60802 D1.3 – Topology Discovery Requirements

6.8 Topology discovery

6.8.1 Topology discovery requirements

...  

Topology verification is an important use case in industrial automation. Checking engineered topologies against discovered topologies is an essential task of the TDE. The check includes the involved IA-stations, ports, and links.

Engineered topologies are created by CAE/CAD systems based on IA-stations’ data sheet provided information. The engineered topologies can then be used by a TDE for topology verification at runtime.

Repair and replacement of an IA-station shall not require an update of the engineered topology for verification. Otherwise the TDE produces a verification error.

Repair and replacement of an IA-station should not require pre-configuration of the replacement IA-station to avoid a TDE topology verification error.
60802 D1.3 - Topology Discovery
LLDP TLV selection

6.8.1.5.2 Chassis ID TLV
...
The Chassis ID subtype field should contain subtype 4, indicating that the Chassis ID field contains a **MAC address** to achieve the Chassis ID’s desired deployment-wide uniqueness.

6.8.1.5.3 Port ID TLV
...
For an IA-station with unique MAC addresses per port, the Port ID subtype field should contain subtype 3, indicating that the Port ID field contains a **MAC address**.

- Chassis IDs and Port IDs are part of the Topology Discovery data.

**BUT:**
- MAC addresses are instance data that is unknown at engineering time and thus, cannot be used for **Topology Verification**!
An approach for **Secure Device Identity** based on manufacturer-supplied content in `ietf-hardware` containers (RFC 8348) was presented at the IEEE 802.1 Mar 22 Plenary ([.../public/docs2022/60802-Pfaff-et-al-Secure-Device-Identity-Profile-0322-v02.pdf](http://)...)

- This data can be used for Topology verification as well.

**Proposal:**

Keep MAC address for LLDP Chassis ID, BUT do not use LLDP Chassis-ID for Topology Verification.

Use `ietf-hardware` data for IA-station identification in Engineering and Topology Discovery/Verification (e.g., `mfg-name`, `model-name`).
60802 D1.3 - Topology Discovery
Port Identification

- Bridge-port data is an augmentation of the IETF interface data (see ieee802-dot1q-bridge YANG model).
- Each IETF interface is identified by an IA-station unique interface name.
- Allowed values for interface names may be restricted ("RFC 8343: A device MAY restrict the allowed values for this leaf, possibly depending on the type of the interface. For system-controlled interfaces, this leaf is the device-specific name of the interface.")

**Proposal:**
Do not use MAC address for LLDP Port ID
Use bridge-port interface name for Port identification in Engineering and Topology Discovery / Verification (i.e., recommend port-id-subtype-type interfaceName in Port ID TLV subclause).
Define some restrictions on bridge-port interface names: e.g.,
- read-only, because System-defined,
- Max length 255 bytes,
- Should match chassis imprinted name,
- ...

March 18, 2022
Josef Dorr, Siemens AG
60802 D1.3 - Topology Discovery Summary

- LLDP Port ID should be based on interface names.
- Topology Verification is not covered sufficiently in 60802 D1.3
- A textual contribution should be provided including:
  - Clarification of Topology Verification requirements,
  - LLDP Port ID TLV with interface names,
  - New subclause describing Topology Verification principles.
chassis-id-type: string of length "1..255"

chassis-id-subtype-type: SYNTAX INTEGER {
  chassisComponent(1), // value of entPhysicalAlias object (RFC 2737) for a chassis component
  interfaceAlias(2),  // value of ifAlias object (RFC 2863) for an interface on the containing chassis
  portComponent(3),   // value of entPhysicalAlias object (RFC 2737) for a port or backplane component
  macAddress(4),      // value of a unicast source address
  networkAddress(5),  // IANA AddressFamilyNumber and the network address value
  interfaceName(6),   // value of ifName object (RFC 2863) for an interface on the containing chassis
  local(7)            // locally defined value
}
60802 D1.3 - Topology Discovery
Port ID type options

... 

port-id-type: string of length "8"

port-id-subtype-type: SYNTAX INTEGER {
  interfaceAlias(1), // value of ifAlias object (RFC 2863)
  portComponent(2), // value of entPhysicalAlias object (RFC 2737) for a port component
  macAddress(3),    // value of a unicast source address
  networkAddress(4), // IANA AddressFamilyNumber and the network address value
  interfaceName(5), // value of ifName object (RFC 2863) for an interface on the containing chassis
  agentCircuitId(6), // agent-local identifier of the circuit (defined in RFC 3046)
  local(7)          // locally defined value
}