IPv6 Kongress 2014

ERNW providing security.

# IPv6 Sicherheit in Enterprise Netzwerken

Enno Rey
erey@ernw.de
Christopher Werny
cwerny@ernw.de

# Who I am

- **Old-school network security guy from**
- **Germany based ERNW GmbH**
  - Independent
  - Deep technical knowledge
  - Structured (assessment) approach
  - Business reasonable recommendations
  - We understand corporate

- **Blog: www.insinuator.net**

- **Conference: www.troopers.de**

# Goal of this Presentation

- **This presentation discusses which specific IPv6 design & configuration approaches might be used for network segments with very high security requirements ("$SEGMENT" from here on), such as DMZs or "secure services areas" or similar networks.**
  - It is an updated (but shortened) version of a talk I gave at the *IPv6 Security Summit* of the *Troopers conference*.
    For the longer version see http://www.insinuator.net/tag/ipv6/ (scroll down to entry from 11 Mar 2013).
- **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.

# Agenda

- **IPv6 Design Goals to Be Kept in Mind & Structural Security Issues of IPv6**
- **Typical Configuration Steps to Address the most Relevant Attacks**
- **Traffic Filtering & Misc**

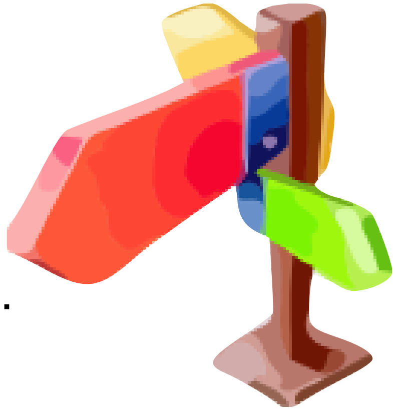
# IPv6 Design Goals to Be Kept in Mind

- **To quite some degree IPv6 was designed for**
  - Large scale systems/networking (*Internet of Things or Sth*)
  - Systems which can act (incl. their IP configuration provisioning) mostly autonomously.
  - Systems which trust their neighbors, or otherwise use crypto.
    - Probably most people here remember the 90s' *crypto-optimism*…

- **So, at least to some degree, it was *not* designed for**
  - Humans' needs for control (as reflected in manual address configuration).
  - Any operations applied on the level of individual systems.
  - Enterprise networks of organizations in general.
    - Which, inherently, strive for optimization of operational effort. Which in turn usually contradicts using much crypto (think key mgmt).
    - I understand that this might hurt, but from an IPv6 perspective "enterprise networks" might be a marginal group…

# “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*?



# Why Do I Tell You all This?

- **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!

# Structurals (Security) Issues of IPv6 (1)

**Network security architectures often rely on**

- **Identification**
  - Be able to identify actors (for security enforcement or audit).
- **Classification**
  - Gather sufficient information to take well-informed decisions.
- **Capabilities**
  - To enhance/assure identification & classification information.
  - To enforce security policy.
- **(Retention of) State**
  - As a supporting tool for classification & enforcement.
- **Simplicity**
  - You all design your networks with RFC 3439 in mind, don’t you?

# Structurals (Security) Issues of IPv6 (2)

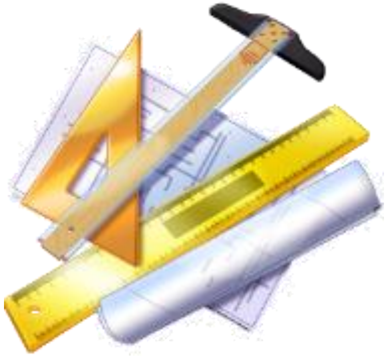
**In an IPv6 world these might become much harder**

- **Identification**
  - There's *Privacy Extensions* (RFC 4941)…
  - Still, we recommend: do *not* fight them. It's tilting at windmills.
- **Capabilities**
  - As of Dec 2013, there's barely *feature parity* on commercial sec. gear.
- **(Retention of) State**
  - Well, with /64 segments, retaining state might become challenging…
- **Simplicity**
  - Are you kidding?
  - Seriously, IPv6 is much more complex (than IPv4) on protocol interaction level. And *dual stack* doesn't help either, complexity-wise.

# Some Design Stuff To be Considered

- **Dedicated /64 for each system might be a good idea**
  - *Might* make filtering easier (or manageable at all).
  - No need to take care of “interface identifier assignment issues” then.
    - Potentially facilitates tracking/auditing/logging.
  - Presumably best control to address all the nasty security problems related to *neighbor discovery* (in the broader sense, as of RFC 4861).
- **Technically this approach**
  - can’t be done in a reasonable way for *dual stack* systems.
  - does not necessarily mean VLAN waste.
    - Think L3 interfaces on top-of-rack switches. Still, this will only work for physical servers. For virtual ones you’ll need VLANs.

# More Design Level Stuff



**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…

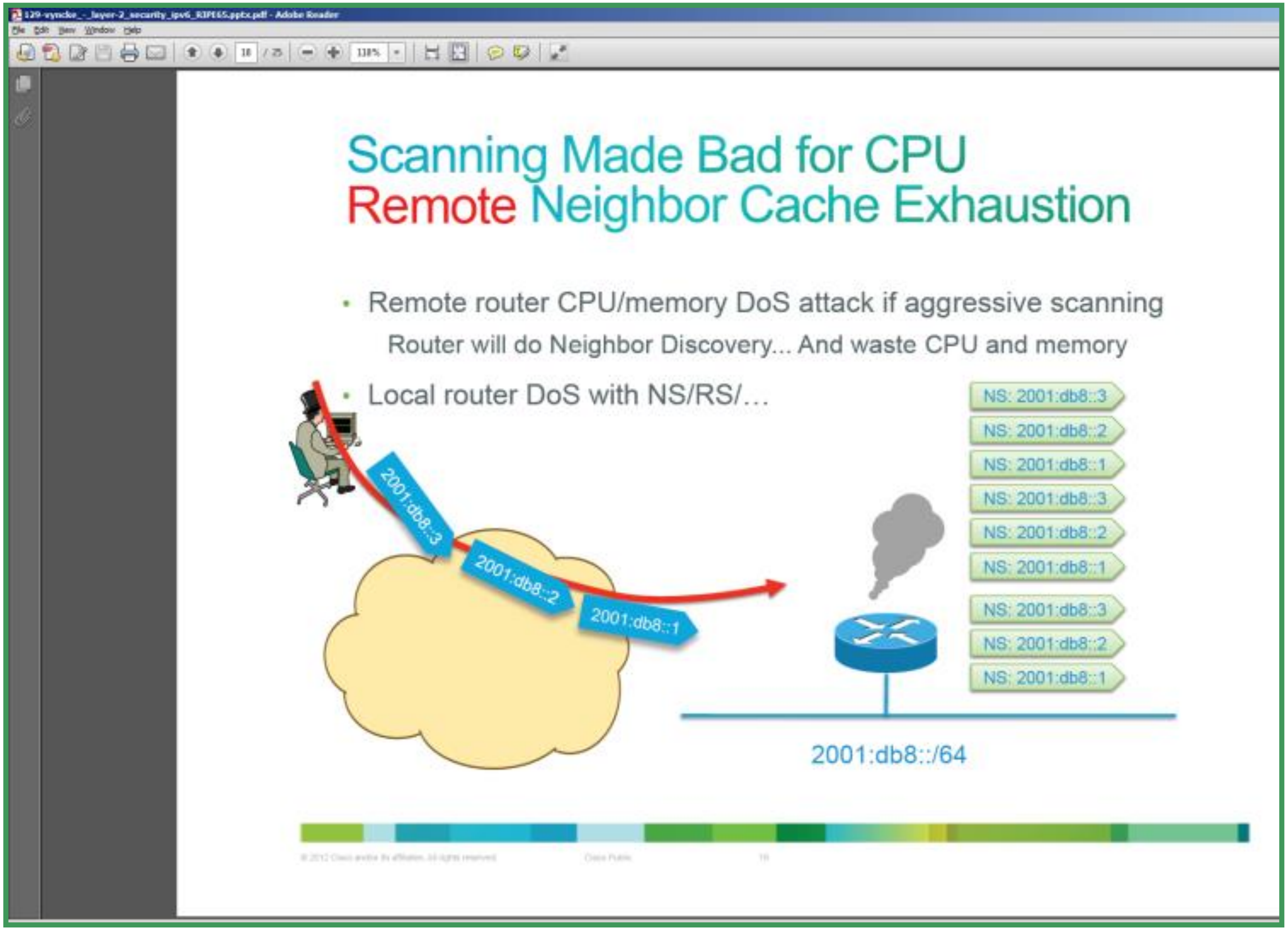
# What Do We Want to Protect $SEGMENT From?

- **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 *ReachableTime* 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 *DELAY_FIRST_PROBE_TIME* seconds. If no message was sent → change state to PROBE. |
| PROBE | A reachability confirmation is actively sought by retransmitting Neighbor Solicitations every *RetransTimer* milliseconds until reachability confirmation is received. |

# Neighbor Cache Exhaustion [this slide stolen from Eric Vyncke]


129-vyncke_-_layer-2_security_ipv6_RIPE65.pptx.pdf - Adobe Reader
File Edit View Window Help
18 / 25
118%

## Scanning Made Bad for CPU Remote Neighbor Cache Exhaustion

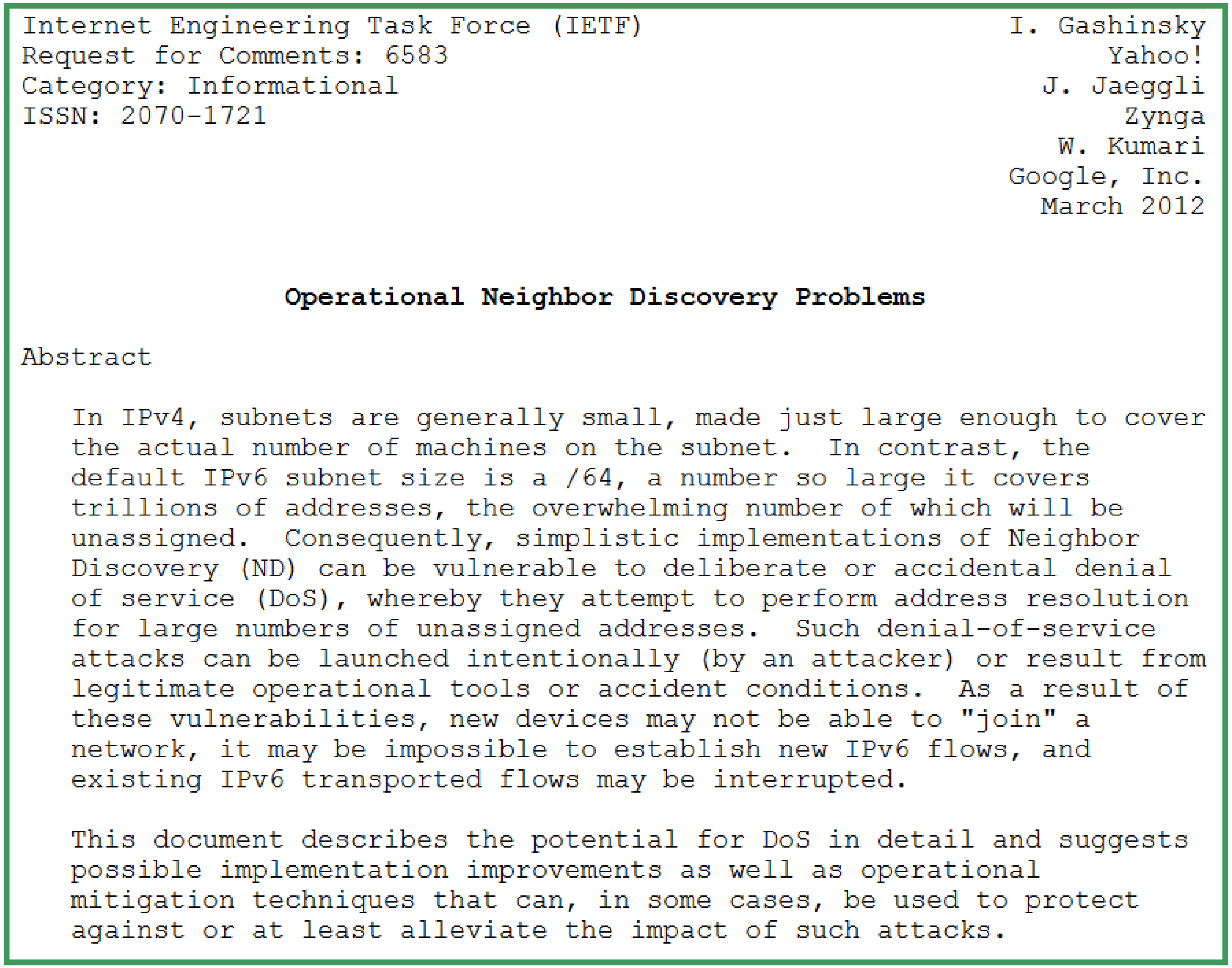
- Remote router CPU/memory DoS attack if aggressive scanning

  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 18

# RFC 6583



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.

# RFC 6583, Potential Controls

- **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.

# RFC 6583, Potential Controls

- ***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).

# RFC 6583, Potential Controls

- **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.”

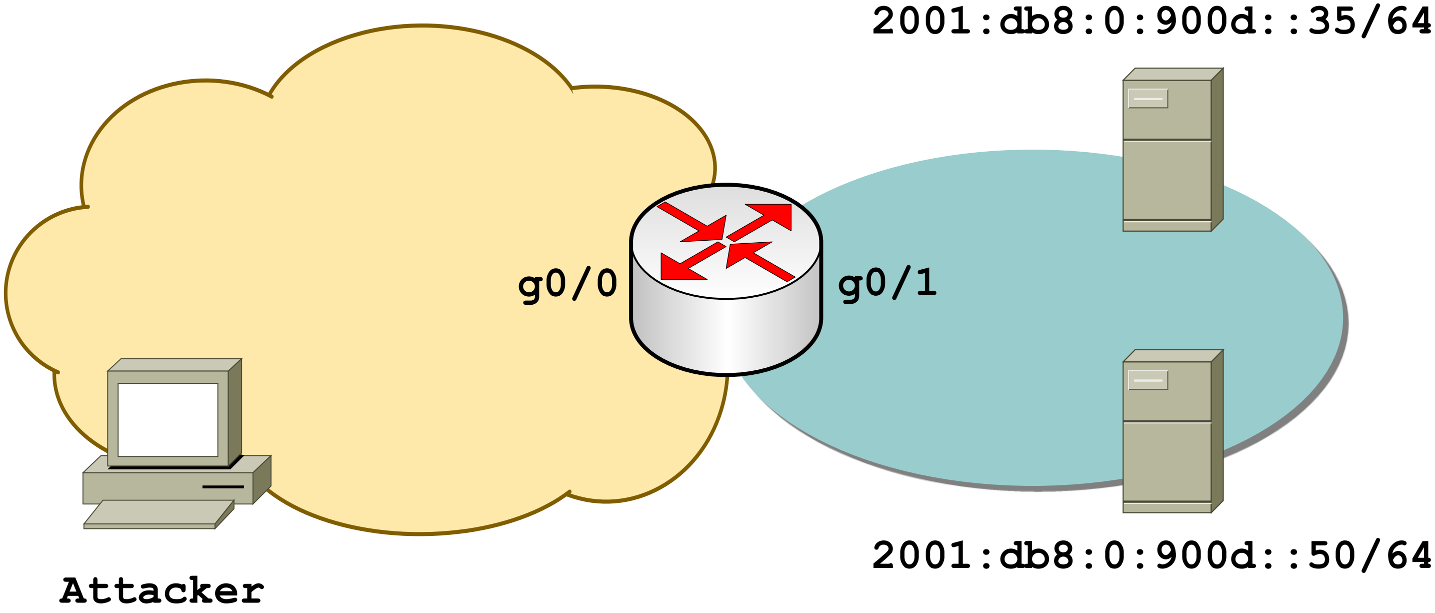
- **Honestly, this approach is so ridiculous both from an architecture and operations perspective, that we’ll not discuss this further.**
  - Anybody remembers the days of `routed` on some Unix systems… and how happy we were to get rid of it?

# RFC 6583, Potential Controls

- **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.
    - After reading some documentation I think this addresses other stuff though.

# NCE, Some Notes from the Lab



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

# NCE, Conclusions from the Lab

- **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/

# Remote Attacks – Scanning

- **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

# Attacks from within $SEGMENT

- **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 Dec 2013)**
  - Due to high complexity of protocols involved 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).

# Did You Get the Message?

- **If you really want to protect a node from (bad) interference with its neighbors, make its segment as small as possible!**

# 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.

# We (still) Don’t Want RAs in $SEGMENT

- **Ok, then there’s three main options:**
  - Suppress emission of RAs on infrastructure level.
    - This is probably not the problem you want to solve ;-)
  - Suppress processing of RAs on hosts.
  - Block forwarding of RAs on infrastructure (L2) level.

# Suppress RA Processing on Hosts

- **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…

# So here We Go

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

- **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 (Dec 2013) not a bullet-proof solution.**
  - Can be circumvented with fragmentation + *extension header* combo.
    - See also http://www.insinuator.net/2011/05/yet-another-update-on-ipv6-security-some-notes-from-the-ipv6-kongress-in-frankfurt/
  - This can *not* be solved by “upgrading firmware”.
    - At some point of time nodes following RFC 6980 will somewhat solve the problem.
- **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
  - Still, currently best protection approach that’s available
    - See also Jim Small’s great presentation at the NA IPv6 Summit 2013
      - http://rmv6tf.org/wp-content/uploads/2013/04/5-IPv6-Attacks-and-Countermeasures-v1.2.pdf.

# RA Guard Config ("old flavor")

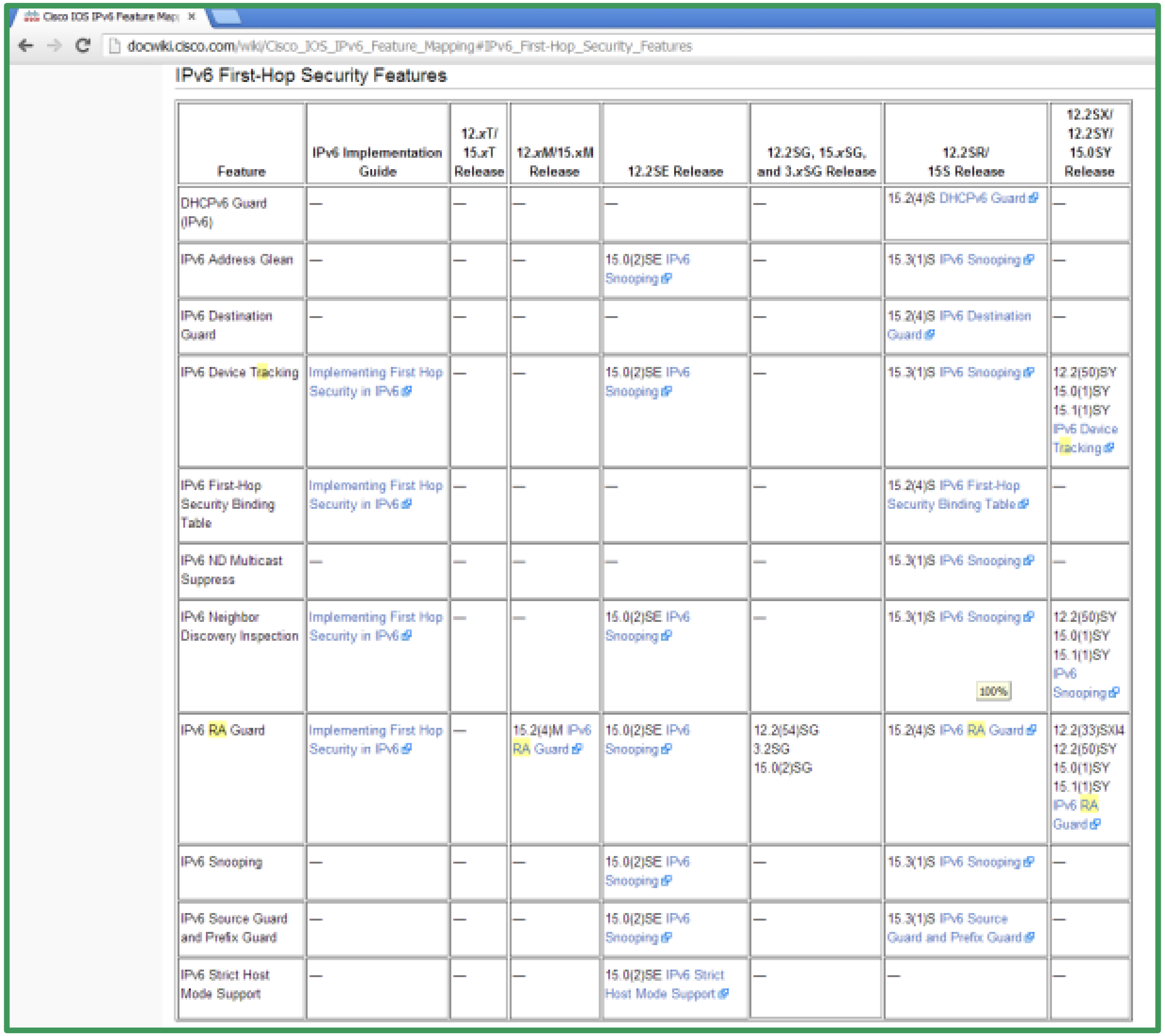
```
Router(config-if)#ipv6 nd ?
  raguard  RA_Guard Configuration Command
Router(config-if)#ipv6 nd raguard ?
  <cr>
Router(config-if)#switchport mode access
Router(config-if)#ipv6 nd raguard
Router(config-if)#exit
Router(config)#exit
```

```
Router# show version
Cisco IOS Software, s3223_rp Software (s3223_rp-IPBASEK9-M),
Version 12.2(33)SXI5, RELEASE SOFTWARE (fc2)
```

# Sample ACL

```
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)


Cisco IOS IPv6 Feature Map ×
docwiki.cisco.com/wiki/Cisco_IOS_IPv6_Feature_Mapping#IPv6_First-Hop_Security_Features
100%

## 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 | 12.2SR/ 15S Release | 12.2SX/ 12.2SY/ 15.0SY Release |
|---|---|---|---|---|---|---|---|
| DHCPv6 Guard (IPv6) | — | — | — | — | — | 15.2(4)S DHCPv6 Guard | — |
| IPv6 Address Glean | — | — | — | 15.0(2)SE IPv6 Snooping | — | 15.3(1)S IPv6 Snooping | — |
| IPv6 Destination Guard | — | — | — | — | — | 15.2(4)S IPv6 Destination Guard | — |
| IPv6 Device Tracking | Implementing First Hop Security in IPv6 | — | — | 15.0(2)SE IPv6 Snooping | — | 15.3(1)S IPv6 Snooping | 12.2(50)SY<br>15.0(1)SY<br>15.1(1)SY<br>IPv6 Device Tracking |
| IPv6 First-Hop Security Binding Table | Implementing First Hop Security in IPv6 | — | — | — | — | 15.2(4)S IPv6 First-Hop Security Binding Table | — |
| IPv6 ND Multicast Suppress | — | — | — | — | — | 15.3(1)S IPv6 Snooping | — |
| IPv6 Neighbor Discovery Inspection | Implementing First Hop Security in IPv6 | — | — | 15.0(2)SE IPv6 Snooping | — | 15.3(1)S IPv6 Snooping | 12.2(50)SY<br>15.0(1)SY<br>15.1(1)SY<br>IPv6 Snooping |
| IPv6 RA Guard | Implementing First Hop Security in IPv6 | — | 15.2(4)M IPv6 RA Guard | 15.0(2)SE IPv6 Snooping | 12.2(54)SG<br>3.2SG<br>15.0(2)SG | 15.2(4)S IPv6 RA Guard | 12.2(33)SXI4<br>12.2(50)SY<br>15.0(1)SY<br>15.1(1)SY<br>IPv6 RA Guard |
| IPv6 Snooping | — | — | — | 15.0(2)SE IPv6 Snooping | — | 15.3(1)S IPv6 Snooping | — |
| IPv6 Source Guard and Prefix Guard | — | — | — | 15.0(2)SE IPv6 Snooping | — | 15.3(1)S IPv6 Source Guard and Prefix Guard | — |
| IPv6 Strict Host Mode Support | — | — | — | 15.0(2)SE IPv6 Strict Host Mode Support | — | — | — |

# RA Guard Availability, Other Vendors

**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

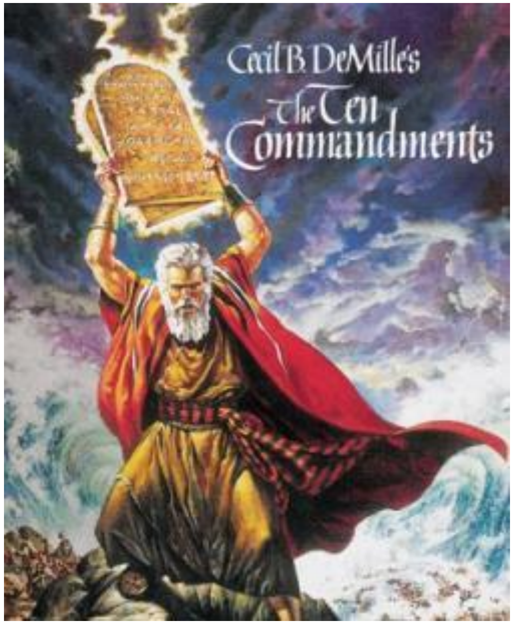
- **Manual configuration**
- **RA Snooping (RA Guard)**
- **Using ACLs**
- **SEcure Neighbor Discovery (SEND)**
- **Router Preference**
- **Relying on Layer 2 Admission Control**
- **Host-Based Packet Filters**
- **Using an "Intelligent" Deprecation Tool**
  - E.g. NDPMon
- **Using Layer 2 Partitioning**

# Traffic Filtering in the IPv6 Age

**There’s three main questions:**

- **Anything specific to allow?**
- **Anything specific to deny?**
- **Can we do it with the same controls & tools as before?**
  - Read: is there *feature parity*?

# Anything Specific to Allow?



**Yes, Sir, there's something:**

**Thou shalt not block**
**ICMPv6 *Packet Too Big* packets!**

# Anything Specific to Deny?

**Well, yes…**

- **There’s a number of severe security issues (incl. IPS evasion scenarios) related to extension headers (EHs) and/or fragmentation.**
  - https://www.ernw.de/download/Advanced%20Attack%20Techniques%20against%20IPv6%20Networks-final.pdf

- **Hence you MUST filter this stuff accordingly!**

# Filtering EHs/Fragmentation

**From our perspective, there are two approaches/policies**

**[and, yes, we understand there's a *tragedy of the commons* here]**

- **STRICT (most "end-user enterprise organizations")**
  - Block all EHs except, maybe, if needed, ESP/AH and/or HBH.
  - Block all fragments.
  - ALWAYS block combination of fragmentation and EHs.
- **LIBERAL (managed service providers)**
  - Block all EHs except HBH, Dest Hdr, Fragment Hdr, ESP/AH.
  - Allow fragments.
  - ALWAYS block combination fragmentation of EHs besides Fragment Hdr.

- **Btw, pls note RFC 7045.**

# Feature Parity

- **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

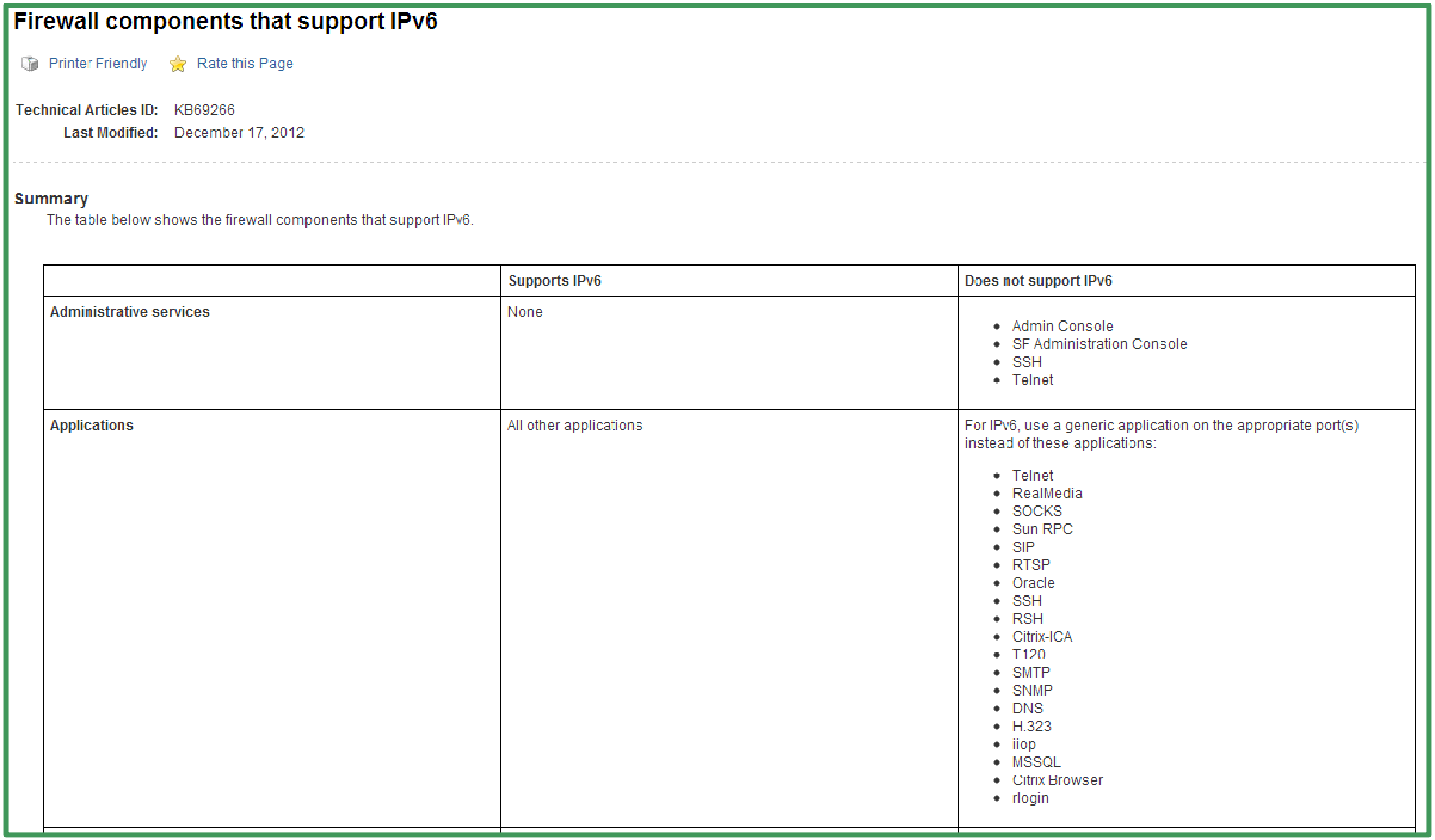
# Feature Parity #2

- **Frankly speaking, do *not* expect full (security) feature parity as of today (Dec 2013).**
  - See, amongst others, our presentation at the *IPv6 Hackers* meeting:

    http://www.ipv6hackers.org/meetings/ipv6-hackers-1/werny-rey-ipv6hackers1-ipv6-security-capabilities.pdf

- ***Feature parity* does not necessarily mean *performance parity*…**

# (Lack of) Feature Parity, Sample



## Firewall components that support IPv6

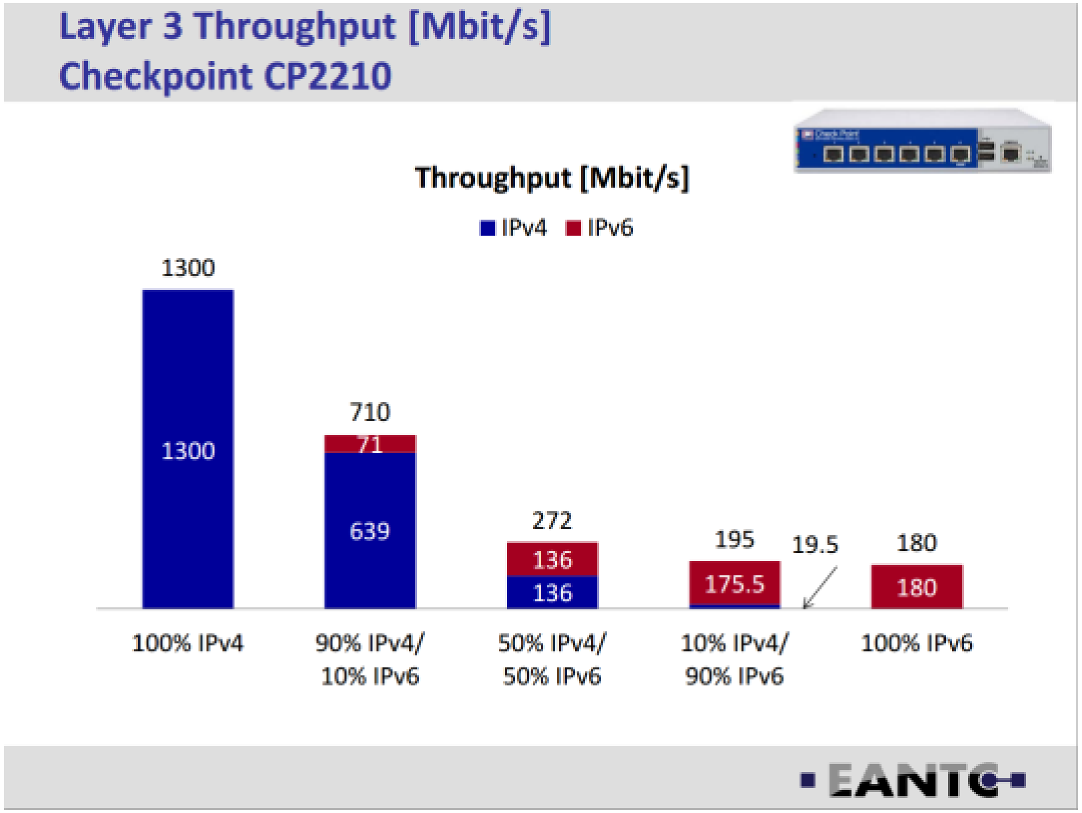
**Technical Articles ID:** KB69266
**Last Modified:** December 17, 2012

**Summary**

The table below shows the firewall components that support IPv6.

| | Supports IPv6 | Does not support IPv6 |
|---|---|---|
| **Administrative services** | None | - Admin Console<br>- SF Administration Console<br>- SSH<br>- Telnet |
| **Applications** | All other applications | For IPv6, use a generic application on the appropriate port(s) instead of these applications:<br>- Telnet<br>- RealMedia<br>- SOCKS<br>- Sun RPC<br>- SIP<br>- RTSP<br>- Oracle<br>- SSH<br>- RSH<br>- Citrix-ICA<br>- T120<br>- SMTP<br>- SNMP<br>- DNS<br>- H.323<br>- iiop<br>- MSSQL<br>- Citrix Browser<br>- rlogin |

# Performance Anyone?



Layer 3 Throughput [Mbit/s]
Checkpoint CP2210

- **This is from Eldad Zack's presentation at the *IPv6 Hackers* meeting at IETF 87.**
  - http://www.ipv6hackers.org/meetings/ipv6-hackers-1

# Conclusions

- **Securing IPv6 networks might follow slightly different rules than securing IPv4 networks.**
- **There's two fundamentally new security issues in IPv6 (*neighbor cache exhaustion* and *rogue router advertisements*). For both problems several mitigation strategies exist, none of which might be optimal for some organizations though.**
- **Filtering of network traffic has some specifics in the IPv6 world. Take care of fragmentation & extension headers!**

# There’s never enough time…

**THANK YOU…**

**...for yours!**

# Questions & Discussion

# 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/

# Evaluating *operational effort*

- **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*.

# Links

- **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

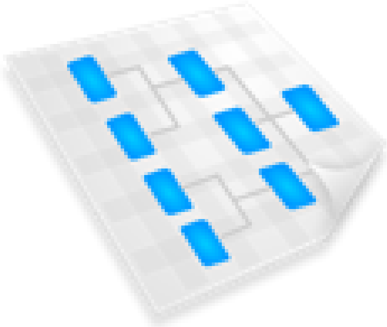
# Links, Filtering

- **ICMP Filtering**
  - http://tools.ietf.org/html/draft-ietf-opsec-icmp-filtering-03

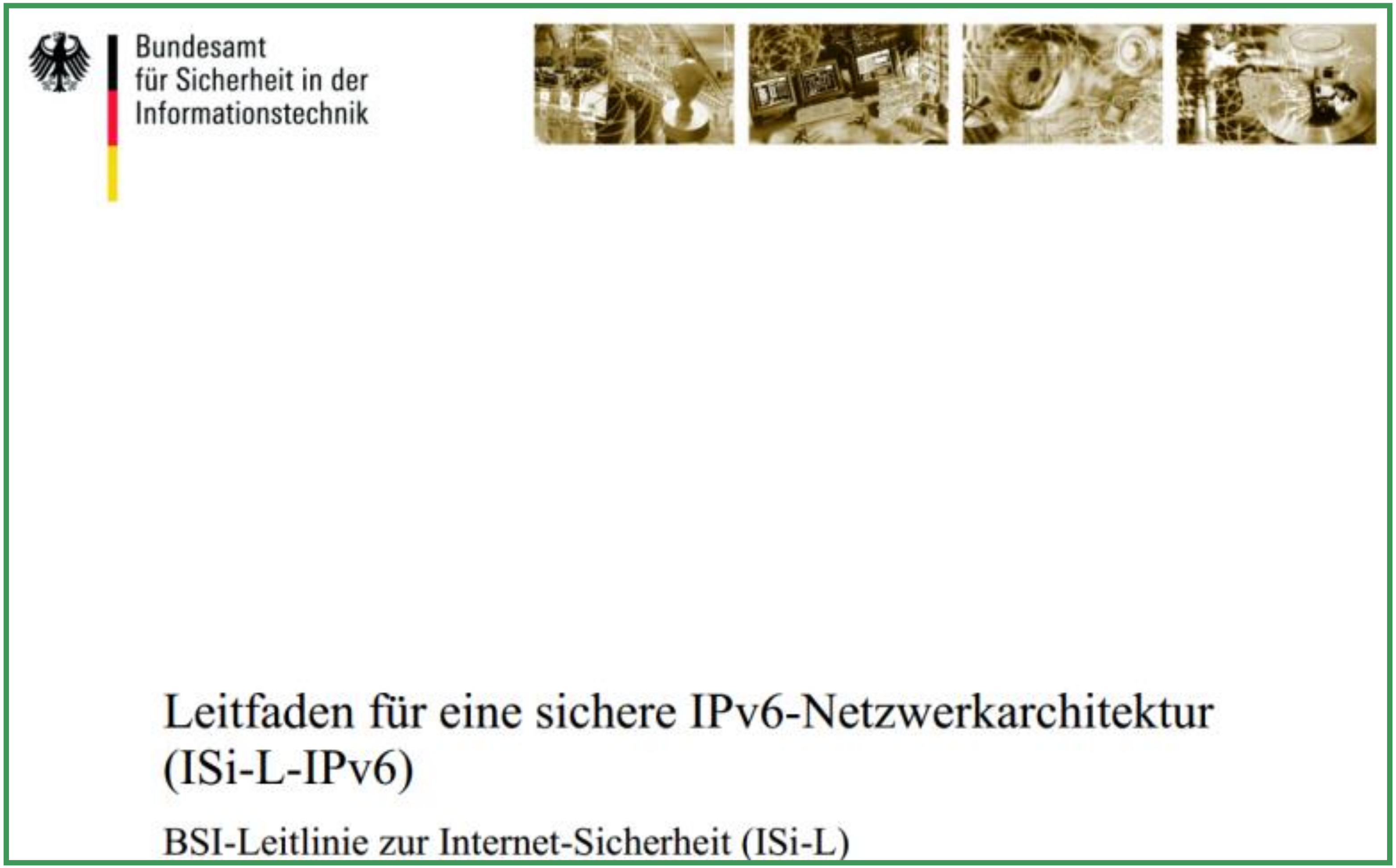
- **Sample ASA config**
  - http://www.cluebyfour.org/ipv6/

# More Links, Cisco

- **First Hop Security**
  - IOS: 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



# BSI



Bundesamt
für Sicherheit in der
Informationstechnik

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

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