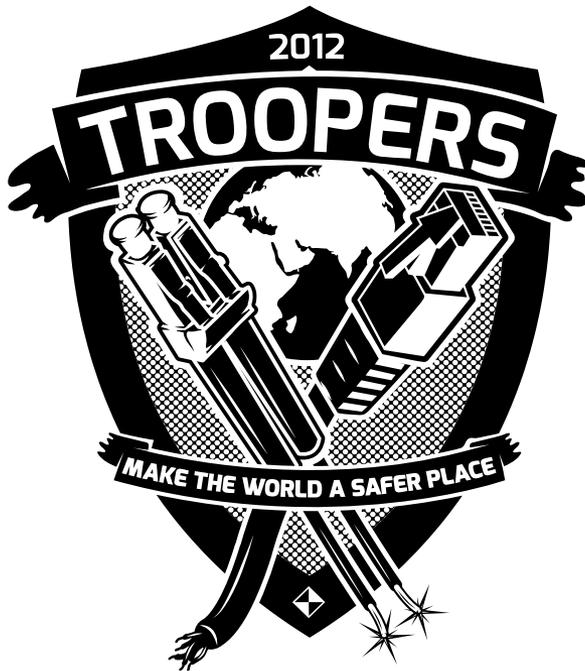


# Privacy Extensions

Enno Rey & Christopher Werny  
{erey, cwerny}@ernw.de



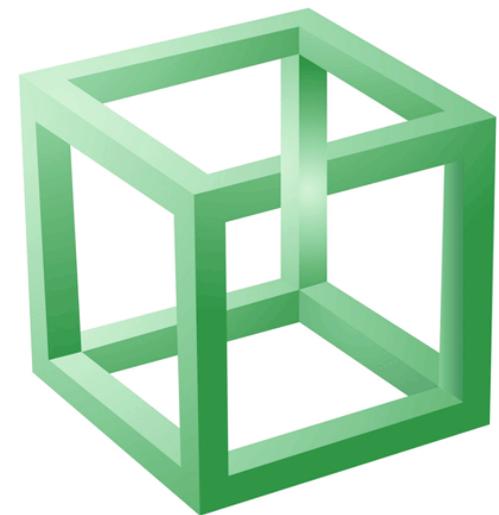


- **Old-school network geeks, working as security researchers for:**
- **Germany based ERNW GmbH**
  - Independent
  - Deep technical knowledge
  - Structured (assessment) approach
  - Business reasonable recommendations
  - We understand corporate
- **Blog: [www.insinuator.net](http://www.insinuator.net)**
- **Conference: [www.troopers.de](http://www.troopers.de)**

- **Intro & Overview**
- **Looking at Privacy Extensions from Different Angles**
- **Operational Aspects**



- **Potentially static instead of dynamic prefixes used in residential subscriber space**
  - Can be addressed by “Privacy Button”.
- **Obfuscating function of NAT no longer available.**
- **Interface identifier generated by [Modified] EUI-64 method.**
  - This is where privacy extensions come into play.



- **Bill Frezza: “Where’s All The Outrage About The IPv6 Privacy Threat?” (1999)**
  - <http://lists.essential.org/random-bits/msg00163.html>
- **RFC 3041 Privacy Extensions for Stateless Address Autoconfiguration in IPv6 (2001)**
- **RFC 4941 Privacy Extensions for Stateless Address Autoconfiguration in IPv6 (2007)**
- **Still ongoing debate on potential enhancements**
  - <https://www.usenix.org/system/files/login/articles/105438-Barrera.pdf>
  - <http://tools.ietf.org/id/draft-gont-6man-stable-privacy-addresses-00.txt>



## ■ RFC 4941

- “This document discusses concerns associated with the embedding of non-changing interface identifiers within IPv6 addresses and describes extensions to stateless address autoconfiguration that can help mitigate those concerns for individual users and in environments where such concerns are significant.”



- **Generate interface identifier in some random way.**

- **Re-generate it periodically.**

TEMP\_VALID\_LIFETIME - 7d

TEMP\_PREFERRED\_LIFETIME - 1d

REGEN\_ADVANCE - 5s

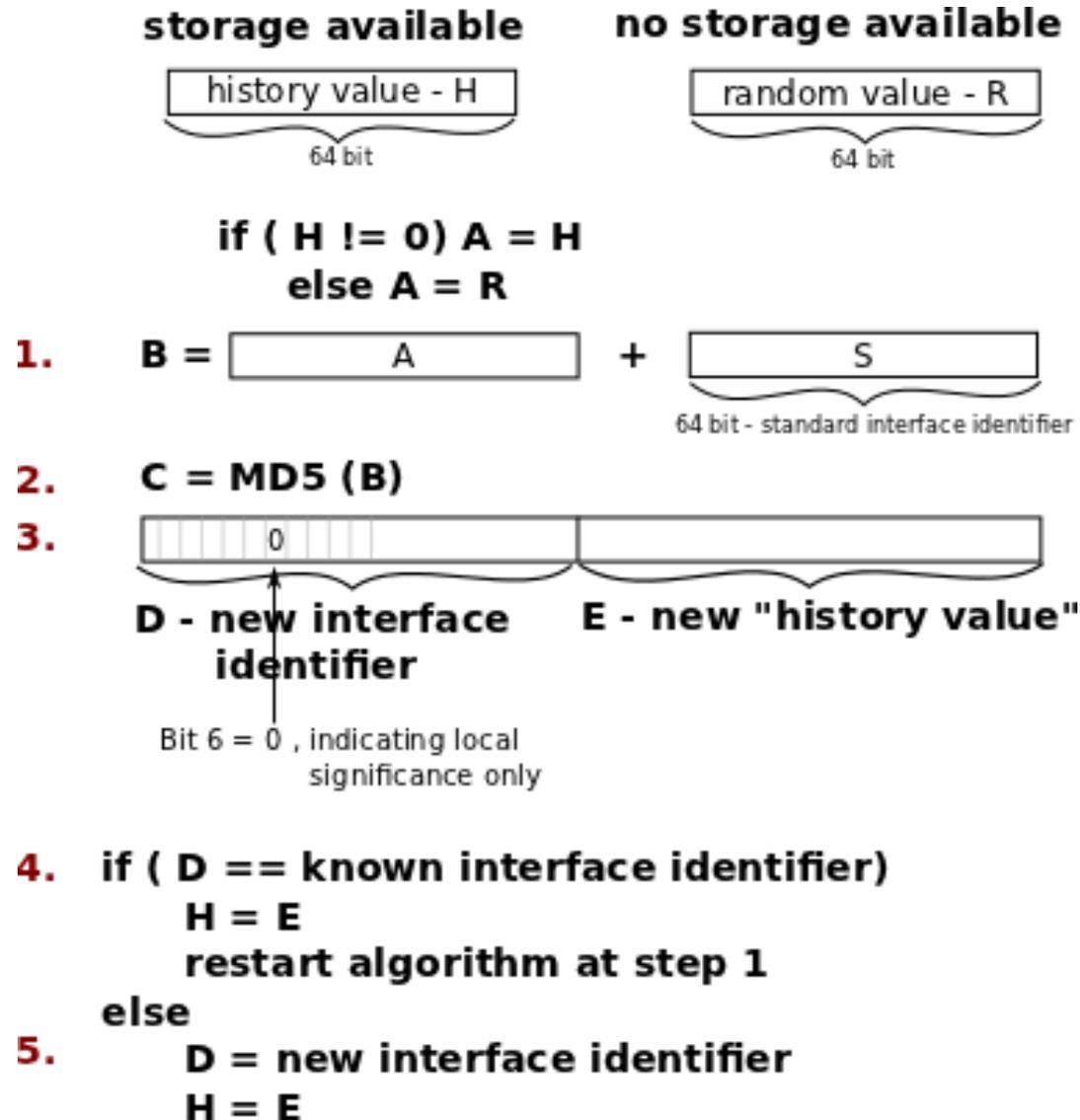
MAX\_DESYNC\_FACTOR - 10m

DESYNC\_FACTOR - random value within range 0- MAX\_DESYNC\_FACTOR

TEMP\_IDGEN\_RETRIES - 3



# RFC 4941, Proposed Algorithm



## ■ (RFC 4941)

- Devices implementing this specification **MUST** provide a way for the **end user** to explicitly enable or disable the use of temporary addresses.
- In addition, a site might wish to disable the use of temporary addresses in order to simplify network debugging and operations. Consequently, implementations **SHOULD** provide a way for **trusted system administrators** to enable or disable the use of temporary addresses.
- Sites might wish to selectively enable or disable the use of temporary addresses for some prefixes.



## ■ What about RFC 3484?

- Rule 7: Prefer public addresses. If SA is a public address and SB is a temporary address, then prefer SA. Similarly, if SB is a public address and SA is a temporary address, then prefer SB.
- Some (work group level) discussion ongoing if RFC 3484 to be enhanced/changed due to larger deployment of priv\_extensions.

- **Not every randomly generated IP address is a “privacy address“.**
- **The idea is that a “privacy address“ periodically changes.**
  - Which applies to “temporary addresses“.
  - → Usually those actually constitute “privacy addresses“.
- In contrast to other (seemingly) randomly generated addresses (e.g. those from DHCPv6).

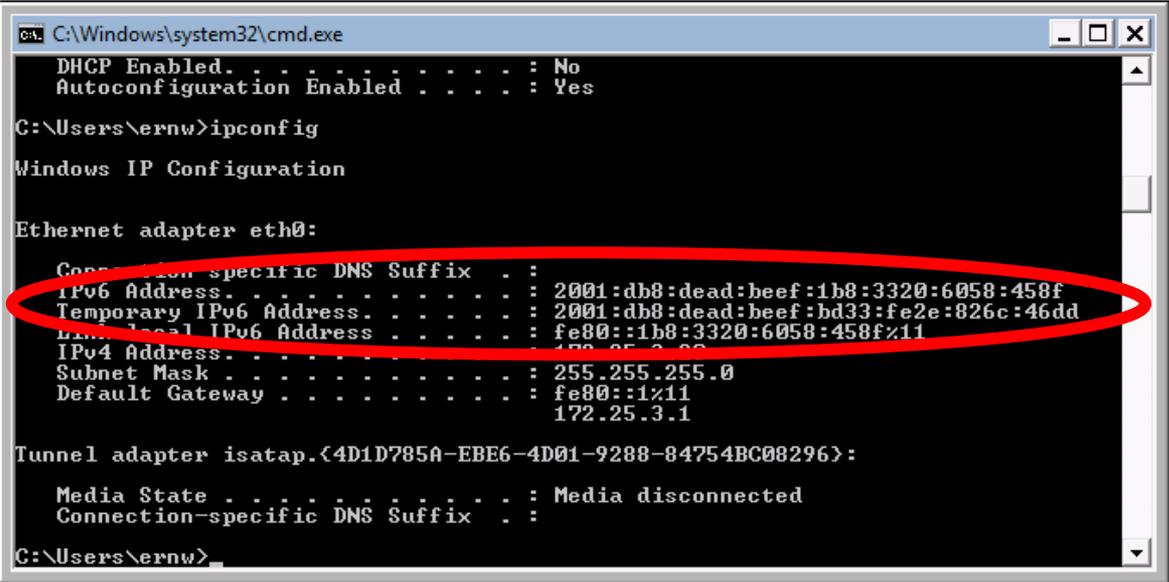




## Let's Have a Quick Look at some Implementations

---

- By default, Windows 7 uses a randomization approach to generate the host part / interface identifier.
- In addition, a second IPv6 address (a *temporary address*) will be configured on the interface.
  - The temporary address will be used for outgoing connections.



```
cmd C:\Windows\system32\cmd.exe
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes

C:\Users\ernw>ipconfig

Windows IP Configuration

Ethernet adapter eth0:

    Connection-specific DNS Suffix . : 
    IPv6 Address. . . . . : 2001:db8:dead:beef:1b8:3320:6058:458f
    Temporary IPv6 Address. . . . . : 2001:db8:dead:beef:bd33:fe2e:826c:46dd
    Link-local IPv6 Address . . . . . : fe80::1b8:3320:6058:458f%11
    IPv4 Address. . . . . : 172.25.3.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : fe80::1%11
                               172.25.3.1

Tunnel adapter isatap.{4D1D785A-EBE6-4D01-9288-84754BC08296}:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : 

C:\Users\ernw>
```

- **The default valid/preferred lifetime in the RAs are set to 30/7 days.**
  - Used in the network we refer to here.
- **The lifetime of the “main“ IPv6 address is reset with every RA received.**
  - Which kind-of makes the IPv6 address static.
- **The lifetime of the temporary address is not reset.**
  - With one exception: During our tests we noticed that as soon as the preferred lifetime is less/smaller than seven days, the lifetime of the temp. address is also reset. These 7d probably related to RAs' lifetime (see above).
  - This means that both addresses are kind-of static.
    - Which might not be the expected/desired behavior.



```
C:\Windows\system32\cmd.exe

C:\Users\ernw>netsh interface ipv6 show addresses

Interface 1: Loopback Pseudo-Interface 1
Addr Type  DAD State  Valid Life  Pref. Life  Address
-----
Other      Preferred  infinite   infinite   ::1

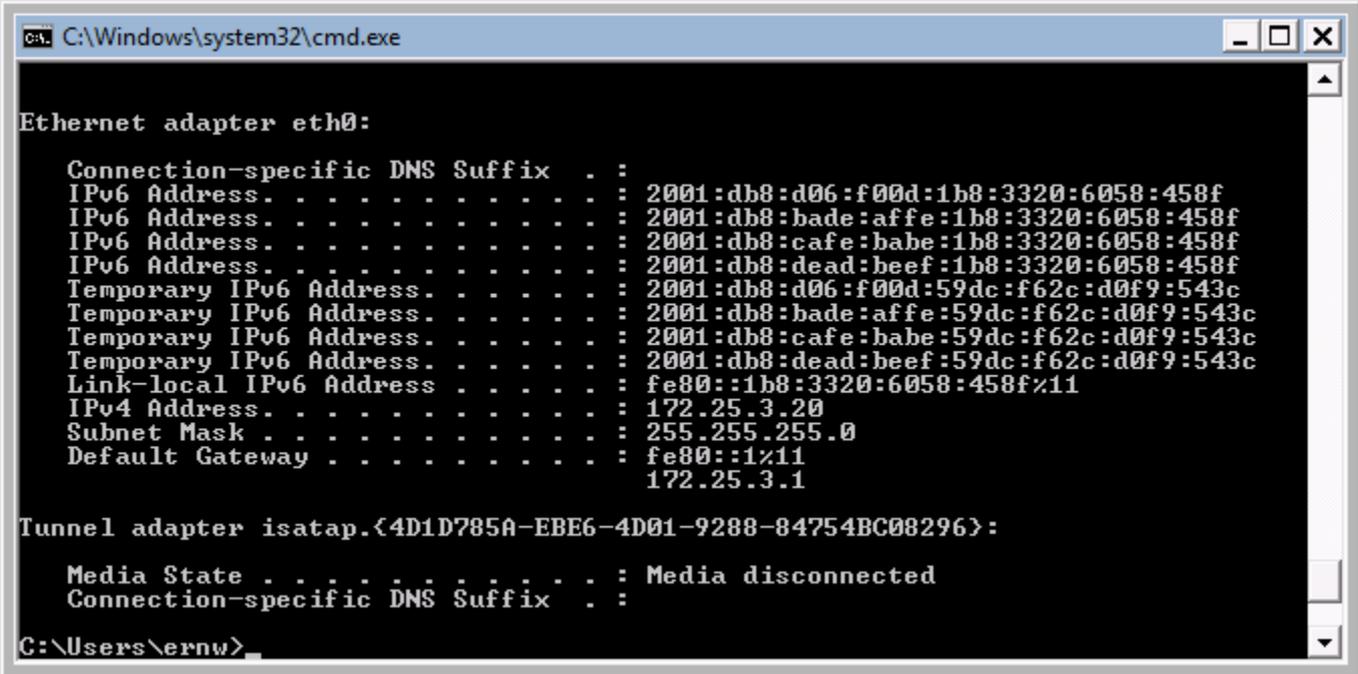
Interface 11: eth0
Addr Type  DAD State  Valid Life  Pref. Life  Address
-----
Public     Preferred  1d23h59m3s 1d23h59m3s 2001:db8:dead:beef:1b8:3320:6058:458f
Temporary Preferred  1d23h59m3s 1d23h59m3s 2001:db8:dead:beef:9573:83ac:191f:1aed
Other      Preferred  infinite   infinite   fe80::1b8:3320:6058:458f%11

C:\Users\ernw>netsh interface ipv6 show addresses

Interface 1: Loopback Pseudo-Interface 1
Addr Type  DAD State  Valid Life  Pref. Life  Address
-----
Other      Preferred  infinite   infinite   ::1

Interface 11: eth0
Addr Type  DAD State  Valid Life  Pref. Life  Address
-----
Public     Preferred  1d23h59m58s 1d23h59m58s 2001:db8:dead:beef:1b8:3320:6058:458f
Temporary Preferred  1d23h59m58s 1d23h59m58s 2001:db8:dead:beef:9573:83ac:191f:1aed
Other      Preferred  infinite   infinite   fe80::1b8:3320:6058:458f%11
```

- Furthermore the host part / interface identifier of the address does not change at all.
- Even when completely different prefixes comes into play.
  - One might ask: what's the (privacy) benefit then?



```
C:\Windows\system32\cmd.exe

Ethernet adapter eth0:

    Connection-specific DNS Suffix  . : 
    IPv6 Address . . . . .           : 2001:db8:d06:f00d:1b8:3320:6058:458f
    IPv6 Address . . . . .           : 2001:db8:bade:affe:1b8:3320:6058:458f
    IPv6 Address . . . . .           : 2001:db8:cafe:babe:1b8:3320:6058:458f
    IPv6 Address . . . . .           : 2001:db8:dead:beef:1b8:3320:6058:458f
    Temporary IPv6 Address . . . . . : 2001:db8:d06:f00d:59dc:f62c:d0f9:543c
    Temporary IPv6 Address . . . . . : 2001:db8:bade:affe:59dc:f62c:d0f9:543c
    Temporary IPv6 Address . . . . . : 2001:db8:cafe:babe:59dc:f62c:d0f9:543c
    Temporary IPv6 Address . . . . . : 2001:db8:dead:beef:59dc:f62c:d0f9:543c
    Link-local IPv6 Address . . . . . : fe80::1b8:3320:6058:458f%11
    IPv4 Address . . . . .            : 172.25.3.20
    Subnet Mask . . . . .            : 255.255.255.0
    Default Gateway . . . . .        : fe80::1%11
                                      172.25.3.1

Tunnel adapter isatap.<4D1D785A-EBE6-4D01-9288-84754BC08296>:

    Media State . . . . .            : Media disconnected
    Connection-specific DNS Suffix  . : 

C:\Users\ernw>
```

- **The Windows default behavior can be changed if desired.**
- **To disable the generation of a temporary address:**
  - `netsh interface ipv6 set privacy state=disabled`
- **To disable randomized address generation at all:**
  - `netsh interface ipv6 set global randomizeidentifiers=disabled`



- **The generated host parts / interface identifiers for both addresses might not change at all.**
  - One would think they do on a regular basis.
  - We mean that's why privacy extensions were introduced, but that's not how it (seemingly) works in Windows.
  - (As usual in Windows space ;- ) rebooting helps... [at least for the temp. address]
- **Reverse DNS Updates for IPv6 work reasonably well in Windows 7/Vista/2008.**
  - Will be covered later in more detail.



- **For most major Linux distributions PE enabled but not turned on by default.**
  - Ubuntu nowadays being a notable exception.

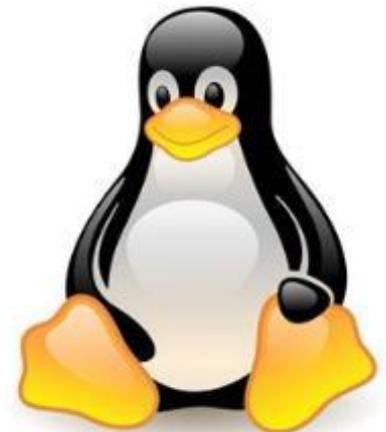
- **Usually modifications in `/etc/sysctl.conf` needed**

```
net.ipv6.conf.all.use_tempaddr = 2  
net.ipv6.conf.default.use_tempaddr = 2
```

- **Restart of network (interfaces) necessary afterwards.**
- **Second (temp.) IPv6 address shows up in `ip addr show`**
- **Configuration of lifetime(s) possible via**

```
sudo sysctl net.ipv6.conf.XX.temp_valid_lft=x  
sudo sysctl net.ipv6.conf.XX.temp_prefered_lft=x
```

**where `XX` stands for the interface.**



- **Before 10.7 not turned on by default.**
- **Currently no GUI way foreseen to tweak PE behavior.**
- **Can be enabled (with elevated privileges) via**  
`sudo sysctl -w net.inet6.ip6.use_tempaddr=1`
- **Longer lasting changes: same procedure as for Linux, except `/etc/sysctl.conf` potentially not existing & needs to be created then.**



# Priv Extensions on Smartphones

---



- **Supporting IPv6 since version 2.1.**
- **For all versions < 4.0 PE not enabled by default; activation needs root privs.**
- **Set sysctl options as super user, but will be restart:**

```
su  
sysctl -w net.ipv6.conf.default.use_tempaddr=2  
sysctl -w net.ipv6.conf.all.use_tempaddr=2
```



ANDROID

- **Permanent solution: edit /data/local/userinit.sh and insert the two commands, will be executed on start up.**
- **No permanent solution without root access foreseen yet by suppliers.**

- **Turned on by default since version 4.3.**
- **For earlier versions can only be enabled via jailbreak.**

```
sudo sysctl -w net.inet6.ip6.use_tempaddr=1
```

- **Permanent fix: adding**

```
net.inet6.ip6.use_tempaddr=1
```

**to /etc/sysctl.conf and restarting network.**



- We performed some testing with an iPhone 4 / iOS v5.1.
- It turns out that there are no GUI elements which provide displaying anything IPv6 related (see above).
- So what's the first approach that comes to your mind?
  - Right, there must be an app which can do that ;-)



- **Fortunately, there is a free app called “ip6config”.**
  - Disclaimer: We are not affiliated with the developer(s).
- **This little app gives at least some info about the IPv6 config of the device.**
- **So we connected the iPhone to an access point to see what happens.**



- **iOS configures two addresses, like Windows.**
- **The first (global) is derived from the MAC address of the device.**
- **The second one is generated with/by means of priv extensions.**



- **iOS generates different host parts when the lifetime expires.**
- **Which differs from the Windows behavior shown before.**
- **When multi. prefixes are received via RAs, iOS generates a dedicated host part for every prefix.**
- **(see picture on the next slide)**



# iOS – Multiple Prefixes



# Android (Ice Cream Sandwich)

No.	Time	Source	Destination	Protocol	Length	Info
7	5.930317	::	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
10	6.740461	::	ff02::1:ffe6:824d	ICMPv6	78	Neighbor solicitation for fe80::a20b:baff:fee6:824d
15	7.832069	fe80::a20b:baff:fee6:824d	ff02::2	ICMPv6	70	Router Solicitation from a0:0b:ba:e6:82:4d
16	7.833135	fe80::1	ff02::1	ICMPv6	214	Router Advertisement from f0:f7:55:2f:8b:c0
17	7.852213	fe80::a20b:baff:fee6:824d	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
18	8.184726	::	ff02::1:ff12:cdf0	ICMPv6	78	Neighbor solicitation for 2001:db8:d06:f00d:2865:550a:c512:cdf0
19	8.406050	::	ff02::1:ffe6:824d	ICMPv6	78	Neighbor solicitation for 2001:db8:bade:affe:a20b:baff:fee6:824d
20	8.406351	::	ff02::1:ffe6:824d	ICMPv6	78	Neighbor solicitation for 2001:db8:cafe:babe:a20b:baff:fee6:824d
21	8.628222	::	ff02::1:ff12:cdf0	ICMPv6	78	Neighbor solicitation for 2001:db8:cafe:babe:2865:550a:c512:cdf0
22	8.628524	::	ff02::1:ff12:cdf0	ICMPv6	78	Neighbor solicitation for 2001:db8:dead:beef:2865:550a:c512:cdf0
23	8.628915	::	ff02::1:ffe6:824d	ICMPv6	78	Neighbor solicitation for 2001:db8:d06:f00d:a20b:baff:fee6:824d
25	8.851195	::	ff02::1:ffe6:824d	ICMPv6	78	Neighbor solicitation for 2001:db8:dead:beef:a20b:baff:fee6:824d
26	8.851470	::	ff02::1:ff12:cdf0	ICMPv6	78	Neighbor solicitation for 2001:db8:bade:affe:2865:550a:c512:cdf0

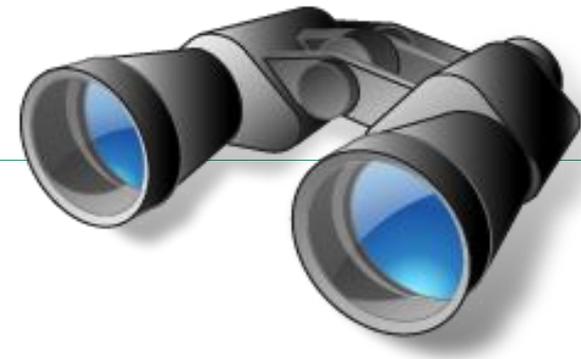
- ⊞ Frame 7: 90 bytes on wire (720 bits), 90 bytes captured (720 bits)
- ⊞ Ethernet II, Src: SamsungE\_e6:82:4d (a0:0b:ba:e6:82:4d), Dst: IPv6mcast\_00:00:00:16 (33:33:00:00:00:16)
- ⊞ Internet Protocol Version 6, Src: :: (::), Dst: ff02::16 (ff02::16)
- ⊞ Internet Control Message Protocol v6

- **In general no (user-) configurable options as for privacy extensions available.**
  - Which clearly violates RFC 4941.
- **Nobody seems to (fully) follow it anyway.**
  - Which makes “full operational control“ of privacy extensions (or, for that matter, addressing in general) much harder/next to impossible.
  - As we said: get over it (the idea of ...).

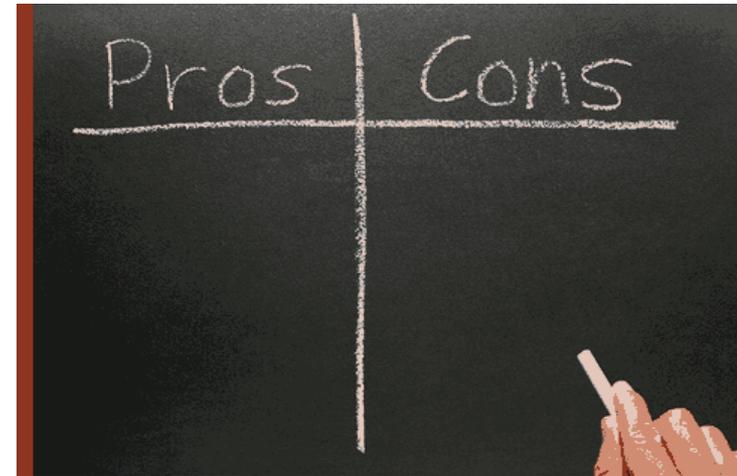


## Different Views on Privacy Extensions

---

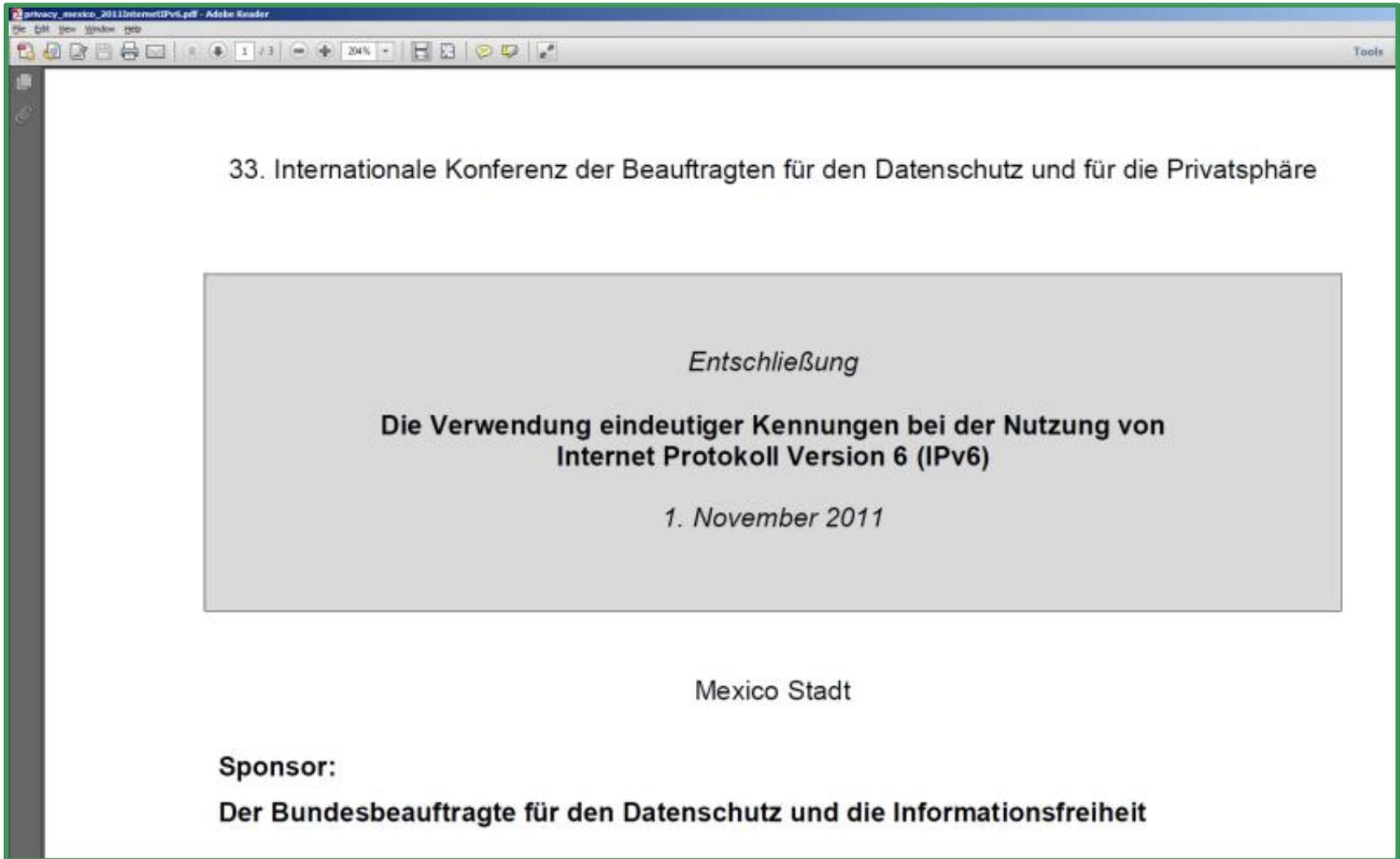


- **Netcitizen**
- **Data Protection Officer**
- **Corporate Information Security**
- **Corporate Network Operations / Management**



- **Does privacy really matter?**
  - “*They can track us anyway*“. (Cookies et.al.)
- **Has anybody noticed that Android 4.0 (Ice Cream Sandwich) now defaults to privacy extensions?**





33. Internationale Konferenz der Beauftragten für den Datenschutz und für die Privatsphäre

*Entscheidung*

**Die Verwendung eindeutiger Kennungen bei der Nutzung von  
Internet Protokoll Version 6 (IPv6)**

*1. November 2011*

Mexico Stadt

**Sponsor:**  
**Der Bundesbeauftragte für den Datenschutz und die Informationsfreiheit**

- Der Einsatz temporärer und nicht permanenter IPv6-Adressen muss mit den IPv6-Autokonfigurationsfunktionen möglich bleiben, indem alle vorhandenen Möglichkeiten der Pseudorandomisierung der Schnittstellenkennung ("Privacy Extensions") genutzt werden. Gerätehersteller – vor allem Hersteller mobiler Geräte – sollten solche Möglichkeiten schnell in ihre Produkte integrieren. Der Einsatz dynamischer Adressen für Endgeräte sollte als Standardfunktion aktiviert werden.
- Als Standardeinstellung sollten Anbieter, Protokolle, Produkte und Dienstleistungen die Nutzung temporärer und nicht permanenter Adressen anbieten.

# What Do those Guys Say?



Bundesamt  
für Sicherheit in der  
Informationstechnik



Leitfaden für eine sichere IPv6-Netzwerkarchitektur  
(ISi-L-IPv6)

BSI-Leitlinie zur Internet-Sicherheit (ISi-L)

## 3.3 Identifizierbarkeit von Geräten und Schutz der Privatsphäre

Wird zur Adresskonfiguration der Mechanismus der Stateless Address Autoconfiguration (siehe Abschnitt 2.3.1) genutzt, wird die MAC-Adresse des Netzwerkinterfaces zur Bildung der IPv6-Adresse herangezogen. Auf diese Weise kann ein Gerät (bspw. ein Notebook oder ein Smartphone) auch dann identifiziert werden, wenn es sich zwischen verschiedenen Netzwerken bewegt. Ein Beobachter (bspw. Anbieter eines Webdienstes) hätte so die Möglichkeit, ein detailliertes „Bewegungsprofil“ eines Gerätes alleine anhand seiner IP-Adresse zu erstellen.

Um diese globale Identifizierbarkeit zu vermeiden, unterstützen die meisten IPv6-Implementierungen sogenannte Privacy Extensions für die Adresskonfiguration. Dabei wird für das Interface in regelmäßigen Abständen eine neue Adresse aus dem jeweiligen Subnetz generiert und für neue Verbindungen verwendet. Auf diese Weise wird es einem Beobachter erschwert, ein Gerät weltweit allein anhand seiner IPv6-Adressen zu verfolgen. Der Nutzen der Privacy Extensions ist jedoch beschränkt, weil die Erstellung von Profilen anhand von Identifizierungsmerkmalen aus höheren Schichten (bspw. HTTP-Cookies) natürlich weiterhin möglich ist.

Im internen Netz erschwert die Nutzung von Privacy Extensions die Segmentierung des Netzes anhand von Paketfilterregeln, denn ein Whitelisting von zugelassenen IP-Adressen an Paketfiltern oder in Anwendungen ist auf diese Weise nicht möglich. Soll ein Teilnetz durch eine IP-Whitelist an einem Paketfilter geschützt werden, so müssen die Privacy Extensions deaktiviert werden oder es muss ein anderer Mechanismus für die Adresskonfiguration gewählt werden.

- In many organizations they don't like `priv_extensions` too much.

- **Main concerns**

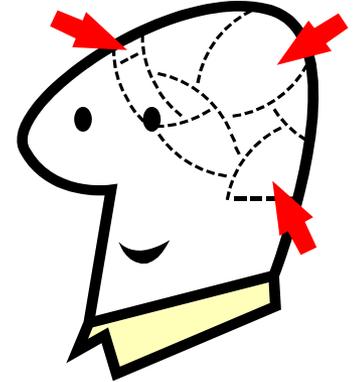
- Loss of traceability.
- Inability to perform forensic examinations.



- **Our stance**

- How many forensic investigations did you actually perform recently?
- With sufficient effort you might be able to find out nevertheless.
- Get over it. You won't be able to avoid `priv_extensions` anyway, given the current behavior of major smartphone OSs (see above).

- **Don't like priv\_extensions either.**
- **Mostly for same overall psychological reasons**
  - Concerned about (perceived) loss of control.



- **All types of lame excuses**
  - “We can't manage the (network|servers|whatever) anymore.”
  - “Logging & monitoring or troubleshooting is a pain then.”
  - Yadda yadda yadda



- **Point is, again: get over it, priv\_ext. will be there anyway.**

Jankiewicz, et al. Informational [Page 11]

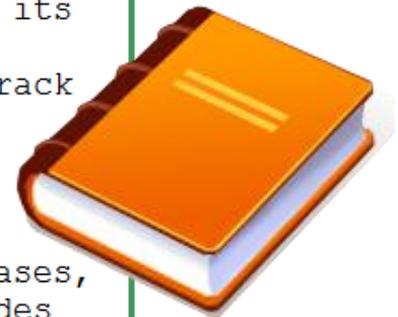
RFC 6434 IPv6 Node Requirements December 2011

### 5.9.3. Privacy Extensions for Address Configuration in IPv6 - RFC 4941

Privacy Extensions for Stateless Address Autoconfiguration [RFC4941] addresses a specific problem involving a client device whose user is concerned about its activity or location being tracked. The problem arises both for a static client and for one that regularly changes its point of attachment to the Internet. When using Stateless Address Autoconfiguration [RFC4862], the Interface Identifier portion of formed addresses stays constant and is globally unique. Thus, although a node's global IPv6 address will change if it changes its point of attachment, the Interface Identifier portion of those addresses remains the same, making it possible for servers to track the location of an individual device as it moves around or its pattern of activity if it remains in one place. This may raise privacy concerns as described in [RFC4862].

In such situations, RFC 4941 SHOULD be implemented. In other cases, such as with dedicated servers in a data center, RFC 4941 provides limited or no benefit.

Implementers of RFC 4941 should be aware that certain addresses are reserved and should not be chosen for use as temporary addresses. Consult "Reserved IPv6 Interface Identifiers" [RFC5453] for more details.



## Operational Implications

---



- **What Corp (NetworkOps|InfoSec) essentially want:**
- **Capability to uniquely identify and contact a given system “now“**
  - Software distribution, remote management, service desk, patch/AV signature distribution, (PXE?)
  - Consistent database needed => DNS
- **Capability to uniquely identify a given system ex post.**
  - Correlation of information needed (already).
  - IPv6 adoption & properties will change game anyway.
    - E.g. malware (disposing of high privs) initializing additional (IPv6) address and removing it once infection/spread job done. Not a fault of priv\_extensions, can happen in EUI-64 oder DHCPv6 based networks too.



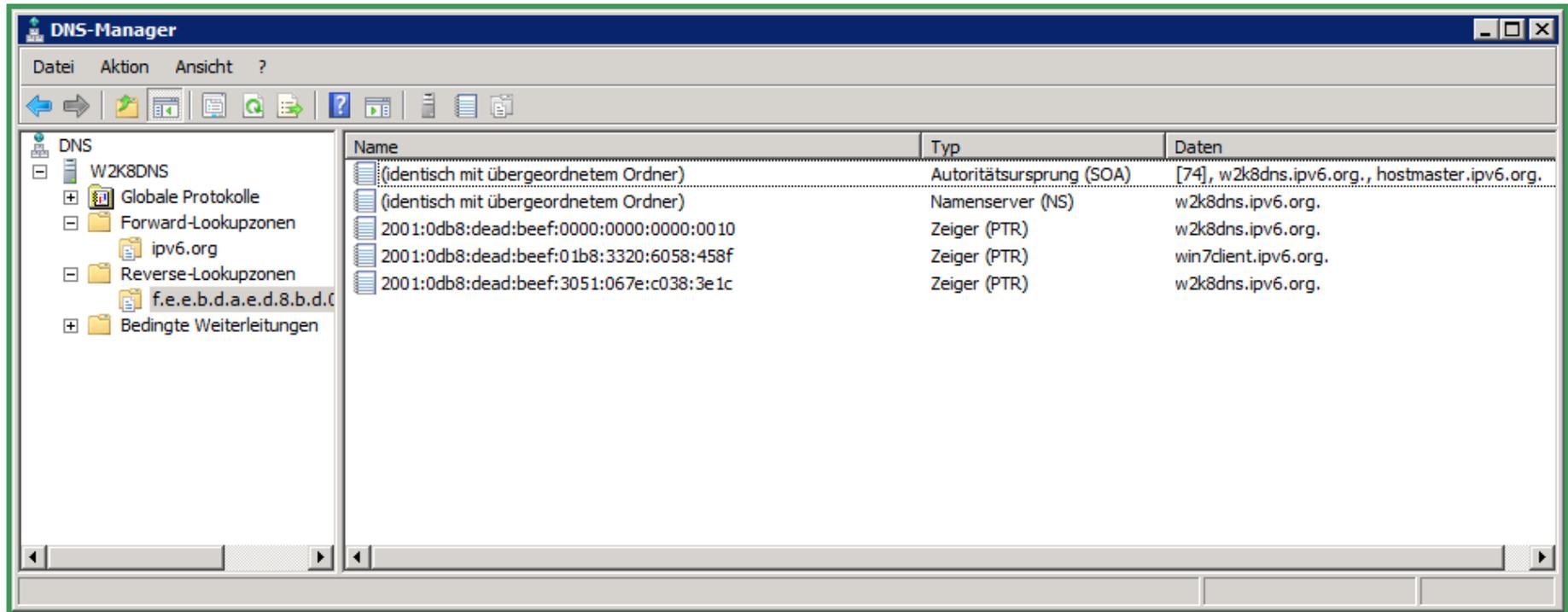
- **“Mature & fully integrated“ DNS needed in the IPv6 world anyway.**
- **In most environments this applies to reverse DNS as well.**
- **=> System mgmt maturity becomes dependent on (reverse) DNS maturity then.**
  - Simply said: if \$SOME\_SYSTEM (fully) supports (reverse) IPv6 DNS, you will be able to manage it. Else you won't.
  - Unfortunately quite some elements “still lack maturity“ when it comes to (IPv6|IPv6 DNS|IPv6 reverse DNS).
  - This is not a specific issue of priv\_extensions though.

- **As soon as the IPv6 address is generated, Windows 7 triggers the dynamic DNS update to create (or update) a PTR record in the reverse lookup zone:**

```
[-] Domain Name System (query)
  [Response In: 242]
  Transaction ID: 0xce32
  [+ Flags: 0x2800 (Dynamic update)
    Zones: 1
    Prerequisites: 1
    Updates: 2
    Additional RRs: 0
  [+ Zone
  [+ Prerequisites
  [- Updates
    [+ f.8.5.4.8.5.0.6.0.2.3.3.8.b.1.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa: type PTR, class ANY
    [- f.8.5.4.8.5.0.6.0.2.3.3.8.b.1.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa: type PTR, class IN, win7client.ipv6.org
      Name: f.8.5.4.8.5.0.6.0.2.3.3.8.b.1.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa
      Type: PTR (Domain name pointer)
      Class: IN (0x0001)
      Time to live: 20 minutes
      Data length: 21
      Domain name: win7client.ipv6.org
```

- **Note: The temporary address will not be inserted into DNS (only the “main address”).**

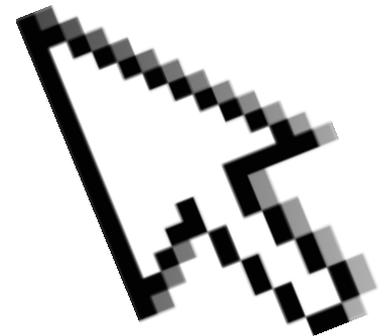
# PTR Records, Example



The screenshot shows the Windows DNS Manager interface. The left pane displays the hierarchy: DNS > W2K8DNS > Forward-Lookupzonen > ipv6.org. The right pane shows a table of records:

Name	Typ	Daten
(identisch mit übergeordnetem Ordner)	Autoritätsursprung (SOA)	[74], w2k8dns.ipv6.org., hostmaster.ipv6.org.
(identisch mit übergeordnetem Ordner)	Namensserver (NS)	w2k8dns.ipv6.org.
2001:0db8:dead:beef:0000:0000:0000:0010	Zeiger (PTR)	w2k8dns.ipv6.org.
2001:0db8:dead:beef:01b8:3320:6058:458f	Zeiger (PTR)	win7client.ipv6.org.
2001:0db8:dead:beef:3051:067e:c038:3e1c	Zeiger (PTR)	w2k8dns.ipv6.org.

- **What happens if an address is no longer used/existent?**
- `netsh interface ipv6 delete address interface=eth0  
2001:db8:dead:beef:1b8:3320:6058:458f`

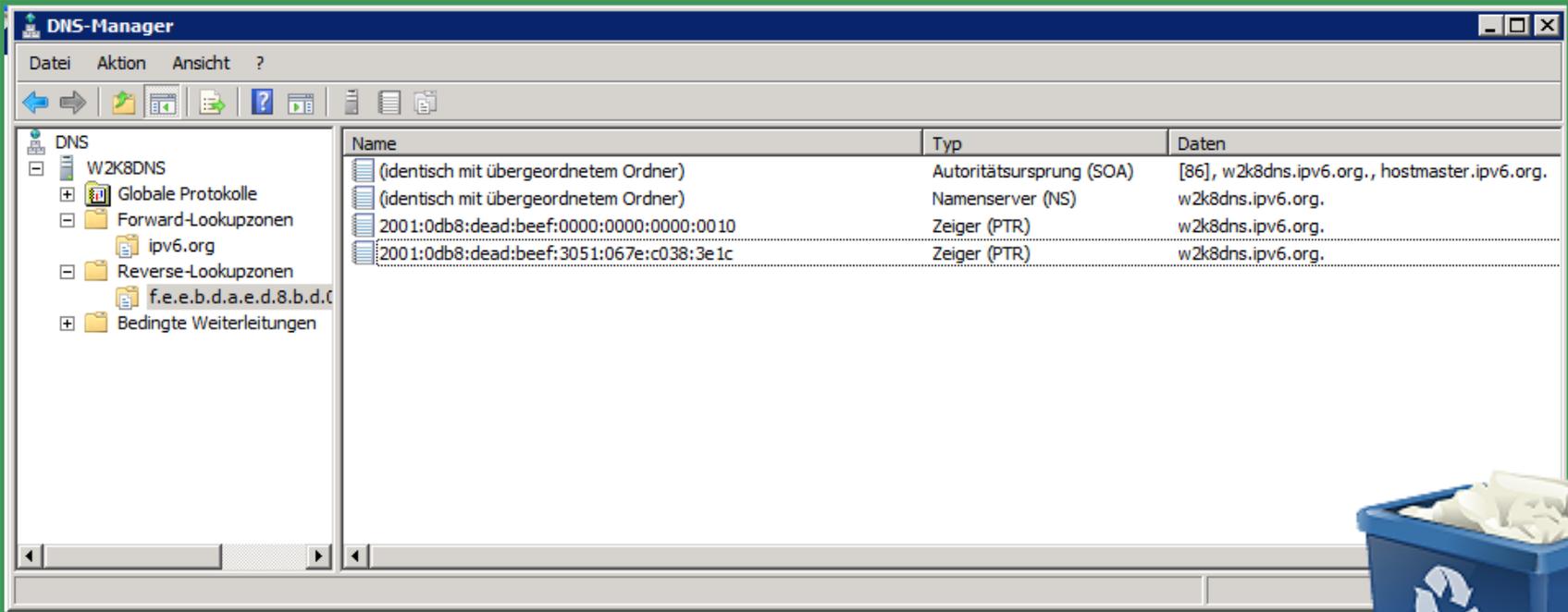


- **As soon as the address is removed, the Windows 7 client sends an DNS update to remove the PTR entry:**

```
[-] Domain Name System (query)
    [Response In: 614]
    Transaction ID: 0xba79
    [-] Flags: 0x2800 (Dynamic update)
        Zones: 1
        Prerequisites: 0
        Updates: 1
        Additional RRs: 0
    [-] Zone
    [-] Updates
        [-] f.8.5.4.8.5.0.6.0.2.3.3.8.b.1.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa: type PTR, class NONE, win7client.ipv6.org
            Name: f.8.5.4.8.5.0.6.0.2.3.3.8.b.1.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa
            Type: PTR (Domain name pointer)
            Class: NONE (0x00fe)
            Time to live: 0 seconds
            Data length: 21
            Domain name: win7client.ipv6.org
```



- ... and the associated PTR record gets removed.



The screenshot shows the DNS Manager interface with the following table of records:

Name	Typ	Daten
(identisch mit übergeordnetem Ordner)	Autoritätsursprung (SOA)	[86], w2k8dns.ipv6.org., hostmaster.ipv6.org.
(identisch mit übergeordnetem Ordner)	Namenserver (NS)	w2k8dns.ipv6.org.
2001:0db8:dead:beef:0000:0000:0000:0010	Zeiger (PTR)	w2k8dns.ipv6.org.
2001:0db8:dead:beef:3051:067e:c038:3e1c	Zeiger (PTR)	w2k8dns.ipv6.org.

- **Unfortunately, it seems not possible that iOS devices register their (privacy) addresses dynamically in DNS.**
- **We couldn't find any information if this functionality will be implemented in the future or not.**
  - Roadmaps in Apple space...

- **We guess the answer is no.**



- **Well... how does an Android system learn IPv6 DNS (servers) at all? ;-)**
  - No manual config possible (at least not without rooting it).
  - DHCPv6 not supported.
  - RRDNS / RFC 6106 (to the best of our knowledge) not supported either.
  
- **So, there's other "issues" to be solved first.**



- **The number of systems using `priv_extensions` by default is growing.**
  - Trying to “correct”/modify this means “deviating from a standard” and hence associated operational effort.



- **Uniquely identifying a system (interface) for system mgmt purposes might still work, mainly by its DNS name.**
  - Might additionally require dynamic DNS, which is (mostly) doable.
- **Ex post identification of systems (interfaces) by their IP address will probably not (and, for the record, never has) be(en), possible in an operationally feasible way.**
  - Don't waste your energy in trying this. There's other areas in IPv6 security space that need your precious resources.

There's never enough time...

**THANK YOU...**



**...for yours!**