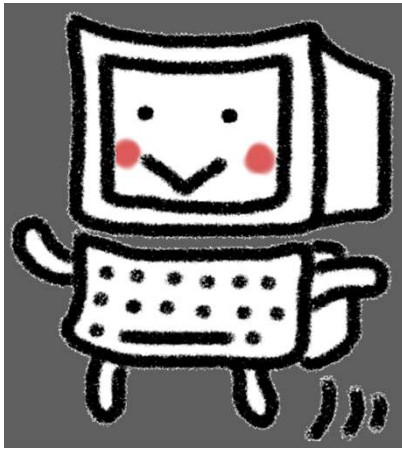


GRE over UDP

Advisor: Prof. Chien-Chao Tseng

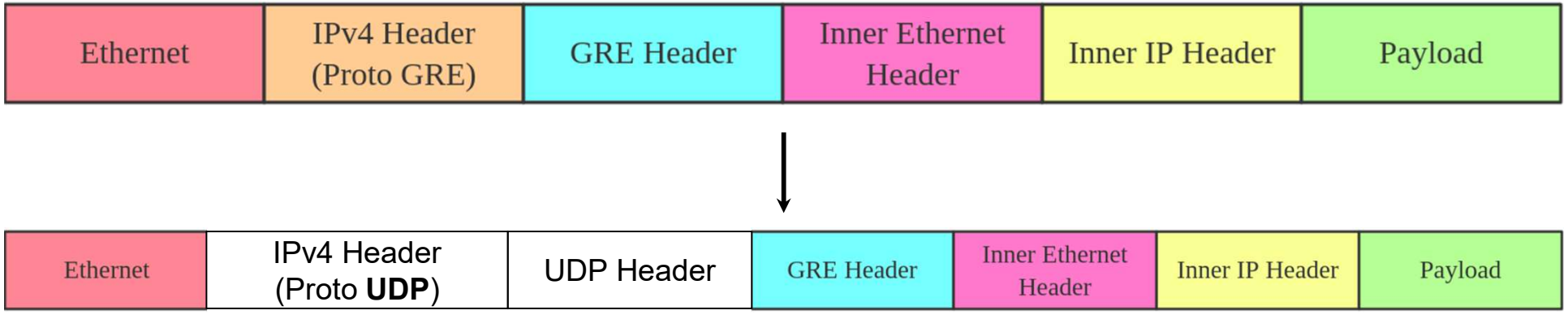
Student: Chenxu Wang

Date: March 24, 2021

Foo over UDP (FOU)

- Foo over UDP (FOU) allows encapsulating packets of an IP protocol directly over UDP
- Compatible with existing tunneling technologies: SSH tunnels (TCP level), GRE/IPIP (IP level)
- Destination port maps to IP Protocol

GRE tunnel packet



GRE tunnel packet with FOU

fou is a Kernel module

```
winlab@server156:~$ modinfo fou
filename:       /lib/modules/4.15.0-128-generic/kernel/net/ipv4/fou.ko
license:        GPL
author:         Tom Herbert <therbert@google.com>
srcversion:     B585EB24ADF9BD720F8B67E
depends:        ip_tunnel,udp_tunnel,ip6_udp_tunnel
retpoline:      Y
intree:         Y
name:           fou
vermagic:       4.15.0-128-generic SMP mod_unload
signat:         PKCS#7
signer:
sig_key:
sig_hashalgo:   md4
winlab@server156:~$
```

Load fou kernel module

$ sudo modprobe fou

Module name Size Used by

```
winlab@server156:~$ lsmod | grep fou
fou                    24576  0
ip_tunnel              24576  2 fou,ip_gre
ip6_udp_tunnel         16384  1 fou
udp_tunnel             16384  1 fou
winlab@server156:~$
```

ip_gre kernel module - fallback device

- Useless

```
winlab@server159:~$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT g
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eno1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT
    link/ether d8:c4:97:86:ad:3b brd ff:ff:ff:ff:ff:ff
3: eno2: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN mode DEF
    link/ether d8:c4:97:86:ad:3c brd ff:ff:ff:ff:ff:ff
4: ens3f0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN mode D
    link/ether 2c:60:0c:ca:58:28 brd ff:ff:ff:ff:ff:ff
5: ens3f1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN mode D
    link/ether 2c:60:0c:ca:58:29 brd ff:ff:ff:ff:ff:ff
6: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
    link/ether 02:42:49:d9:a0:61 brd ff:ff:ff:ff:ff:ff
winlab@server159:~$ lsmod | grep gre
winlab@server159:~$ sudo modprobe ip_gre
winlab@server159:~$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT g
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eno1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT
    link/ether d8:c4:97:86:ad:3b brd ff:ff:ff:ff:ff:ff
3: eno2: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN mode DEF
    link/ether d8:c4:97:86:ad:3c brd ff:ff:ff:ff:ff:ff
4: ens3f0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN mode D
    link/ether 2c:60:0c:ca:58:28 brd ff:ff:ff:ff:ff:ff
5: ens3f1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN mode D
    link/ether 2c:60:0c:ca:58:29 brd ff:ff:ff:ff:ff:ff
6: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
    link/ether 02:42:49:d9:a0:61 brd ff:ff:ff:ff:ff:ff
7: gre0@NONE: <NOARP> mtu 1476 qdisc noop state DOWN mode DEFAULT group default q
    link/gre 0.0.0.0 brd 0.0.0.0
8: gretap0@NONE: <BROADCAST,MULTICAST> mtu 1462 qdisc noop state DOWN mode DEFAULT
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
9: erspan0@NONE: <BROADCAST,MULTICAST> mtu 1450 qdisc noop state DOWN mode DEFAULT
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
winlab@server159:~$ lsmod | grep gre
ip_gre                 28672  0
ip_tunnel              24576  1 ip_gre
gre                    16384  1 ip_gre
winlab@server159:~$
```

GRE / GRETAP / ERSPAN

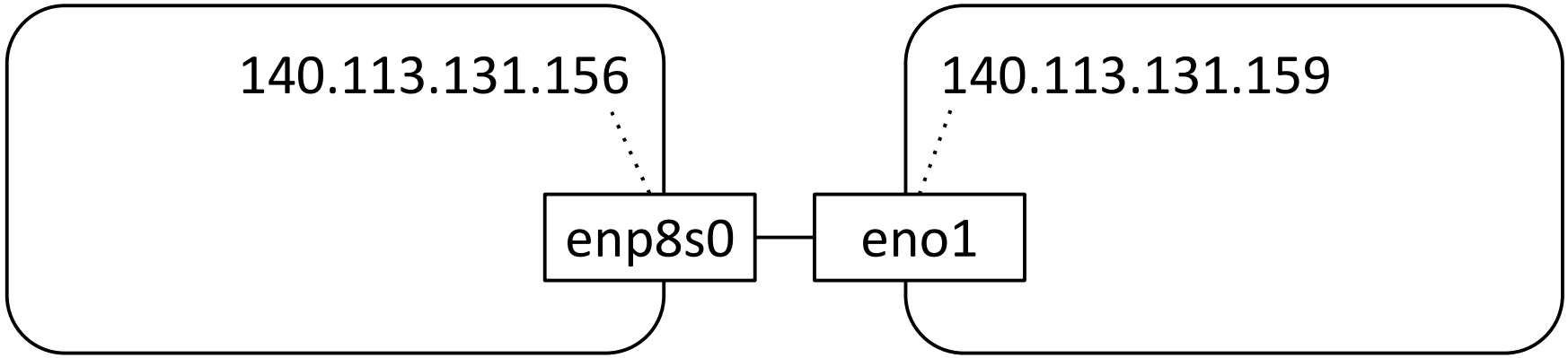
- They are GRE tunnel

gre

gretap

erspan (Encapsulated Remote Switch Port Analyzer)

Testing environment



Create tunnel endpoint (1/3)

$ ip link add my-gre type gretap remote 140.113.131.156 local **140.113.131.159** encap fou encap-sport 33333 encap-dport 55555

- Syntax from man page

```
ip link add DEVICE type { gre | gretap }  remote ADDR
local ADDR [ [no][i|o]seq ] [ [i|o]key KEY | no[i|o]key ]
[ [no][i|o]csum ] [ ttl TTL ] [ tos TOS ] [ [no]pmtudisc ]
[ [no]ignore-df ] [ dev PHYS_DEV ] [ encap { fou | gue |
none } ] [ encap-sport { PORT | auto } ] [ encap-dport
PORT ] [ [no]encap-csum ] [ [no]encap-remcsum ] [ external
]
```

https://man7.org/linux/man-pages/man8/ip-link.8.html

Create tunnel endpoint (2/3)

$ ip link add my-gre type gretap remote 140.113.131.156 local **140.113.131.159** encap fou encap-sport 33333 encap-dport 55555

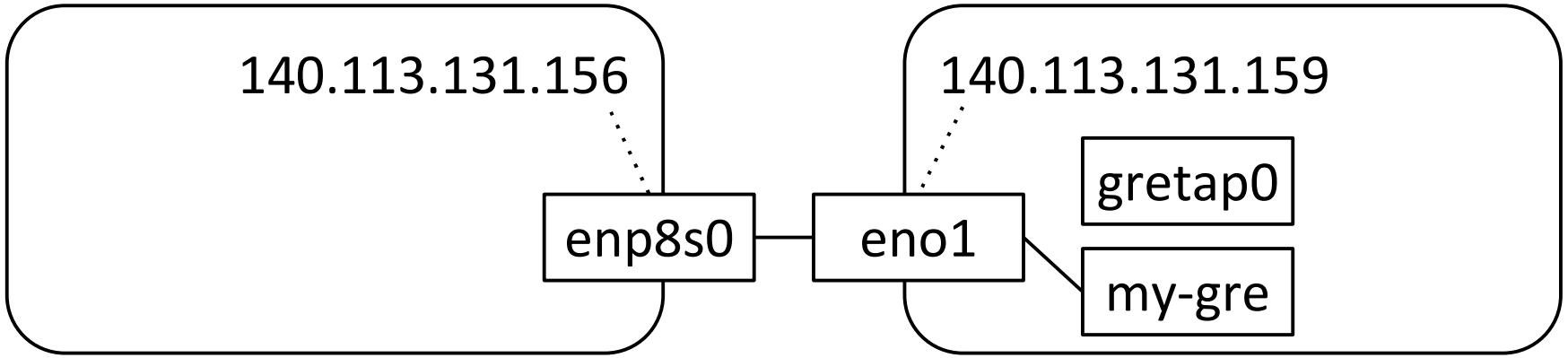
- Have to load fou kernel module manually

```
winlab@server159:~$ sudo ip link add my-gre type gretap remote 140.113.
RTNETLINK answers: Invalid argument
winlab@server159:~$ sudo modprobe fou
winlab@server159:~$ sudo ip link add my-gre type gretap remote 140.113.
winlab@server159:~$
```

- Bring device up

```
winlab@server159:~$ ip link show my-gre
10: my-gre@NONE: <BROADCAST,MULTICAST> mtu 1454 qdisc noop state DOWN mode DEFAULT group
    link/ether da:13:8a:ef:ab:69 brd ff:ff:ff:ff:ff:ff
winlab@server159:~$ sudo ip link set my-gre up
winlab@server159:~$ ip link show my-gre
10: my-gre@NONE: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1454 qdisc fq_codel state UNKNOWN
    link/ether da:13:8a:ef:ab:69 brd ff:ff:ff:ff:ff:ff
```

Create tunnel endpoint (3/3)



- Let’s send something!

Sending test

```
winlab@server159:~$ arping -I my-gre 192.168.123.123 -c 1
ARPING 192.168.123.123 from 140.113.131.159 my-gre
Sent 1 probes (1 broadcast(s))
Received 0 response(s)
winlab@server159:~$
```

Outer Eth

Inner Eth

```
winlab@server156:~$ sudo tcpdump -i enp8s0 udp -XX -n -c 1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp8s0, link-type EN10MB (Ethernet), capture size 262144 bytes
18:13:51.899629 IP 140.113.131.159.33333 > 140.113.131.156.55555: UDP, length 46
        0x0000:  8cea 1b30 da17 d8c4 9786 ad3b 0800 4500  ...0.......;..E.
        0x0010:  004a 85ea 4000 4011 949a 8c71 839f 8c71  .J..@.@....q...q
        0x0020:  839c 8235 d903 0036 0000 0000 6558 ffff  ...5...6....eX..
        0x0030:  ffff ffff da13 8aef ab69 0806 0001 0800  .........i......
        0x0040:  0604 0001 da13 8aef ab69 8c71 839f ffff  .........i.q....
        0x0050:  ffff ffff c0a8 7b7b                      ......{{
1 packet captured
1 packet received by filter
0 packets dropped by kernel
winlab@server156:~$
```

0x8235 = 33333

0xd903 = 55555

Outer UDP

GRE header

Create tunnel endpoint for decapsulation (1/2)

- Syntax from man page

```
ip fou add port PORT { gue | ipproto PROTO } [ local IFADDR ] [
        peer IFADDR ] [ peer_port PORT ] [ dev IFNAME ]
```

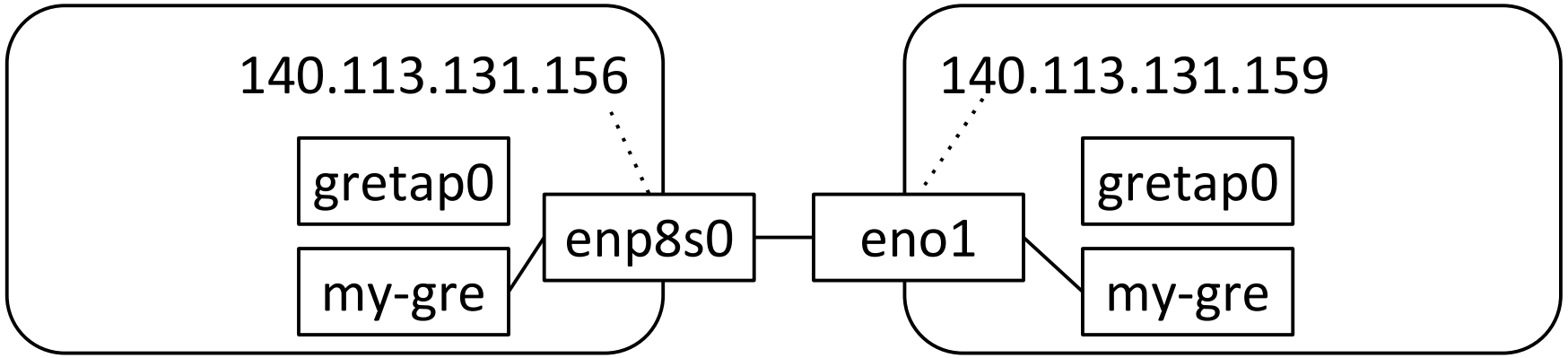
- Configure a FOU receive port for GRE

$ ip fou add port 55555 ipproto 47

$ ip link add my-gre type gretap remote 140.113.131.156 local **140.113.131.159** encap fou encap-sport 33333 encap-dport 55555

$ ip link add my-gre type gretap remote 140.113.131.159 local **140.113.131.156** encap fou encap-sport 55555 encap-dport 33333

Create tunnel endpoint for decapsulation (2/2)



- Let’s receive something!

Receiving test

```
winlab@server159:~$ arping -I my-gre 192.168.123.123 -c 1
ARPING 192.168.123.123 from 140.113.131.159 my-gre
Sent 1 probes (1 broadcast(s))
Received 0 response(s)
winlab@server159:~$
```

```
winlab@server156:~$ sudo tcpdump -i enp8s0 udp -XX -n -c 1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp8s0, link-type EN10MB (Ethernet), capture size 262144 bytes
18:44:37.386372 IP 140.113.131.159.33333 > 140.113.131.156.55555: UDP, length 46
        0x0000:  8cea 1b30 da17 d8c4 9786 ad3b 0800 4500  ...0.......;..E.
        0x0010:  004a 7d02 4000 4011 9d82 8c71 839f 8c71  .J}.@.@....q...q
        0x0020:  839c 8235 d903 0036 0000 0000 6558 ffff  ...5...6....eX..
        0x0030:  ffff ffff da13 8aef ab69 0806 0001 0800  .........i......
        0x0040:  0604 0001 da13 8aef ab69 8c71 839f ffff  .........i.q....
        0x0050:  ffff ffff c0a8 7b7b                      ......{{
1 packet captured
1 packet received by filter
0 packets dropped by kernel
winlab@server156:~$
```

```
winlab@server156:~$ sudo tcpdump -i my-gre -XX -c 1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on my-gre, link-type EN10MB (Ethernet), capture size 262144 bytes
18:44:37.386406 ARP, Request who-has 192.168.123.123 (Broadcast) tell 140.113.131.159, length 28
        0x0000:  ffff ffff ffff da13 8aef ab69 0806 0001  ...........i....
        0x0010:  0800 0604 0001 da13 8aef ab69 8c71 839f  ...........i.q..
        0x0020:  ffff ffff ffff c0a8 7b7b                 ........{{
1 packet captured
1 packet received by filter
0 packets dropped by kernel
winlab@server156:~$
```

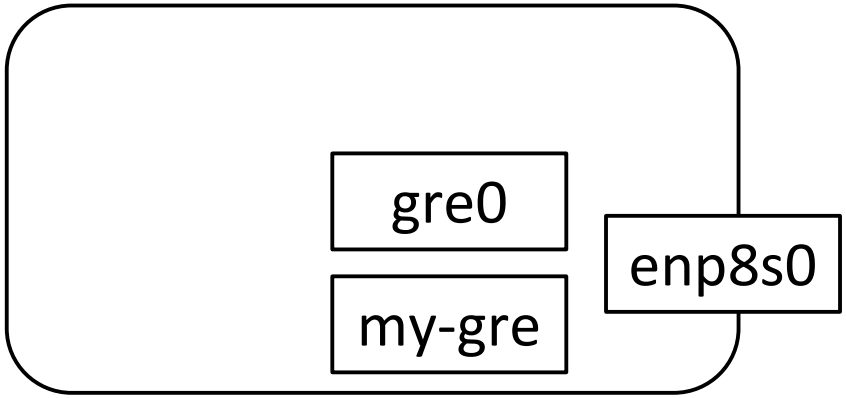
FOU uses UDP socket

- To implement the receive path, an implementation creates an **in-kernel UDP socket** and binds the local port to the port number specified for encapsulation.
- $ ip fou add port 55555 ipproto 47

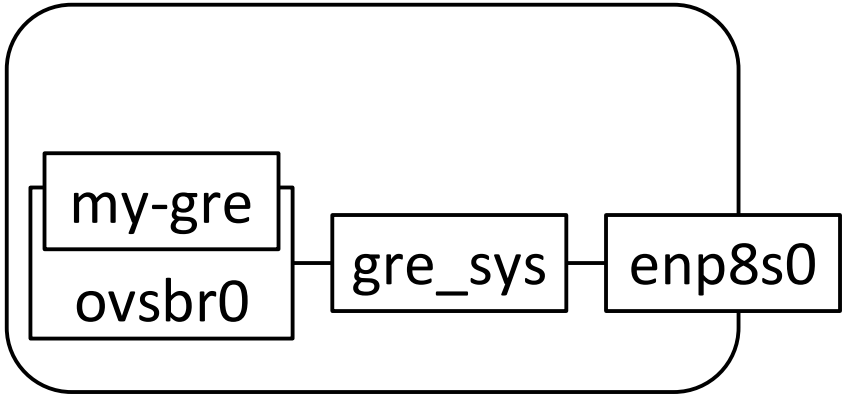
- 55555 UDP port (FOU) has NO PID!!

```
winlab@server156:~$ sudo netstat -ulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
udp        0      0 0.0.0.0:55555           0.0.0.0:*                           -
udp        0      0 127.0.0.53:53           0.0.0.0:*                           1231/systemd-resolv
winlab@server156:~$ nc -u -l localhost 55555
nc: Address already in use
winlab@server156:~$ sudo lsof -i UDP:55555
winlab@server156:~$ sudo lsof -i UDP:53
COMMAND    PID            USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
systemd-r 1231 systemd-resolve   12u  IPv4  15756      0t0  UDP localhost:domain
winlab@server156:~$
```

Appendix: Linux way vs OVS way



One tunnel per netdevice



Flow-based tunnelling where one device is used for all tunnels