



# DESIGN & CONFIGURATION OF IPv6 SEGMENTS WITH HIGH SECURITY REQUIREMENTS

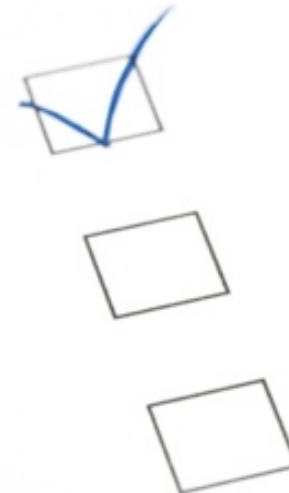
Enno Rey  
[erey@ernw.de](mailto:erey@ernw.de)

- **Old-school network security guys from**
- **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) (← You're here ;-)**



- **This presentation discusses which specific IPv6 design & configuration approaches might be used for network segments with very high security requirements, such as DMZs or “secure services areas” or similar networks.**
  - The fictional segment in question is called \$SEGMENT from here on.
- **This presentation is *not* about securing access networks or segments with many (client) systems.**
- **It is assumed that you already have a solid understanding of IPv6.**
  - Still, feel free to ask questions at any point.

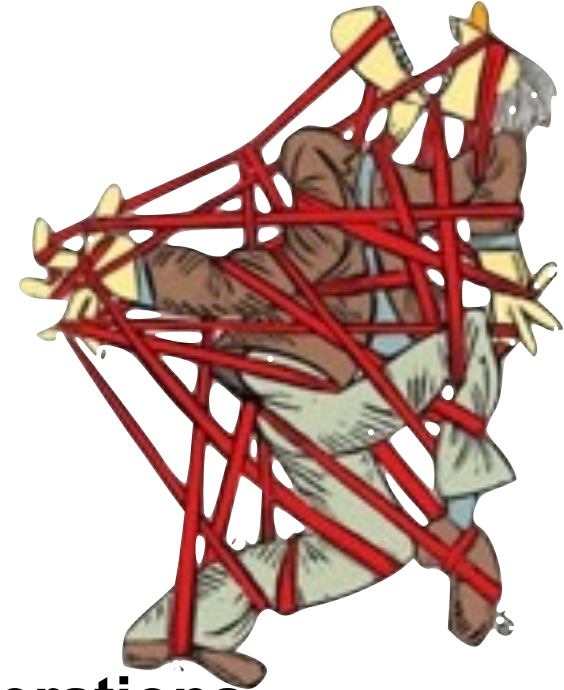
- **Design stuff**
- **Typical configuration steps to address the most relevant attacks.**
- **Filtering & misc**



- **Quite some stuff discussed in this presentation (namely in the “host configuration” sections) heavily contradicts traditional IPv6 networking paradigms**
  - Configuring static addresses for hosts.
  - Potential use of prefix lengths  $> /64$ .
  - Deactivation of RA processing on hosts.
- **So you might apply some specific toolset in a world that tends/expects to follow completely different rules.**
  - Do not underestimate the operational impact of this. Do not!



- You knew this, did you?



- Whatever you do... pls keep in mind:  
**Complexity *always* has an impact on operations.  
And thereby impedes overall security.**
- Everybody involved in securing complex networks & systems: please read RFC 3439. Please!

# “Deviation from Default”

- **By this term we designate any deviation from a default setting of any IT system which happens by means of some configuration step(s).**
  - Change some parameter from “red” to “black” or 0 to 1 or ...
- ***Deviation from default always requires OpEx.***
  - In particular if to be maintained through affected systems’ lifecycle.
  - Even more so if affected system base is heterogeneous.
  - By its very nature, OpEx is limited. You knew that, right? ;-)
- ***Deviation from default doesn’t scale.***
  - \$SEGMENT might have 20 systems today. And tomorrow?
- ***Deviation from default adds complexity.***
  - In particular if it’s “just some small modifications” combined...
    - Remember RFC 3439’s *Coupling Principle*?





# Sorry, Guys, One More Thing



- **Where not stated otherwise ;-)** all the configuration pieces presented here have been validated in real-world customer environments or in our lab.
- **Still, given the “current VUCA-like state of RFC compliance” in the IPv6 offerings of some vendors, things might behave differently in your environments.**
  - Thorough testing required.
    - You do this in your IPv6 setups anyway, don't you? ;-)



# Seven Sisters

*Dei sju søstre, Norway*





Access Control



Isolation



Restriction



Encryption



Entity Protection



Secure Management



Visibility

See also: [bit.ly/SevenSisters](https://bit.ly/SevenSisters) [insinuator.net]

- **Dedicated /64 for each system might be a good idea**
  - Will make filtering easier
    - Networks *might* be easier to handle than hosts on some filtering devices.
      - Could depend on number of “networks” though ;-)
    - No need to take care of “interface identifier assignment issues”.
    - Potentially facilitates tracking/auditing/logging.
  - Impact on/interference with clustering approaches?



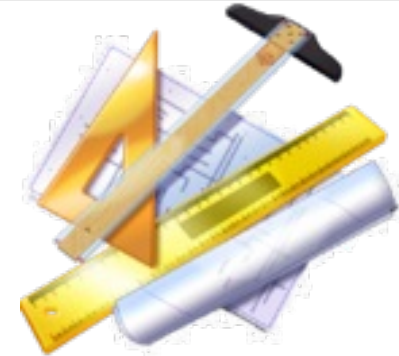
# Some Design Stuff To be Considered

## #2

### ■ **First hop redundancy approach to be discussed**

- In some setups HSRP/VRRP can be replaced by RAs
  - Reduces complexity ;-)
- Obviously you need RAs then ⇔ might contradict general RA handling approach for \$SEGMENT (see below).
- See also:
  - <http://blog.ioshints.info/2012/12/do-we-need-fhrp-hsrp-or-vrrp-for-ipv6.html>
  - <http://packetlife.net/blog/2011/apr/18/ipv6-neighbor-discovery-high-availability/>





**As for addressing, here's our 0.02**

- **Limit the number of addresses on any given interface.**
  - You do not really expect stacks (and services/applications!) to follow RFC 6724/3484, do you?
  - This not only applies to \$SEGMENT, but to all IPv6 deployments.
- **Hence, only use ULAs when connections to GUA\_world *proxied* somewhere.**
  - Did you get that? Do *not* use both on any given interface.
- **We prefer going with GUAs everywhere**
  - But, well, that's yet another of those IPv6 debates...

# What Do We Want to Protect \$SEGMENT From?

- **Attacks from outside**
  - Neighbor cache exhaustion (NCE)
  - Scanning
  
- **Attacks from within a segment**
  - NDP spoofing / flooding
  - Rogue router advertisements / flooding



# Neighbor Cache Entries

State	Description
INCOMPLETE	Neighbor Solicitation has been sent, but no Neighbor Advertisement has been retrieved.
REACHABLE	Positive confirmation was received within the last <i>ReachableTime</i> milliseconds, no special actions necessary.
STALE	ReachableTime milliseconds have elapsed, no actions takes place. This is entered upon receiving an unsolicited Neighbor Discovery message → entry must actually be used.
DELAY	ReachableTime milliseconds have elapsed and a packet was sent within the last <i>DELAY_FIRST_PROBE_TIME</i> seconds. If no message was sent → change state to PROBE.
PROBE	A reachability confirmation is actively sought by retransmitting Neighbor Solicitations every <i>RetransTimer</i> milliseconds until reachability confirmation is received.

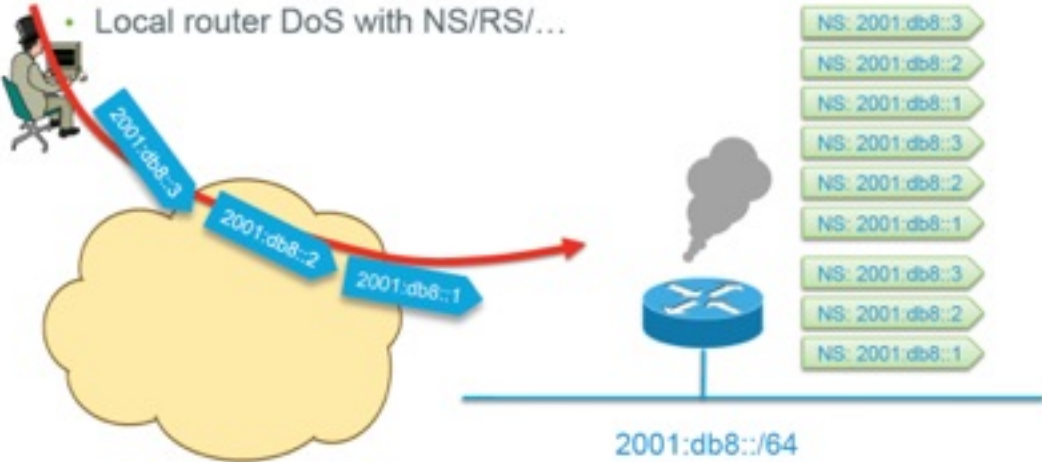


# Neighbor Cache Exhaustion

[this slide stolen from Eric Vyncke]

Scanning Made Bad for CPU  
Remote Neighbor Cache Exhaustion

- Remote router CPU/memory DoS attack if aggressive scanning  
Router will do Neighbor Discovery... And waste CPU and memory
- Local router DoS with NS/RS/...



© 2012 Cisco and/or its affiliates. All rights reserved. Cisco Public

Internet Engineering Task Force (IETF)  
Request for Comments: 6583  
Category: Informational  
ISSN: 2070-1721

I. Gashinsky  
Yahoo!  
J. Jaeggli  
Zynga  
W. Kumari  
Google, Inc.  
March 2012

## **Operational Neighbor Discovery Problems**

### **Abstract**

In IPv4, subnets are generally small, made just large enough to cover the actual number of machines on the subnet. In contrast, the default IPv6 subnet size is a /64, a number so large it covers trillions of addresses, the overwhelming number of which will be unassigned. Consequently, simplistic implementations of Neighbor Discovery (ND) can be vulnerable to deliberate or accidental denial of service (DoS), whereby they attempt to perform address resolution for large numbers of unassigned addresses. Such denial-of-service attacks can be launched intentionally (by an attacker) or result from legitimate operational tools or accident conditions. As a result of these vulnerabilities, new devices may not be able to "join" a network, it may be impossible to establish new IPv6 flows, and existing IPv6 transported flows may be interrupted.

This document describes the potential for DoS in detail and suggests possible implementation improvements as well as operational mitigation techniques that can, in some cases, be used to protect against or at least alleviate the impact of such attacks.

## ■ Filtering of Unused Address Space

- RFC 6583: “it is fully understood that this is ugly (and difficult to manage); but failing other options, it may be a useful technique especially when responding to an attack.”



## ■ Obviously this requires static addressing.

## ■ If you do this, use *stateless* filtering.

- ACLs might be your friend.
- Do *not* induce additional state by stateful filtering!
  - The more overall state maintained, the higher the overall vulnerability for DoS.

## ■ ***Minimal Subnet Sizing***

- RFC 6583: “this approach is not suitable for use with hosts that are not statically configured.”

## ■ **Well, this violates the /64 paradigm.**

- Doesn't RFC 6164 “allow” this violation anyway?
- Still, this is about leaving “a standard path”. Be careful!
  - “Organization’s culture” may play a role here.
- Yes, we are aware of sect. 3 of RFC 5375.
  - We don't regard this as relevant here though.



- **Overall this approach might have quite good *operational feasibility*. Provided nothing breaks due to deviation f. /64.**
- **If you do this, still assign full /64, but configure /120 or sth.**
  - So you can revert to /64 in case of problems or once better solutions are available (see below).

## ■ Routing Mitigation

- “For obvious reasons, host participation in the IGP makes many operators uncomfortable, but it can be a very powerful technique if used in a disciplined and controlled manner. One method to help address these concerns is to have the hosts participate in a different IGP (or difference instance of the same IGP) and carefully redistribute into the main IGP.”



## ■ **Honestly, this approach is so ridiculous both from an architecture and operations perspective, that we'll not discuss this further.**

- Anybody remembers the days of `routed` on some Unix systems... and how happy we were to get rid of it?

## ■ **Tuning of the NDP Queue Rate Limit**

- “It is worth noting that this technique is worth investigating only if the device has separate queues for resolution of unknown addresses and the maintenance of existing entries.”

## ■ **We expect this to become “the main approach”**

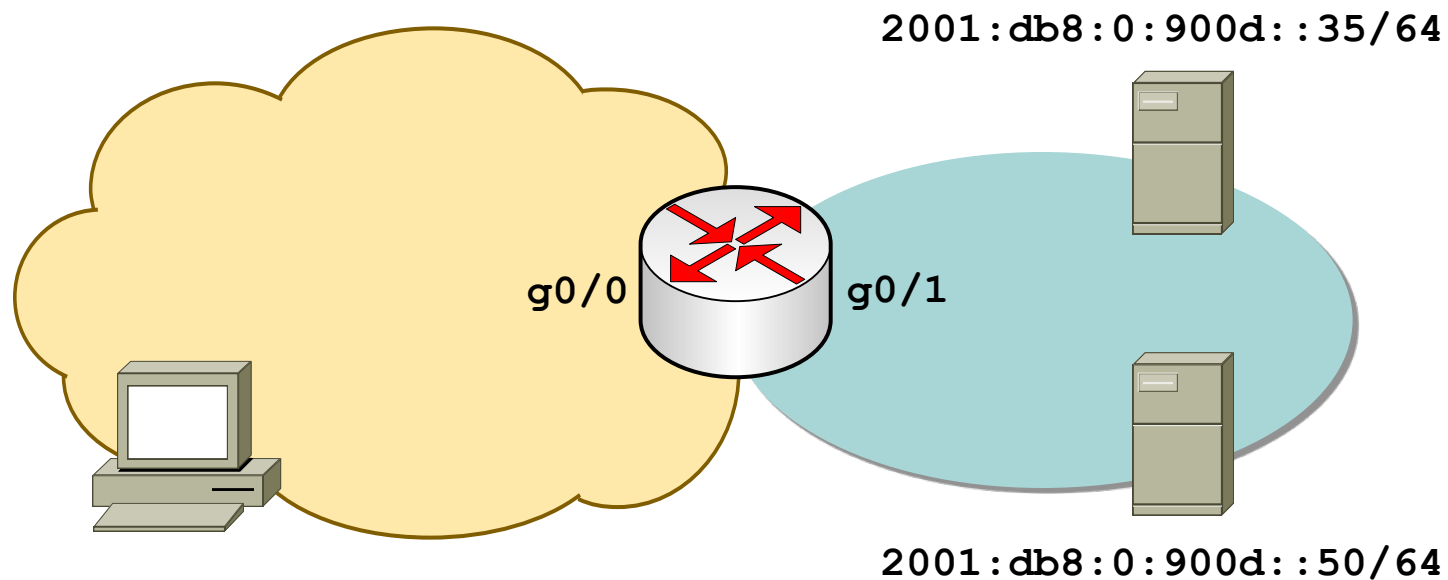
- Vendors already start to implement this. (see below)

## ■ **In Cisco land:**

- `ipv6 nd cache interface-limit`
  - See also <http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6/command/ipv6-i3.html#GUID-FC37F82B-5AAC-4298-BB6C-851FB7A06D88>
  - This one provides some logging, too. Might come in handy for attack detection.
    - Mar 10 15:11:51.719: %IPV6\_ND-4-INTFLIMIT: Attempt to exceed interface limit on GigabitEthernet0/1 for 2001:DB8:0:900D::2:329A (So use it in any case!)
- on IOS-XE 2.6: `ipv6 nd resolution data limit`
  - Thanks to Jim Small for this hint. Might address another problem though.



## ■ **Another suggestion: lowering retrans-timer to a sub-second value (suggestion f. Benedikt Stockebrand. Thx!)**



**Attacker**

```
GigabitEthernet0/0
  FE80::BAAD:1
  2001:DB8:0:BAAD::1/64
GigabitEthernet0/1
  FE80::900D:1
  2001:DB8:0:900D::1/64
```



- **All tested Cisco devices do not store more than 512 INCOMPLETE entries in neighbor cache, at any given time.**
  - Four different IOS-based medium-end devices tested.
- **Furthermore reading RFC 4861 sect. 7.2.2 indicates INCMP entries will be deleted after three seconds anyway.**
- **So NCE *seems* not to be a major problem here (C land).**
  - Various sources told us that Juniper space actually *is* susceptible to (NCE) problems.
  - We'll do some lab testing with an M7i and keep you posted.
    - Right now we can't comment on this further.
- **Details of testing to be found here**
  - <http://www.insinuator.net/2013/03/ipv6-neighbor-cache-exhaustion-attacks-risk-assessment-mitigation-strategies-part-1/>

- **If a system has a DNS record it will be found anyway.**
  - Derive your own conclusions...
- **See also**
  - <http://7bits.nl/blog/2012/03/26/finding-v6-hosts-by-efficiently-mapping-ip6-arpa>
  - Full thread on *IPv6 hackers* mailing list: <http://lists.si6networks.com/pipermail/ipv6hackers/2012-March/000526.html>

- **Here's the most common ones:**
  - NDP spoofing / flooding
  - Rogue router advertisements / flooding
  
- **Be aware: protecting from DoS attacks from within \$SEGMENT is *very hard* (at least as of Mar 2013)**
  - Due to high complexity of protocols involved (MLDv2...) and immature implementations.
  - Many tools available
    - THC IPv6 suite (<http://www.thc.org/thc-ipv6/>)
    - SI6 Network's IPv6 toolkit (<http://www.si6networks.com/tools/ipv6toolkit/>)
  
- **General mitigation approach**
  - Segmentation!
  - Prevent compromise in the first place.
  - Use infrastructure security controls on L2 devices (see below).

# The Rogue Router Advertisement Problem Statement

- **Router advertisements (as part of autoconfig approach) fundamental part of “IPv6 DNA”.**
  - Modifying this behavior (e.g. by deactivating their processing on the host level) is a severe “deviation from default” and as such “operationally expensive”.
  - Such an approach might be hard to maintain through a system’s lifecycle as well.
    - Think service packs in MS world, kernel updates, installation of libs/tools/apps.
- **By default, local link regarded trustworthy in IPv6 world**
  - All ND related stuff (which includes RAs) unauthenticated, by default.



## ■ RAs mainly have two functions:

- Advertise prefix(es) for SLAAC
  - Usually one does not want SLAAC in \$SEGMENT.
- Advertise default route
  - Well, a default route might be handy, at times ;-)
  - Sure, this *can* be configured statically/manually
    - See above as for discussion on first hop redundancy handling options.



## ■ In most scenarios these two can be handled differently.

- No SLAAC, but get default route by RAs.
- E.g. on Linux play with `accept_ra_defrtr` **vs.** `accept_ra_pinfo`.

- **Ok, then there's three main options:**
  - Suppress emission of RAs on infrastructure level.
  - Suppress processing of RAs on hosts.
  - Block forwarding of RAs on infrastructure (L2) level.



# Suppress Emission of RAs on Infrastructure Level

Comes in different flavors (full suppress vs. clearing A-flag)

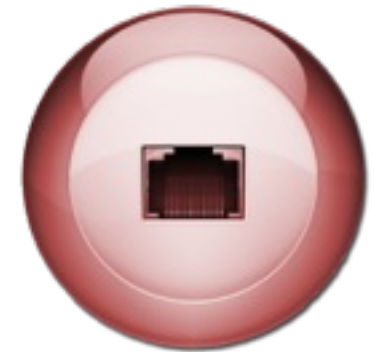
Will just prevent “benign” host processing, but not prevent attacks against hosts from their (potentially compromised) neighbors.

## ■ Full suppression

- Cisco:

```
L3_device(config-if)#ipv6 nd ra suppress [all]
```

- On some devices/OSs RAs might still be triggered by some host on local link sending router solicitation (RS) packets.
  - E.g. in Cisco land different behavior between 12.4 and 15.x releases. See also CSCth90147.
- Default route will have to be configured statically on hosts then, too.
  - Might have influence on first hop redundancy approach.  
Probably not relevant for these types of networks though.
- Must be kept in mind for future activities in \$SEGMENT.
  - People (other admins...) might expect it (RAs) “just to be there”.
  - We don’t like the suppress\_RAs approach anyway. Deviation from default...





- **Do not send out prefixes at all**
  - `L3_device(config-if)#ipv6 nd prefix default no-advertise`
- **Do not set A-flag in prefixes sent**



- **A (*Autonomous*) -Flag in router advertisements indicates “use prefix(es) for address autoconfig”.**

- See RFC 4862, sect.  
5.5.3 Router Advertisement Processing



- **Clearing A-flag**

- Cisco (ASAs only [?]):
  - `ipv6 nd prefix default no-autoconfig`
- Router advertisements will still be used for learning the default route(s).
- If using this approach, always combine with `router-preference high` (which you should use anyway, anywhere).

- **Some interesting, theory-only discussion on this recently**
  - Considered to potentially provide “private VLAN” type connectivity.
- **Obviously this can be circumvented by additional/manual configuration once a host is compromised with high privs.**
  - Still might be helpful to counter application based host-to-host stuff.
- **We don't know any practical way of removing the L-flag**
  - No operational experience so far with this approach => use w/ caution!
  - Removing full prefix from RAs might have same impact/benefit.
    - And (much) less operational impact/effort.
- **See also:**
  - <http://blog.ioshints.info/2012/11/ipv6-router-advertisements-deep-dive.html>
  - <http://blog.ioshints.info/2012/11/ipv6-on-link-determination-what-is-it.html>
  - RFC 5942 IPv6 Subnet Model: The Relationship between Links and Subnet Prefixes





- **Operationally expensive & severe deviation from default.**
- **Note: just assigning a static IP address might not suffice.**
  - E.g. MS Windows systems can still generate additional addresses/interface identifiers.
- **Still we know and – somewhat – understand that most of you have a strong affinity to this approach**
  - Human (and in particular: sysadmin) nature wants to *control* things...

## ■ MS Windows



- `netsh int ipv6 set int [index] routerdiscovery=disabled`

## ■ FreeBSD



FreeBSD®

- `sysctl net.inet6.ip6.accept_rtadv=0`
- Do not run/invoke `rtsold`. (but the above prevents this anyway).



## ■ Linux

- Sth like: `echo 0 > /proc/sys/net/ipv6/conf/*/accept_ra`
- See also IPv6 sect. of <https://www.kernel.org/doc/Documentation/networking/ip-sysctl.txt>

# Block Forwarding of RAs on Infrastructure (L2) Level

- **RA Guard or ACLs**
  - `_Or_!`
- **RA Guard currently (Mar 2013) not a bullet-proof solution.**
  - -DF switch in THC's `fakerouter6` does the trick.
    - See also <http://www.insinuator.net/2011/05/yet-another-update-on-ipv6-security-some-notes-from-the-ipv6-kongress-in-frankfurt/>
- **ACLs might be operationally expensive.**
  - Probably port based ACLs not part of your current ops model, right?
  - HW support needed
    - [http://docwiki.cisco.com/wiki/Cisco\\_IOS\\_IPv6\\_Feature\\_Mapping#IPv6\\_Features](http://docwiki.cisco.com/wiki/Cisco_IOS_IPv6_Feature_Mapping#IPv6_Features)
  - Still, currently best protection approach that's available
    - See also <http://www.insinuator.net/2012/03/the-story-continues-another-ipv6-update/>
- **RA Guard will (hopefully) evolve**
  - Some IETF drafts out there to address evasion problem
    - <http://tools.ietf.org/html/draft-ietf-v6ops-ra-guard-implementation-07>

# RA Guard Config (“old flavor”)

- Router(config-if)#ipv6 nd ?
- raguard   RA\_Guard Configuration Command
- Router(config-if)#ipv6 nd raguard ?
- <cr>
- Router(config-if)#switchport mode access
- Router(config-if)#ipv6 nd raguard
- Router(config-if)#exit
- Router(config)#exit
  
- Router# show version
- Cisco IOS Software, s3223\_rp Software (s3223\_rp-IPBASEK9-M) ,  
Version 12.2(33)SX15, RELEASE SOFTWARE (fc2)





```
4948E(config)#ipv6 access-list IPv6
4948E(config-ipv6-acl)#deny ipv6 any any undetermined-transport
4948E(config-ipv6-acl)#permit ipv6 any any
4948E(config)#interface g1/19
4948E(config-if)#ipv6 traffic-filter IPv6 in
```

# RA Guard Availability, Cisco Land

## [the other params covered in tomorrow's ws]

Cisco IOS IPv6 Feature Map: X

docs.cisco.com/wiki/Cisco\_IOS\_IPv6\_Feature\_Mapping#IPv6\_First-Hop\_Security\_Features

### IPv6 First-Hop Security Features

Feature	IPv6 Implementation Guide	12.xT/ 15.xT Release	12.xM/15.xM Release	12.2SE Release	12.2SG, 15.xSG, and 3.xSG Release	12.2SA/ 15S Release	12.2SX/ 12.2SY/ 15.8SY Release
DHCPv6 Guard (IPv6)	—	—	—	—	—	15.2(4)S DHCPv6 Guard <a href="#">d</a>	—
IPv6 Address Guard	—	—	—	15.0(2)SE IPv6 Snooping <a href="#">d</a>	—	15.3(1)S IPv6 Snooping <a href="#">d</a>	—
IPv6 Destination Guard	—	—	—	—	—	15.2(4)S IPv6 Destination Guard <a href="#">d</a>	—
IPv6 Device Tracking	Implementing First Hop Security in IPv6 <a href="#">d</a>	—	—	15.0(2)SE IPv6 Snooping <a href="#">d</a>	—	15.3(1)S IPv6 Snooping <a href="#">d</a>	12.2(50)SY 15.0(1)SY 15.1(1)SY IPv6 Device Tracking <a href="#">d</a>
IPv6 First-Hop Security Binding Table	Implementing First Hop Security in IPv6 <a href="#">d</a>	—	—	—	—	15.2(4)S IPv6 First-Hop Security Binding Table <a href="#">d</a>	—
IPv6 ND Multicast Suppress	—	—	—	—	—	15.3(1)S IPv6 Snooping <a href="#">d</a>	—
IPv6 Neighbor Discovery Inspection	Implementing First Hop Security in IPv6 <a href="#">d</a>	—	—	15.0(2)SE IPv6 Snooping <a href="#">d</a>	—	15.3(1)S IPv6 Snooping <a href="#">d</a>	12.2(50)SY 15.0(1)SY 15.1(1)SY IPv6 Snooping <a href="#">d</a>
IPv6 RA Guard	Implementing First Hop Security in IPv6 <a href="#">d</a>	—	15.2(4)M IPv6 RA Guard <a href="#">d</a>	15.0(2)SE IPv6 Snooping <a href="#">d</a>	12.2(54)SG 3.2SG 15.0(2)SG	15.2(4)S IPv6 RA Guard <a href="#">d</a>	12.2(33)SX4 12.2(50)SY 15.0(1)SY 15.1(1)SY IPv6 RA Guard <a href="#">d</a>
IPv6 Snooping	—	—	—	15.0(2)SE IPv6 Snooping <a href="#">d</a>	—	15.3(1)S IPv6 Snooping <a href="#">d</a>	—
IPv6 Source Guard and Prefix Guard	—	—	—	15.0(2)SE IPv6 Snooping <a href="#">d</a>	—	15.3(1)S IPv6 Source Guard and Prefix Guard <a href="#">d</a>	—
IPv6 Strict Host Mode Support	—	—	—	15.0(2)SE IPv6 Strict Host Mode Support <a href="#">d</a>	—	—	—

**Last time we checked (late 2012):**

- **Juniper (EX series): not available.**
- **HP: on some platforms since Dec 2011**
  - Release K.15.07.0002 for the 5400, 8200 and 3500 series switches.
  - Configuration is pretty straightforward:
    - `no] ipv6 ra-guard ports <port-list> [log]`
- **H3C: RA Guard available on many platforms.**

# For Completeness' Sake: Spoofed RA protection as of RFC 6104

- **Manual configuration**
- **RA Snooping (RA Guard)**
- **Using ACLs**
- **SEcure Neighbor Discovery (SEND)**
- **Router Preference**
- **Relying on Layer 2 Admission Control**
- **Host-Based Packet Filters**
- **Using an “Intelligent” Deprecation Tool**
  - E.g. NDPMon
- **Using Layer 2 Partitioning**



- In RFC 4191 an additional flag was introduced within RA messages to indicate the preference of a default router in case more than one are present on the local link.





- The *preference* values are encoded as a two-bit signed integer with the following values:
  - 01 High
  - 00 Medium (default)
  - 11 Low
  - 10 Reserved

- When the *preference* is set, the RA messages look like:

```
Internet Control Message Protocol v6
  Type: 134 (Router advertisement)
  Code: 0
  Checksum: 0xded0 [correct]
  Cur hop limit: 64
  Flags: 0x08
    0... .... = Not managed
    .0.. .... = Not other
    ..0. .... = Not Home Agent
    ...0 1... = Router preference: High
  Router lifetime: 1800
  Reachable time: 0
  Retrans timer: 0
  + ICMPv6 option (Source link-layer address)
  + ICMPv6 option (MTU)
  + ICMPv6 option (Prefix information)
```

```
Internet Control Message Protocol v6
  Type: 134 (Router advertisement)
  Code: 0
  Checksum: 0xcdc6 [correct]
  Cur hop limit: 64
  Flags: 0x00
    0... .... = Not managed
    .0.. .... = Not other
    ..0. .... = NOT Home Agent
    ...0 0... = Router preference: Medium
  Router lifetime: 1800
  Reachable time: 0
  Retrans timer: 0
  + ICMPv6 option (Source link-layer address)
  + ICMPv6 option (MTU)
  + ICMPv6 option (Prefix information)
```

- **The configuration of the preference is done with the following command:**



- Router(config)# interface f0/1
  - Router(config-if)# ipv6 nd router-preference {high | medium | low}
- **If the command is not configured, the default value of medium will be used in the RA messages.**
- **Command available since IOS Version 12.4(2)T**



# Evaluation of RFC 6104 Controls

## [Devices capable of RA Guard assumed]

Control	Sec Benefit	Operational Feasibility
Manual configuration	4	1
RA Snooping (RA Guard)	4	4
Using ACLs	5	3
SEcure Neighbor Discovery (SEND)	5	1
Router Preference	2	5
Relying on Layer 2 Admission Control	5	2
Host-Based Packet Filters	3	1
Using an “Intelligent” Deprecation Tool	2	1
Using Layer 2 Partitioning	4	3

- **In case there's additional \$CONTROLS in \$SEGMENTS, these should provide the same security benefit for IPv6, right?**
  - Let's call this *feature parity*.
  - \$SEC\_CONTROLS: IPSs, WAFs, EmailSec, ContentFilters

- **Frankly speaking, do *not* expect (security) feature parity as of today (Mar 2013).**
  - See workshop in the afternoon.
- ***Feature parity* does not necessarily mean *performance parity*...**

# (Lack of) Feature Parity, Sample

## Firewall components that support IPv6

 [Printer Friendly](#)  [Rate this Page](#)

Technical Articles ID: KB69266

Last Modified: December 17, 2012

### Summary

The table below shows the firewall components that support IPv6.

	Supports IPv6	Does not support IPv6
Administrative services	None	<ul style="list-style-type: none"><li>• Admin Console</li><li>• SF Administration Console</li><li>• SSH</li><li>• Telnet</li></ul>
Applications	All other applications	<p>For IPv6, use a generic application on the appropriate port(s) instead of these applications:</p> <ul style="list-style-type: none"><li>• Telnet</li><li>• RealMedia</li><li>• SOCKS</li><li>• Sun RPC</li><li>• SIP</li><li>• RTSP</li><li>• Oracle</li><li>• SSH</li><li>• RSH</li><li>• Citrix-ICA</li><li>• T120</li><li>• SMTP</li><li>• SNMP</li><li>• DNS</li><li>• H.323</li><li>• IIS</li><li>• MSSQL</li><li>• Citrix Browser</li><li>• rlogin</li></ul>



- **We're not aware of other stuff currently done**
  - Like tweaking sysctls (other than the RA processing related ones).
- **As for local / host based packet filtering**
  - Windows Firewall, iptables, pf all have capabilities.
  - Keep in mind:
    - You MUST understand NDP...
    - Following RFC 4890 as for ICMP might be a good idea.



- **IPv6 fragmentation can help in evading \$MAJOR\_IPSs (as of Mar 2013)**
  - Remember Antonios' presentation one hour ago?
- **If you have dual-stack in \$SEGMENT (which is probable ;-), be aware of**
  - Attackers combining IPv4 and IPv6 based stuff, to evade IPSs/WAFs.



- **There's some design and configuration approaches that can help to increase the security of IPv6 networks/segments.**
- **These come at the price of added complexity and operational effort.**
- **Choose them wisely.**

- **Think about microsegmentation.**
  - Decreases attack surface and helps steering things.
- **Neighbor cache exhaustion can be mitigated, in many cases with reasonable operational effort.**
- **Host based stuff includes static addresses and disabling local RA processing.**
  - Understand the implications, on the operations level.
- **Layer 2 is the first line of defense.**
  - (Next generation) RA Guard is essential.
- **Do not expect feature parity as for security products, as of Mar 2013.**



# Appendix

# When thinking about security controls...

## ■ Two essential factors must be evaluated:

### ■ *Security benefit*

- “How much do we gain, security-wise?”
- “What’s the risk reduction of this control?”



### ■ Operational feasibility

- “What’s the **operational** effort to do it?”
- Pls note: *opex*, not *capex*, counts!



## ■ For some more discussion on these see also:

- <http://www.insinuator.net/2011/05/evaluating-operational-feasibility/>
- <http://www.insinuator.net/2010/12/security-benefit-operational-impact-or-the-illusion-of-infinite-resources/>

## ■ For each potential control the following points should be taken into account

- How many lines of code/configuration does it need?
  - Can it be implemented by means of templates or scripts? Effort needed for this?
- To what degree does the implementation differ in different scenarios?
  - Per system/subnet/site?
  - Can “the difference” be scripted?
    - Taken from another source (e.g. central database)
    - “Calculated” (e.g. neighboring routers on local link)



- How much additional configuration is needed for previous functionality?
  - E.g. to pass legitimate traffic in case of (“new”) application of ACLs?
- “Business impact” incl. number of associated support/helpdesk calls.
- Cost for deployment of additional hardware/licenses.
  - Cost for their initial procurement is *capex*.

- **IETF Draft Operational Security Considerations:**
  - <http://tools.ietf.org/html/draft-ietf-opsec-v6-01>
- **Design Guidelines for IPv6 Networks**
  - <http://tools.ietf.org/html/draft-matthews-v6ops-design-guidelines-01>
- **Enterprise IPv6 Deployment Guidelines**
  - <http://tools.ietf.org/html/draft-ietf-v6ops-enterprise-incremental-ipv6-01>
- **DC Migration to IPv6**
  - <http://tools.ietf.org/html/draft-lopez-v6ops-dc-ipv6-02>
- **Sicherheitsanforderungen DTAG**
  - <http://www.telekom.com/static/-/155996/4/technische-sicherheitsanforderungen-si>
  - <http://www.telekom.com/verantwortung/sicherheit/155994>



## ■ ICMP Filtering

- <http://tools.ietf.org/html/draft-ietf-opsec-icmp-filtering-03>

## ■ Sample ASA config

- <http://www.cluebyfour.org/ipv6/>



## ■ First Hop Security

- IOS: [http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6\\_fhsec/configuration/15-1sg/ip6f-15-1sg-book.html](http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6_fhsec/configuration/15-1sg/ip6f-15-1sg-book.html)
- IOS XE: [http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6\\_fhsec/configuration/xe-3s/asr1000/ip6f-xe-3s-asr1000-book.html](http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6_fhsec/configuration/xe-3s/asr1000/ip6f-xe-3s-asr1000-book.html)



Bundesamt  
für Sicherheit in der  
Informationstechnik



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

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