Netzwerke und Kommunikation
B-LS-MI 004
Homework: Network Address Translation
NAT

rolf.schmutz@fhnw.ch

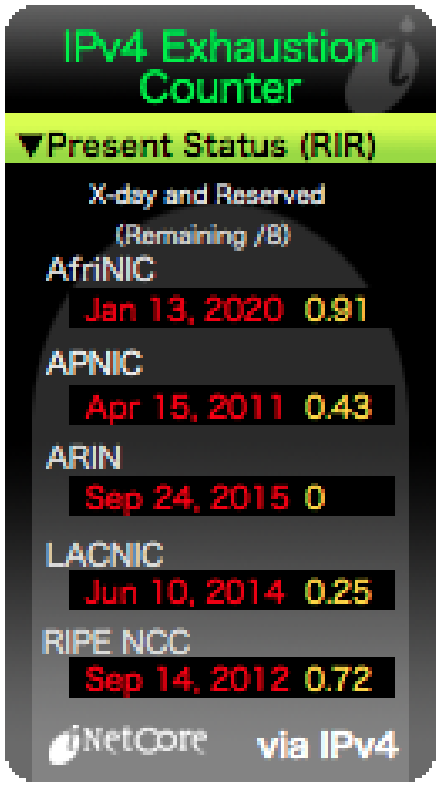
FHNW

21. Oktober 2020

Ziele

- Sie kennen die Verwendung der RFC1918 *private addresses* und die dazu nötigen Vorkehrungen
- . . .die Adressbereiche
- die nötige Statustabelle
- die "Vorschriften" zur Verwendung
- die Einsatzmöglichkeiten

IPv4 Adressenknappheit



http://inetcore.com/project/ipv4ec/en-us/index.html

→ IPv4-Adressen sind eine sehr knappe Resource

IPv4 Adressenknappheit Problem

- Grosse Adressblöcke werden intern[1] verwendet
- Traditionell reservierte Blöcke[2]

→ eigentlich noch viele Adressen "frei" aber nicht zur öffentliche Nutzung

Update/2020:
Die Situation mit den "grossen privaten" Adressblöcken hat sich entspannt/verlagert:

- grosse Netzwerke wurden freigegeben (1/8, 8/8, 9/8, 3/8, 5/8, etc)
- ...und wurden gleich wieder von grossen Akteuren[3] "geschluckt"

[1] d.h. werden im Internet nicht "geroutet" und sind somit nicht eigentlich Bestandteil des Internets → Verschwendet z.B. die Schweizerische Post: 138.191/16=intern, Apple Inc 17/8 meist intern verwendet

[2] z.B. 127/8, 0/8, etc

[3] z.B. Cloud-Infrastruktur-Betreiber, die Verwendung ist aber ähnlich wie bei Internet-Service-Provider

IPv4 Adressenknappheit Lösung[9]: *private networks*

- Reine Client-Computer[4] müssen *aus* dem Internet nicht sichtbar sein[5]
  → diese Adressen müssen nicht eindeutig sein
- Um trotzdem das korrekte Routing im Internet sicherzustellen *müssen* solche Adressen spätestens beim Verbindungsrouter ins Internet in eine *öffentliche*[6] IP-Adresse umgewandelt werden
- das private Netzwerk wird gegenüber dem Internet hinter einer[7] *öffentlichen* Adresse "verborgen"

→ drei Netzwerke sind in RFC1918[8] zur *internen* Verwendung freigegeben:

192.168.0.0/16
172.16.0.0/12
10.0.0.0/8

[4] Arbeitsstationen zum surfen, mailen, etc

[5] Bzw. der Verbindungsaufbau geht immer *von* diesen Computer aus

[6] *public*

[7] oder mehrere. . .

[8] http://www.rfc-editor.org/rfc/pdfrfc/rfc1918.txt.pdf

[9] sort of. . .

NAT: "Vorschriften"

- die drei RFC1918-Bereiche 10/8, 172.16/12 und 192.168/16 dürfen uneingeschränkt in einem privaten/lokalen Netzwerk-Verbund benutzt werden
- es darf nie eine solche Adresse "ins Internet" gelangen – d.h. in den öffentlichen Teil des Internets[10]

[10] source: Antwort/Rückweg würde nicht funktionieren, destination: kann nicht in Routing-Tables im Internet gefunden werden (per Definition)

NAT: Spielarten

- "Masquerading": n:1 dynamisches NAT[11], one-way. Das, was Sie zuhause benutzen
- static n:n oder n:m NAT: feste Zuordnung[12] von *private* zu *public* Adressen
- dynamic n:m NAT[13]: ein (grösserer) *private* Bereich wird auf einen (kleineren) *public* Bereich "gemappt"
- PAT[14], port-address-translation: wie *masquerading* aber mit spezifischen Umleitungen von *eingehenden* Verbindungen auf spezielle Ports zu festen *private* Adressen[15]

[11] also PAT, port-address-translation

[12] . . .dies spart natürlich keine public-IPs

[13] das ist in der FHNW für Clients so eingerichtet

[14] . . .ja, da herrscht ein wenig Begriffsverwirrung. Viele Hersteller von SOHO-Router (Zyxel, Netgear, etc) benutzen diese Terminologie

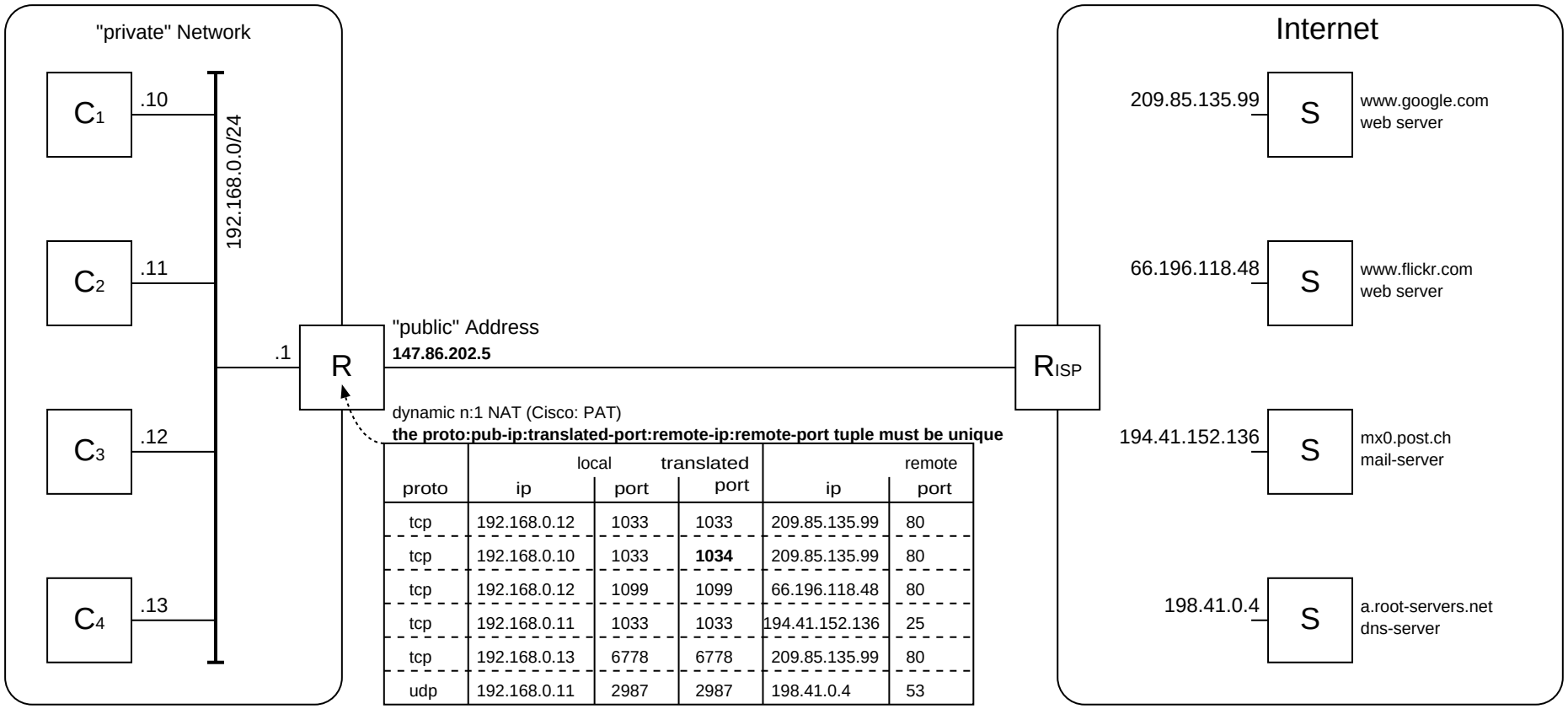
[15] z.B. um zuhause auf einem PC mit *private* Adresse einen Webserver zu betreiben

NAT: Problemstellung

- Ausgehende *private* IP-Adressen müssen auf *public* umgeschrieben werden
- bei eingehenden Paketen muss die Zieladresse[16] vor der Weiterleitung in das interne Netz wieder in die entsprechende *private* Adresse umgeschrieben werden
  $\rightarrow$ d.h. wir brauchen eine Statustabelle
- Einträge in der Statustabelle müssen *eindeutig* aus den eingehenden Paketen identifiziert werden können
  $\rightarrow$ dazu wird meistens ein 5 Tuple:
  {protocol, local-translated-ip, local-translated-port, remote-ip, remote-port}
  verwendet
- Statuseinträge werden bei TCP nach Verbindungsabbau, bei UDP nach einer konfigurierbaren Zeitspanne gelöscht
- eingehende Pakete ohne passenden Eintrag in der Statustabelle werden verworfen

[16] das ist die im ersten Schritt eingesetzte *public* Adresse

IPv4 NAT – *Masquerading*



dynamic n:1 NAT (Cisco: PAT)

the proto:pub-ip:translated-port:remote-ip:remote-port tuple must be unique

	local		translated		remote
proto	ip	port	port	ip	port
tcp	192.168.0.12	1033	1033	209.85.135.99	80
tcp	192.168.0.10	1033	**1034**	209.85.135.99	80
tcp	192.168.0.12	1099	1099	66.196.118.48	80
tcp	192.168.0.11	1033	1033	194.41.152.136	25
tcp	192.168.0.13	6778	6778	209.85.135.99	80
udp	192.168.0.11	2987	2987	198.41.0.4	53

IPv4 NAT – *Masquerading*

- man beachte den möglichen Konflikt und die Lösung durch Ersetzen des Source-Ports für den öffentlichen Teil (Zeile 2)
- eine solche Statustabelle dient auch als *einfache* “Firewall”:
  - es werden nur Pakete entgegengenommen, die in der Tabelle einer Verbindung zugeordnet werden können
  - die Einträge in der Tabelle erfolgen nur bei ausgehenden Verbindungen
  - d.h. keine “unzuweisbaren”/unaufgeforderte Pakete können in das private Netzwerk gelangen

NAT: Beispiel GNU/Linux (edited)

```
root@zaphod:~# cat /proc/net/ip_conntrack
udp      17 136   src=188.40.65.199 dst=109.75.190.27 sport=123 dport=123
                  src=109.75.190.27 dst=188.40.65.199 sport=123 dport=123 [ASSURED]
tcp      6 431999 ESTABLISHED
                  src=77.56.89.75 dst=188.40.65.199 sport=1104 dport=22
                  src=188.40.65.199 dst=77.56.89.75 sport=22 dport=1104 [ASSURED]
tcp      6 344782 ESTABLISHED
                  src=77.56.89.75 dst=188.40.65.199 sport=8994 dport=22
                  src=188.40.65.199 dst=77.56.89.75 sport=22 dport=8994 [ASSURED]
udp      17 16    src=188.40.65.199 dst=213.165.64.1 sport=6449 dport=53
                  src=213.165.64.1 dst=188.40.65.199 sport=53 dport=6449
icmp     1 20     src=188.40.65.199 dst=207.46.19.190 type=8 code=0 id=32527 [UNREPLIED]
                  src=207.46.19.190 dst=188.40.65.199 type=0 code=0 id=32527
tcp      6 431979 ESTABLISHED
                  src=212.60.51.243 dst=188.40.65.199 sport=54054 dport=80
                  src=188.40.65.199 dst=212.60.51.243 sport=22 dport=54054 [ASSURED]
tcp      6 261956 ESTABLISHED
                  src=77.56.89.75 dst=188.40.65.199 sport=7120 dport=22
                  src=188.40.65.199 dst=77.56.89.75 sport=22 dport=7120 [ASSURED]
udp      17 14    src=188.40.65.199 dst=62.219.186.11 sport=27015 dport=53
                  src=62.219.186.11 dst=188.40.65.199 sport=53 dport=27015
tcp      6 13501 ESTABLISHED
                  src=77.56.89.75 dst=188.40.65.199 sport=40459 dport=22
                  src=188.40.65.199 dst=77.56.89.75 sport=22 dport=40459 [ASSURED]
udp      17 7     src=188.40.65.199 dst=188.40.65.199 sport=53745 dport=53
                  src=188.40.65.199 dst=188.40.65.199 sport=53 dport=53745
tcp      6 8 TIME_WAIT
                  src=41.117.24.238 dst=188.40.65.199 sport=63745 dport=25
                  src=188.40.65.199 dst=41.117.24.238 sport=25 dport=63745 [ASSURED]
tcp      6 425142 ESTABLISHED
                  src=71.103.253.162 dst=188.40.65.199 sport=4017 dport=25 [UNREPLIED]
                  src=188.40.65.199 dst=71.103.253.162 sport=25 dport=4017
```

NAT: neue Probleme...

- Protokolle, die IP-Adressinformationen als *payload*[17] transportieren müssen speziell behandelt werden
  → d.h. zusätzlich zu den Adressen im IP-Header müssen auch die Adressen *im* Paket umgeschrieben werden
  Beispiele: FTP, SIP[18]
- manche Protokolle haben keine Portnummern, die umgeschrieben oder in der Statustabelle eingetragen werden können: ICMP
  → durch die Paarung `protocol=ICMP` im Tuple wird meistens trotzdem eine eindeutige Zuweisung erreicht

[17] in den Nutzdaten

[18] Session-Initiation-Protocol, IP-Telefonie

References

- NAT: http://en.wikipedia.org/wiki/Network_address_translation
- private address space http://www.rfc-editor.org/rfc/pdfrfc/rfc1918.txt.pdf