# Stateless IPv6 Prefix Delegation for IPv6 enabled networks

*draft-savolainen-stateless-pd-01*

Teemu Savolainen, Jouni Korhonen

**V6OPS at IETF #77**

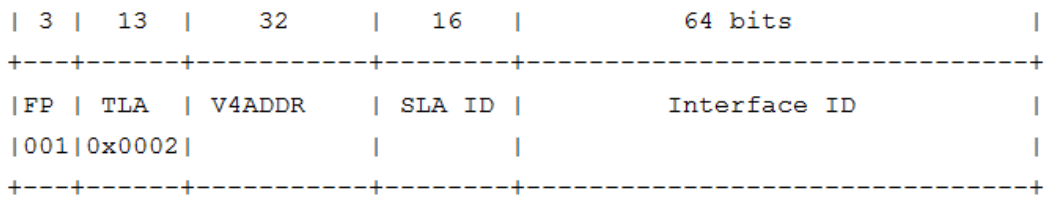
**22-03-2010**

NOKIA

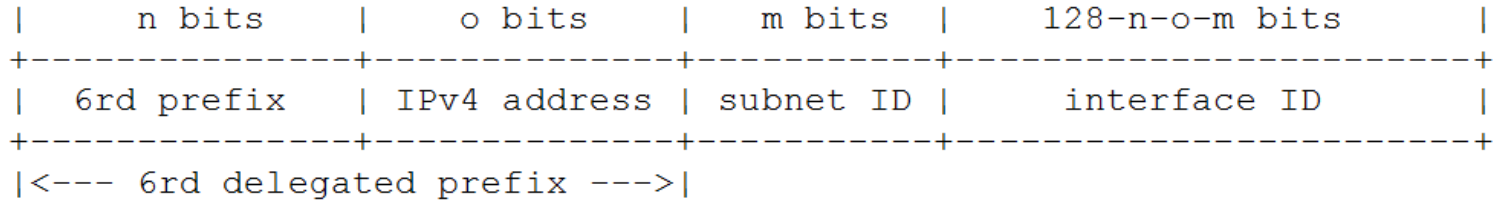
Nokia Siemens Networks

# Background

- In 6to4 and 6RD a site gets an IPv6 prefix by **automatically combining** a 6to4/6RD prefix with an IPv4 address
- 6to4 uses fixed **2002:V4ADDR::/48** prefix (RFC3056)




- 6RD uses **service provider specific prefix** (draft-ietf-softwire-ipv6-6rd-07):




- In 6RD the required information is configured to customer edge device with DHCPv4, IPCP, or similar tools
- IPv6 packets are encapsulated over IPv4

# Background continued

- Routing in 6to4 and 6RD is **based on IPv4 addresses** embedded within IPv6 address
- A relay / border router encapsulates downlink IPv6 packets to IPv4 and sends resulting IPv4 packets to the IPv4 addresses fetched from the packets' destination IPv6 addresses

# Observation and a question

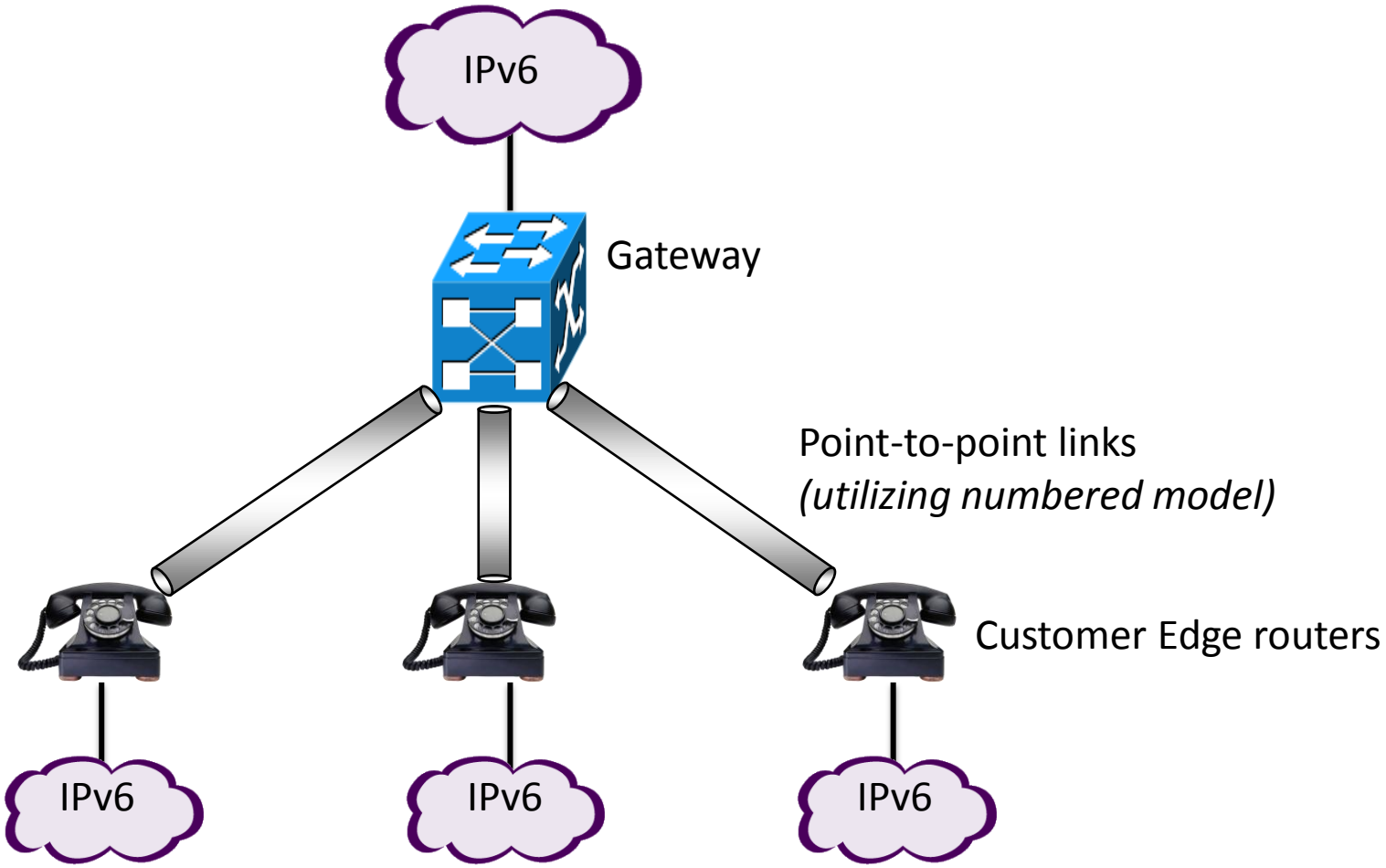
- In 6to4 and 6RD technologies IPv4 addressed host/customer edge gets a “delegated” IPv6 prefix without explicit allocation from the network – in automatic and stateless manner
- The prefix shares faith with the IPv4 address

Would it be nice to have similar technology for IPv6 without dependency to IPv4, and/or without IPv4 encapsulation?

# Overview of the proposal

- Does not touch SLAAC!
  - Sorry for confusion caused by naming..
- Does not propose replacement for DHCPv6 PD!
- In Stateless PD the customer edge router calculates the prefixes it has been allocated based on configuration information received (possibly via stateless DHCPv6). Gateway does the same calculation.
  - In DHCPv6 PD (RFC3633) the network explicitly tells which prefixes are allocated

# Targeted network topology

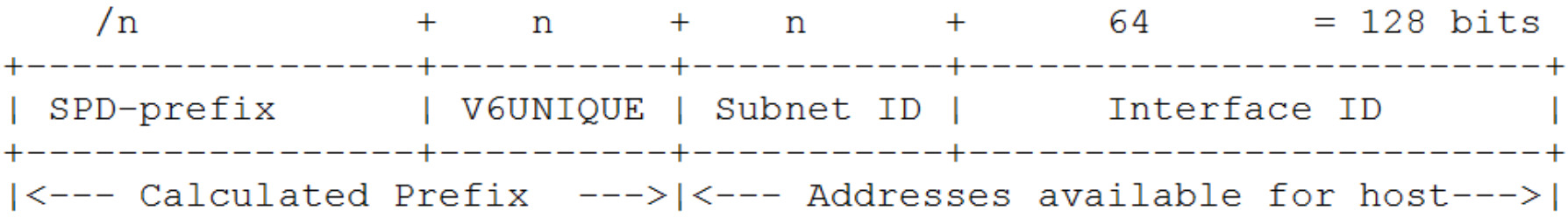


# Alternatives to IPv4 address in delegated prefix construction

- If the locally **unique bits** for prefix calculation are not obtained from IPv4 address, then from where?
- Possibilities could be:
  - Unique /64 prefix available on customer edge devices' uplink interfaces (e.g. 3GPP architecture)
  - Layer 2 identifier (GTP TEID, GRE key)
  - Interface Identifier (in point-to-point links interface identifier can be ensured to be locally unique (e.g. do configure-NAK described in RFC5072))
- **Note!** If customer edge is provisioned with dual-stack connection, an IPv4 address could also be used as described in 6to4/6RD but **without encapsulation**

# Stateless PD addressing proposal

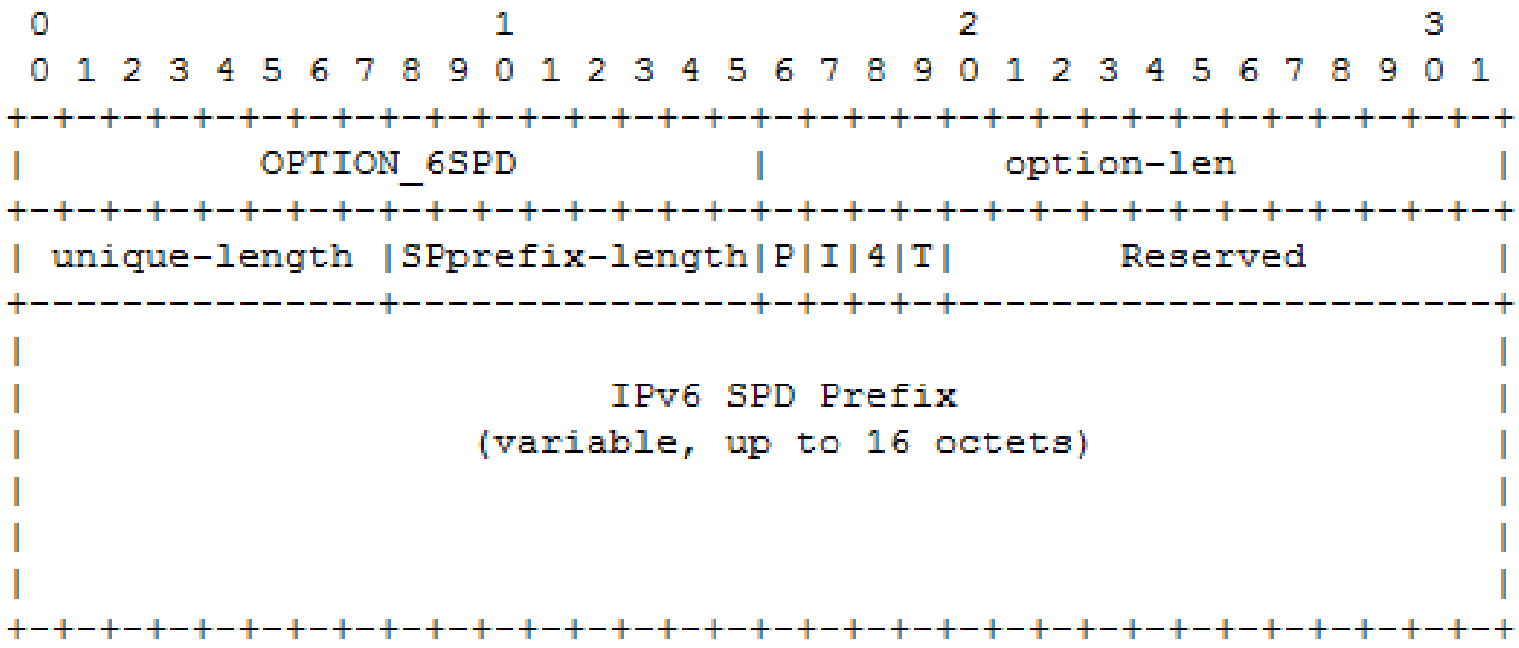
- Essentially the same as in 6RD, but with generic V6UNIQUE instead of an IPv4 address




- **SPD-Prefix**: SP's prefix chosen for PD use
- **V6UNIQUE**: the locally unique bits, which can be taken from IPv4 address as well
- **Subnet ID**: the bits available for router to use

# Stateless PD configuration

- For example, with a new DHCPv6 option:



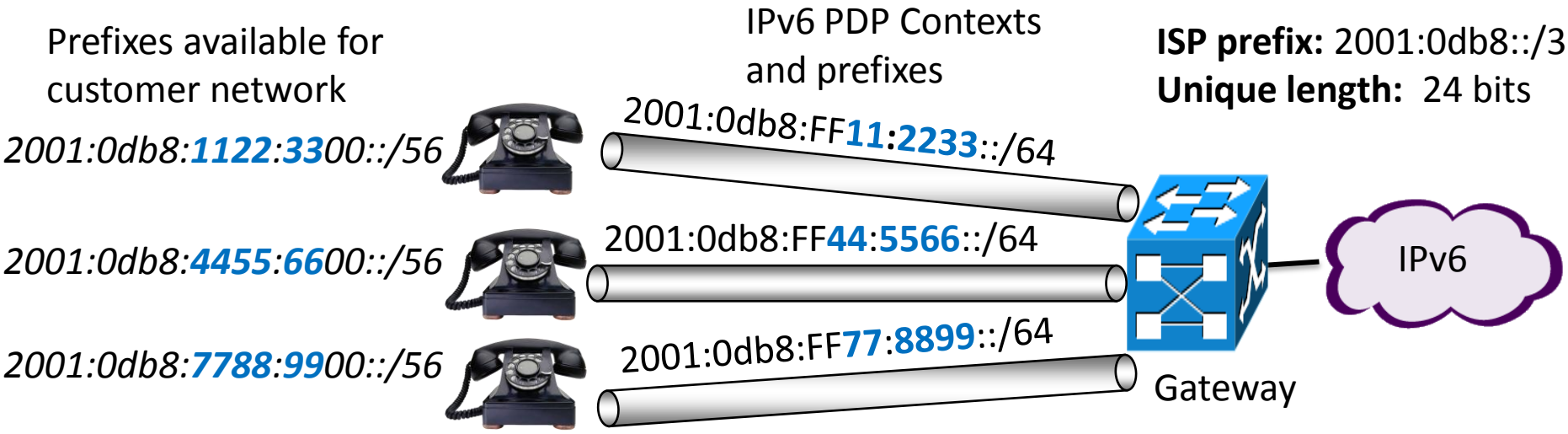

- Length of service provider's prefix, **where the unique bits are taken**, length of unique bits, and the service provider's prefix itself

# Few notes on functionality

- A relay/gateway/border router uses the ”unique bits” fetched from downlink packets destination IPv6 addresses in its routing decisions
- A prefix calculation on customer edge happens as in 6RD, but now instead of IPv4 address the CE device is configured with the information used in the deployment at hand
- Customer edge device can do connectivity test with one or more addresses from the calculated prefix space to verify network really routes correctly
- Best explained via examples on following pages..

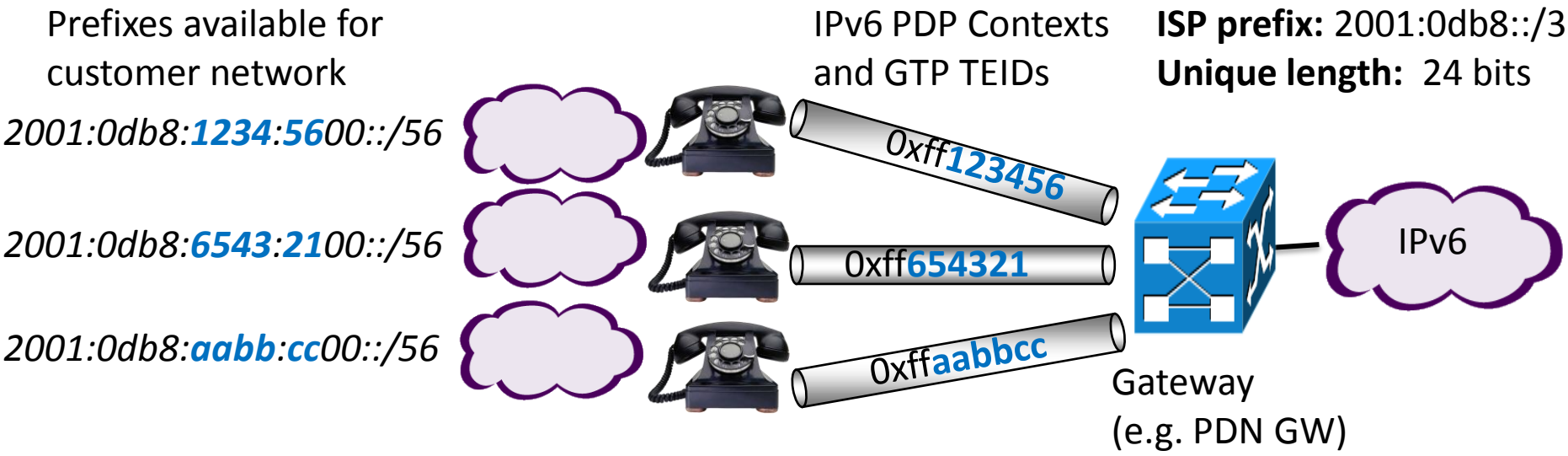
# Example: bits from unique /64 prefixes as unique identifiers

- In 3GPP(2)/WiMAX and elsewhere where ”customer edges” connect to a gateway via point-to-point connections that are allocated with unique /64 prefixes
- Locally unique bits taken from the /64s could be used to calculate delegated prefixes and on the gateway to see to whom downlink packets should be routed to



# Example: GTP TEID as an identifier

- In 3GPP setup ”customer edges”, i.e. ”phones”, connect via point-to-point connections identified uniquely by the gateway by GPRS Tunneling Protocol (GTP) Tunnel Endpoint Identifiers
- The gateway could use GTP TEID embedded into IPv6 addresses as identifier in downlink IPv6 packet forwarding decisions



# Possible benefits

- Network does not need to explicitly allocate prefixes for each customer, to authenticate requests, or keep a note who had what and for how long *(iff information can be calculated later on – depends on used source of unique bits).*
- Full DHCPv6 client and server are not necessary
- Prefix aggregation: as this is new proposal, the customer edge device could support prefix aggregation = allow CE's uplink interface have a prefix taken from the "delegated block"
- No encapsulation

# Possible issues

- Address space is wasted as prefixes are allocated even if not explicitly requested (can be mitigated)
- Buggy software on CE or network side may result in synchronization problems
- Local network renumbering if the unique bits of used unique source changes, e.g. due CE uplink re-establishment
- Multiple prefixes in use on CE's uplink interface – which one is used for prefix calculation?
- Limited applicability when compared to DHCPv6, but as said, no intention to replace DHCPv6 PD
  - Less flexibility, e.g. if customer edge requires multiple different delegated prefixes for different kinds of services
- Risks caused by mix of layers, if L2 identifiers are used
- Generally, being another tool..

# Next Steps..

- Excellent feedback on v6ops list already, more welcome – also if interested on technology☺
- Digging furher if DHCPv6 PD can be essentially utilized in a manner that brings similar benefits *(e.g. lightweight DHCPv6 server on a gateway using stateless means to come up with prefixes to delegate and means to aggregate prefixes)*
- Discussions ongoing in relation to draft-krishnan-intarea-pd-epc regarding applicability in 3GPP