A Tale of Two Checksums

draft-ietf-fairhurst-udp-options-cco-00

Gorry Fairhurst, Tom Jones, Raffaele Zullo

tom@erg.abdn.ac.uk
UDP Option Area

IP transport payload

| IP Hdr | UDP Hdr | UDP user data | surplus area |

RFC793
UDP Option TLV

+--------+ | Kind=0 | EOL
+--------+        +--------+ | Kind=1 | NOP
         +--------+ | Kind=2 | Ones8 | OCS
         +--------+ | Kind=3 | Len=4 | CRC16sum | ACS

An innocuous little bug

```c
void in_delayed_cksum(struct mbuf *m) {
    struct ip *ip;
    uint16_t csum, offset, ip_len;

    ip = mtod(m, struct ip *);
    offset = ip->ip_hl << 2;
    ip_len = ntohs(ip->ip_len);
    csum = in_cksum_skip(m, ip_len, offset);
    if (m->m_pkthdr.csum_flags & CSUM_UDP && csum == 0)
        csum = 0xffff;
    offset += m->m_pkthdr.csum_data; /* checksum offset */
    /* find the mbuf in the chain where the checksum starts*/
    while ((m != NULL) && (offset >= m->m_len)) {
        offset -= m->m_len;
        m = m->m_next;
    }
    *(u_short *)(m->m_data + offset) = csum;
}
```
An innocuous little bug

```c
void
in_delayed_cksum(struct mbuf *m)
{
    struct ip *ip;
    uint16_t csum, offset, ip_len;

    ip = mtod(m, struct ip *);
    offset = ip->ip_hl << 2;
    ip_len = ntohs(ip->ip_len);
    csum = in_cksum_skip(m, ip_len, offset);
    if (m->m_pkthdr.csum_flags & CSUM_UDP && csum == 0)
        csum = 0xffff;
    offset += m->m_pkthdr.csum_data; /* checksum offset */
    /* find the mbuf in the chain where the checksum starts*/
    while ((m != NULL) && (offset >= m->m_len)) {
        offset -= m->m_len;
        m = m->m_next;
    }
    *(u_short *)(m->m_data + offset) = csum;
}
```
• Fixed in FreeBSD by r334705

• The IETF - *Making the Internet Better!*
• Fixed in FreeBSD by r334705

• The IETF - *Making the Internet Better!*
Measuring UDP Options

• There are no UDP Options hosts on the internet (yet!)

• Measurements with Mobile Tracebox Core*

• UDP is difficult to measure
  • STUN
  • DNS
  • NTP

• HTTP tricks

*https://erg.abdn.ac.uk/~raffaele
More dangerous than gator wrestling
Middlebox Pathologies

“... middle boxes can silently discard packets for other reasons. For example, on the Juniper SRX, the default behavior for a stateful firewall is to discard all packets with incorrect checksums.”

- Ron Bionica
Middlebox Pathologies

- Works
- Full Payload Checksum
- Full Payload Checksum, UDP length Pseudoheader
- UDP Length Checksum, IP length Pseudoheader
- Only passes 0s as options space
- Only passes IP payload length == UDP Length
The CCO Option

<table>
<thead>
<tr>
<th>Kind=xx</th>
<th>Len=4</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 byte</td>
<td>1 byte</td>
<td>2 bytes</td>
</tr>
</tbody>
</table>

UDP CCO Option Format
The Magic CCO Option
It works against CPE too!

- Dlink: DIR-655-A2, A3, A4, B1; DIR 619-Ax; DI-614+-B2
- Jensen: AirLink WBR 7954 v2, v3; AirLink 1000Gv2 (A)
- Linksys: E2500, WRT54G/GL/GS v1.1, WRT54G, E4200
- Netgear: WGR 614v7, v9; WNDR3400
- Topcom; WBR 254G, BR 604
- TP-Link: TL-MR3020 v1, TL-WR703N
- 3g modem: WR3G050-02
- ZyXEL: P-2812HNU-F3
- Xiaomi: Router 3C

17 Pass UDP Options, 6 Drop UDP Options

Courtesy of Runa Barik, University of Oslo
Please read
draft-ietf-fairhurst-udp-options-cco-00

This work is partially supported by the European Commission under Horizon 2020 grant agreement no. 688421 Measurement and Architecture for a Middleboxed Internet (MAMI).
The Full Picture

Failure pathologies

IPv4 STUN  IPv4 DNS  IPv4 HTTP  IPv6 DNS  IPv6 HTTP
The Full Picture

- No CS
- Other
- Zero CS only
- 4th CS only
- 3rd CS only
- Zero-Padded Options only
- Correct UDP CS AND Full IP Payload CS (Compensated CS Only)
- Full IP Payload CS
- Correct UDP CS OR Full IP Payload CS
- Correct UDP CS only
- Any CS
The Full Picture

- No CS
- Other
- Zero CS only
- 4th CS only
- 3rd CS only
- Zero-Padded Options only
- Correct UDP CS AND Full IP Payload CS (Compensated CS Only)
- Full IP Payload CS
- Correct UDP CS OR Full IP Payload CS
- Correct UDP CS only
- Any CS
The Full Picture
The Full Picture