

Mapping of Address and Port algorithms

Softwires Interim Beijing

Ole Trøan

2011-09-27



“How many ways can we map an IPv4 address into an IPv6 address?”

Purpose of this discussion:

Consensus on requirements and selection criteria for stateless address and port mapping algorithms

Is one unified mapping algorithm still possible?

Next: Assign weights and evaluate each solution against the criteria

Documents:

Analysis:

<http://tools.ietf.org/html/draft-bsd-softwire-stateless-port-index-analysis-00>

<http://tools.ietf.org/html/draft-despres-softwire-stateless-analysis-tool-00>

Requirements

<http://tools.ietf.org/html/draft-boucadair-softwire-stateless-requirements-00>

<http://tools.ietf.org/html/draft-ietf-softwire-stateless-4v6-motivation-00>

Mapping of Address and Port

“Forwarding function”:

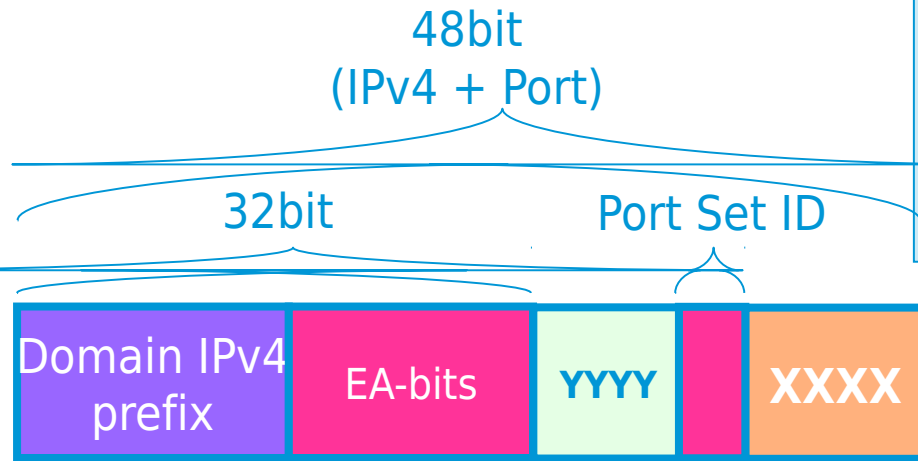
IPv4 address + TCP/UDP port => IPv6 address

Provisioning function:

IPv6 address + algorithm configuration information

=> IPv4 address / prefix + port range

Mapping of Address and Port



- Provisioning based on elements that are:
- Well known
 - Derived from IPv6 provisioning
 - Provisioned via DHCPv6
 - Embedded in address

Proposals:

Algorithm

A1: Port-range

A2: Non contiguous port-range

A3: divi

A4: divi-pd

A5: Port heads

A6: Port heads with mask

A7: 4rd B1

A8: 4rd B2

A9: Prefix

A10: Chen 4v6-pd

Reference

<http://tools.ietf.org/html/draft-boucadair-behave-ipv6-portrange-04>

<http://tools.ietf.org/html/draft-boucadair-behave-ipv6-portrange-04>

<http://tools.ietf.org/html/draft-xli-behave-divi-03>

<http://tools.ietf.org/html/draft-murakami-softwire-4rd-00>

<http://tools.ietf.org/html/draft-murakami-softwire-4rd-01>

<http://tools.ietf.org/html/draft-despres-softwire-4rd-addmapping-00>

<http://tools.ietf.org/html/draft-despres-softwire-4rd-addmapping-01>

E.g. /8 of port space.

<http://www.ietf.org/internet-drafts/draft-chen-softwire-4v6-pd-00.txt>

Evaluation criteria

Criteria/Requirements

{Must have, Nice, Don't care}

C1: Efficient bit representation. Address + Port range in $\leq /56$ Must

C2: Algorithm complexity. (Mapping must be done per packet.) Nice

C3: Minimum / Maximum sharing ratio

C4: Multiple rules Must

C5: Differentiated port ranges Must

C6: Domain prefix of any length (0..128) Nice

C7: Can excludes well known ports <1024 Must

C8: Does not require IPv4 routing imported in IPv6 Must

C9: Old RTP/RTCP compatibility Nice / TBD

C10: UPnP 1.0 friendly Don't care

C11: Port guessing complexity Nice

C12: "Unshared" address/prefix (provisioning) Must TBD

C13: Assign IPv4 address from an SP IPv4 block Must

C14: IPv4 + port id in the IPv6 IID TBD

C15: Port ID at the end Must

Evaluation:

Criteria	Weight	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
C1											
C2											
C3											
C4											
C5											
C6											
C7											
C8											
C9											
C10											
C11											

