New Results for the PTB-PTS Attack on Tunneling Gateways

Vincent Roca, Ludovic Jacquin, Saikou Fall, Jean-Louis Roch

▶ **To cite this version:**

Vincent Roca, Ludovic Jacquin, Saikou Fall, Jean-Louis Roch. New Results for the PTB-PTS Attack on Tunneling Gateways. GreHack 2015, Cédric Lauradoux, Florent Autréau, Nov 2015, Grenoble, France. hal-01245629v1

HAL Id: hal-01245629

https://inria.hal.science/hal-01245629v1

Submitted on 17 Dec 2015 (v1), last revised 20 May 2016 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution - NonCommercial - ShareAlike 4.0 International License

New Results for the PTB-PTS Attack on Tunneling Gateways

Vincent Roca

Ludovic Jacquin

Saikou Fall

Jean-Louis Roch

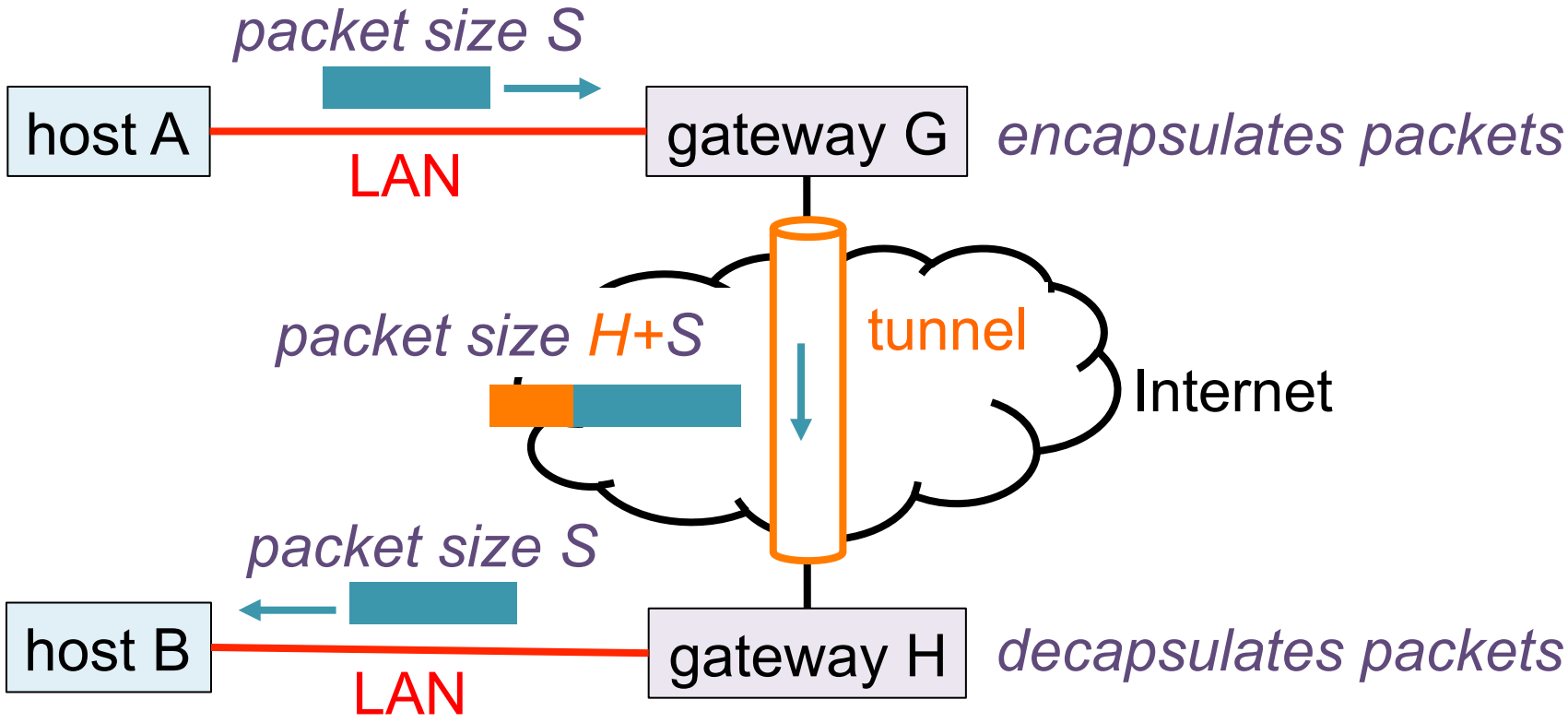
GreHack'15, Grenoble, November 20th 2015

Packet Too Big (PTB) or Packet Too Small (PTS)?

The underlying idea

About packet sizes and tunnel

- two gateways establish a tunnel to connect two remote LANs (or sites)

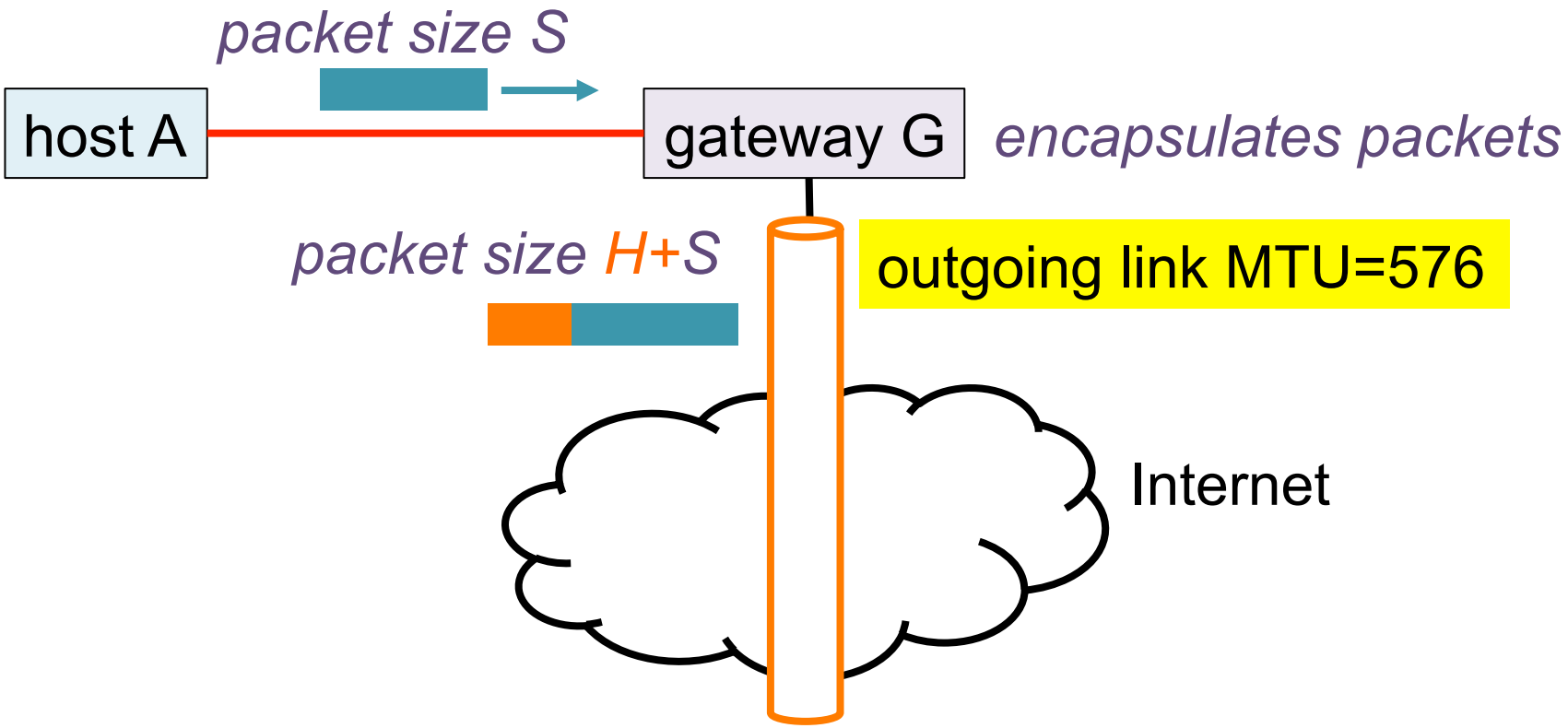


About packet sizes and tunnel… (cont')

- each link has a Maximum Transmission Unit (MTU)
  - **maximum allowed frame size on that link**
  - **e.g. 1500 bytes for Ethernet (i.e., 1460 b. or less at TCP level)**
- Path MTU (PMTU) is the min. MTU along the path
- a packet larger than a link's MTU is either
  - **dropped and an error ICMP "Packet Too Big" (PTB) message containing the MTU is returned to sender, or**
  - **fragmented if feasible (iff. IPv4 with DF bit clear)**
- each link MUST guaranty a minimum MTU
  - **IPv4 576 bytes**
  - **IPv6 1280 bytes**
  - **essentially here for performance reasons**

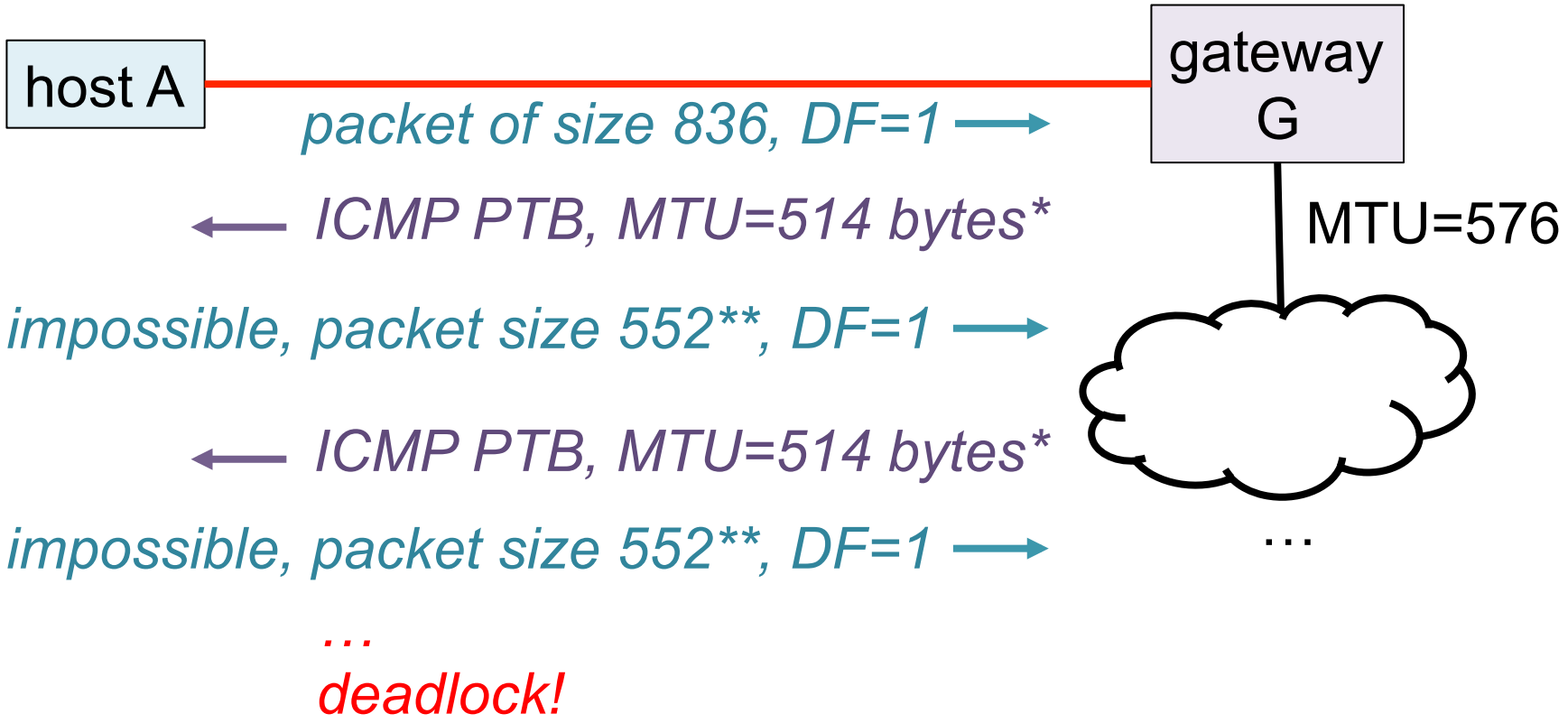
The issue

- what happens if G's outgoing link is already at MTU 576 bytes (IPv4)?
  - then we need H+S ≤ 576, which implies that S ≤ 576 - H



The issue – an experimental example

- G tunneling A's traffic using IPsec (Linux/Debian)



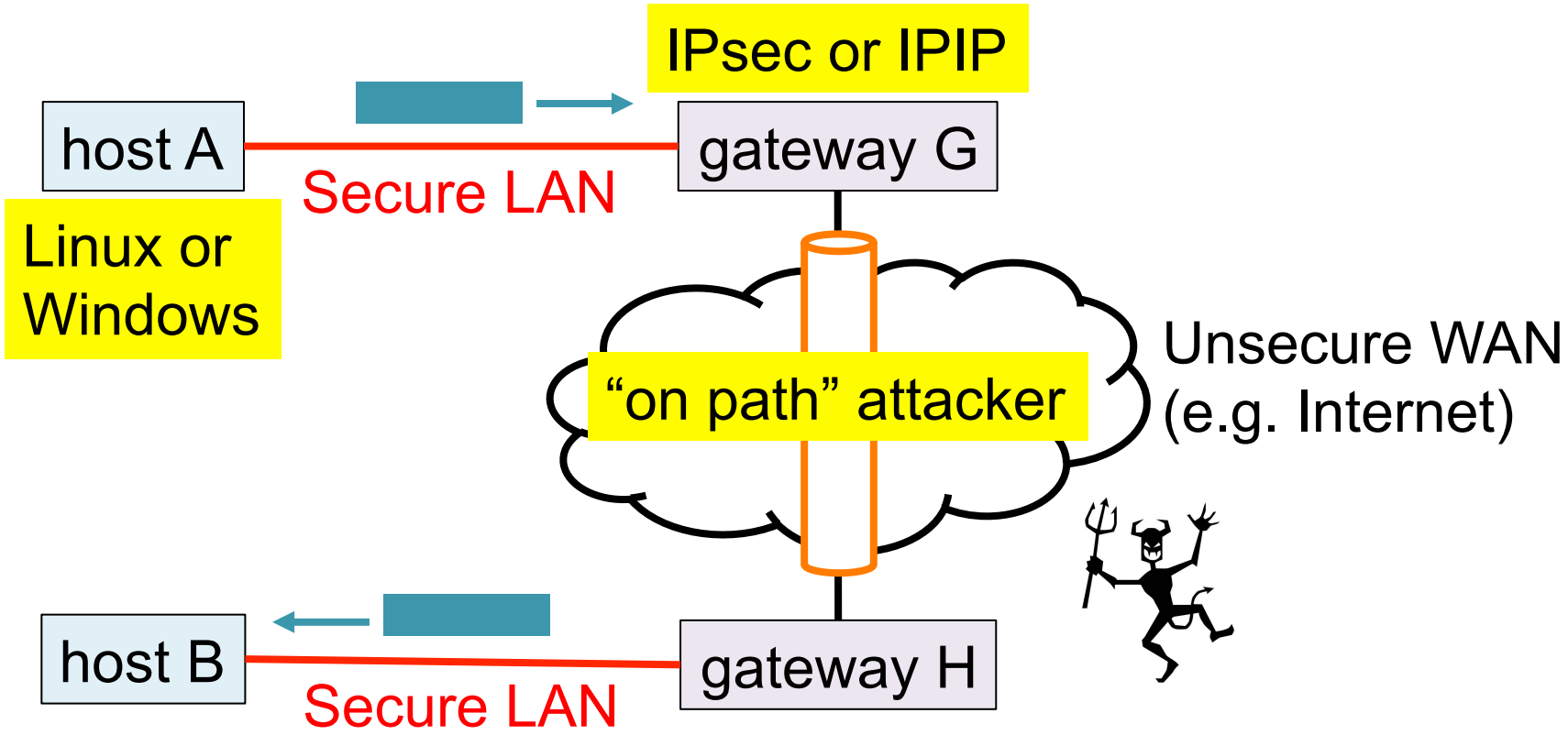
* 514 bytes because of IPsec ESP header

** 552 is minimum PMTU value on Linux/Debian

And now the exploit!

Attacker model

- "On path" attacker
  - Eavesdrop and inject traffic on the WAN
  - IPsec cryptographic ciphers deemed secure

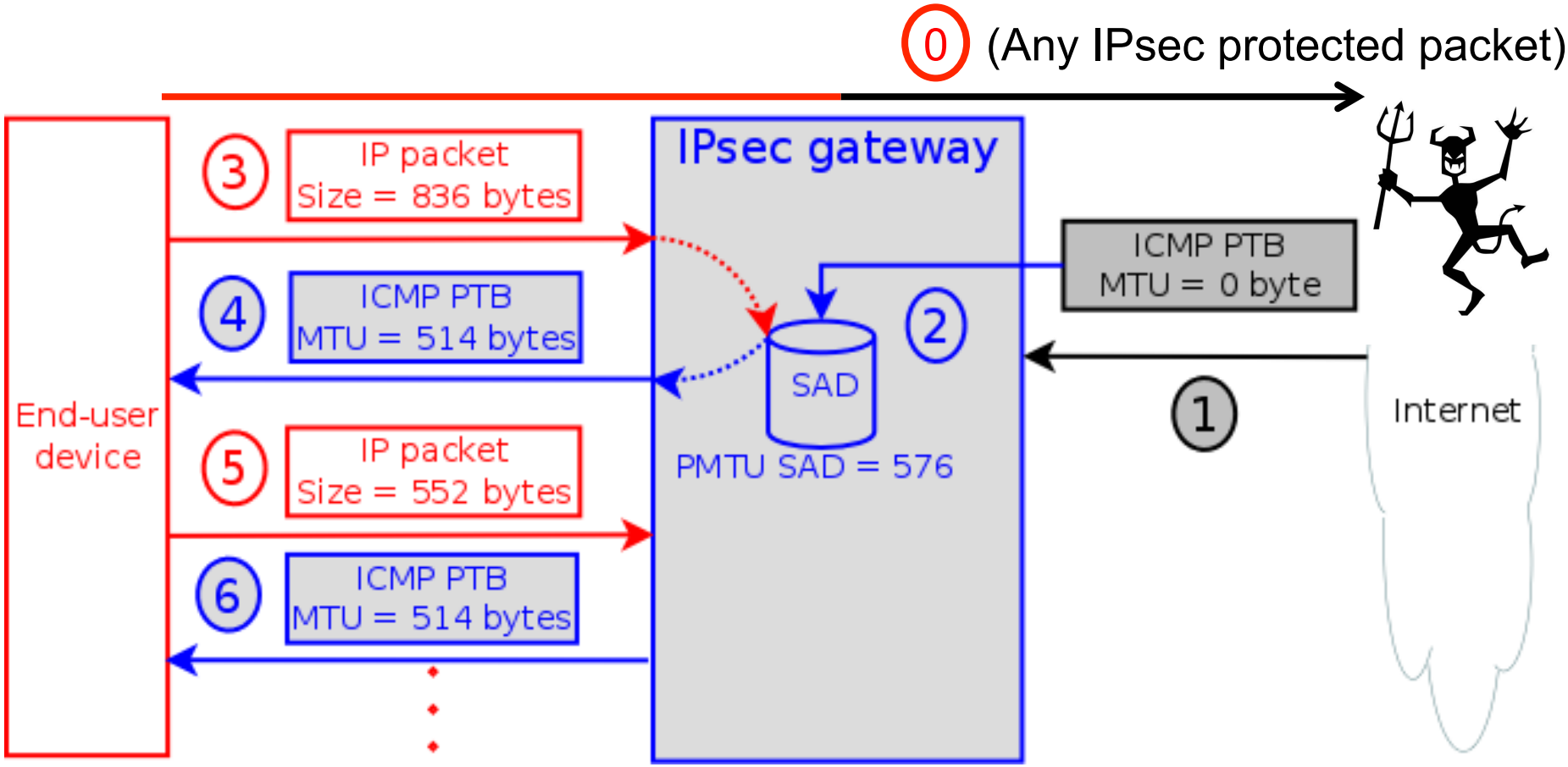


Description of the exploit

- Resetting gateway G's PMTU
  - the attacker needs to **be on the tunnel path**
    - **eavesdrops a tunneled packet**
    - **forges an ICMP PTB message**
      - Including a copy of the eavesdropped packet to bypass IPsec security check w.r.t. ICMP error messages
  - the attacker can use a compromised router…
  - … or be a simple host attached to a **non-encrypted WiFi**
    - **if a user uses a tunnel from a laptop (no gateway H) to a remote network, and is attached to a non-encrypted WiFi, then we can attack the remote tunnel gateway**
  - a **single** "well formed" ICMP PTB packet is sufficient to launch the attack!

Detail of the exploit

- Debian IPsec gateway
- Ubuntu client, TCP traffic, IPv4 with PMTUD

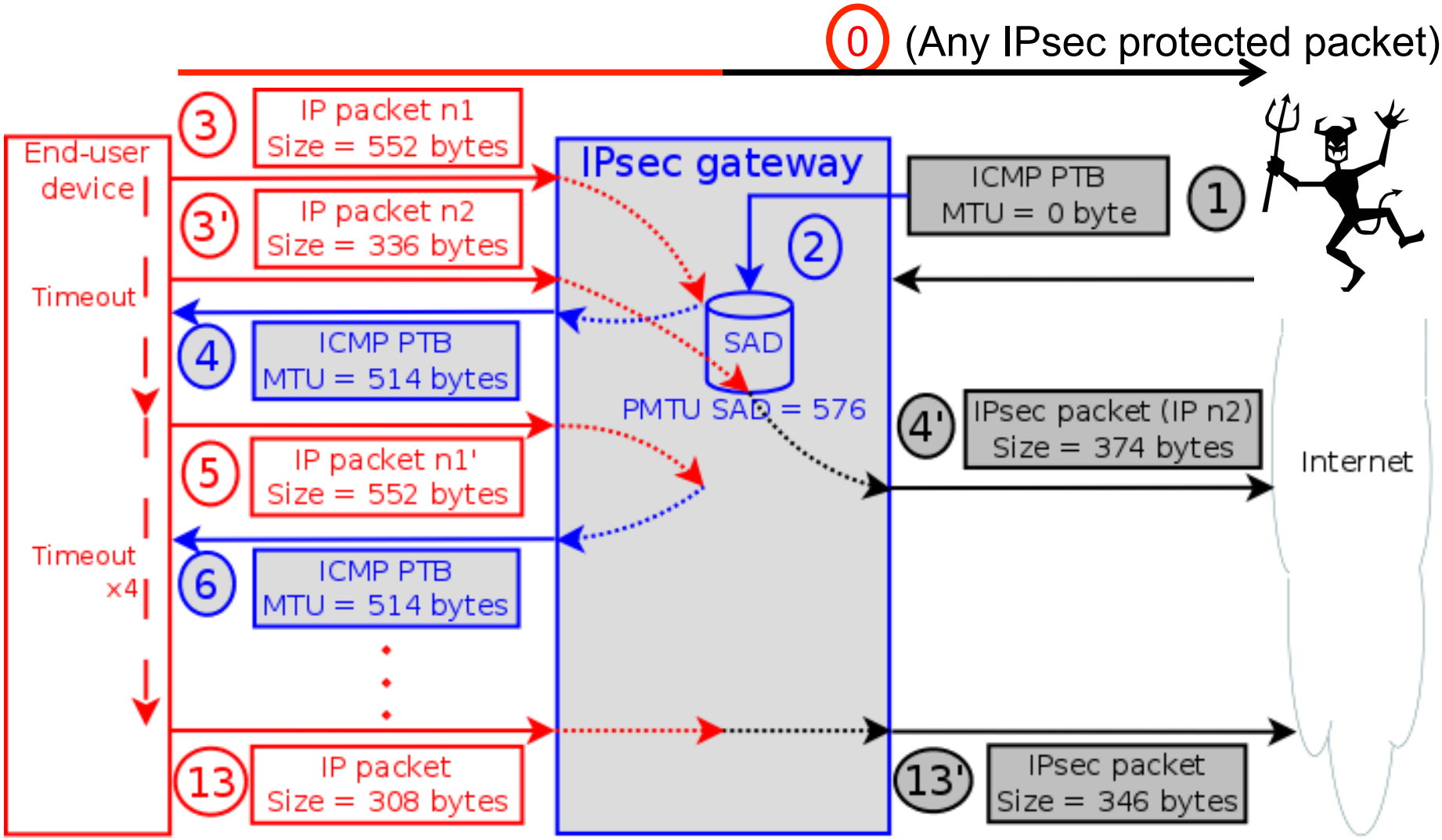


Another PMTU discovery to the rescue?

- Packetization Layer Path MTU Discovery (PLPMTUD)
  - Developed to mitigate ICMP “black holes”
    - **no dependency on ICMP any more**
  - Relies on “probes” and “feedbacks” to adjust packet sizes
  - compatible with TCP
    - **TCP ACK are used as feedbacks**
  - the TCP packet size can be reduced below the 576 minimum MTU (in IPv4) if needed
    - **e.g., 256 bytes + headers**

PLPMTUD only mitigates the exploit

- Ubuntu client, TCP traffic, IPv4 with PLPMTUD



Some additional tests

- UDP traffic with PMTUD
- IPv6
- Windows 7, with default configuration
- IPIP tunnel

Ubuntu client results

TCP, IPv4, PMTUD IPsec tunnel	**DoS:** no connection possible any more (TCP closes after 2 min.)
TCP, IPv4, PLPMTUD IPsec tunnel	**Major performance impacts:** 6.5s initial freeze, tiny packets (MSS = 256)
UDP, IPv4, PMTUD IPsec tunnel	**Major performance impacts:** tiny packets
TCP, IPv6, PMTUD IPsec tunnel	**DoS:** no connection possible any more (TCP closes after 2 min.)
TCP, IPv6, PLPMTUD IPsec tunnel	**Major performance impacts:** 3.3s initial freeze, small packets (MSS = 504)
TCP, IPv4, PMTUD IPIP tunnel	**Major performance impacts:** 7 min. initial freeze, tiny packets (MSS = 256)
TCP, IPv4, PLPMTUD IPIP tunnel	**Major performance impacts:** 6.7s initial freeze, small packets

Windows 7 client results

TCP, IPv4 IPsec tunnel	**Major performance impacts:** fragmented packets (548 and 120)
TCP, IPv6 IPsec tunnel	**DoS:** no connection possible any more (TCP closes after 21 sec.)
TCP, IPv4 IPIP tunnel	**DoS:** no connection possible any more (TCP closes after 35 sec.)

- Really strange behavior in TCP/IPv4/IPsec tests
  - Windows reset the “Don’t Fragment” bit after the first error
  - It keeps increasing TCP segment size… up to ~64 kB!!!
  - The gateway needs to fragment into smaller packet which is highly inefficient
- Similar results with Windows 10

Conclusions

A highly effective attack

- A **single** packet is enough to launch the attack
  - Only needs to eavesdrop one packet of the tunnel
- The gateway and client cannot agree
  - Once the attacker created confusion he can pull out
- Works on all client OSes
  - Highly effective, no matter the client configuration, leading either to DoS or major performance impacts
  - There is no good solution to deal with it!

Two issues highlighted

- Tunnels and small PMTU
  - The client rejects request to use an MTU smaller than the "minimum guaranteed"
    - **The client does not know this is motivated by IPsec or IPIP tunneling at the gateway**
    - **… and in any case it infringes the minimum MTU**

- Legitimacy of untrusted ICMP PTB packets
  - IPsec sanity check is not fully reliable and is by-passed if the attacker is on the path

Some counter-measures

- Trivial and unsatisfying
  - Ignore DF bit at a tunneling gateway
    - **E.g., as suggested by CISCO IPsec configuration guide!**
  - Ignore any ICMP PTB at the gateway and let clients use PLPMTUD
    - **But PLPMTUD won't work with UDP!**
- Two proposed counter-measures at a gateway
  - A gateway must not blindly accept an ICMP PTB advertising a tiny MTU
    - **The gateway needs room to add tunneling headers**
  - A gateway should assess untrusted ICMP PTB
    - **Add a probing scheme between tunneling gateways, similarly to PLPMTUD, to check the Path MTU**

Thank you