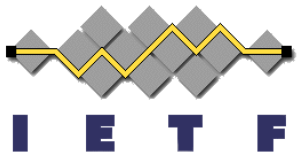
Discovery of a Network-Specific NAT64 Prefix using a Well-Known Name

IETF Behave WG interim meeting
June 21, 2012



draft-ietf-behave-nat64-discovery-heuristic

Teemu Savolainen,
Jouni Korhonen,
Dan Wing

WGLC

- Received five reviews with lots of comments
  - Aaron Yi Ding
  - Simon Perreault
  - Marc Petit-Huguenin
  - Andrew Sullivan
  - Dave Thaler
- Following pages split the comments into categories:
  - Fully open - need be discussed what to do
  - Proposal exist – I have proposal how to handle the comment

WGLC comments - open

- What is best address for NAT64 FQDN?
  - Pref64::/n with padded zeros *(current)*
  - Pref64::WKA/n with well-known IPv4 address in the RFC6052 location used by the DNS64?
- Prioritization of multiple Pref64::/n:
  - Pick one Pref64::/n pseudo-randomly
  - Use all Pref64::/n in a node in round-robin fashion
- Connectivity check default:
  - STUN or ICMPv6?
- Should we have figure showing example MSC?

WGLC comments - open

- If no AAAA response, how often node should repeat the query? Any thoughts?
  - I guess DNS64 server quite rarely suddenly appears on the link, so maybe when node reconnects or learns about new DNS64 servers? (i.e. No ”periodical” learning)
- Appendix question by Dave:”*Comment [DT27]: Is the only difference the extra RRSIGs? If so, then I don’t think we need the duplication.”*
  - Dan?
- Mark Andrews commented on some issues related to intermediate DNS servers. Is this something that needs to be taken into account with heuristic draft, or something to be handled separately?

WGLC comments with fix proposals

- Need for A query?
  - Proposal to add to the section 3: "*In case a node does not receive positive AAAA reply, the node MAY perform A query for the well-known name. If the node receives positive reply to the A query it means the used recursive DNS server is not DNS64 server.*"
- Decide how many, and what, addresses for the well-known name.
  - Proposal for two*: "192.0.0.170 and 192.0.0.171"* (binary 10101010 and 10101011)
- Be more exact with TTL for well-known name.
  - Proposal to section 4:" The authoritative name server for the well-known name SHALL have DNS record Time-To-Live (TTL) set to at least 60 minutes in order to improve effectiveness of DNS caching. The exact TTL value will be determined and tuned based on operational experiences."
- When to refresh cached Pref64::/n?
  - Proposal for ten seconds:"*SHOULD repeat the discovery process ten seconds before the Time-To-Live of the Well-Known Name's synthetic AAAA DNS response expires*"
- 3.1.2 step 5 verification should say instead:
  - " in which case the node MUST check if any of the responses matches the Pref64::/n obtained in step 1"

WGLC comments with fix proposals

- Modified step 2 for node DNSSEC behavior:
  - ”*Send DNS PTR query for the IPv6 address of the translator (for "ipv6.arpa"), using the Pref64::/n from the step 1 and zeroes for the elements after the actual prefix length. CNAME and DNAME results should be followed according to the rules in [RFC1034], [RFC1035], and [RFC6672]. The ultimate response will include one or more NAT64 FQDNs*.”
- Clarified in security considerations that these issues are inherit to DNS64:
  - *“The security considerations follow closely those of RFC 6147. The possible attacks are very similar in the case where attacker controls DNS64 server and returns tampered IPv6 addresses to a node and in the case where attacker causes the node to use tampered Pref64::/n for local address synthesis. The DNSSEC cannot be used to validate responses created by a DNS64 server the node has no trust relationship with. Hence this document does not change the big picture for untrusted network scenarios. If an attacker mangles with Pref64::/n used by a DNS64 server or a node, the traffic generated by the node will be delivered to an altered destination.”*

WGLC comments with fix proposals

- Connectivity check works only with NSP, as with WKP there cannot be domain specific A record.
  - Added to connectivity check section:"*The Pref64::/n -based connectivity check approach works only with NSP, as it is not possible to register A record for each different domain using WKP.*"
- Domain hijacking issue. Added to section 3:
  - *"A DNS reply with one or more AAAA records indicates that the access network is utilizing IPv6 address synthesis. In some scenarios NXDOMAIN and NXRRSET hijacking may result in a false positive. One method to detect such hijacking is to query a FQDN that is known to be invalid (and normally return an empty response or an error response) and see if it returns a valid resource record. However, the as long as the hijacked domain does not result in AAAA responses that contain well-known IPv4 address in any location defined by RFC6052, the response will not disturb Pref64::/n learning procedure"*

WGLC nits from Dave (pdf) and others

- Fixed all nits, comments to following:
  - Clarified where to search for:
    - *“In the case of NSPs, the node SHALL determine the used address format by searching the received IPv6 addresses for the WKN's well-known IPv4 addresses. The node SHALL assume the well-known IPv4 addresses might be found at the locations specified by RFC6052 section 2.2.””*
  - Only one instance of IPv4, shall search for the other:
    - ”*The node MUST ensure a 32-bit well-known IPv4 address value is present only once in an IPv6 address. In case another instance of the value is found inside the IPv6 address, the node SHALL repeat the search with the other well-known IPv4 address*.”

WGLC nits from Dave (pdf) and others

- 3.1.2. renamed to ”*NSSEC Requirements for the node*”
- Changed ”vendor” to ”implementation” in various places

Next steps

- I can upload new I-D early next week
- New WGLC to complete by Vancouver, or to IESG if changes look good?