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 $H + S \leq 576$, which implies that $S \leq 576 - H$
The issue – an experimental example

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

packet of size 836, DF=1 →

ICMP PTB, MTU=514 bytes*

impossible, packet size 552**, DF=1 →

ICMP PTB, MTU=514 bytes*

impossible, packet size 552**, DF=1 →

... deadlock!

* 514 bytes because of IPsec ESP header
** 552 is minimum PMTU value on Linux/Debian
And now the exploit!
New Results for the PTB-PTS Attack on Tunneling Gateways

**Attacker model**

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

![Diagram of network setup](Image)
**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

![Diagram of the exploit](image)
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
New Results for the PTB-PTS Attack on Tunneling Gateways

**PLPMTUD only mitigates the exploit**

- Ubuntu client, TCP traffic, IPv4 with PLPMTUD

---

**Diagram Description**

1. (Any IPsec protected packet)
2. ICMP PTB
3. IP packet n1
   - Size = 552 bytes
4. ICMP PTB
   - MTU = 514 bytes
5. IP packet n1'
   - Size = 552 bytes
6. ICMP PTB
   - MTU = 514 bytes
7. IP packet n2
   - Size = 336 bytes
8. IPsec packet (IP n2)
   - Size = 374 bytes
9. IPsec packet (IP n2)
   - Size = 346 bytes
10. Router performance (number of packets per second) is relatively independent
11. Let us now consider a UDP flow, where the application
12. To conclude a huge delay of 6.59s was required for data to
13. The ssh connection finishes after a few additional segments
14. Beginning of the attack is the same. Then the host sends a
15. The initial large UDP datagram is transmitted in the
16. The host after this initial delay contains at most 256 byte of data,
17. and a prompt appears in the terminal.
18. In five medium size or tiny IP packets (548, 60, 548, 60 and
19. In three medium size or tiny IP packets (548, 60 and
20. The end of the attack is the same. Then the host sends a
21. and returns a prompt appears in the terminal.
Some additional tests

- UDP traffic with PMTUD
- IPv6
- Windows 7, with default configuration
- IPIP tunnel
### Ubuntu client results

<table>
<thead>
<tr>
<th>Protocol Configuration</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP, IPv4, PMTUD IPsec tunnel</td>
<td><strong>DoS:</strong> no connection possible any more (TCP closes after 2 min.)</td>
</tr>
<tr>
<td>TCP, IPv4, PLPMTUD IPsec tunnel</td>
<td><strong>Major performance impacts:</strong> 6.5s initial freeze, tiny packets (MSS = 256)</td>
</tr>
<tr>
<td>UDP, IPv4, PMTUD IPsec tunnel</td>
<td><strong>Major performance impacts:</strong> tiny packets</td>
</tr>
<tr>
<td>TCP, IPv6, PMTUD IPsec tunnel</td>
<td><strong>DoS:</strong> no connection possible any more (TCP closes after 2 min.)</td>
</tr>
<tr>
<td>TCP, IPv6, PLPMTUD IPsec tunnel</td>
<td><strong>Major performance impacts:</strong> 3.3s initial freeze, small packets (MSS = 504)</td>
</tr>
<tr>
<td>TCP, IPv4, PMTUD IPIP tunnel</td>
<td><strong>Major performance impacts:</strong> 7 min. initial freeze, tiny packets (MSS = 256)</td>
</tr>
<tr>
<td>TCP, IPv4, PLPMTUD IPIP tunnel</td>
<td><strong>Major performance impacts:</strong> 6.7s initial freeze, small packets</td>
</tr>
</tbody>
</table>
Windows 7 client results

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Tunnel Type</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP, IPv4</td>
<td>IPsec tunnel</td>
<td><strong>Major performance impacts:</strong> fragmented packets (548 and 120)</td>
</tr>
<tr>
<td>TCP, IPv6</td>
<td>IPsec tunnel</td>
<td><strong>DoS:</strong> no connection possible any more (TCP closes after 21 sec.)</td>
</tr>
<tr>
<td>TCP, IPv4</td>
<td>IPIP tunnel</td>
<td><strong>DoS:</strong> no connection possible any more (TCP closes after 35 sec.)</td>
</tr>
</tbody>
</table>

- 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