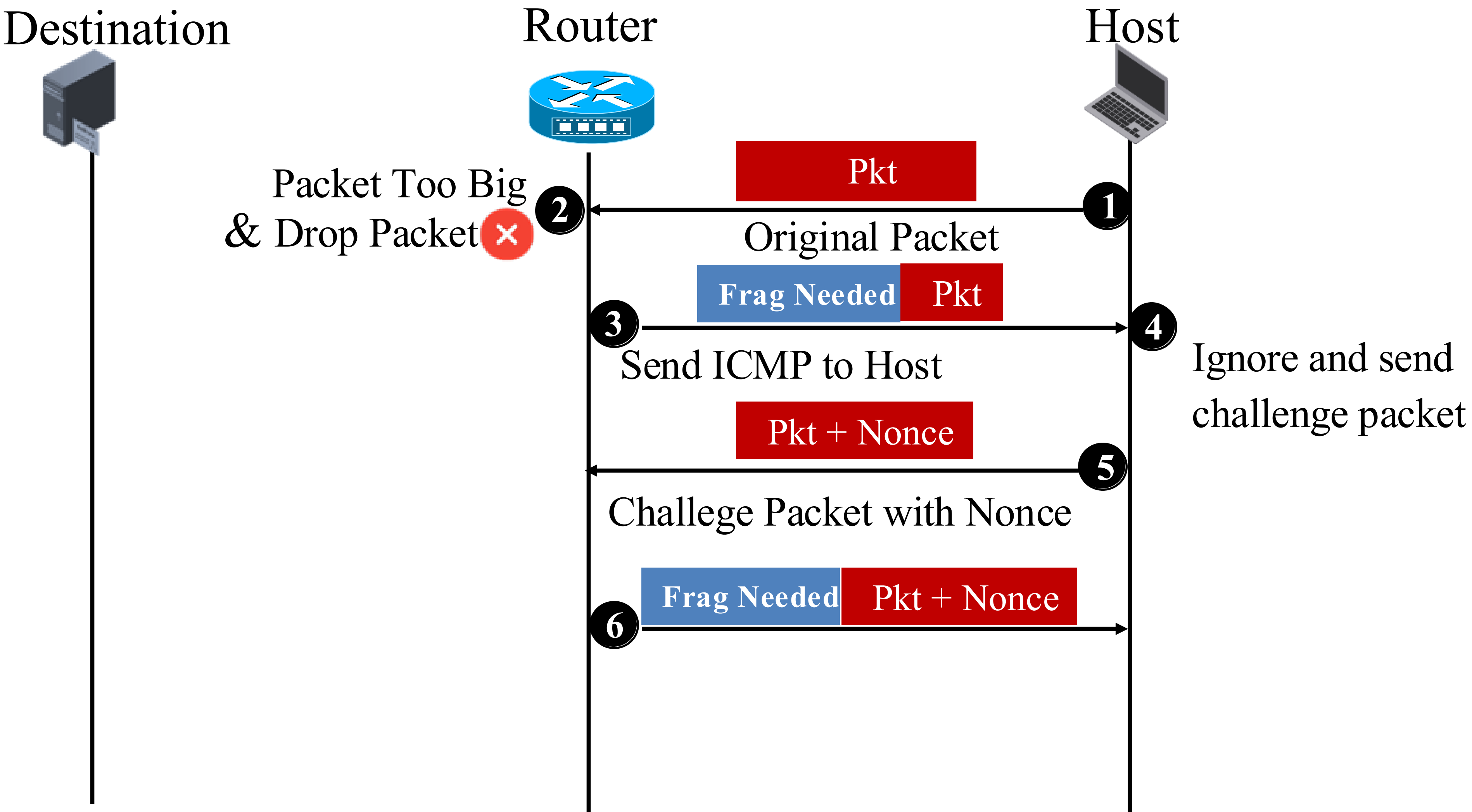
Case Study

- The mechanism when dealing with ICMP Fragmentation Needed messages



Case Study

□ The mechanism when dealing with ICMP Fragmentation Needed messages



Destination

Router

Host

Pkt

1 Original Packet

2 Packet Too Big & Drop Packet

Frag Needed Pkt

3 Send ICMP to Host

4 Ignore and send challenge packet

Pkt + Nonce

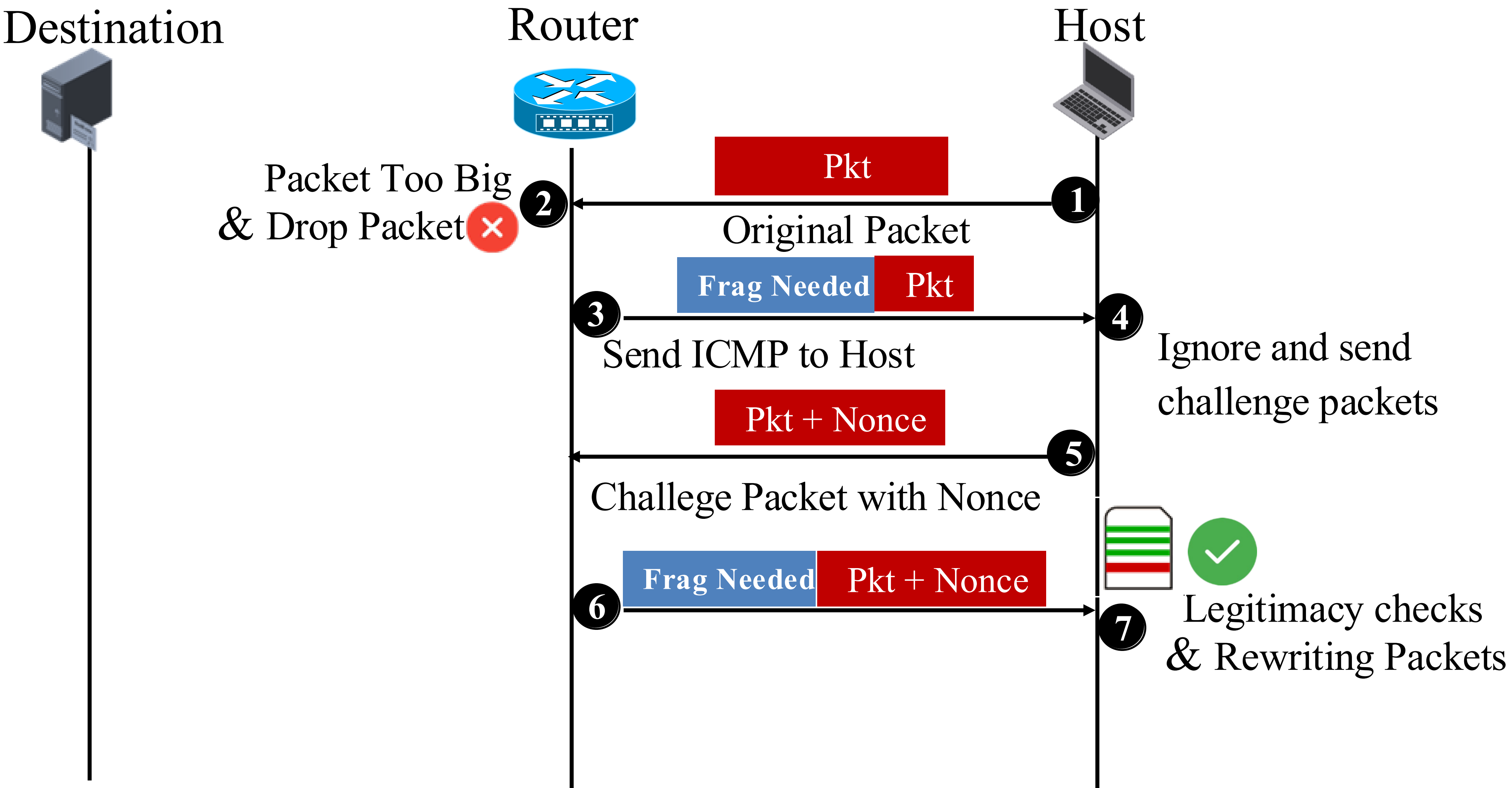
5 Challege Packet with Nonce

Frag Needed Pkt + Nonce

6

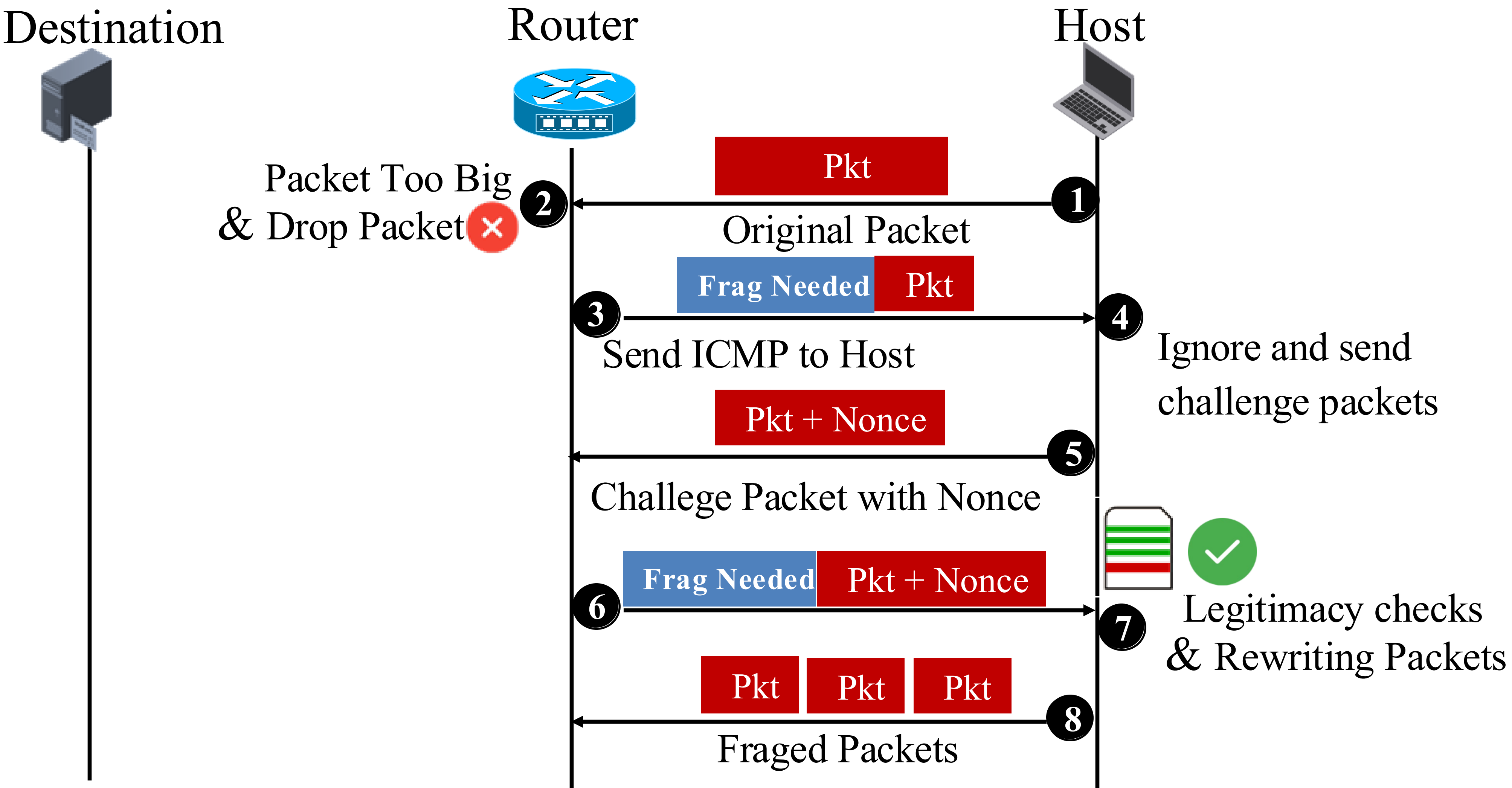
Case Study

- The mechanism when dealing with ICMP Fragmentation Needed messages



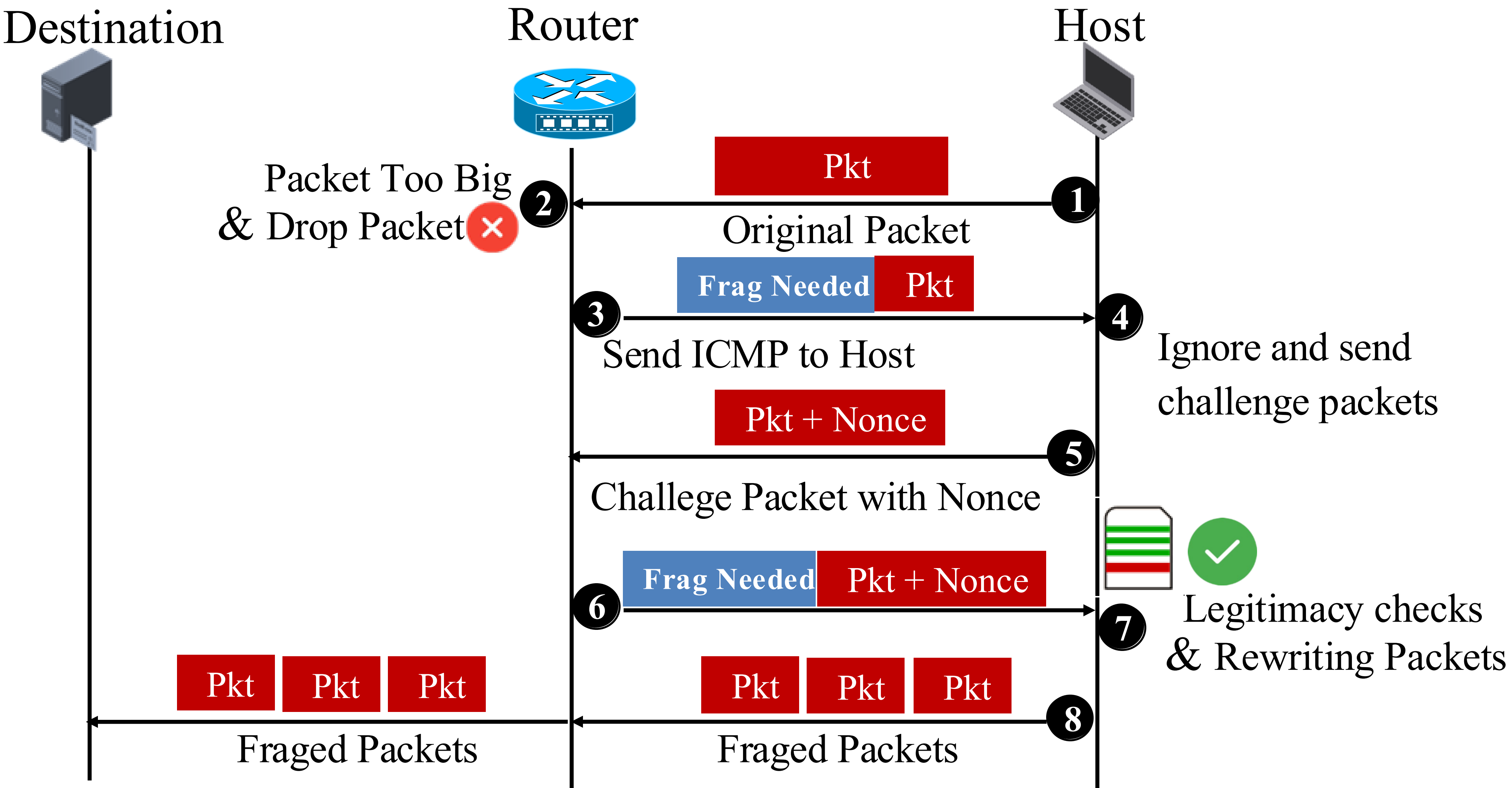
Case Study

- The mechanism when dealing with ICMP Fragmentation Needed messages



Case Study

□ The mechanism when dealing with ICMP Fragmentation Needed messages



Considerations

- □ Authentication Strength
  - ◆ Utilizing high-entropy random numbers to ensure that challenges are unpredictable and resistant to forgery.
- □ Replay Attack Mitigation
  - ◆ Assigning unique random numbers to each challenge and implementing expiration timers to mitigate the risk of replay attacks.
- □ Denial of Service Prevention
  - ◆ Rate limiting and challenge frequency controls should be implemented to prevent potential DoS attacks.
- □ Backward Compatibility
  - ◆ The proposed mechanism only requires updates solely to the ICMP error message verification on end hosts. Intermediate routing devices remain unaffected.

Next Step

□ Collaboration is welcome!

□ Your comments and suggestions are welcome

Thanks!