Packet in Packet - Cisco HDLC

Michael Samuel - @mik235

Packet in Packet

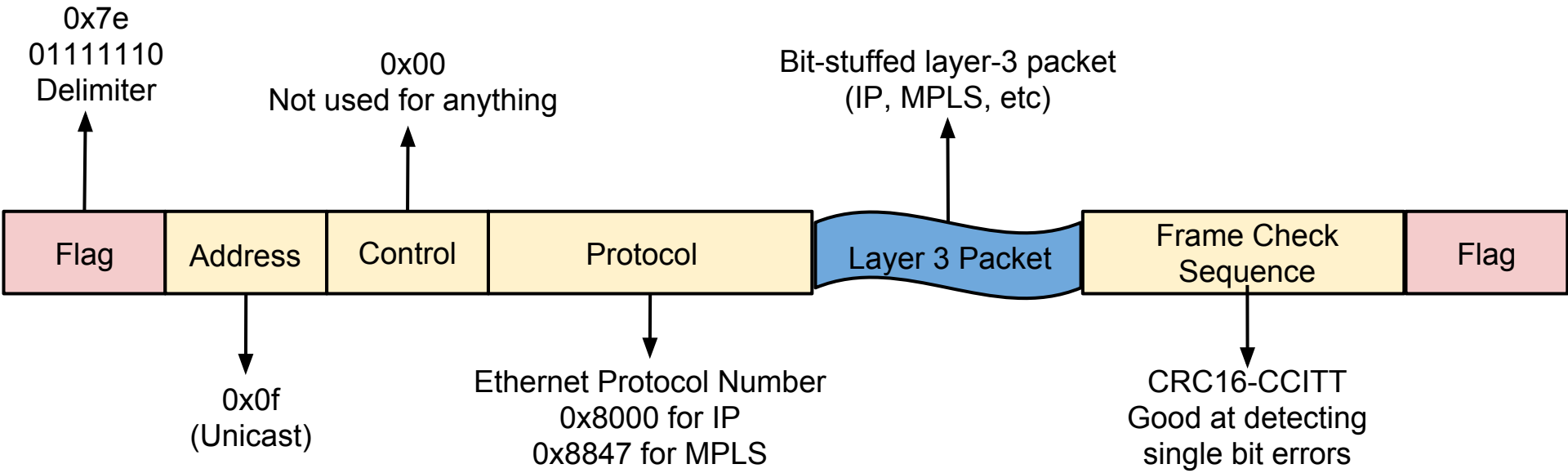
Like SQL Injection, but with packets

Travis Goodspeed did an excellent talk at Ruxcon on sending packets for one wireless protocol inside packets for another.

This is a variation of his technique, but on a wire protocol that some telcos and service providers use.

Practicality warning: this attack is a really long shot, especially on slow links

Cisco HDLC



- Flag always sent before and after a packet
- Flag is continuously sent on idle synchronous link
- Address can be 0x8f for broadcast
- Both Address and Control are basically pointless

Bit Stuffing

If 5 simultaneous 1 bits are sent on the link, a 0 bit is sent, which is ignored by the remote end.

```
01111110
```

becomes

```
011111010
```

Same concept as URL encoding

This prevents people like me from putting flags inside packets, unless...

Bit-errors Happen

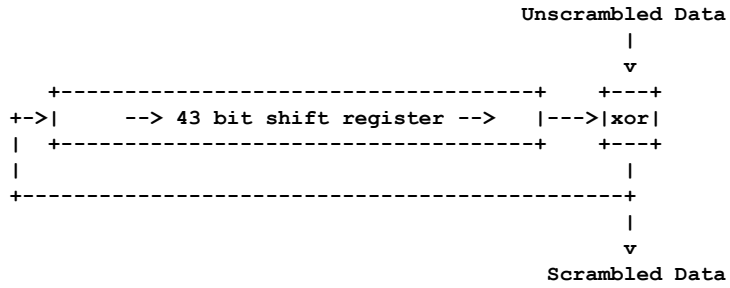
Bit errors are very rare on good clean optical links.

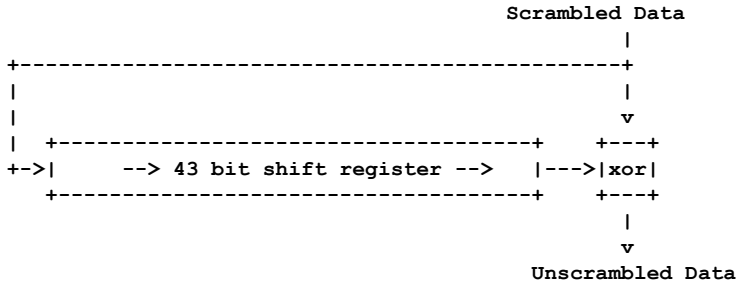
Many microwave links are using POS, which would usually use a HDLC-like encapsulation

If there's a Cisco router on each end they will probably be talking cHDLC (the default setting)

0x7e = 01111110
0x5e = 01011110
That's a bit-flip away from a flag

The ATM Self-Synchronous Scrambler





`draft-ferguson-pppsonet-selfsync-00.txt`

43 bits was chosen as the size because the ATM committee was batshit crazy

- Bit errors are repeated 43 bits later
- Burst errors (> 43 bits) randomize packet data
- Recovers after 43 bits of clean data

The most likely way you'll get a working exploit is for a burst error to scramble the first 3 bits of your faux-flag. If such a lucky coincidence occurred, there'd be a 12.5% chance of this exploit working.

A random bit-error 40 bits before our faux-flag would also work

Injection

Information you need to collect

- What you want your packet to look like
  - If the target is MPLS, you need to know label values for the BGP route for the target router on that link
  - If providers forget to enter 'no mpls ip propagate-ttl [forwarded]', this can be done with traceroute (on Cisco MPLS networks)
- What your packet does look like
  - TTL
  - MPLS label values (traceroute)
  - Random evil firewalls
  - ISP tampering

Collision!

One flag is hard, two flags are impossible

Most layer 2 protocols use a variation of CRC at the end of the packet

Cisco HDLC uses CRC16-CCITT

CRC16 can be reset to any value by manipulating 16 bits

You want it so that after your faux-flag, the CRC will be at 0xffff

Just brute force it (one less slide = 1 more beer)

The leaning tower of packet

```
[cHDLC header]
  [mpls transport tag]
  [mpls vpn tag]
    [ip]
      [tcp/udp/icmp]
        [2 bytes of crc collision]
        [faux flag byte]
        [cHDLC header]
          [evil mpls transport tag]
          [evil mpls vpn tag (optional!)]
            [ip]
              ....
[cHDLC FCS]
```

Stuff you need to know

Stuff you control

Other Layer-2 Protocols

- PPP
  - Address is 0xff
  - The *ACFC* option removes address and control
- Ethernet
  - Bit errors are very rare
  - Only -T is likely to be vulnerable at all
  - 10G-BaseW (aka WAN-PHY) is not vulnerable (but still insane)

Watch Travis Goodspeed's Ruxcon Talk:

`http://youtu.be/iQk0GHXs8NY`

Thanks!

Twitter: @mik235

http://www.miknet.net/