ERNW
providing security.

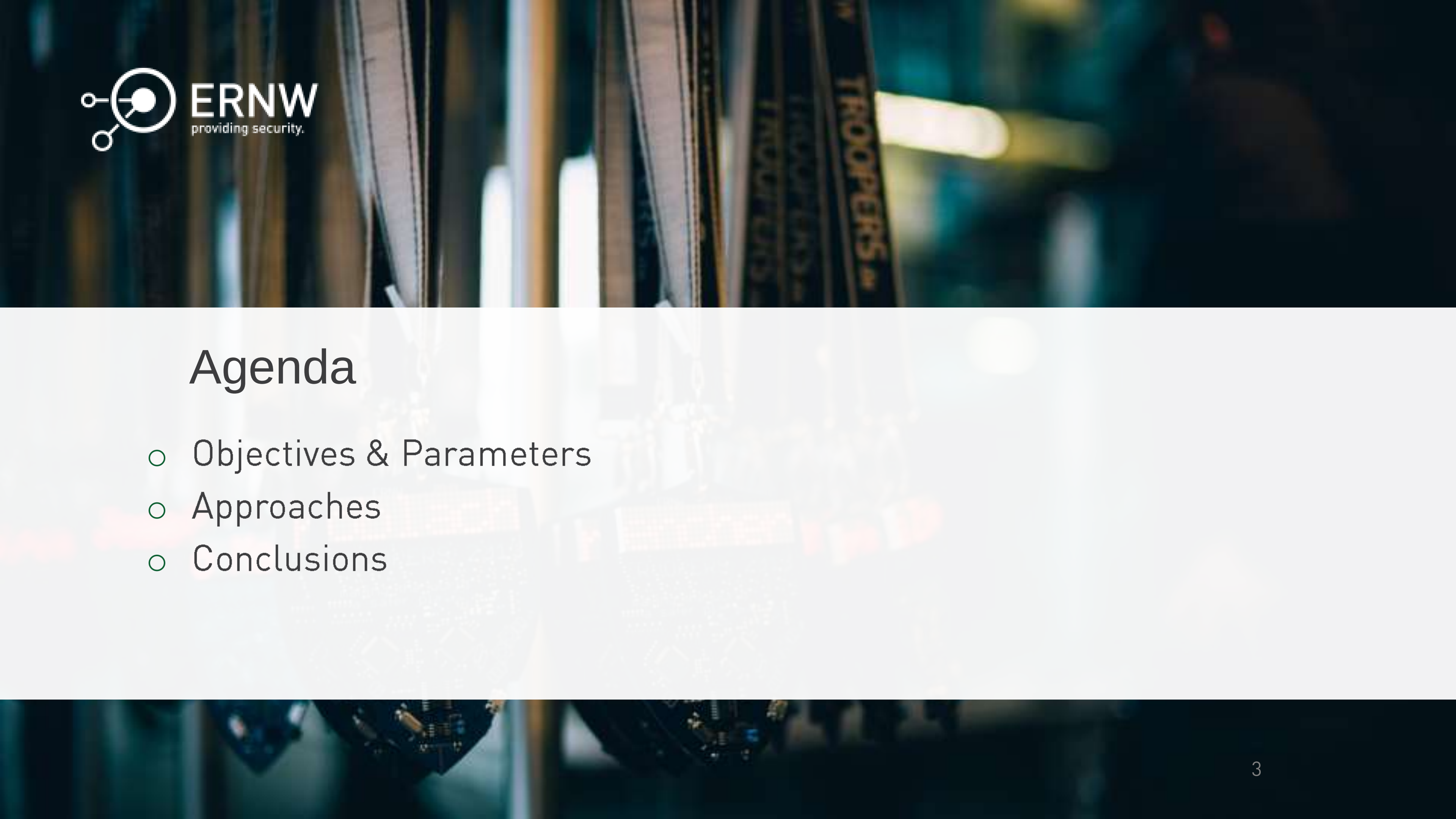
IPv6 Configuration Approaches for Servers

Enno Rey, erey@ernw.de

#whoami

- Infosec as full-time profession since 1997
- Founder of ERNW in 2001
- TROOPERS since 2008
- Blogs about IPv6 at
  - https://insinuator.net/tag/ipv6/

TROOPERS10 workout ;-)



Agenda

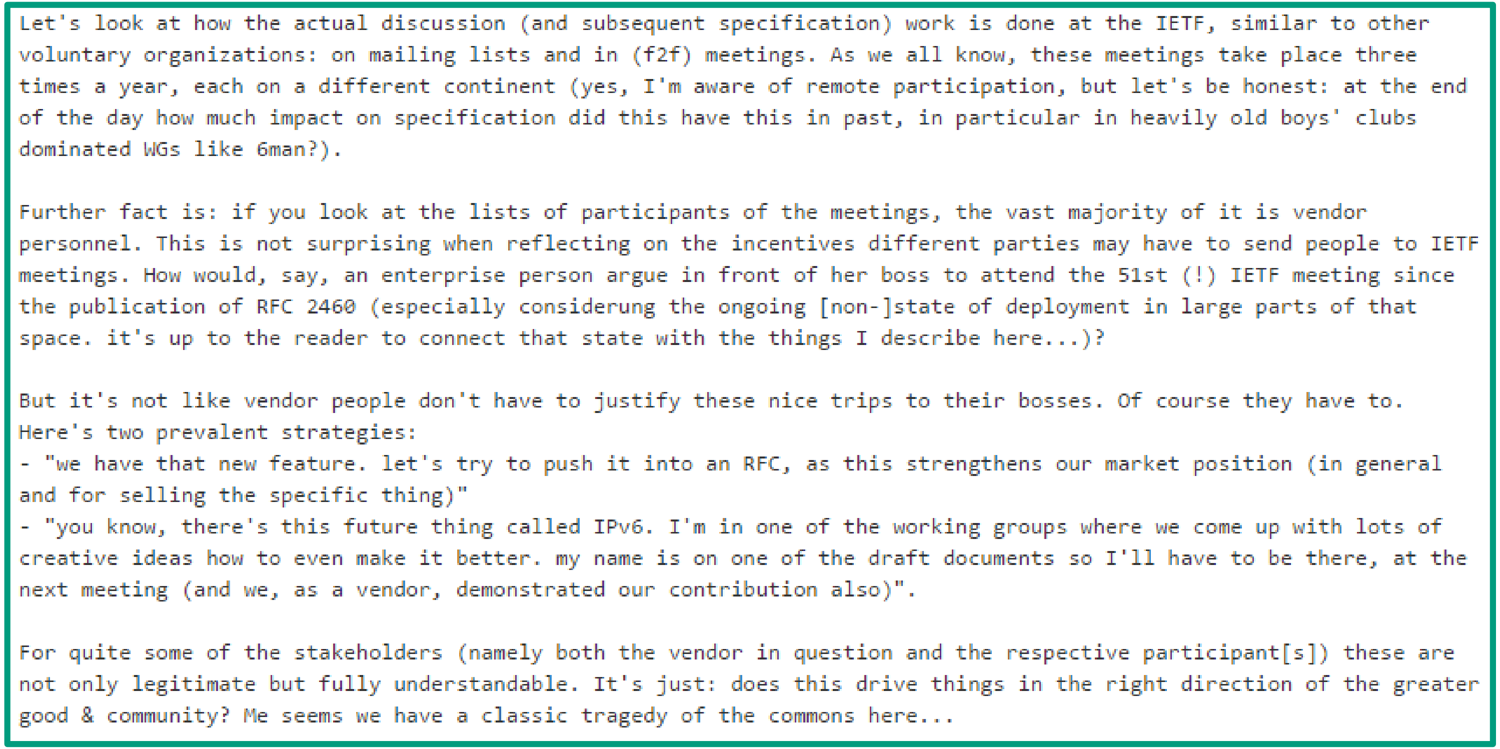
- Objectives & Parameters
- Approaches
- Conclusions

Introduction

- In the IPv4 age usually “static” IP parameters were used for certain systems
  - e.g. servers in data centers

- More options to do the task in IPv6, but
  - each with advantages/disadvantages
  - You must understand implications
  - Not all OSs support all parameters needed
    - Actually it’s a huge mess.
    - Testing & documentation needed
    - Many IPv6 people at the IETF ignore the problem, usually for political reasons.

Quick Rant on How Things Work @ IETF



Let's look at how the actual discussion (and subsequent specification) work is done at the IETF, similar to other voluntary organizations: on mailing lists and in (f2f) meetings. As we all know, these meetings take place three times a year, each on a different continent (yes, I'm aware of remote participation, but let's be honest: at the end of the day how much impact on specification did this have this in past, in particular in heavily old boys' clubs dominated WGs like 6man?).

Further fact is: if you look at the lists of participants of the meetings, the vast majority of it is vendor personnel. This is not surprising when reflecting on the incentives different parties may have to send people to IETF meetings. How would, say, an enterprise person argue in front of her boss to attend the 51st (!) IETF meeting since the publication of RFC 2460 (especially considerung the ongoing [non-]state of deployment in large parts of that space. it's up to the reader to connect that state with the things I describe here...)?

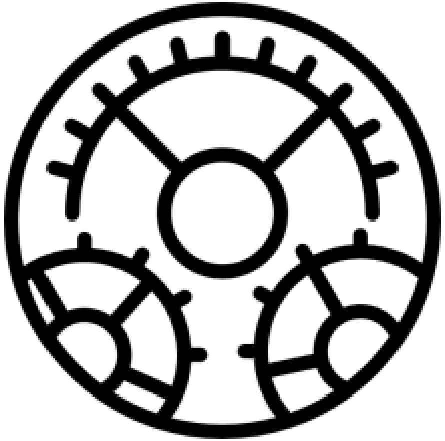
But it's not like vendor people don't have to justify these nice trips to their bosses. Of course they have to.
Here's two prevalent strategies:
- "we have that new feature. let's try to push it into an RFC, as this strengthens our market position (in general and for selling the specific thing)"
- "you know, there's this future thing called IPv6. I'm in one of the working groups where we come up with lots of creative ideas how to even make it better. my name is on one of the draft documents so I'll have to be there, at the next meeting (and we, as a vendor, demonstrated our contribution also)".

For quite some of the stakeholders (namely both the vendor in question and the respective participant[s]) these are not only legitimate but fully understandable. It's just: does this drive things in the right direction of the greater good & community? Me seems we have a classic tragedy of the commons here...

https://mailarchive.ietf.org/arch/msg/ipv6/tNR24ZN6o9APFEedk1_vTJe09Mc

Common Objectives of “Static“ Configuration

- Completeness & correctness of IP configuration
- Predictability (specific IP address at point of time)
- Traceability (identify systems, in real-time or hindsight)
- Security (resistance against link-local attacks)
- Operational feasibility (do it with a finite amount of resources)



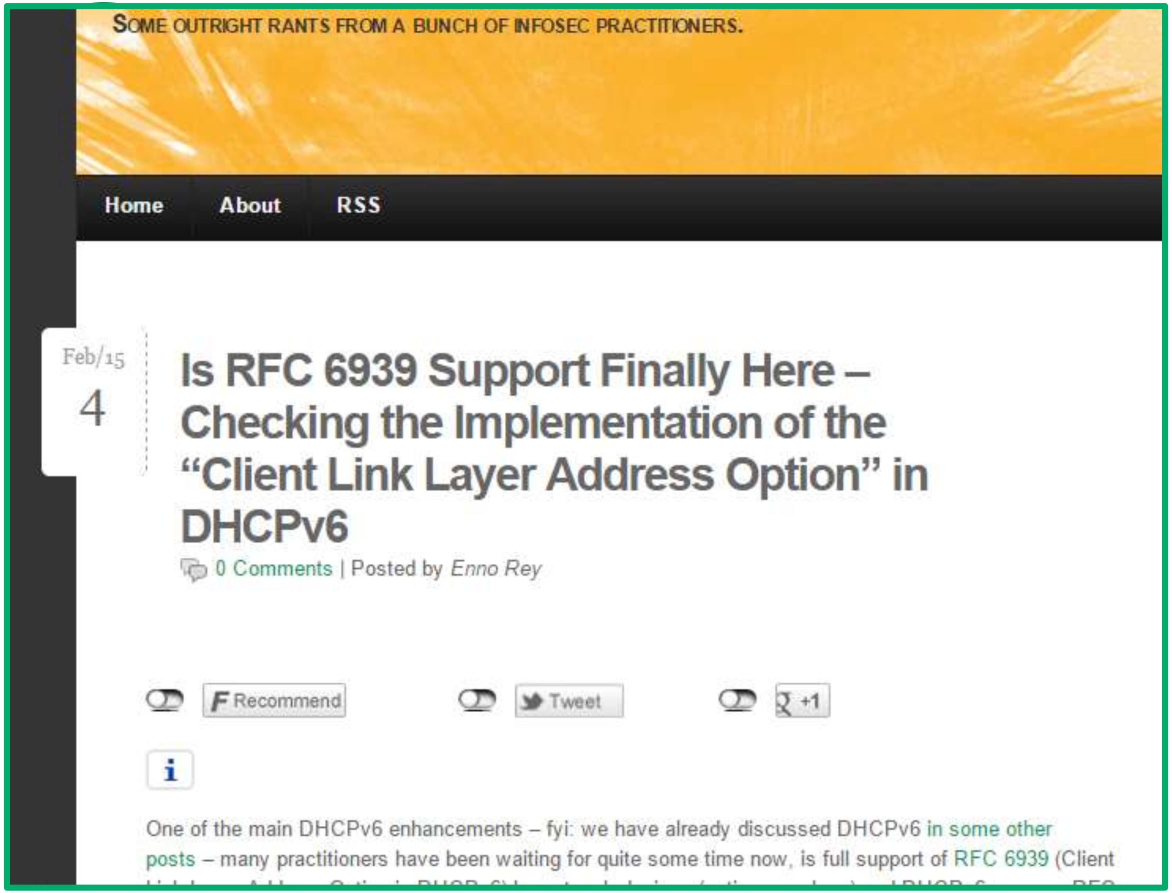
Parameters to Be Provisioned

- Global IPv6 address(es)
- Default gateway/route
- DNS resolver(s)
- NTP server(s)

Relevant Other IPv6 Properties of $OS (I)

- Address generation approach
  - Stable IIDs (RFC 7217)
  - Temporary addresses added (RFC 4941)

- DHCPv6
  - DUID generation (which method? changeable?)
    - See also comments of https://insinuator.net/2017/01/ipv6-properties-of-windows-server-2016-windows-10
  - DHCPv6 client enabled by default? behavior?
  - RFC 6939 support of nw infrastructure



SOME OUTRIGHT RANTS FROM A BUNCH OF INFOSEC PRACTITIONERS.

Feb/15
4

Is RFC 6939 Support Finally Here – Checking the Implementation of the “Client Link Layer Address Option” in DHCPv6

0 Comments | Posted by *Enno Rey*

One of the main DHCPv6 enhancements – fyi: we have already discussed DHCPv6 in some other posts – many practitioners have been waiting for quite some time now, is full support of RFC 6939 (Client

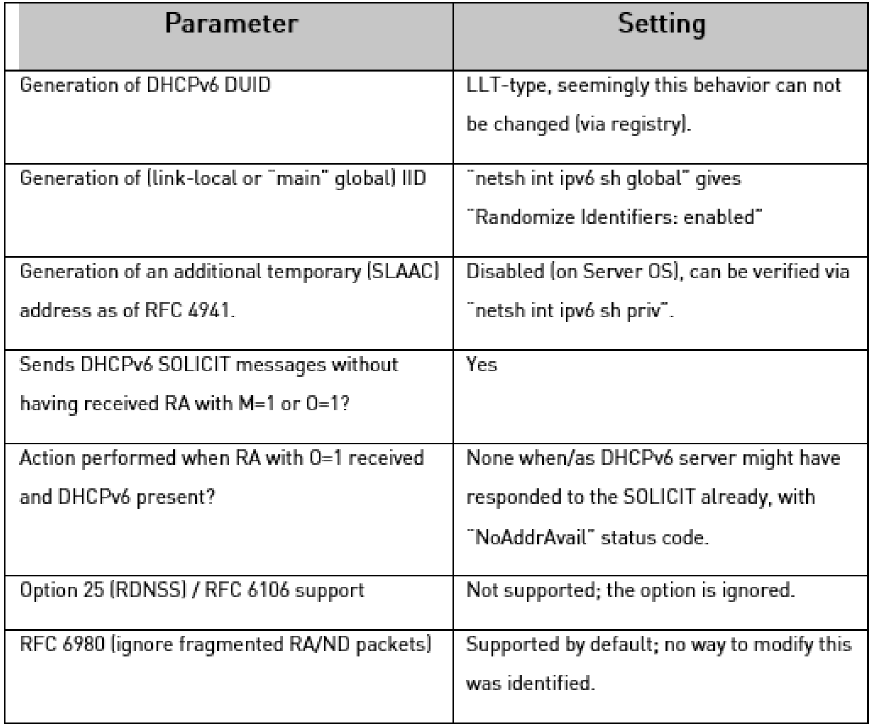
RFC 6939

http://www.insinuator.net/2015/02/is-rfc-6939-support-finally-here-checking-the-implementation-of-the-client-link-layer-address-option-in-dhcpv6/

Relevant Other IPv6 Properties of $OS (II)

- Support of Option 25 (RDNSS) in RAs
- Support of / behavior wrt RFC 6980
  - Does it drop fragmented RAs?
  - Rly?
  - Does it matter?
    - Many virtualized switches still do not support *RA Guard*.

Sample of OS Properties I Recently Looked At (Win Srv 2016)



Parameter	Setting
Generation of DHCPv6 DUID	LLT-type, seemingly this behavior can not be changed (via registry).
Generation of (link-local or “main” global) IID	“netsh int ipv6 sh global” gives “Randomize Identifiers: enabled”
Generation of an additional temporary (SLAAC) address as of RFC 4941.	Disabled (on Server OS), can be verified via “netsh int ipv6 sh priv”.
Sends DHCPv6 SOLICIT messages without having received RA with M=1 or O=1?	Yes
Action performed when RA with O=1 received and DHCPv6 present?	None when/as DHCPv6 server might have responded to the SOLICIT already, with “NoAddrAvail” status code.
Option 25 (RDNSS) / RFC 6106 support	Not supported; the option is ignored.
RFC 6980 (ignore fragmented RA/ND packets)	Supported by default; no way to modify this was identified.

How Can You Find Out?

- Quite often vendor documentation is, well, somewhat sparse.

- Digging through mailing lists (of $OS) can be helpful.
  - Or ask people who should know, like Fernando re: RFC 7217.

- RFC 6980 (support) usually requires actual testing.

IPv6 Security Testing

- Mainly four toolkits which can be used
  - Antonios Atlasis‘ Chiron
  - Marc Heuse’s THC-IPV6
  - Fernando Gont‘s IPv6 Toolkit
  - Scapy (whose IPv6 capabilities are, afaik, mainly maintained by Guillaume Valadon)
- Each has specific strenghts & limits
- For RFC 6980 testing we use Chiron because of its most powerful options as for IPv6 Extension Headers and fragmentation.

Recent Lab Testing (Srv 2016) – Main Phases and Baseline Attack

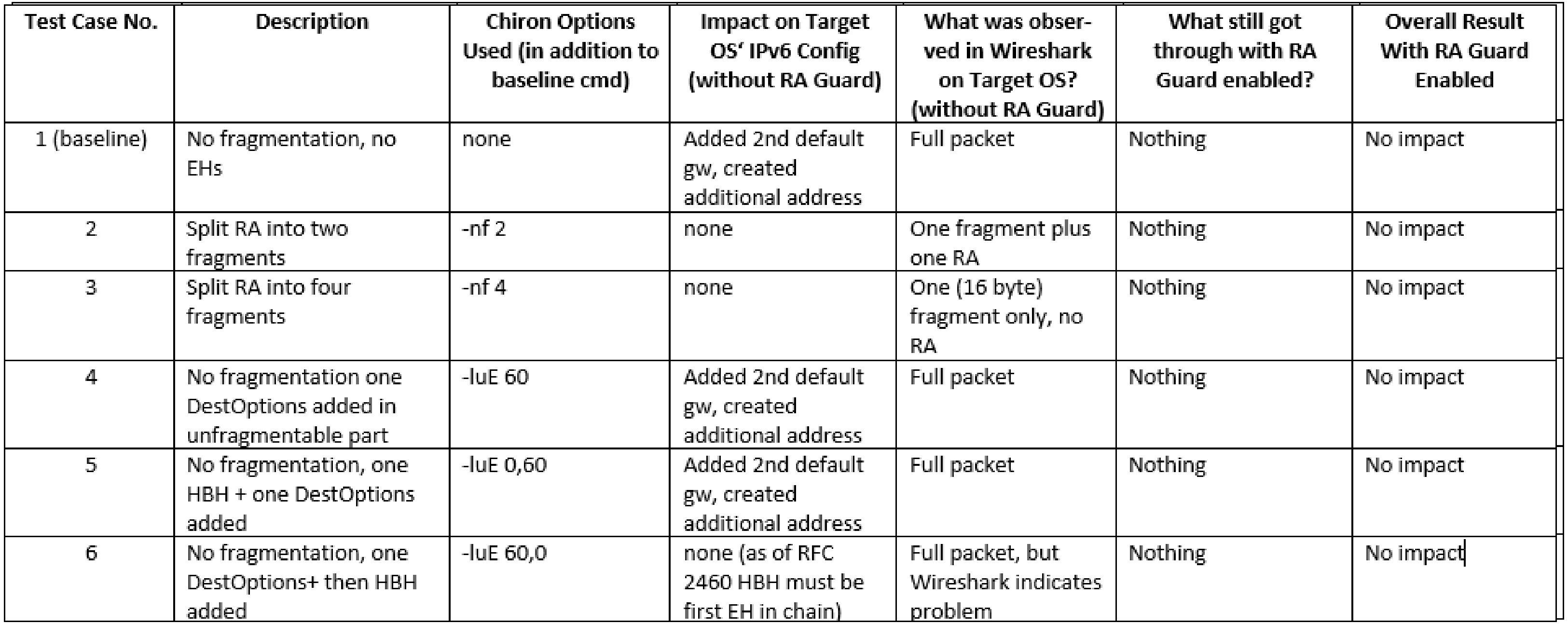
- Phase 1: without RA Guard on the switch

- Phase 2: enabled RA Guard and repeat (only) the "successful" variants from Phase 1
  - used "debug ipv6 snooping raguard" on the device (detection/blocking capabilities)

- Chiron command to attack the victim:

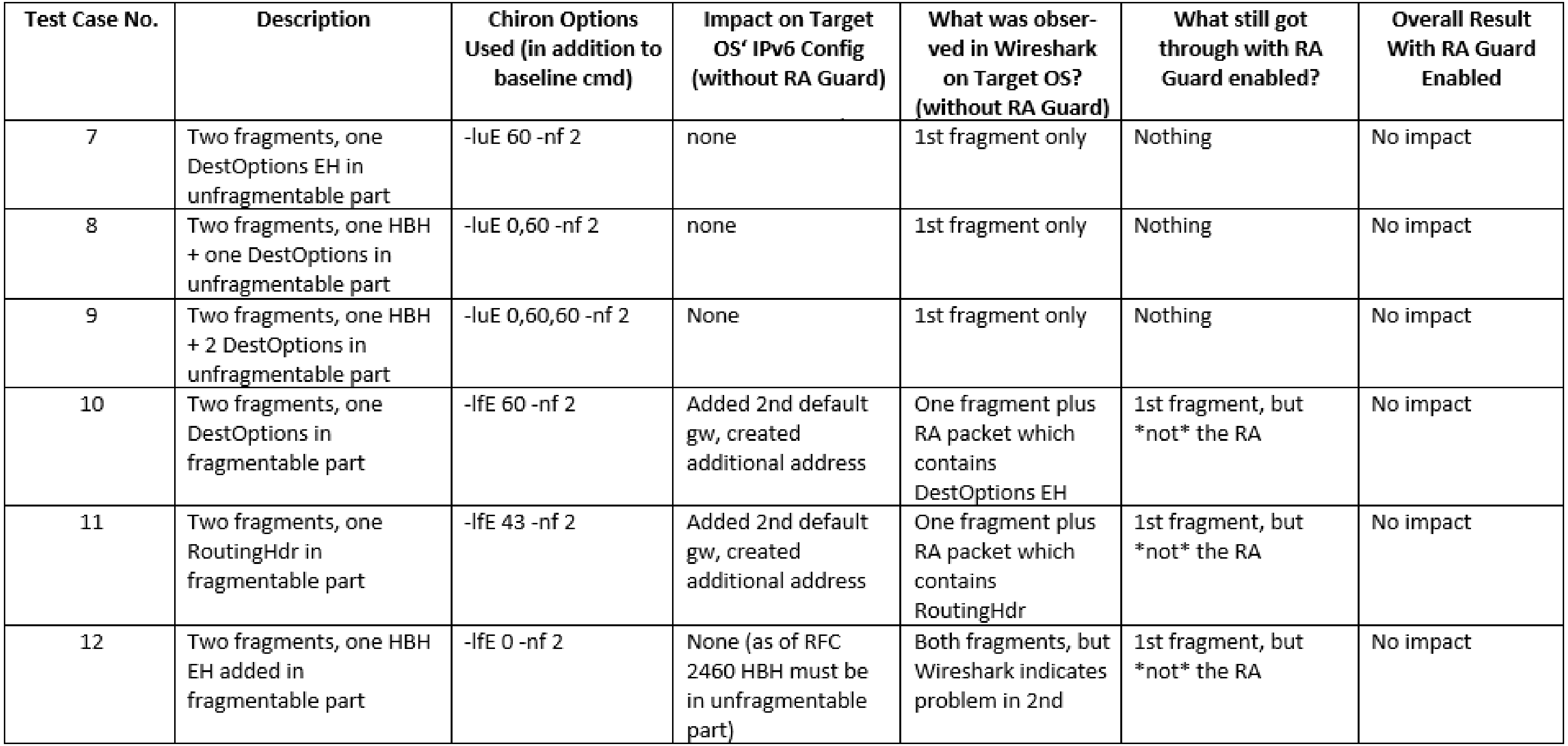
  -
    ```
    chiron_local_link.py eth0 -ra
    -pr 2001 1:db8:10:50:: -pr-length 64 -mtu 1400
    -s fe80::3aea:a7ff:fe85:c926
    ```

Relevant Results Part 1 (Sample)



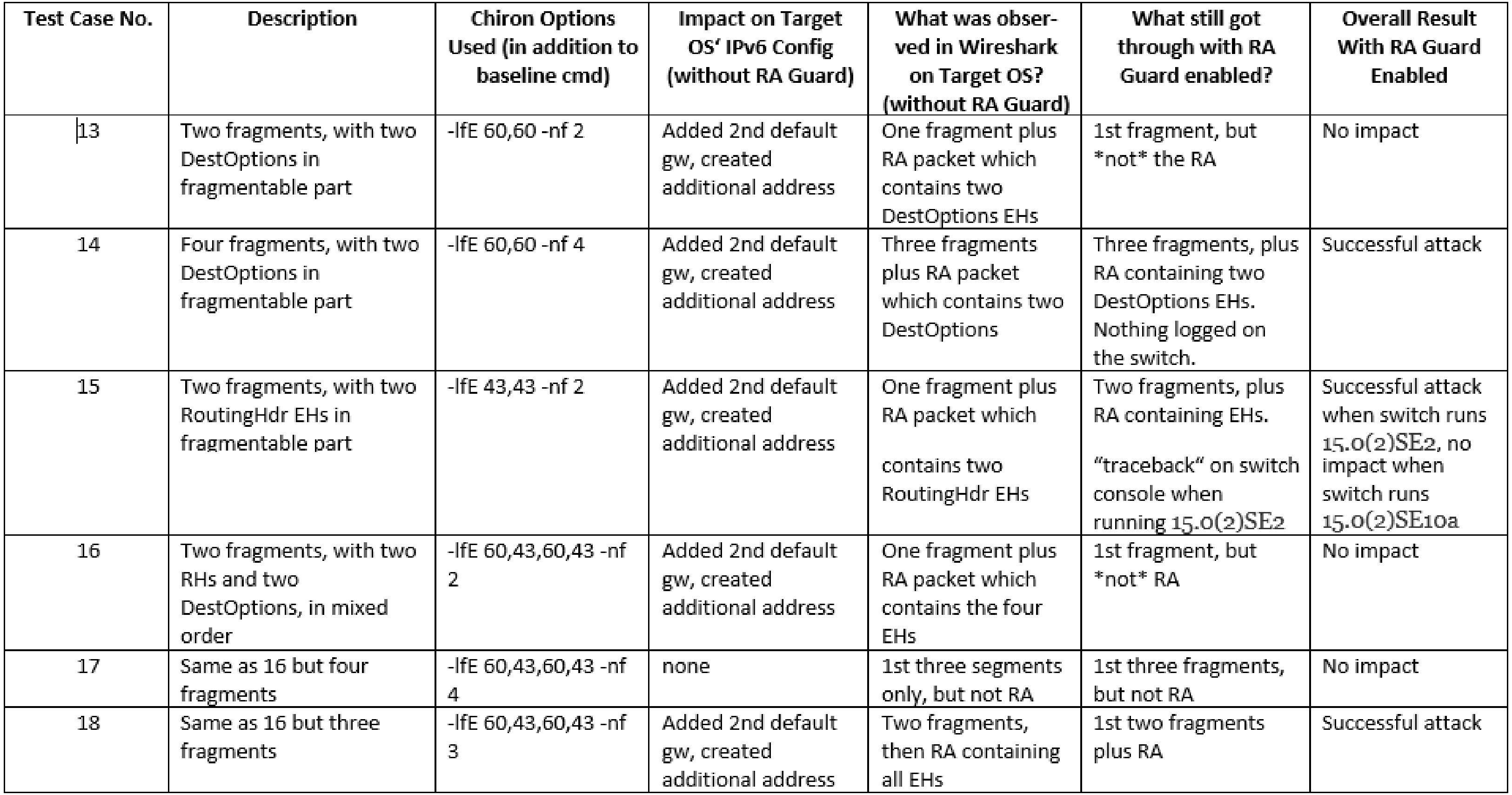
Test Case No.	Description	Chiron Options Used (in addition to baseline cmd)	Impact on Target OS‘ IPv6 Config (without RA Guard)	What was obser-ved in Wireshark on Target OS? (without RA Guard)	What still got through with RA Guard enabled?	Overall Result With RA Guard Enabled
1 (baseline)	No fragmentation, no EHs	none	Added 2nd default gw, created additional address	Full packet	Nothing	No impact
2	Split RA into two fragments	-nf 2	none	One fragment plus one RA	Nothing	No impact
3	Split RA into four fragments	-nf 4	none	One (16 byte) fragment only, no RA	Nothing	No impact
4	No fragmentation one DestOptions added in unfragmentable part	-luE 60	Added 2nd default gw, created additional address	Full packet	Nothing	No impact
5	No fragmentation, one HBH + one DestOptions added	-luE 0,60	Added 2nd default gw, created additional address	Full packet	Nothing	No impact
6	No fragmentation, one DestOptions+ then HBH added	-luE 60,0	none (as of RFC 2460 HBH must be first EH in chain)	Full packet, but Wireshark indicates problem	Nothing	No impact

Relevant Results Part 2



Test Case No.	Description	Chiron Options Used (in addition to baseline cmd)	Impact on Target OS‘ IPv6 Config (without RA Guard)	What was obser-ved in Wireshark on Target OS? (without RA Guard)	What still got through with RA Guard enabled?	Overall Result With RA Guard Enabled
7	Two fragments, one DestOptions EH in unfragmentable part	-luE 60 -nf 2	none	1st fragment only	Nothing	No impact
8	Two fragments, one HBH + one DestOptions in unfragmentable part	-luE 0,60 -nf 2	none	1st fragment only	Nothing	No impact
9	Two fragments, one HBH + 2 DestOptions in unfragmentable part	-luE 0,60,60 -nf 2	None	1st fragment only	Nothing	No impact
10	Two fragments, one DestOptions in fragmentable part	-lfE 60 -nf 2	Added 2nd default gw, created additional address	One fragment plus RA packet which contains DestOptions EH	1st fragment, but *not* the RA	No impact
11	Two fragments, one RoutingHdr in fragmentable part	-lfE 43 -nf 2	Added 2nd default gw, created additional address	One fragment plus RA packet which contains RoutingHdr	1st fragment, but *not* the RA	No impact
12	Two fragments, one HBH EH added in fragmentable part	-lfE 0 -nf 2	None (as of RFC 2460 HBH must be in unfragmentable part)	Both fragments, but Wireshark indicates problem in 2nd	1st fragment, but *not* the RA	No impact

Relevant Results Part 3



Test Case No.	Description	Chiron Options Used (in addition to baseline cmd)	Impact on Target OS‘ IPv6 Config (without RA Guard)	What was obser-ved in Wireshark on Target OS? (without RA Guard)	What still got through with RA Guard enabled?	Overall Result With RA Guard Enabled
13	Two fragments, with two DestOptions in fragmentable part	-lfE 60,60 -nf 2	Added 2nd default gw, created additional address	One fragment plus RA packet which contains two DestOptions EHs	1st fragment, but *not* the RA	No impact
14	Four fragments, with two DestOptions in fragmentable part	-lfE 60,60 -nf 4	Added 2nd default gw, created additional address	Three fragments plus RA packet which contains two DestOptions	Three fragments, plus RA containing two DestOptions EHs. Nothing logged on the switch.	Successful attack
15	Two fragments, with two RoutingHdr EHs in fragmentable part	-lfE 43,43 -nf 2	Added 2nd default gw, created additional address	One fragment plus RA packet which contains two RoutingHdr EHs	Two fragments, plus RA containing EHs. “traceback“ on switch console when running 15.0(2)SE2	Successful attack when switch runs 15.0(2)SE2, no impact when switch runs 15.0(2)SE10a
16	Two fragments, with two RHs and two DestOptions, in mixed order	-lfE 60,43,60,43 -nf 2	Added 2nd default gw, created additional address	One fragment plus RA packet which contains the four EHs	1st fragment, but *not* RA	No impact
17	Same as 16 but four fragments	-lfE 60,43,60,43 -nf 4	none	1st three segments only, but not RA	1st three fragments, but not RA	No impact
18	Same as 16 but three fragments	-lfE 60,43,60,43 -nf 3	Added 2nd default gw, created additional address	Two fragments, then RA containing all EHs	1st two fragments plus RA	Successful attack

Four Possible Approaches in Enterprise Space

- Fully Static Configuration
- "Hybrid" with Static Address but Default Route via RA
- Stable Addresses [RFC 7217] with Dynamic DNS Updates
- DHCPv6 with Reservations

Fully Static Configuration

- All parameters are configured in a static manner.
- Disable dynamic IPv6 mechanisms on the local system
  - e.g. processing of router advertisements
  - DHCPv6 client.
- Results in a “deviation from default”
- Might pose operational challenges
  - Keep system state over the whole lifecycle

Fully Static Configuration

- Advantages
  - Accommodates human desire to be in control
  - High level of predictability and traceability if properly applied
  - Good resistance against RA-related attacks

Fully Static Configuration

- Disadvantages
  - Requires significant operations effort to be configured
  - Networks dynamics might lead to the need to renumber which *still needs work* (RFC 5887).
  - Violates core IPv6 principles (RAs being the “source of life” for an IPv6 stack).

Fully Static Configuration

- Accompanying Configuration of (L3) Devices
  - No RAs needed at all.
    - e.g. “ipv6 nd suppress all” on Cisco IOS
  - Occurrence of RAs in the local subnet would raise suspicion
    - Combination of RA guard and “ipv6 snooping logging packet drop”

config

ERNW
providing security.

“Hybrid” with Static Address but Default Route via RA

- Configure IP address and NTP server(s) in a static manner.
- Configure DNS server(s) in a static manner or learn them from RAs (currently not supported on Windows OS).
- Get default gateway from RAs.

“Hybrid” with Static Address but Default Route via RA

- Advantages
  - Does not require disabling of local RA processing
  - Higher degree of flexibility (e.g. changing DNS resolvers)
  - Local system interaction with FHRP protocols may work more smoothly (mentioned by some people, but not confirmed yet!)

“Hybrid” with Static Address but Default Route via RA

- Disadvantages
  - Reasonable RA-related security only with RA guard AND RFC 6980 support on the server OSs
    - RFC 6980 support on Linux systems by default, but “it depends” for Windows OS.
    - Alternatively Port- or VLAN-based ACLs
    - ACLs potentially not being desirable option from operations perspective ;-)

"Hybrid" with Static Address but Default Route via RA

- Accompanying Configuration of (L3) Devices
  - Clear prefix information option (PIO) in RAs
    - e.g. "ipv6 nd prefix 2001:db8:1:1::/64 no-advertise" on Cisco IOS
  - Add option 25 (RDNSS) and potentially option 31 (DNSSL) to RAs
    - e.g. "ipv6 nd ra dns server 2001:db8:1:1::53" on Cisco IOS

Stable Addresses [RFC 7217] with Dynamic DNS Updates

- Generate a static SLAAC address
  - As long as the system is in the same subnet
- Populate DNS with this address
  - Assuming communication only via systems DNS name
- NTP servers have to be distributed by other means.
- Also: RFC 8064 *Recommendation on Stable IPv6 Interface Identifiers*

Stable Addresses [RFC 7217] with Dynamic DNS Updates

- Advantages
  - Doesn’t need much tweaking of router advertisements.
  - Serves objectives predictability and traceability.
  - Minimal renumbering effort in case of network changes (system moves to another subnet).

Stable Addresses [RFC 7217] with Dynamic DNS Updates

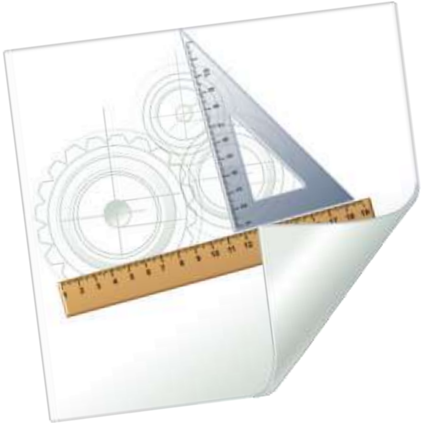
- Disadvantages
  - Involved systems must support RFC 7217.
  - Unauthenticated DNS updates will be required.
  - Windows systems currently not fulfill the requirements (RFC 6106 and 7217).
  - Protection from rogue RAs requires RA Guard AND RFC 6980 (or Port-/VLAN-based ACLs).
  - Distribution of NTP servers to be done by some other means.

Stable Addresses [RFC 7217] and Dynamic DNS Updates

- Accompanying Configuration of (L3) Devices
  - Default, but RDNSS must be distributed via RAs
  - Optionally clear L-flag in PIO for PVLAN-like behavior.
    - This type of isolation can easily be circumvented by an attacker with high privileges on local system (by adding host routes [for systems to-be-attacked] or a “network“ route for the local subnet).

DHCPv6 with Reservations

- To some extent familiar from the IPv4 world.
- Reservations needed on $DHCP_SERVER.
  - Initial DHCPv6 procedure might require somewhat manual interaction.
  - Or heavy configuration tweaking.
- Support of RFC 6939 is usually necessary.



DHCPv6 with Reservations

- Advantages
  - The closest you can come to centralized IP parameter provisioning (and administration).
  - Supports predictability and traceability.
  - IPv6 Addresses, DNS resolver(s) and NTP server(s) can all be distributed.

DHCPv6 with Reservations

- Disadvantages
  - DHCPv6 often turns out to be a somewhat unreliable/immature beast.
  - Need of RFC 6939 support might be a show-stopper.
    - To the best of our knowledge it's not supported in Cisco IOS so far.
    - But available (and even enabled by default) in IOS-XE.

DHCPv6 with Reservations

- Accompanying Configuration of (L3) Devices
  - Configure everything that’s needed to operate DHCPv6 (m-flag in RAs et al.)…
  - Optional L-flag in the PIO if needed.

ERNW
providing security.

Conclusions

- In the IPv6 world there's several options to configure systems with a "proper set of IP parameters"
  - Carefully consider advantages/disadvantages.
  - Going with "fully static" usually *not* a good option.
  - Several parameters have to be considered.

- The task is hindered by inconsisted support of parameters by different OSs.
  - As so often in IPv6 thorough testing is key.

ERNW
providing security.

Thank you for your Attention!

✉ erey@ernw.de

www.ernw.de

@Enno_Insinuator

www.insinuator.net

References

- [1] https://insinuator.net/2016/12/ipv6-configuration-approaches-for-servers/

Image Source:

- Icons made by Freepik from www.flaticon.com