IPv6

oder

„Sind doch nur längere Adressen“
„Funktioniert Out of the Box“
„Ist modernes, sicheres Protokoll“

Motivation

Warum jetzt mit IPv6 beschäftigen?

- IPv6 kommt … so langsam wirklich
- Provider liefern IPv6 an Endkunden
- Betriebssysteme haben IPv6 per default aktiv (auch DSL Router!)
- **IPv6 ist bevorzugter Transport, falls verfügbar**
- „IoT“ / immer mehr vernetzte Gadgets
- IPv6 kommt erst so langsam in der Praxis an
- „Eingebaute Sicherheit?!?“
- Läßt sich nur noch unhandlich (mit Nebenwirkungen?) Abschalten

IPv6 – Sicher by design?

- Zentrale RFCs von 1995/1998 => Sicherheitsmodell der 90er!
- Ziel: Flexibilität & „Zukunftssicherheit“
- Ziel: direkte Ende zu Ende Kommunikation
- => Extension Header, Netz „dummes“ Transportmedium
- Viele „SHOULD“ / „MAY“ in RFCs => Interprätationsspielraum
- „Sicherheitsprobleme lassen sich mit Crypto lösen“

ABER:

- Steigende Verbreitung deckt Design & Implementierungsfehler auf
- Flexibilität erschwert / verhindert effektive Sicherheitsmaßnahmen
- Crypto skaliert nicht / ist operativer Alptraum
- IPsec => RFC 6434: „… SHOULD for all nodes“, nur für VPN verwendet
- SeND => geringe Verbreitung, Skaliert nicht

Hersteller/Entwickler auf aktuellem Stand?

„IPv6 Ready Logo“ (https://www.ipv6ready.org)

- Core Protocols: Revision 4.0.7, 2016
  - RFC 1981 (Sept 1996)
  - RFC 2460 (Dez 1998)
  - RFC 4443 (Maerz 2006)
  - RFC 4861 (Sept 2007)
  - RFC 4862 (Sept 2007)
- IPv6 RFCs seit RFC 4443: 313, letztes: RFC 8219, August 2017
- Sicherheitsrelevant:
  - 6105 IPv6 Router Advertisement Guard. February 2011.
  - 6980 Security Implications of IPv6 Fragmentation with IPv6 Neighbor Discovery. August 2013.
  - 7112 Implications of Oversized IPv6 Header Chains. January 2014.
  - 7113 Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard). February 2014.
  - 7123 Security Implications of IPv6 on IPv4 Networks. February 2014.

Hersteller/Entwickler auf aktuellem Stand?

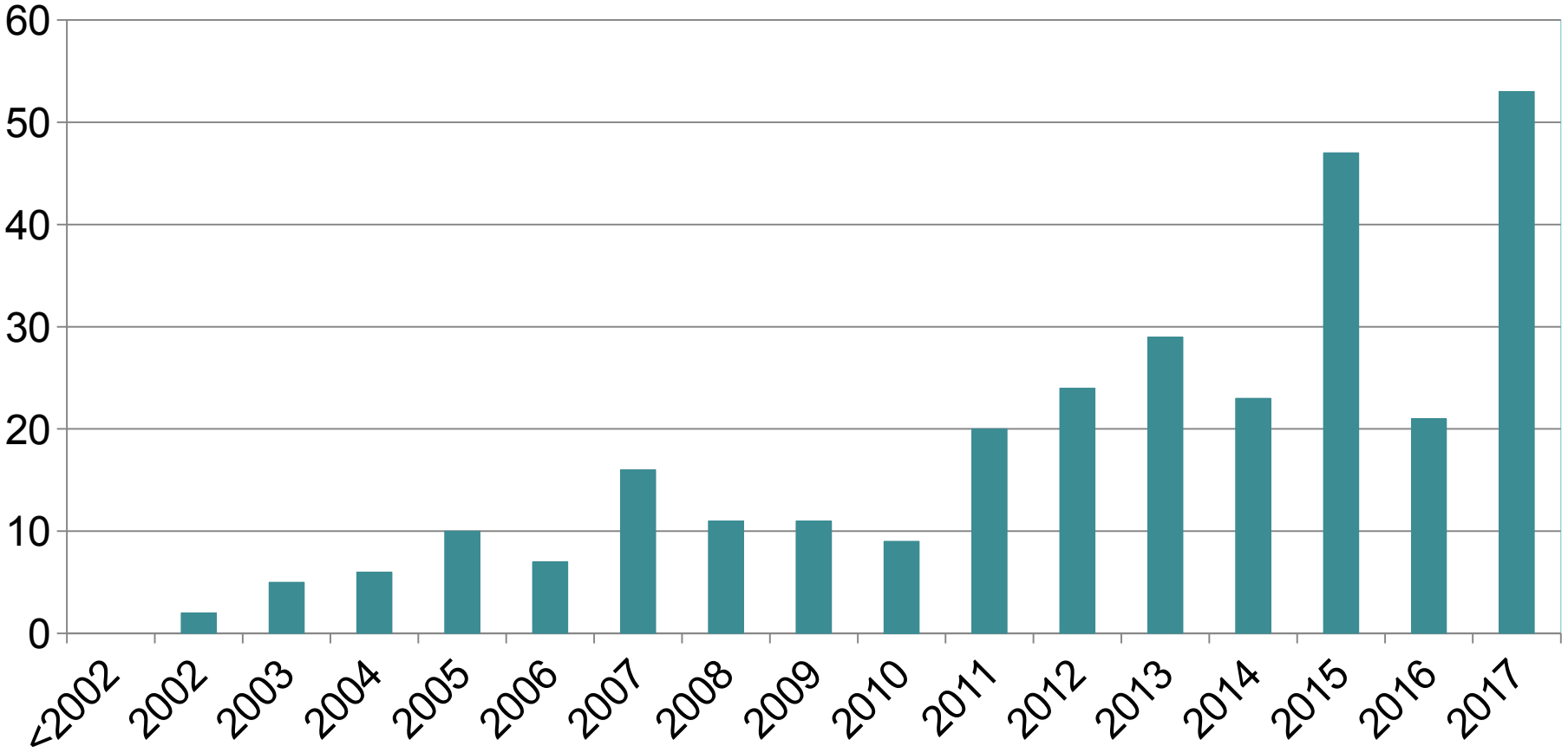
„IPv6 Ready Logo“ (https://www.ipv6ready.org)

- DHCPv6: Revision 1.1.4, Juli 2011
  - RFC 3315 (Juli 2003)
  - RFC 3646 (Dez 2003)
  - RFC 3633 (Dez 2003)
  - RFC 3736 (April 2004)

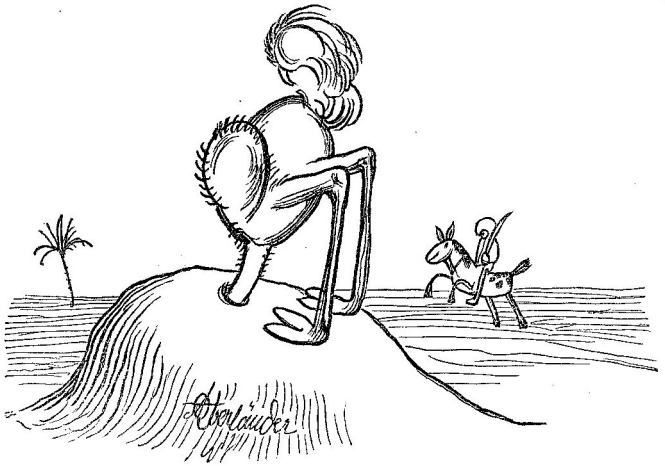
- DHCP RFCs seit RFC 3736: 48, letztes: RFC 8168, Mai 2017

- Sicherheitsrelevant:
  - 7610 DHCPv6-Shield: Protecting against Rogue DHCPv6 Servers. August 2015

Schwachstellenmeldungen mit IPv6 (CVE)



Was tun?



IPv6 im Schnelldurchgang

- Bis vor kurzem Mindset 90er
- Netzwerk wird „einfache“ Paketschubse
- Direkte Ende-zu-Ende Verbindung (kein NAT mehr)
- Ultimative Flexibilität durch Header Extensions
- Steuerung / Konfiguration durch ICMPv6
- „Sicherheit mit Crypto“

Notation

- 128 Bits, 4-bitweise Hexadezimale Notation (case insensitv)
- 4-er Gruppen (Hextet) von Hex-Ziffern durch ':' getrennt
- Beispiel: 2001:0DB8:0000:0000:0066:0000:ABC0:0001
- Kürzen führender Nullen möglich
- „Komprimieren“ **einer** Folge von 0000-Hextets durch „::“ möglich

	2001:0DB8:0000:0000:6660:0000:0000:0001
Ohne führende 0en	2001:DB8:0:0:6660:0:0:1
Komprimiertes Format	2001:DB8::6660:0:0:1
oder	
Komprimiertes Format	2001:DB8:0:0:6660::1

=> Wie Behandlung unterschiedlicher Formate in Logfiles?

Notation

- IPv6 Adressen setzen sich zusammen aus
  - Netzwerk Präfix (mit Präfix-Länge statt Netzmaske)
  - Interface-ID
- Netzwerk Präfix kann unterteilt sein in
  - Routing Prefix (globales Routing durch Provider)
  - Subnet-ID (internes Routing)
- Beispiel IPv6-Adresse
  - 2001:0DB8:0000:0000:0066:0000:ABC0:0001/64
  - 2001:0DB8::/56 Provider Präfix
  - 256 lokale Subnetze (/64) möglich (00 - FF)
- Beispiel IPv6-Netz
  - 2001:0DB8::/32

IPv6 Adress-Typen

Unicast

- Global Unicast Address
- Unique Local Address (ULA)
  - **FD00::/8**
- Link Local **FE80::/64**
- Loopback: **::1/128** bzw. **::1**
- „Unspezifiziert“: **::/128** bzw **::**

Multicast (Gruppen)

- **Devices joinen aktiv**
- **Präfix FFxx::/8**
- **FF02::1 „All-nodes“**
- **FF02::2 „All routers“**

Anycast Adressen

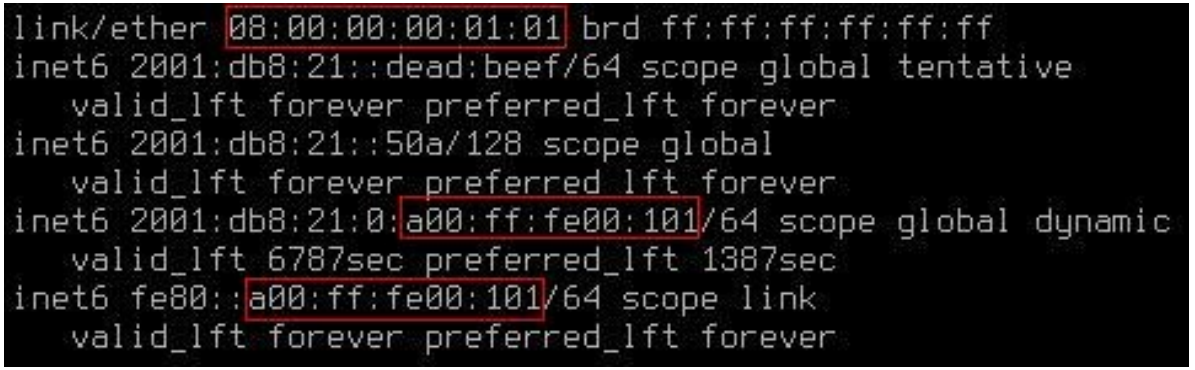
- Selbe, statische Unicast Adresse auf mehreren Interfaces

Keine Broadcast Adressen!

- Wird durch Multicast gelöst

Adresskonfiguration

- Manuell / statisch
- Dynamisch basierend auf Präfix
  - Stateless Address Autoconfiguration (SLAAC)
  - Präfix + EUI-64 (aus MAC berechnet)
  - DHCPv6
  - Privacy Extensions (temporäre, zufällige Adressen)
- Immer auch Link-Local Adresse



- Mehrere IPv6-Adressen / Präfixe auf Interface normal!

=> Welche wird genommen? Welche von Firewall geschützt?

DHCPv6

- DHCP Unique Identifier (DUID) (RFC 3315, 6355)
  - Clients and servers MUST treat DUIDs as opaque values and MUST only compare DUIDs for equality. Clients and servers MUST NOT in any other way interpret DUIDs. Clients and servers MUST NOT restrict DUIDs to the types defined in this document, as additional DUID types may be defined in the future.
- Bis zu 128 Byte lang
- Generierung OS-Abhängig
  - Empfohlen: Einmalig bei erstem Systemstart auswürfeln.
  - Unterschiedliche UUIDs bei Multiboot!
  - Änderung bei Neuinstallation!
  - Wie in größeren Umgebungen Client vorab eintragen? (Pre Deploy)
- Probleme bei Prefix Delegation (Präfix-Änderung bei Routerwechsel)

ICMPv6

- Im Gegensatz zu ICMP unter IPv4 **verpflichtend**!
- Steuert IPv6 Kommunikation & Konfiguration
  - Destination Unreachable (Type 1)
  - Packet Too Big (Type 2)
  - Time Exceeded (Type 3, Code 0)
  - Prarameter Problem (Type 4, Codes 1 & 2)
  - Router Solicitation (Type 133)
  - Router Advertisement (Type 134)
  - Neighbor Solicitation (Type 135)
  - Neighbor Advertisement (Type 136)
  - ….

=> Firewalls müssen bestimmte ICMPv6 Typen weiterleiten

=> Firewalls müssen auf bestimmte ICMPv6 Typen antworten

Neighbor Discovery Protocol (NDP)

- Ersetzt & Erweitert ARP-Protokoll
- **Nur auf Local Link**
- Router Solicitation (RS)
  - Client fragt am Local Link nach Routern
- Router Advertisement (RA)
  - Router senden auf Anfrage / periodisch Konfigurationsinformationen (Präfixe, Routen, ….)
- Neighbor Solicitation (NS)
  - Anfrage nach Erreichbarkeit und MAC-Adresse
- Neighbor Advertisement (NA)
  - Antwort auf NS oder unnachgefragtes Update
- Redirect
  - Verweis auf „besseren“ Router

Ablauf Stateless Address Autoconfiguration

- Link-Local Address (EUI-64 ID) generieren
- Neighbor Solicitation (NS) für Duplicate Address Detection (DAD) senden
- Autoconfiguration abbrechen, falls ein Neighbor Advertisement (NA) einen Adresskonflikt anzeigt
- Router Solicitation (RS) aussenden
- Falls kein Router Advertisement (RA) empfangen wird, starte DHCPv6
- Falls ein Router Advertisement empfangen wird
  - Generiere Adressen für jeden enthaltenen Präfix; danach DAD
- M Flag == 1 im Router Advertisement:
  - Starte DHCPv6, um weitere Adressen und Parameter zu erhalten
- M Flag == 0 und O Flag == 1 im Router Advertisment:
  - Starte DHCPv6, um weitere Konfigurationsparameter (z.B. DNS) zu erhalten

Was kann dabei schon schief gehen...

- Erst mal keinerlei Absicherung der NDP Meldungen
- Mit gefälschten Neighbor Advertisments
  - bei DAD immer „hier“ schreien => **DOS**
  - ungefragt / als Reaktion auf NS des Opfers antworten => **MiTM**, **DOS**
- Mit gefälschten Router Advertisments
  - Fake Präfixe senden => Last bei Opfer erzeugen
  - Neue (priorisierte) Default Route => MiTM, DOS
  - DHCP / DNS Werte überschreiben => MiTM, Redirect, DOS
- Abwehr prinzipbedingt (Standard) schwierig
  - Höherwertige Switche mit „First Hop Security“ notwendig
  - Krypto: Secure Neighbor Discovery (SeND), aber skaliert nicht
- Tools zum testen:
  - Ip6 Tools
  - Thc-ipv6 attack toolkit

Spass mit thc-ipv6 Tools

- Vorher

```
# cat /etc/resolv.conf
nameserver 2001:db8:21::1
search lan
```

```
# ip -6 addr show eth1
    inet6 2001:db8:21::50a/128 scope global
       valid_lft forever preferred_lft forever
    inet6 2001:db8:21:0:a00:ff:fe00:101/64 scope global dynamic
       valid_lft 6733sec preferred_lft 1333sec
    inet6 fe80::a00:ff:fe00:101/64 scope link
       valid_lft forever preferred_lft forever
```

```
# ip -6 route show
2001:db8:21::50a dev eth1  proto kernel  metric 256
2001:db8:21::/64 dev eth1  proto kernel  metric 256  expires 7146sec
fe80::/64 dev eth1  proto kernel  metric 256
default via fe80::a00:ff:fe00:b01 dev eth1  proto static  metric 1
default via fe80::a00:ff:fe00:b01 dev eth1  proto ra  metric 1024  expires 1746sec hoplimit 64
```

angreifer# fake_router26 -a 10 -A 2001:db0:1234::1/64 -D 1234::1 eth1

- Nachher

```
# cat /etc/resolv.conf
nameserver 1234::1
nameserver 2001:db8:21::1
search lan
```

```
# ip -6 addr show eth1
    inet6 2001:db0:1234:0:a00:ff:fe00:101/64 scope global dynamic
       valid_lft 6sec preferred_lft 1sec
    inet6 2001:db8:21::50a/128 scope global
       valid_lft forever preferred_lft forever
    inet6 2001:db8:21:0:a00:ff:fe00:101/64 scope global dynamic
       valid_lft 7158sec preferred_lft 1758sec
    inet6 fe80::a00:ff:fe00:101/64 scope link
       valid_lft forever preferred_lft forever
```

```
# ip -6 route show
2001:db0:1234::/64 dev eth1  proto kernel  metric 256  expires 6sec
2001:db8:21::50a dev eth1  proto kernel  metric 256
2001:db8:21::/64 dev eth1  proto kernel  metric 256  expires 7122sec
fe80::/64 dev eth1  proto kernel  metric 256
default via fe80::a00:ff:fe00:b01 dev eth1  proto static  metric 1
default via fe80::a00:ff:fe00:b01 dev eth1  proto ra  metric 1024  expires 1722sec hoplimit 64
default via fe80::a00:ff:fe00:201 dev eth1  proto ra  metric 1024  expires 2044sec hoplimit 255
```

- Thc-ipv6 enthält 80 weitere Tools ;)

Ich mache nur IPv4 und bin sicher...

- Sicher dass nur IPv4 da ist?
- IPv6 auf modernen Systemen per Default „dabei“
- Sobald Router Advertisements im lokalen Netz auftauchen aktiviert sich ein vorhandener IPv6 Stack!
  - Fehlkonfigurierte / bösartige Endgeräte im LAN reichen
- Eigener IPv6 Uplink nicht nötig, IPv6-Tunnel reicht!

=> viele IPv6 Tunneltechniken (Teredo, 6rd, 6in4, 6to4, …)

=> Wie siehts mit sonstigen VPN-Tunneln aus?

Apropos VPN...

- Szenario Dual-Stack (IPv4 + IPv6 parallel aktiv)
- Aktiver VPN-Tunnel (nur IPv4 oder nur IPv6), Ziel sowohl per IPv4 als auch IPv6 erreichbar

  => geht der Datenverkehr zum Ziel durch Tunnel oder vorbei?

  => Welcher DNS wird benutzt?
- Wie sieht es aus, wenn ein Ziel nur per IPv4 oder IPv6 erreichbar ist?
- **Minimale IPv6 MTU: 1280 Byte!**

Firewalling

IPv4

- **INPUT**
  - Von LAN erlauben
  - Von WAN nur Established
- **OUTPUT**
  - Alles erlauben
- **FORWARD**
  - LAN → WAN alles erlauben
  - WAN → LAN nur Established
- **NAT**
  - MASQUERADE

IPv6

- RFC 4890
- ICMPv6 **MUSS** behandelt werden
- Kein NAT mehr
- Router/Firewall muss von aussen per ICMPv6 ansprechbar sein
- Endgeräte müssen per ICMPv6 erreichbar sein
- Script mit 331 Zeilen um ICMPv6 korrekt zu behandeln
- Ein-/Ausgehende Regeln notwendig, da Endgeräte nun öffentliche IP haben!
- Problematik Privacy Extensions / SLAAC / **Präfix-Wechsel**

Zusätzlich Kombination IPv4 + IPv6 Routing / Firewalling zu beachten!

Firewalling - ICMPv6

Transit Traffic

4.3.1 MUST NOT be dropped

- Destination Unreachable (Type 1) - All codes
- Packet Too Big (Type 2)
- Time Exceeded (Type 3) - Code 0 only
- Parameter Problem (Type 4) - Codes 1 and 2 only

- Echo Request (Type 128)
- Echo Response (Type 129)

4.3.2. Traffic That Normally Should Not Be Dropped

- Time Exceeded (Type 3) - Code 1
- Parameter Problem (Type 4) - Code 0

Firewalling - ICMPv6

Lokale Erreichbarkeit

4.4.1. Traffic That Must Not Be Dropped

- Destination Unreachable (Type 1) - All codes
- Packet Too Big (Type 2)
- Time Exceeded (Type 3) - Code 0 only
- Parameter Problem (Type 4) - Codes 1 and 2 only

- Echo Request (Type 128)
- Echo Response (Type 129)

- Router Solicitation (Type 133)
- Router Advertisement (Type 134)
- Neighbor Solicitation (Type 135)
- Neighbor Advertisement (Type 136)
- Inverse Neighbor Discovery Solicitation (Type 141)
- Inverse Neighbor Discovery Advertisement (Type 142)
- Listener Query (Type 130)
- Listener Report (Type 131)
- Listener Done (Type 132)
- Listener Report v2 (Type 143)

4.4.2. Traffic That Normally Should Not Be Dropped

- Time Exceeded (Type 3) - Code 1
- Parameter Problem (Type 4) - Code 0

NAT

- NAT sollte entfallen („direkte Ende-Ende Kommunikation“)

  => Entfall der „trivial Netzfirewall“

  => korrekte Firewall an Internet und Host nun Pflicht!

- keine festen internen IPs „dank“ Präfixwechseln

  => intern parallel private IPv6 Adressen notwendig

  => Problematik Firewallfreigaben bei Präfixwechseln

- NAT-kaputte Protokolle sind auch bei Firewall-Einsatz kaputt
- Im Nachgang NAT66 definiert (lokale Inseln an Internet)

Privacy / Privacy Extensions?

Abgesehen von SLAAC per MAC...

- sorgt für viele Probleme beim Netzbetrieb
  - Firewalling
  - Logging
  - Troubleshooting
  - Je nach OS nicht deaktivierbar!
- Hilft nicht gegen das effizientere Tracken auf Applikationsebene (insbesondere Browser)
- Hilft nicht im „lokalen“ Netz (LTE, Firma, ...)
- Abhilfe gegen MAC-basiertes Netzübergreifende Tracking:
  - Adressvergabe per DHCPv6
  - RFC 7217 “Semantically Opaque Interface Identifiers”

Fazit

- Grundsätzlich funktioniert es, aber man muss die Nebenwirkungen kennen
- Es wird kommen!
- Endgeräte haben unterschiedliche Implementierung / Settings
- IoT?
- Basisschutz
  - Segmentierung im LAN (/64 vom Provider / Hoster reicht nicht)
  - Firewalling **im LAN** und **ausgehend**
  - IPv6 deaktivieren wo unnötig / nicht handlebar
  - Ab und an mal das lokale LAN auf komische Pakete untersuchen
  - Firmware / OS Updates
  - Altgeräte raus
  - Bei VPN-Tunneln testen ob Verkehr korrekt fliesst

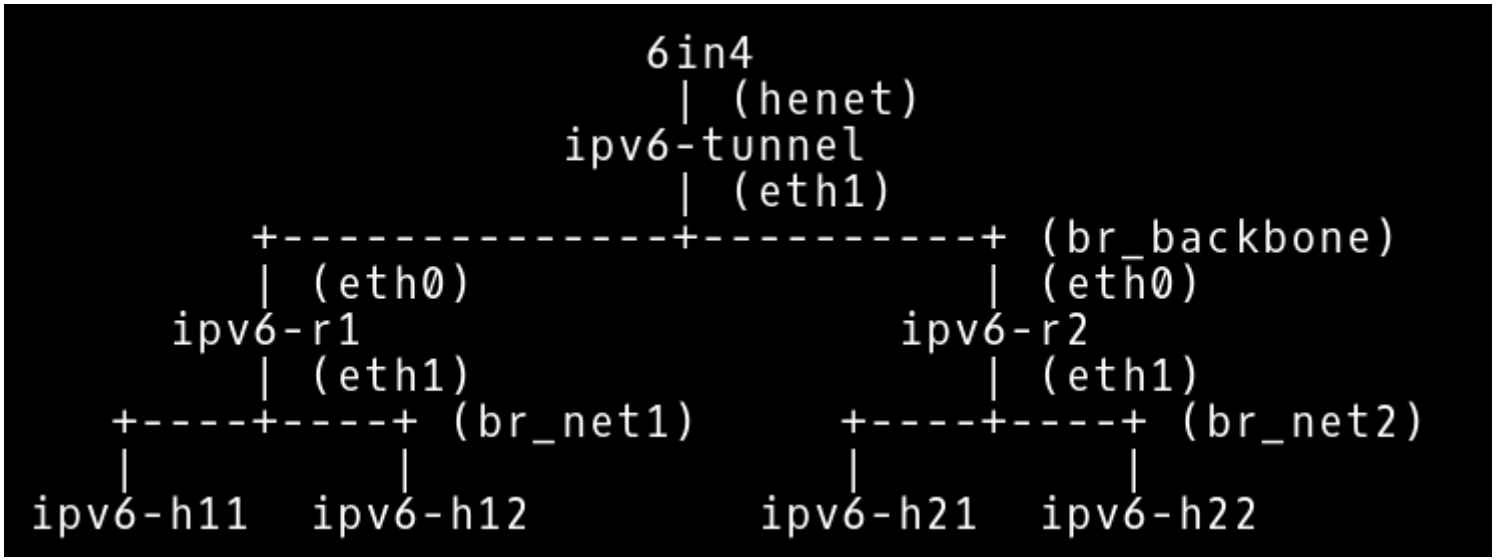
=> Angriffe werden steigen!

=> Jetzt damit beschäftigen!

IPv6 testen

Testen ohne sich ins Knie zu schiessen

- Nicht am produktiven DSL-Router aktivieren!
- Abgetrennte Testinfrastruktur (auch vom Host!)
- IPv6 Aussenanbindung über 6in4 Tunnel: tunnelbroker.net
  - Notwendig: Erreichbarkeit auf IPv4 per Ping & IP Protokoll 41
  - Virtualbox scheidet aus
- Erste Schritte mit KVM + LEDE als Router



Relevante RFCs & Best Practices

IPv6 Neighbor Discovery (ND) Trust Models and Threats (2004!)

- https://tools.ietf.org/html/rfc3756

Security Implications of IPv6 Fragmentation with IPv6 Neighbor Discovery (2013)

- https://tools.ietf.org/html/rfc6980

Implications of Oversized IPv6 Header Chains (2014)

- https://tools.ietf.org/html/rfc7112

Security Implications of IPv6 on IPv4 Networks (2014)

- https://tools.ietf.org/html/rfc7123

Layer 3 Virtual Private Network (VPN) Tunnel Traffic Leakages in Dual-Stack Hosts/Networks (2014)

- https://tools.ietf.org/html/rfc7359

Recommendations for Filtering ICMPv6 Messages in Firewalls (2007)

- https://tools.ietf.org/html/rfc4890

Operational Security Considerations for IPv6 Networks (21.3.2016)

- https://tools.ietf.org/wg/opsec/draft-ietf-opsec-v6/

Network Reconnaissance in IPv6 Networks (09.03.2016)

- https://tools.ietf.org/html/rfc7707

Deep Dive

Tools für Pentesting / Verifizierung der Implementierung

- http://www.si6networks.com/tools/ipv6toolkit/
- https://www.thc.org/thc-ipv6/
- http://www.secdev.org/projects/scapy/

***Die* Experten zum Thema**

- **si6networks / Fernando Gont**
  - https://www.si6networks.com/publications/index.html
  - https://www.si6networks.com/presentations/index.html
- **Marc Heuse**
  - http://www.mh-sec.de/downloads/mh-ipv6_vulnerabilities.pdf