Privacy Considerations for 6lo Nodes

draft-ietf-6lo-privacy-considerations-01

Dave Thaler <dthaler@microsoft.com>

Reminder: Purpose of Document

- Primary Audience: IP-over-foo spec writers
- Key message: what to cover in Security (Privacy) Considerations

- There are no mandatory protocol requirements in the doc
- But if a spec doesn't address at all an issue covered in this document, it's deficient

General advice (NOT in doc)

Whenever possible:

- IP-over-foo protocols should be designed without limiting which upper-layer protocols should be run over them
- IP-over-foo protocols should be designed without assuming that there's always a firewall separating nodes from the Internet
  - But a firewall is often a way to mitigate some privacy concerns

Updates since -00 per Kerry Lynn's review

- Address scanning not important for link-locals, so need fewer bits of entropy
- Better distinguish between issues relevant to on-link vs off-link attackers
- Reduce emphasis on "46 bits" since that's just an example for links >8 yrs long. Key point is to derive # from max expected link lifetime.
- Minor editorial fixes

<u>No changes (at least in -01) for:</u>

- EUI-48/EUI-64 are trademarked, any reason to note this?
  - Other RFCs don't
- Proposed "privacy IIDs are RECOMMENDED for routable addresses"
  - If lifetime is <1 second (e.g., NFC?), then may be overkill, depending on what "privacy IIDs" means.

Link-Local Addresses (1/2)

- “Specifications should not simply construct an IPv6 interface identifier by padding a short address with a set of other well-known constant bits, unless the link lifetime is guaranteed to be extremely short.”

- Does this apply to link-local addresses?
  - Correlation over time: n/a
  - **Location tracking: relevant if short address is unique enough**
  - Vulnerability fingerprinting: n/a
  - Address scanning: usually n/a

Link-Local Addresses (2/2)

- “Specifications should make sure that an IPv6 address can change over long periods of time. For example, the interface identifier might change each time a device connects to the network (if connections are short), or might change each day (if connections can be long). This is necessary to mitigate correlation over time.”

- Does this apply to link-local addresses?
  - For on-link attackers, requires changing link-**layer** address (too).
  - For off-link attackers, applies when link-locals leak in a higher layer protocol

Using DHCPv6 for temporary addresses (RFC 7824 section 4.1)

[RFC3315] defines a mechanism for a client to request temporary addresses. The idea behind temporary addresses is that a client can request a temporary address for a specific purpose, use it, and then never renew it (i.e., let it expire).

There are a number of serious issues, both related to protocol and its implementations, **that make temporary addresses nearly useless for their original goal**. First, [RFC3315] does not include T1 and T2 renewal timers in IA_TA (a container for temporary addresses). However, in Section 18.1.3, it explicitly mentions that temporary addresses can be renewed. Client implementations may mistakenly renew temporary addresses if they are not careful (i.e., by including the IA_TA with the same IAID in Renew or Rebind requests, rather than a new IAID -- see Section 22.5 of [RFC3315]), thus forfeiting short liveness. [RFC4704] **does not explicitly prohibit servers from updating DNS for assigned temporary addresses**, and there are implementations that can be configured to do that. However, this is not advised as publishing a client's IPv6 address in DNS that is publicly available is a major privacy breach.