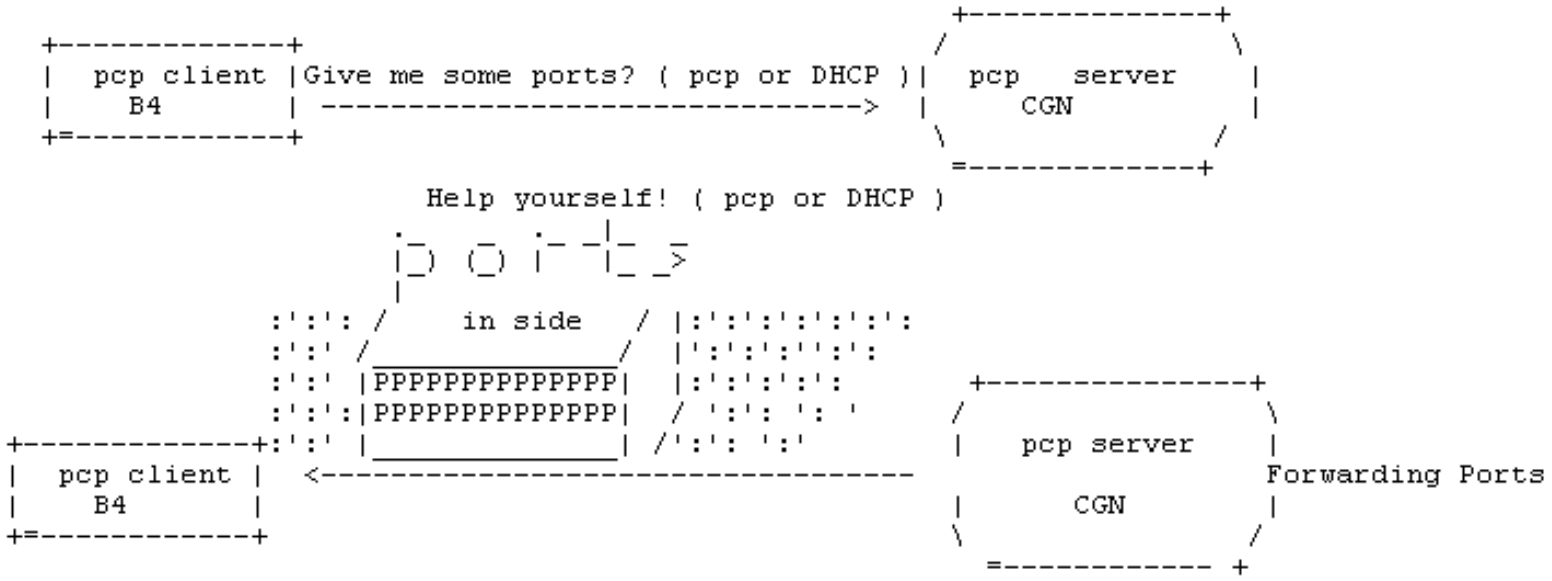
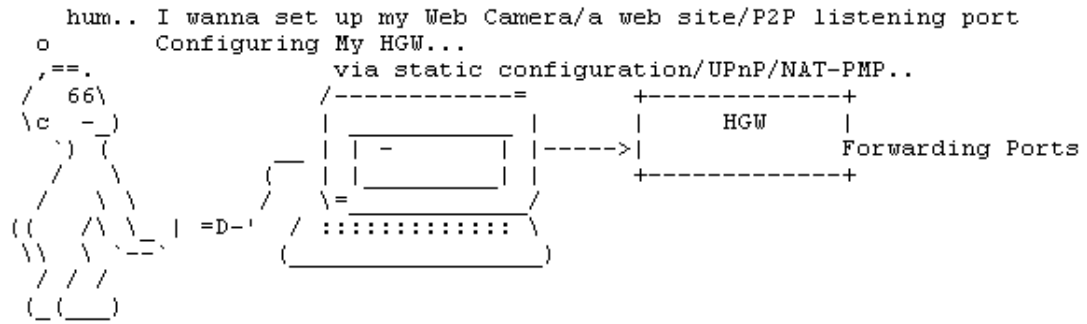
CGN NAT Bypass

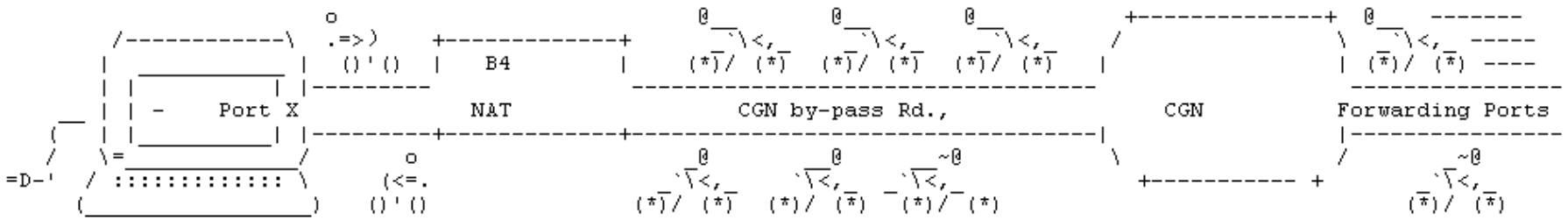
X. Deng, M. Boucadair
France Telecom
C. Zhou Huawei Technologies
T. Tsou Huawei Technologies (USA)
G. Bajko Nokia

Core idea


pcp client
B4
Give me some ports? (pcp or DHCP)
pcp server
CGN
Help yourself! (pcp or DHCP)
in side
pcp client
B4
pcp server
CGN
Forwarding Ports


hum.. I wanna set up my Web Camera/a web site/P2P listening port
Configuring My HGW...
via static configuration/UPnP/NAT-PMP..
HGW
Forwarding Ports

How it works



- Outbound Session
  - B4 behaviors
    - NAT & Encapsulation
  - AFTR behaviors
    - De-capsulation & forward
- Inbound Session
  - AFTR behaviors
    - Encapsulation & port forwarding
  - B4 behaviors
    - De-capsulation & NAT

Scattered ports provisioning

- What's the benefits of provisioning scattered ports?
  - For incoming ports
  - Scattered ports allocation is more likely to satisfy the random incoming port requests from applications
    - such as eMule, uTorrent,sharez, using UPnP 1.0

- A solution
  - to distribute bulks of non-continuous ports among subscribers,
  - also takes port randomization into account

How to provision scattered ports?

- Only two parameters
- Subscribers ID pattern

1st	2nd	3rd	4th	5th	6th	7th	8th
0	s	0	0	s	0	s	0
9th	**10th**	**11th**	**12th**	**13th**	**14th**	**15th**	**16th**
s	0	s	0	0	0	0	0

- Subscribers ID value

1st	2nd	3rd	4th	5th	6th	7th	8th
1	**0**	1	1	**0**	1	**0**	1
9th	**10th**	**11th**	**12th**	**13th**	**14th**	**15th**	**16th**
0	1	**1**	1	1	1	1	1

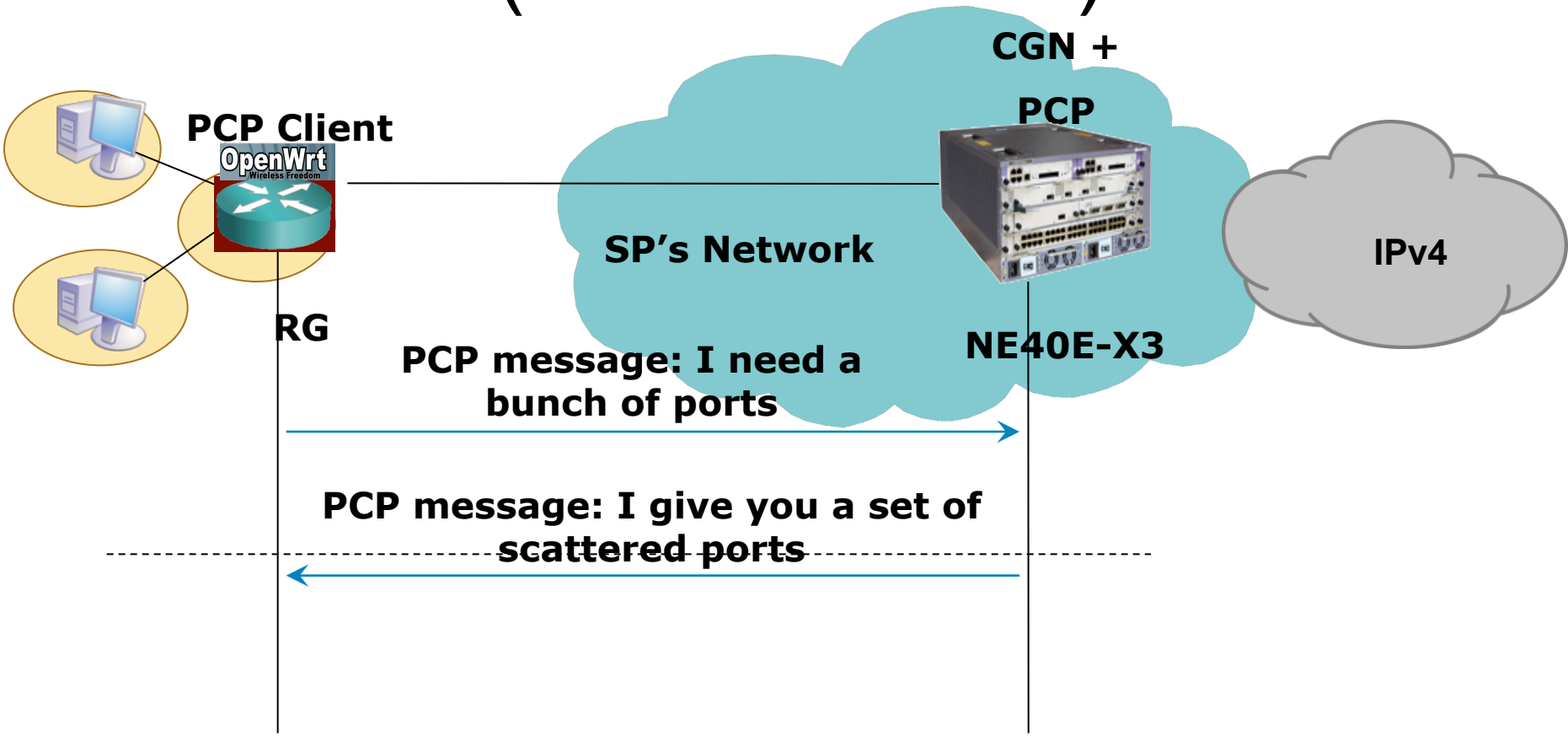
Random ephemeral port selection within the restricted port for CPE NAT

- Subscribers ID pattern
- Subscribers ID value

Only one line code needs to be changed!

```
do{
    restricted_next_ephemeral = (random()|subscriber_ID_pattern)
                                & subscriber_ID_value;
    if(five-tuple is unique)
    return restricted_next_ephemeral;
}
```

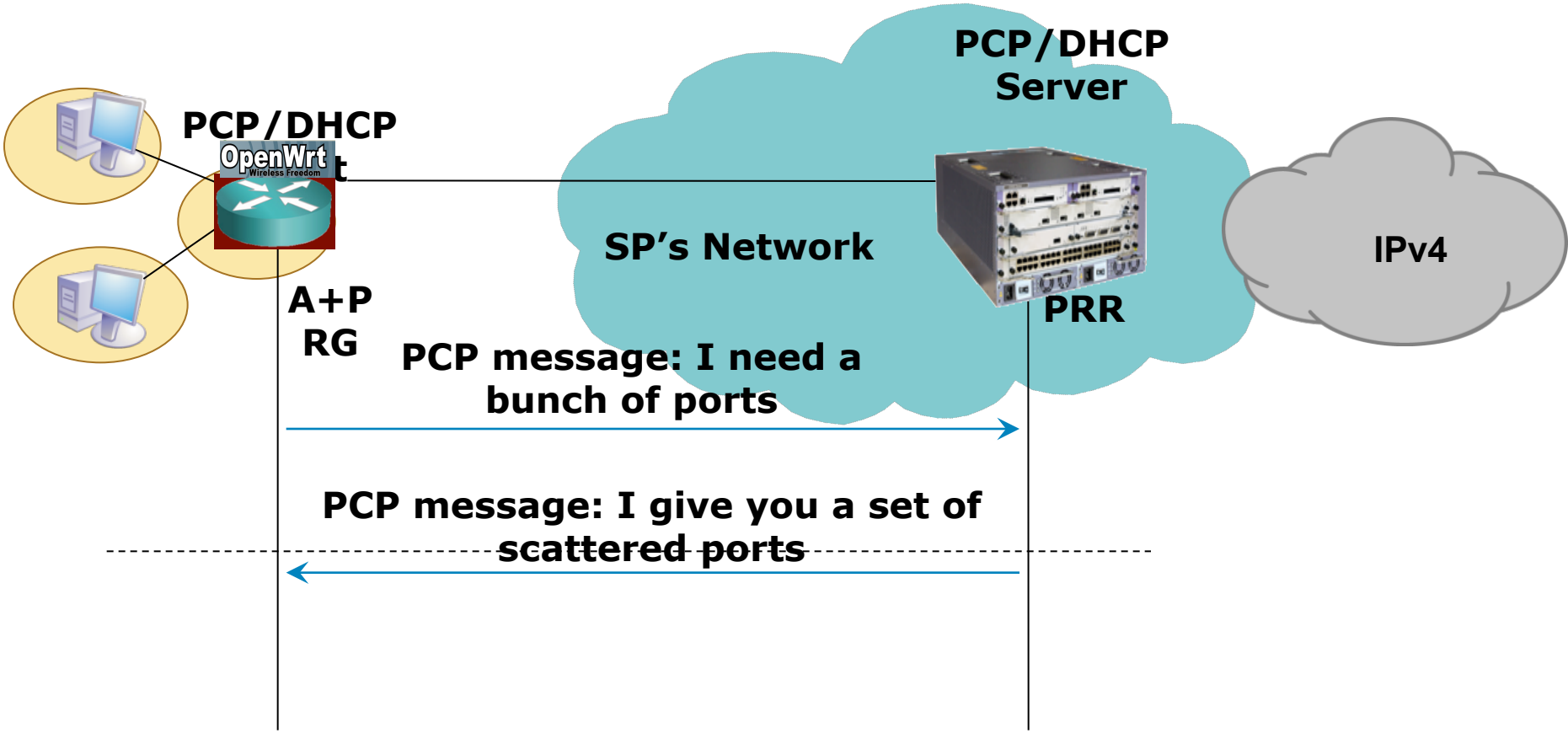
An Implementation DS-Lite encapsulation (demonstrated)

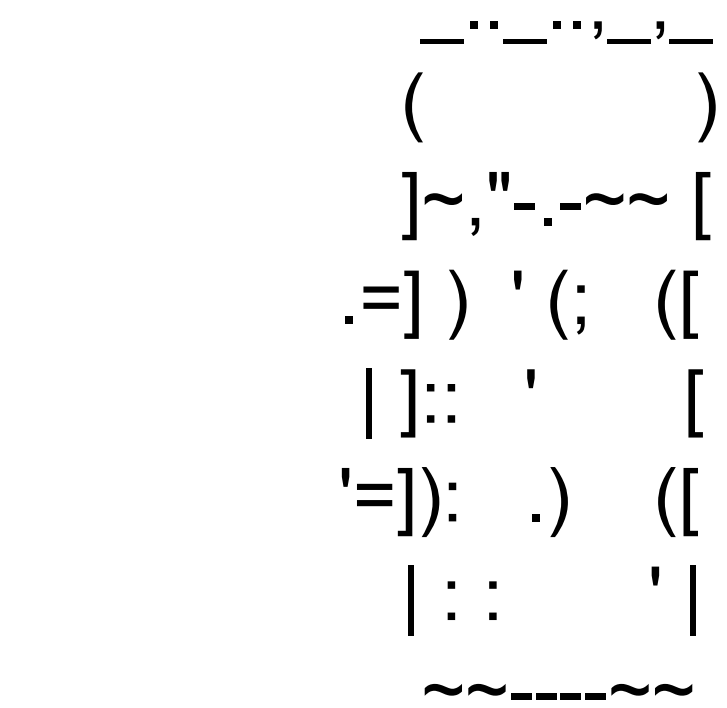


Location: **2000D**

Check out website for this demo: http://130.129.48.23:35328/

Alternative implementation A+P encapsulation





Thoughts?