Linaro connect
Vancouver 2018

VPN from Upstream Kernel in AOSP

Sam Protsenko

VPN in Android

Two ways to implement VPN in Android:

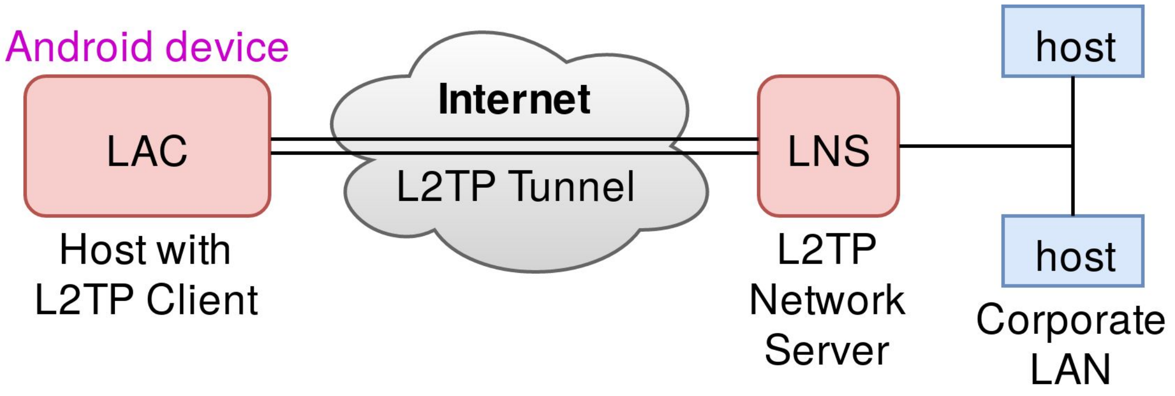
- Application-based VPN
- Legacy VPN

Legacy VPN implementation:

- Protocols: PPTP, L2TP+IPSec
- Kernel drivers (PPPoPNS, PPPoLAC)
- Daemons (mtpd, racoon)

L2TP use-case

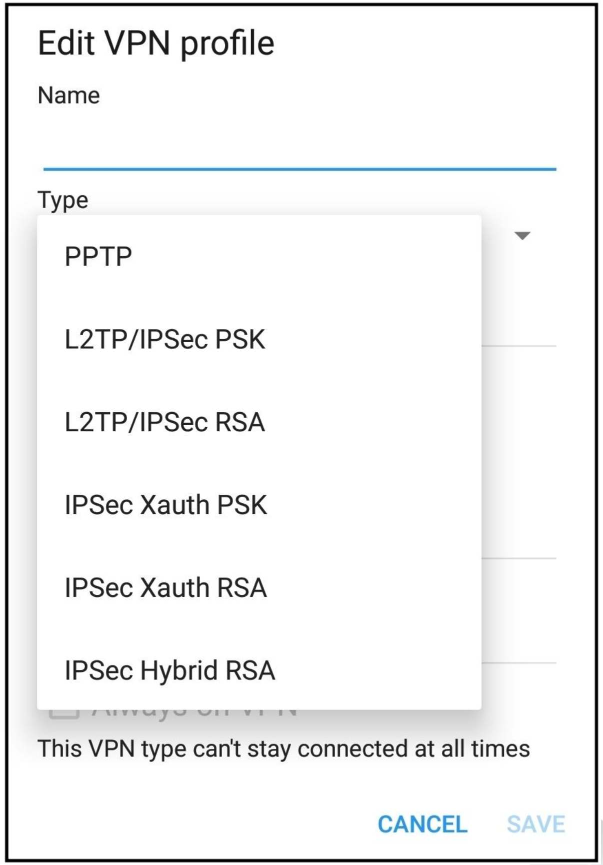
L2TP connection:



Similar scheme exists for PPTP:

- LAC => PNS (client, requests to establish a connection)
- LNS => PAC (server)

VPN via UI



Edit VPN profile

Name

Type

PPTP

L2TP/IPSec PSK

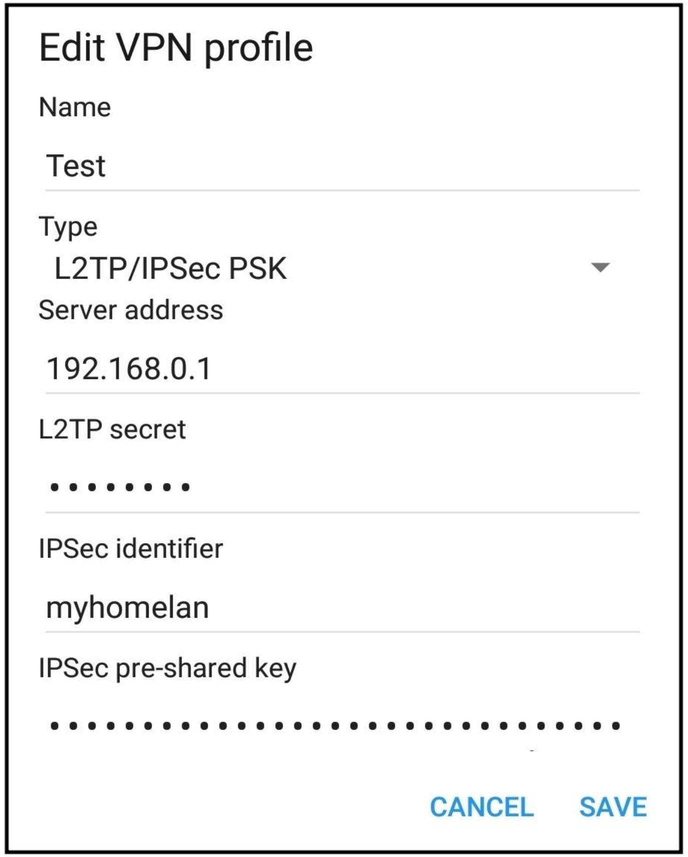
L2TP/IPSec RSA

IPSec Xauth PSK

IPSec Xauth RSA

IPSec Hybrid RSA

This VPN type can't stay connected at all times



Edit VPN profile

Name: Test

Type: L2TP/IPSec PSK

Server address: 192.168.0.1

L2TP secret: ••••••••

IPSec identifier: myhomelan

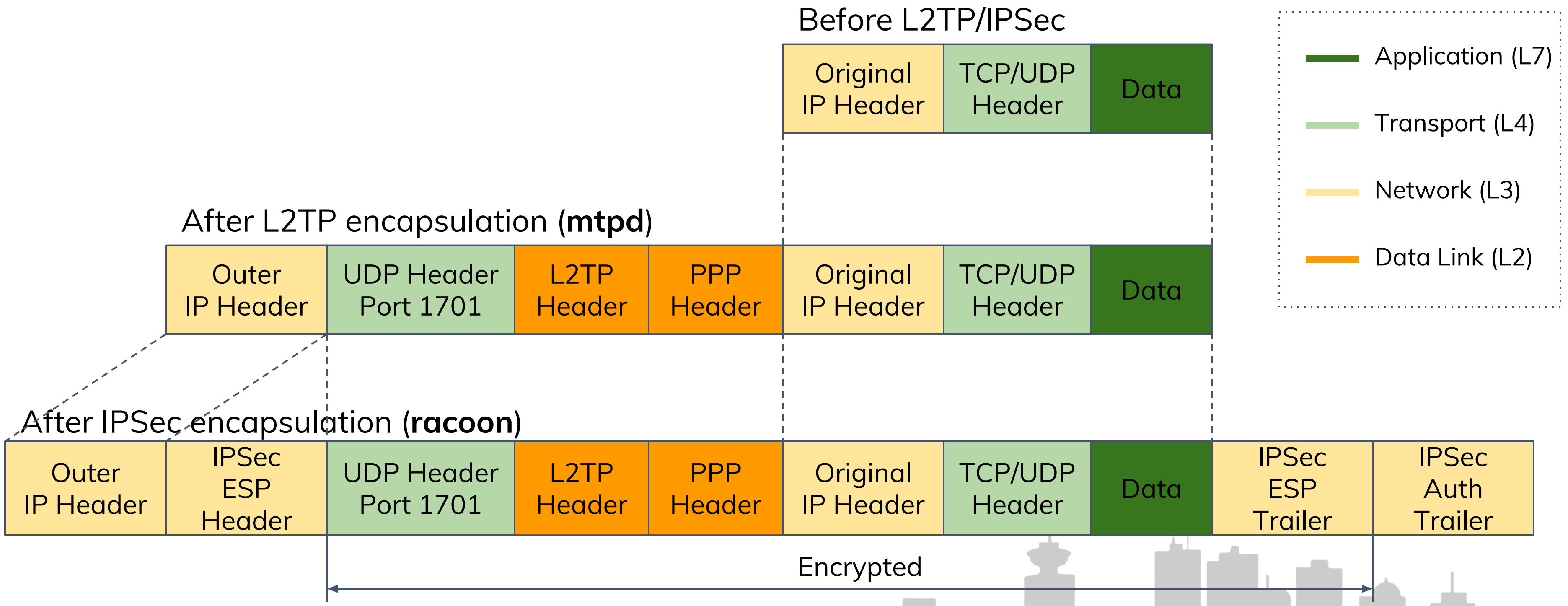
IPSec pre-shared key: ••••••••••••••••••••••••••••••

VPN via console

```
# racoon eth0 192.168.0.1 udppsk myhomelan \
       d41d8cd98f00b204e980 1701 &

# mtpd eth0 l2tp 192.168.0.1 1701 "" linkname vpn name joe \
      password test1234 refuse-eap nodefaultroute \
      usepeerdns idle 1800 mtu 1400 mru 1400 &
```

L2TP/IPSec packet



The problem

Protocol	Android kernel	Upstream kernel
PPTP	drivers/net/ppp/pppopns.c (PX_PROTO_OPNS)	drivers/net/ppp/pptp.c (PX_PROTO_PPTP)
L2TP	drivers/net/ppp/pppolac.c (PX_PROTO_OLAC)	net/l2tp/l2tp_ppp.c (PX_PROTO_OL2TP)

Now that upstream kernel implementation exists, we can adopt it.

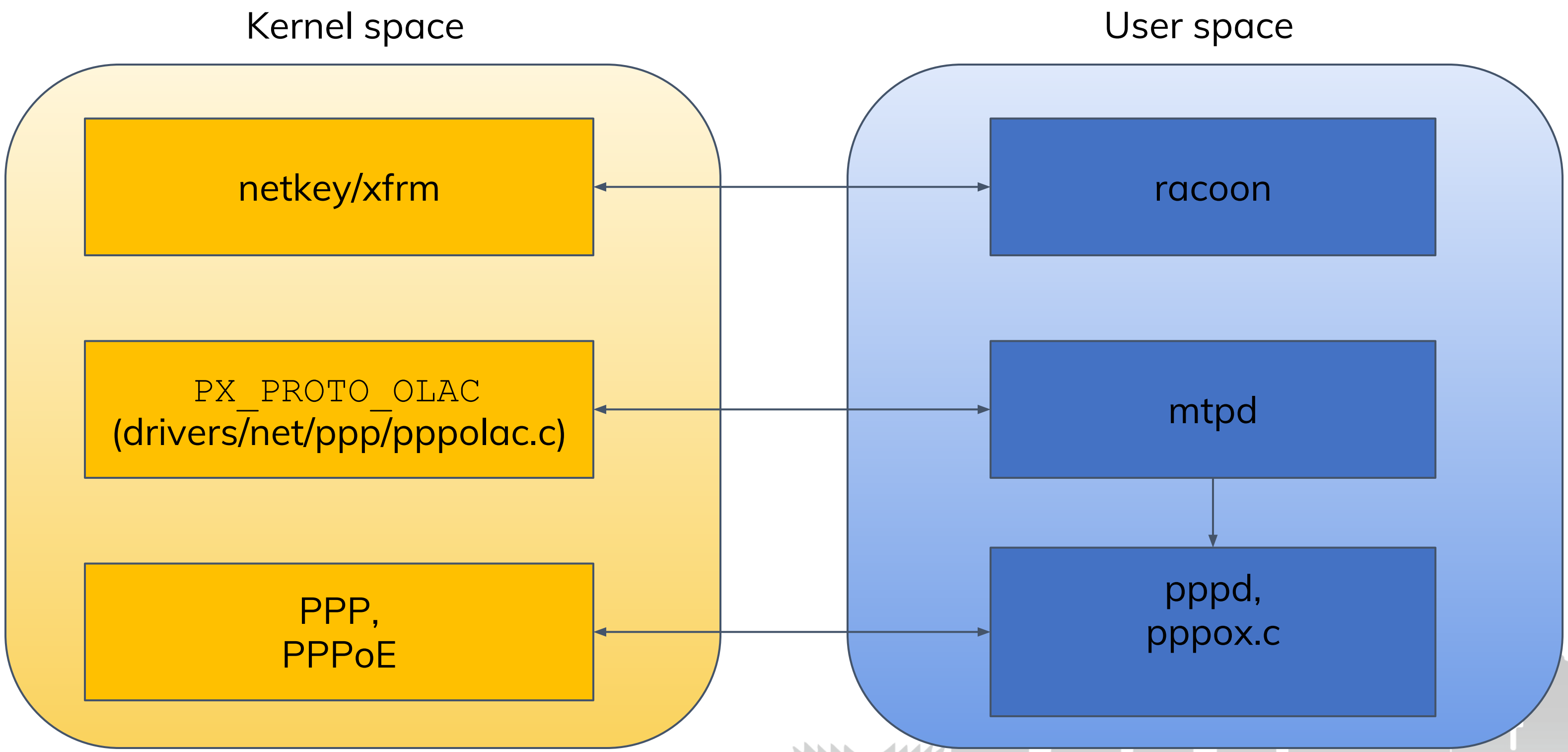
Benefits of using upstream

- Reduce maintenance costs
- Improved security (see CVE lists for L2TP/PPTP in kernel)
- More possible features (L2TPv3, IPv6, etc)
- Avoid code duplication
- More review from related engineers
- **Ultimate goal**: make Android kernel closer to upstream kernel

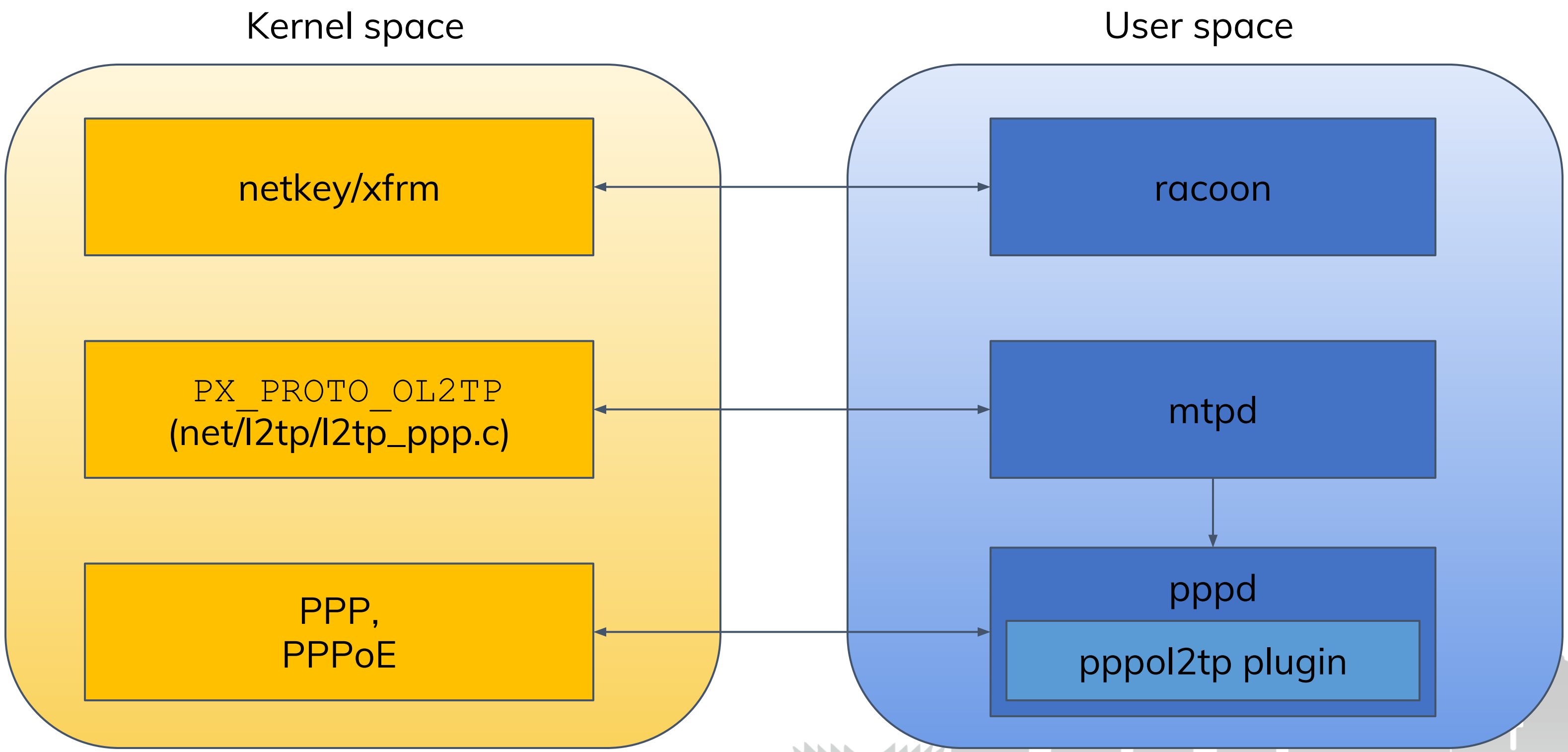
Work process

- Initial patches
  - Technical difficulties, but eventually it's done
  - Submission to Gerrit
- Discussion with Google
- Testing:
  - BeagleBoard X15 (ARMv7)
  - HiKey (ARMv8)
  - Testing in Google lab
- Rebasing, keeping up with new Android features
- Review process, fixing code flaws
- Merging into AOSP/master
- **Current status**: Patches are merged, will be used in Android-Q

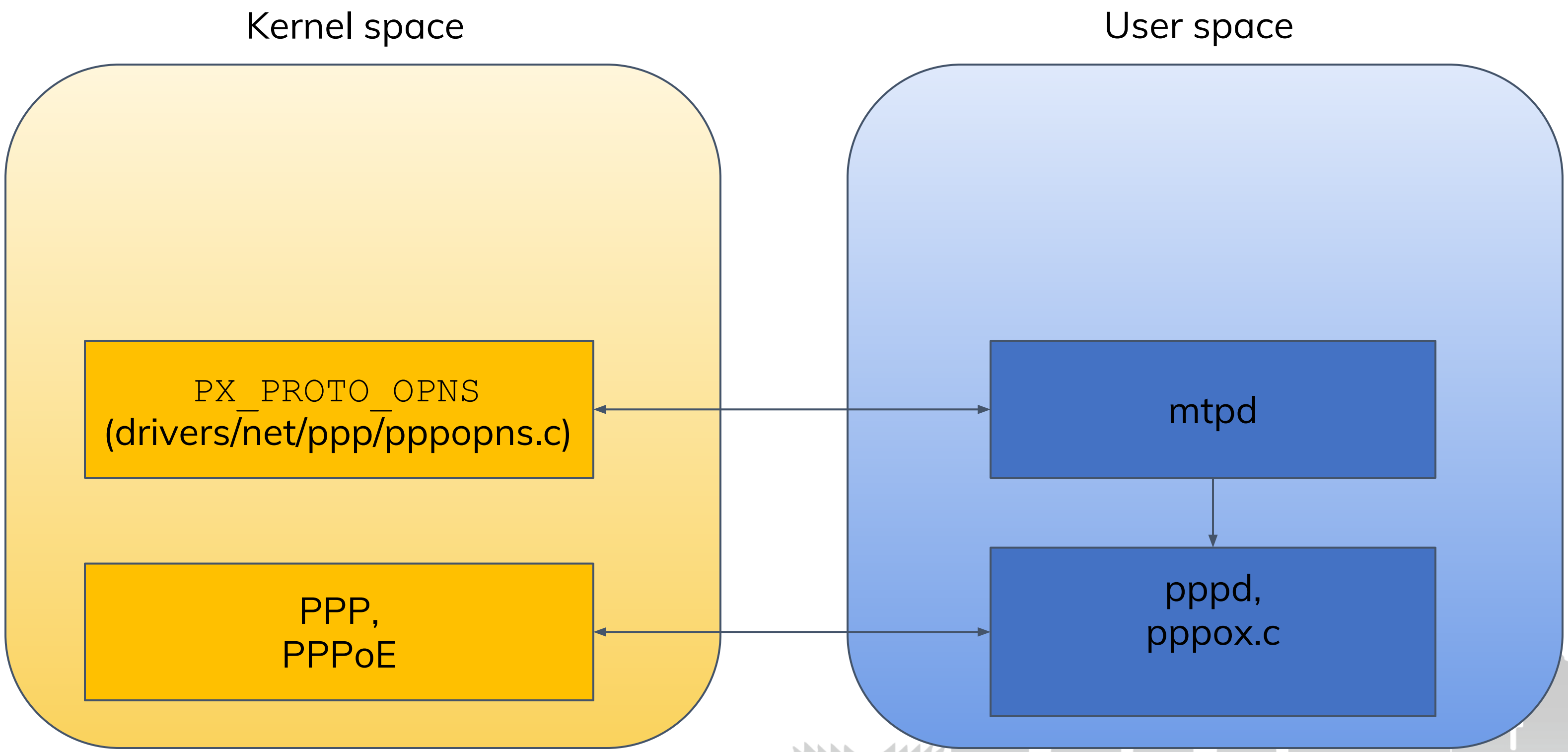
Old L2TP implementation



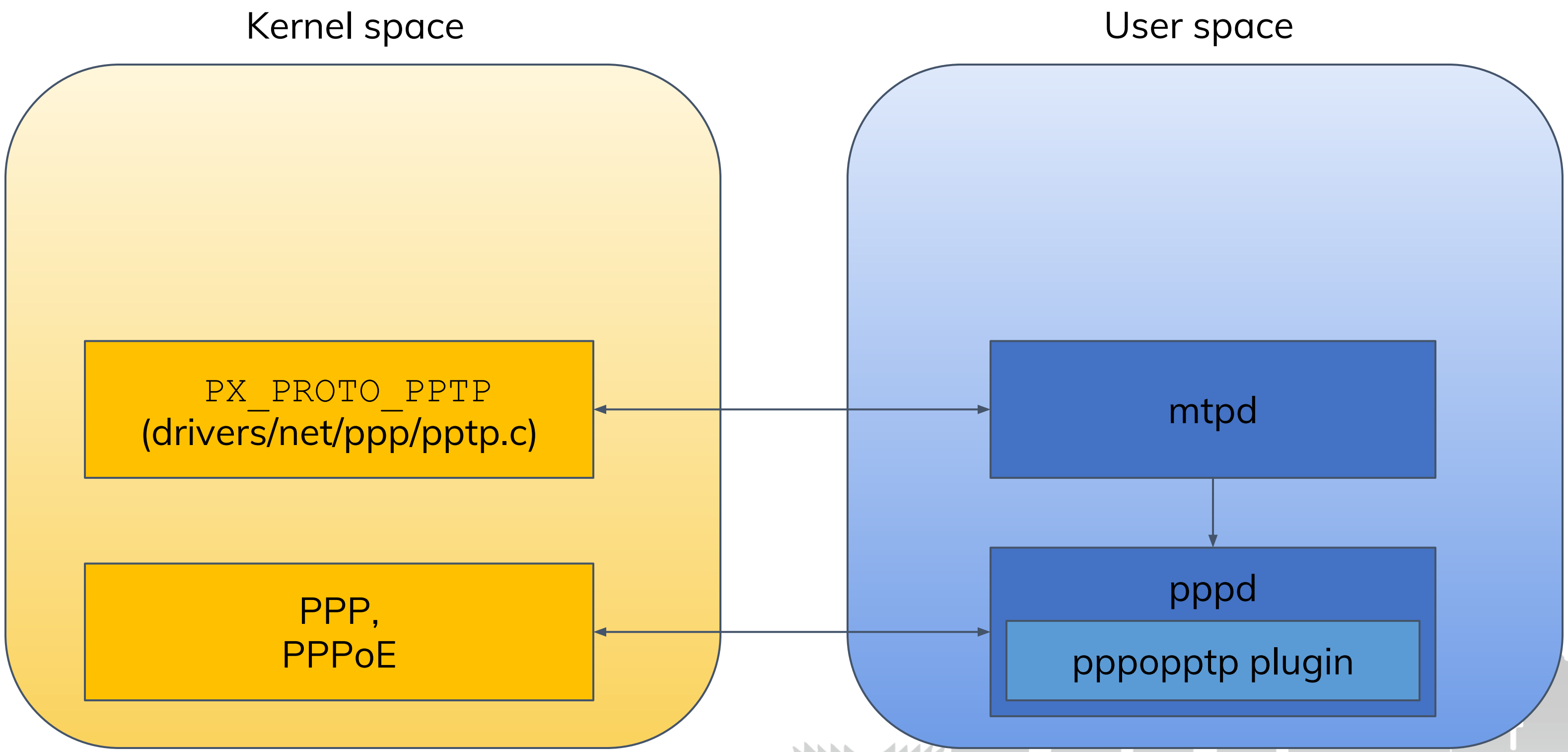
New L2TP implementation



Old PPTP implementation

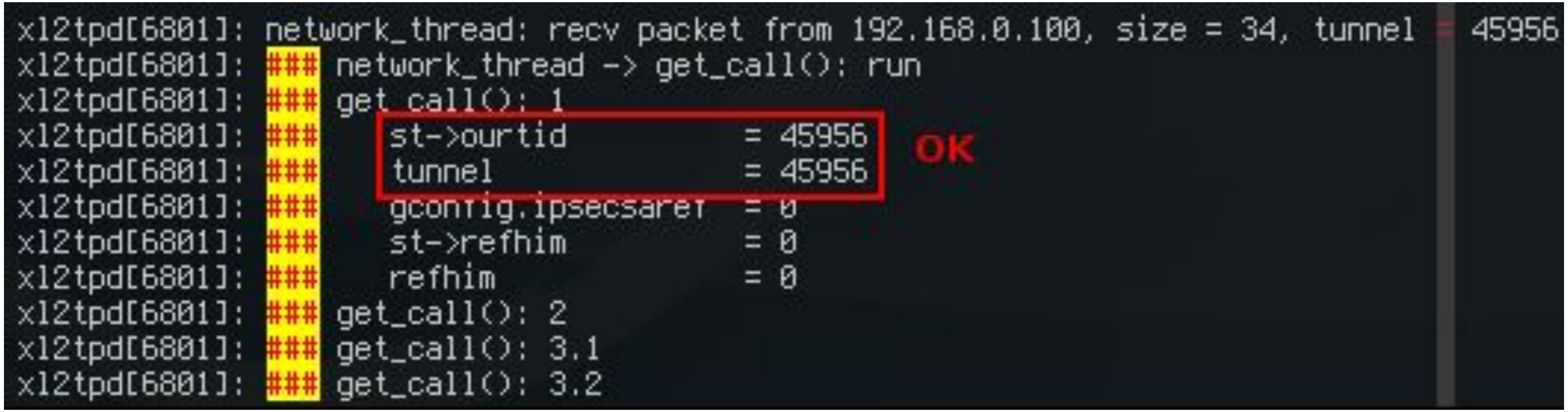


New PPTP implementation



Endianness issues

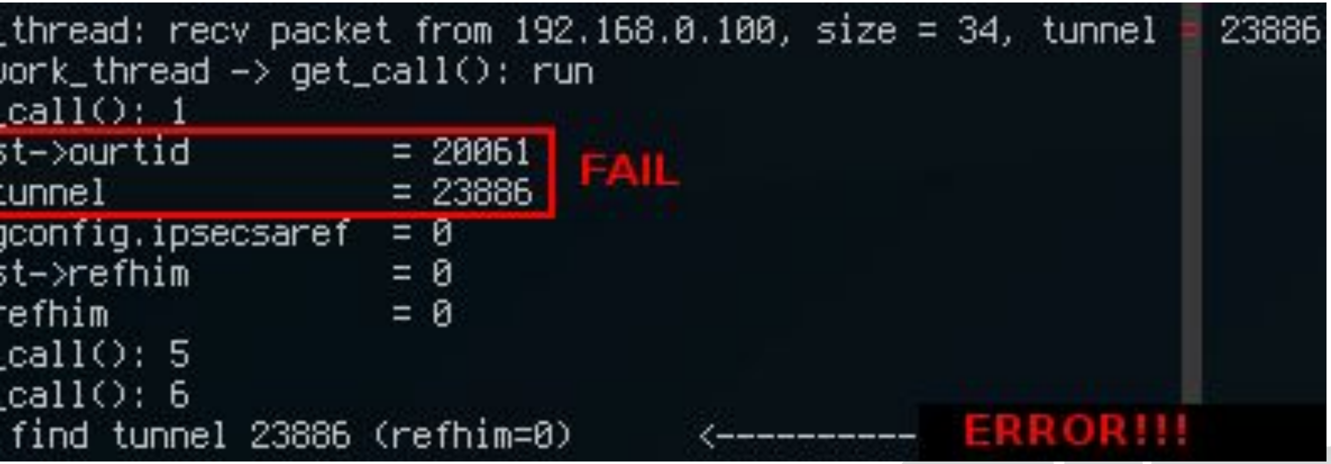
Correct L2TP exchange (using old Android drivers):



```
xl2tpd[6801]: network_thread: recv packet from 192.168.0.100, size = 34, tunnel = 45956
xl2tpd[6801]: ### network_thread -> get_call(): run
xl2tpd[6801]: ### get_call(): 1
xl2tpd[6801]: ###     st->ourtid              = 45956
xl2tpd[6801]: ###     tunnel                  = 45956
xl2tpd[6801]: ###     gconfig.ipsecsaref      = 0
xl2tpd[6801]: ###     st->refhim              = 0
xl2tpd[6801]: ###     refhim                  = 0
xl2tpd[6801]: ### get_call(): 2
xl2tpd[6801]: ### get_call(): 3.1
xl2tpd[6801]: ### get_call(): 3.2
```

Endianness issues (cont'd)

Wrong L2TP exchange (using upstream drivers):



```
_thread: recv packet from 192.168.0.100, size = 34, tunnel = 23886
work_thread -> get_call(): run
_call(): 1
st->ourtid          = 20061
tunnel              = 23886
gconfig.ipsecsaref  = 0
st->refhim          = 0
refhim              = 0
_call(): 5
_call(): 6
 find tunnel 23886 (refhim=0)        <----------  ERROR!!!
```

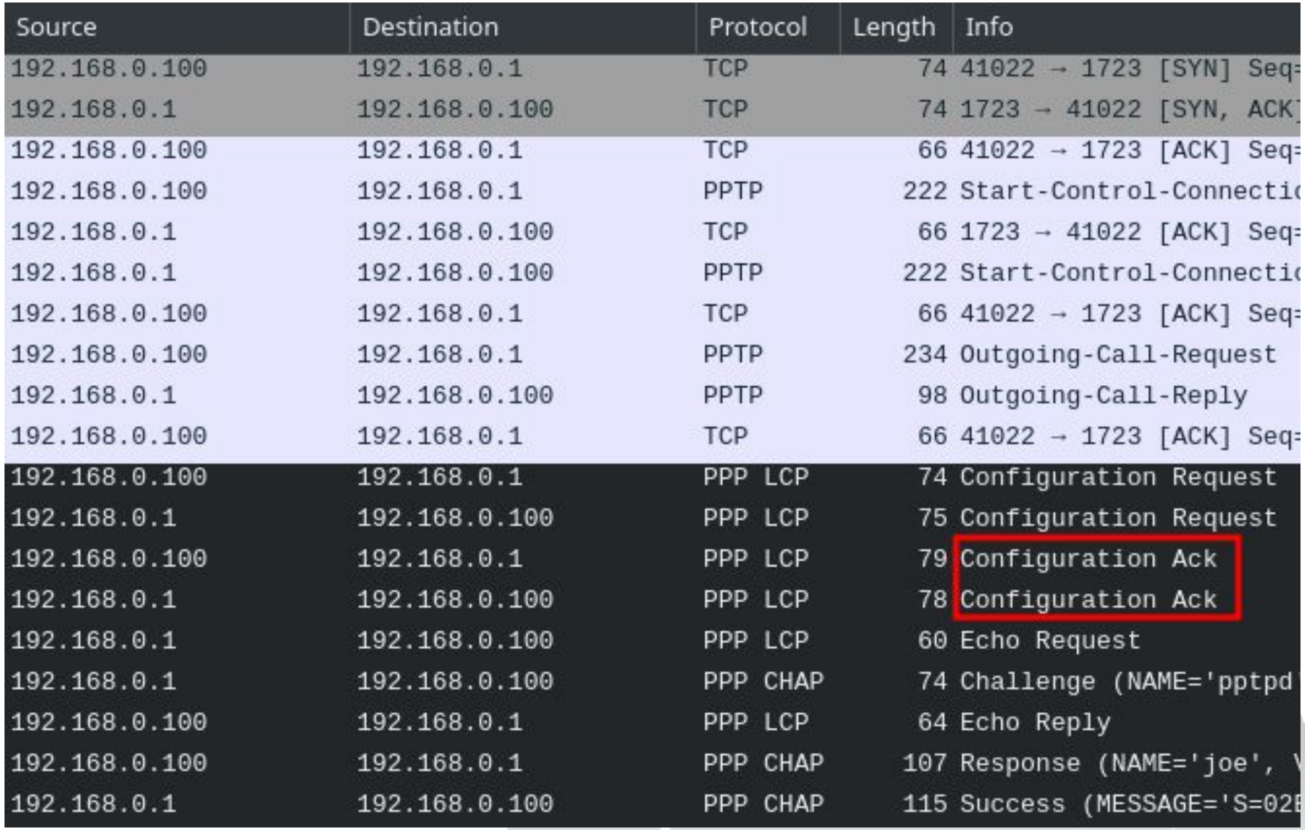
Endianness issues (cont'd)

```
struct sockaddr_pppolac address = {
    .sa_family = AF_PPPOX,
    .sa_protocol = PX_PROTO_OLAC,
    .udp_socket = the_socket,
    .local = {
      .tunnel = local_tunnel,
      .session = local_session
    },
    .remote = {
      .tunnel = remote_tunnel,
      .session = remote_session
    },
};
```

```
session_sa.sa_family = AF_PPPOX;
session_sa.sa_protocol = PX_PROTO_OL2TP;
session_sa.pppol2tp.fd = the_socket;
session_sa.pppol2tp.s_tunnel =
    ntohs(local_tunnel);
session_sa.pppol2tp.s_session =
    ntohs(local_session);
session_sa.pppol2tp.d_tunnel =
    ntohs(remote_tunnel);
session_sa.pppol2tp.d_session =
    ntohs(remote_session);
```

Missing fields in struct

Good:

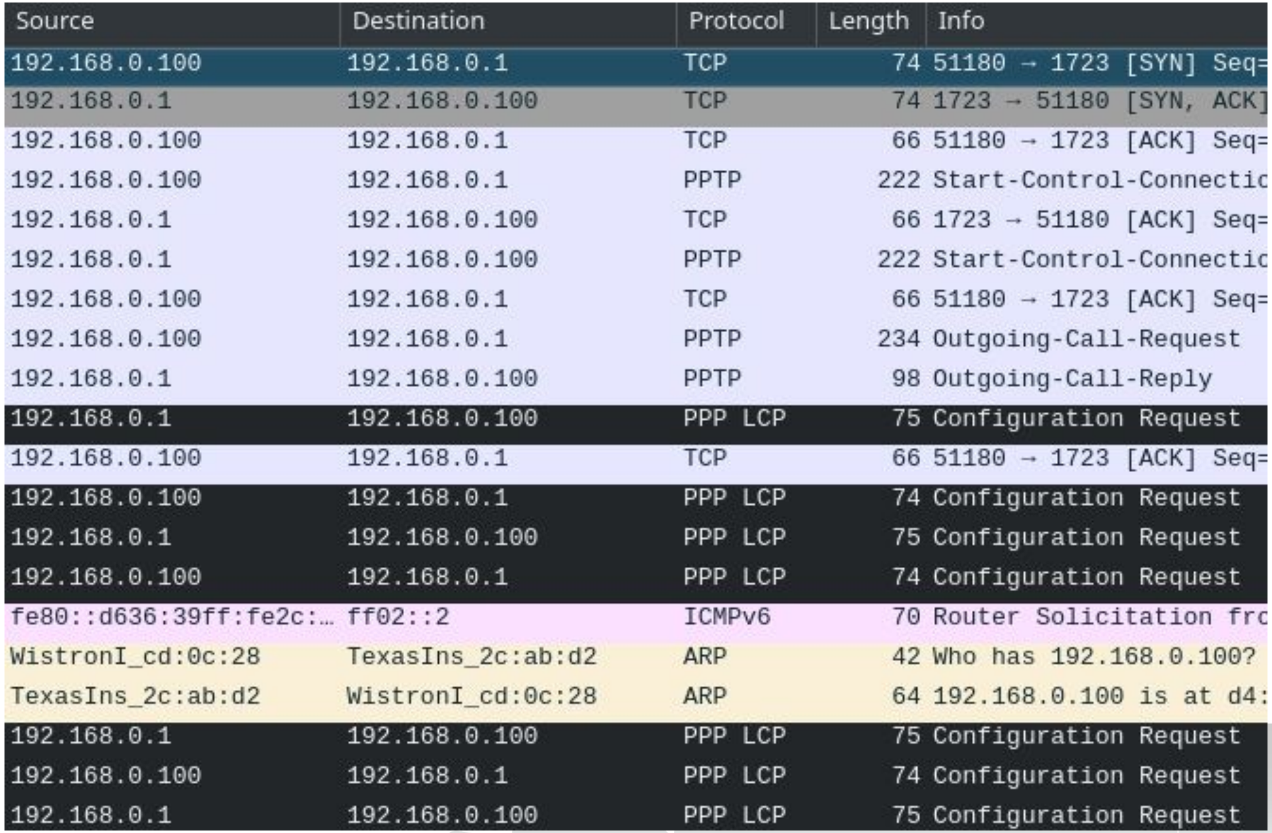


Source	Destination	Protocol	Length	Info
192.168.0.100	192.168.0.1	TCP	74	41022 → 1723 [SYN] Seq=
192.168.0.1	192.168.0.100	TCP	74	1723 → 41022 [SYN, ACK]
192.168.0.100	192.168.0.1	TCP	66	41022 → 1723 [ACK] Seq=
192.168.0.100	192.168.0.1	PPTP	222	Start-Control-Connecti
192.168.0.1	192.168.0.100	TCP	66	1723 → 41022 [ACK] Seq=
192.168.0.1	192.168.0.100	PPTP	222	Start-Control-Connecti
192.168.0.100	192.168.0.1	TCP	66	41022 → 1723 [ACK] Seq=
192.168.0.100	192.168.0.1	PPTP	234	Outgoing-Call-Request
192.168.0.1	192.168.0.100	PPTP	98	Outgoing-Call-Reply
192.168.0.100	192.168.0.1	TCP	66	41022 → 1723 [ACK] Seq=
192.168.0.100	192.168.0.1	PPP LCP	74	Configuration Request
192.168.0.1	192.168.0.100	PPP LCP	75	Configuration Request
192.168.0.100	192.168.0.1	PPP LCP	79	Configuration Ack
192.168.0.1	192.168.0.100	PPP LCP	78	Configuration Ack
192.168.0.1	192.168.0.100	PPP LCP	60	Echo Request
192.168.0.1	192.168.0.100	PPP CHAP	74	Challenge (NAME='pptpd'
192.168.0.100	192.168.0.1	PPP LCP	64	Echo Reply
192.168.0.100	192.168.0.1	PPP CHAP	107	Response (NAME='joe', V
192.168.0.1	192.168.0.100	PPP CHAP	115	Success (MESSAGE='S=02E

Missing fields in struct

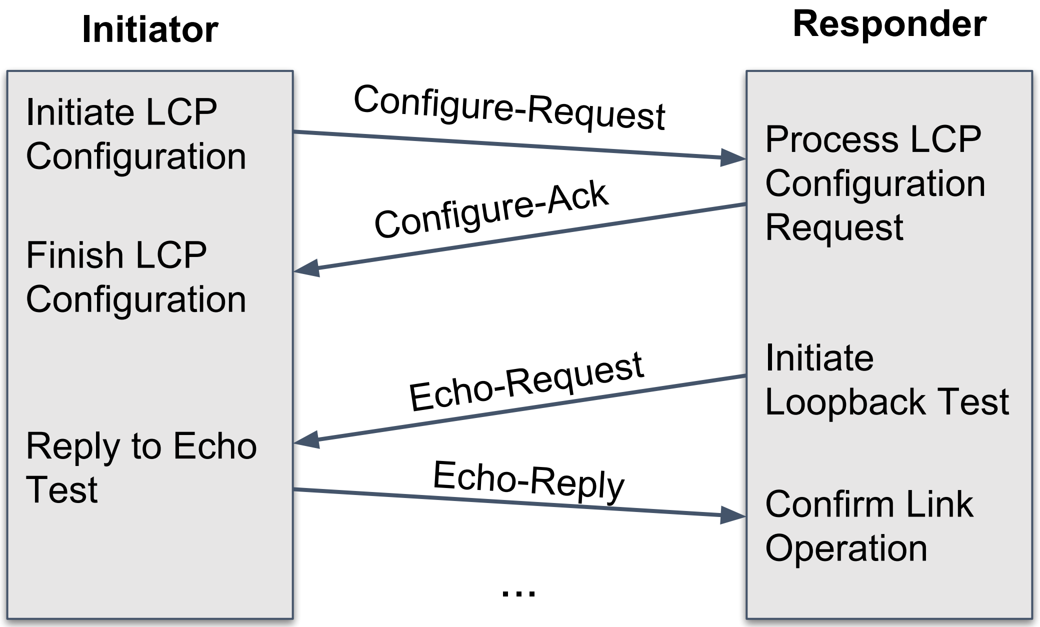
Bad:

(no ACK)



Source	Destination	Protocol	Length	Info
192.168.0.100	192.168.0.1	TCP	74	51180 → 1723 [SYN] Seq=
192.168.0.1	192.168.0.100	TCP	74	1723 → 51180 [SYN, ACK]
192.168.0.100	192.168.0.1	TCP	66	51180 → 1723 [ACK] Seq=
192.168.0.100	192.168.0.1	PPTP	222	Start-Control-Connectic
192.168.0.1	192.168.0.100	TCP	66	1723 → 51180 [ACK] Seq=
192.168.0.1	192.168.0.100	PPTP	222	Start-Control-Connectic
192.168.0.100	192.168.0.1	TCP	66	51180 → 1723 [ACK] Seq=
192.168.0.100	192.168.0.1	PPTP	234	Outgoing-Call-Request
192.168.0.1	192.168.0.100	PPTP	98	Outgoing-Call-Reply
192.168.0.1	192.168.0.100	PPP LCP	75	Configuration Request
192.168.0.100	192.168.0.1	TCP	66	51180 → 1723 [ACK] Seq=
192.168.0.100	192.168.0.1	PPP LCP	74	Configuration Request
192.168.0.1	192.168.0.100	PPP LCP	75	Configuration Request
192.168.0.100	192.168.0.1	PPP LCP	74	Configuration Request
fe80::d636:39ff:fe2c:…	ff02::2	ICMPv6	70	Router Solicitation fro
WistronI_cd:0c:28	TexasIns_2c:ab:d2	ARP	42	Who has 192.168.0.100?
TexasIns_2c:ab:d2	WistronI_cd:0c:28	ARP	64	192.168.0.100 is at d4:
192.168.0.1	192.168.0.100	PPP LCP	75	Configuration Request
192.168.0.100	192.168.0.1	PPP LCP	74	Configuration Request
192.168.0.1	192.168.0.100	PPP LCP	75	Configuration Request

PPP LCP Message Exchange



Missing fields in struct (cont'd)

```
struct sockaddr_pppox src, dst;

src.sa_family = AF_PPPOX;
src.sa_protocol = PX_PROTO_PPTP;
src.sa_addr.pptp.call_id = ntohs(local);
src.sa_addr.pptp.sin_addr = local_addr;

dst.sa_family = AF_PPPOX;
dst.sa_protocol = PX_PROTO_PPTP;
dst.sa_addr.pptp.call_id = ntohs(remote);
dst.sa_addr.pptp.sin_addr = remote_addr;
```

Compatibility concerns

- If upstream kernel L2TP is not enabled, mtpd will fallback to Android one:

```
if (check_ol2tp()) {
    create_pppox_ol2tp(...);
    start_pppd_ol2tp(...);
} else {
    start_pppd(create_pppox_olac());
}
```

- The same goes for PPTP

Patches: external/ppp

- pppd: Remove obsolete way of receiving args from mtpd
- pppd: Enable plugin support in pppd
- pppd: Convert Android.mk to Android.bp

- pppd: Add pppol2tp-android plugin
- pppd: Fix pppol2tp-android.so build
- pppd: Add rules for building the pppol2tp-android plugin

- pppd: Add pppopptp-android plugin

Patches: external/mtpd

- mtpd: Remove obsolete way of passing args to pppd

- mtpd: l2tp: Fix endianness issues in log prints
- mtpd: Use L2TP implementation from mainline kernel

- mtpd: pptp: Fix endianness issues in log prints
- mtpd: Use PPTP implementation from upstream kernel

Patches: kernel/configs

- Enable L2TP and PPTP from upstream kernel:

```
- CONFIG_PPPOLAC=y
- CONFIG_PPPOPNS=y
+ CONFIG_PPPOL2TP=y
+ CONFIG_PPTP=y
```

References

1. “Android Security Internals” by Nikolay Elenkov
2. “Linux Networking Architecture” by Klaus Wehrle
3. https://wiki.linaro.org/LMG/Kernel/PPP
4. https://android-review.googlesource.com/q/topic:pppolac+status:merged
5. https://android-review.googlesource.com/q/topic:pppopns+status:merged

Thank you!

Questions?

Sam Protsenko

<semen.protsenko@linaro.org>