July 2023

AUTOMATED ONBOARDING AND TIME-SENSITIVE NETWORK (TSN) FLOW CREATION OF DEVICES IN A TSN NETWORK ENVIRONMENT

Robert Barton

Indermeet Gandhi

Maik Gunter Seewald

AUTOMATED ONBOARDING AND TIME-SENSITIVE NETWORK (TSN) FLOW CREATION OF DEVICES IN A TSN NETWORK ENVIRONMENT

AUTHORS:
Robert Barton
Indermeet Gandhi
Maik Gunter Seewald

ABSTRACT

Techniques described herein provide for automating a time-sensitive network (TSN) configuration using an edge inventory system. In particular, according to techniques described herein, an asset inventory system discovers a TSN-capable device, identifies the requirements of the TSN-capable device, and adds a TSN tag to the asset inventory system that describes the function of the device. The asset inventory system provides instructions to on-board the device to a network as TSN-mandatory and sets the latency bounds for the network based on known or discovered characteristics of the device.

DETAILED DESCRIPTION

Internet of Things (IoT) technologies are driving changes in Industry 4.0 by reshaping system architectures across all domains in manufacturing and other industries. However, changes in Industry 4.0 present new challenges that require various manufacturers (e.g., food and beverage industry, oil and gas, utility providers, etc.) and industrial equipment vendors (e.g., original equipment manufacturers (OEMs) and original design manufacturers (ODMs)) to improve existing real-time compute capabilities and the vectors of time synchronization (time sync) and timeliness across devices. For example, next generation automation controllers must process various workloads, such as video stream and control traffic, in parallel, provide for deep learning capabilities all in one machine, and simultaneously communicate with other controllers in a factory's network in a timely manner.

As per the definition in IEEE P802.1Qcc/P802.1Qdj, it is customary for the Centralized User Configuration (CUC) to obtain stream requirements from end stations (e.g., talker and listeners). The end stations are configured by humans using engineering tools. The CUC performs the following functions – reconciling requirements from talkers and listeners to stream requirements, if necessary; recommending a StreamID group (based on the stream requirement); sending the stream requirements to the Central Network Control (CNC); receiving the end station communication-configuration from the CNC; and distributing the end station communication-configuration to the talkers and listeners. Such a manual interaction of the CUC by humans is error prone and can result in a mismatch of the time-sensitive network (TSN) flows in network devices and switches. The CUC additionally interfaces with a TSN CNC using a user-network interface (UNI) to program the network devices (e.g., switches, etc.). The CNC is responsible for scheduling of TSN devices, identifying the latency of each hop, and setting end-to end-latency bounds.

An automated approach to automate the onboarding of TSN endpoints and TSN flow creation is desirable for increased efficiency of Industrial Internet of Things (IIoT) networks and converged networks (OT and IT), which are less error prone to human interactions than the CUC function described above. Techniques described herein provide for automating the TSN configuration using an edge inventory system. In particular, techniques described herein provide for automated onboarding of Operational Technology (OT) and IIoT devices and their associated policies and configurations using an inventory management system and its associated network sensors in a TSN controller.

Figure 1, below, illustrates an example system for performing a method for automating the TSN configuration using an inventory management system.

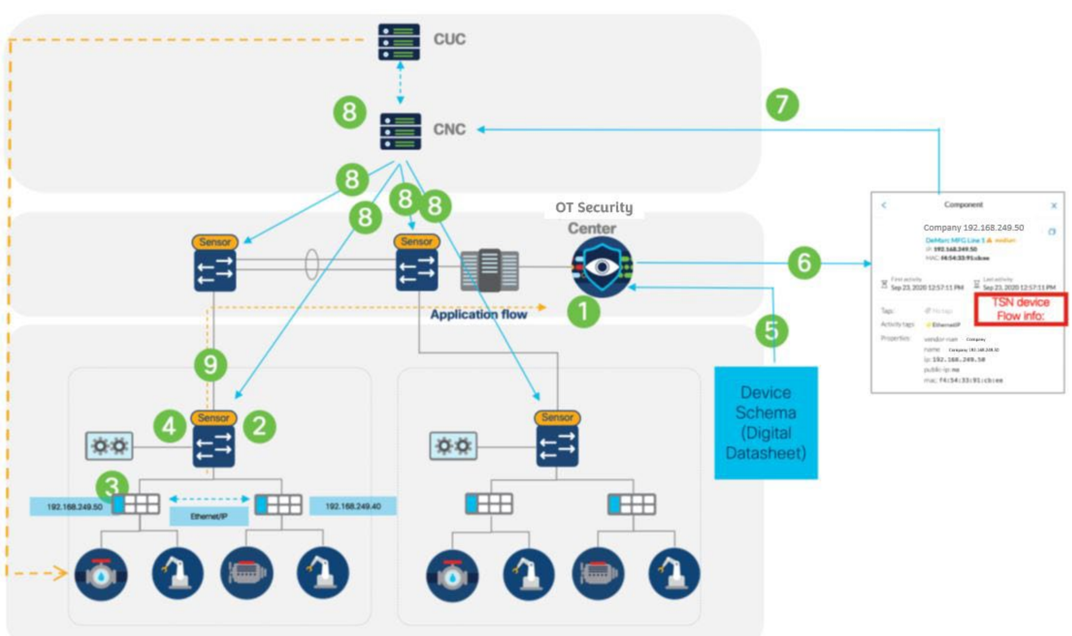

Figure 1: Example System for Automating the TSN Configuration

As illustrated in Figure 1, in a first step, an OT inventory or asset management system is used where sensors are deployed at the edge (e.g., embedded in network devices such as the IE3K switches at the edge or an IR1101 Gateway, etc. or deployed on a Switched Port Analyzer (SPAN) port connected to a compute node, such as the IC3000, etc.). Within the OT network, TSN end stations may be connected and have latency sensitive requirements.

In a second step, the sensors profile traffic coming and going from industrial assets. The profile for the traffic may be passive or active. For a passive profile, visibility is built by listening to network traffic as it passes through the sensor. For an active profile, sensors probe devices to discover their properties (and possibly also query the device for data stored as part of a local Yet Another Next Generation (YANG) data store in the device). As devices are profiled, the sensor uses two types of tags to classify devices: asset tags and activity tags. The asset tag describes what the device is (e.g., what kind of a device it is, make, model, etc.). The activity tag describes what the device is doing on the network (e.g., protocols, etc.).

In a third step, as endpoints are profiled by the sensors, devices that are known to be TSN compatible are identified, and their topologies are discovered. According to techniques described herein, the asset inventory system is enhanced with a TSN-centric Topology Discovery function. For example, if a TSN-capable automation system is detected, the asset inventory system is used to discover not only the TSN devices, but also an application protocol in use and its specific latency bound controls. Once a device is discovered, an OT inventory management system or asset inventory system may validate the TSN requirements through a central policy system, such as an identity services engine, to determine if the device supports TSN capabilities (e.g., obtained from the local YANG datastore of the device or offline from a digital data sheet). If it is confirmed that the device supports TSN capabilities, the device is registered as a TSN-compliant device. For example, if the sensor detects an automation control system connected to a network switch, it will first confirm whether the device supports TSN through a central lookup and it will then mark flows from this device as TSN-mandatory.

According to techniques described herein, the device schema outlining its TSN capabilities and requirements may be available through a Manufacturer Usage Description (MUD) file. In this case, the MUD file would be a centrally hosted digital data sheet. Such a device schema may include a preference for TSN handling, the actual latency bound requirements for communication (e.g., the listeners must receive the communication within 600µs from start of transmission), the maximum size of the Ethernet packet that will be sent (e.g., 100 Bytes), and other dependencies (e.g., whether there is a sequence order to the TSN flows).

In a fourth step, if the device is confirmed to be TSN-mandatory, then the asset tag is enhanced with a new custom attribute (assetTSNflowtypes) used to identify TSN-mandatory support and the attribute is displayed in a user interface. If the sensor detects other devices that do not require or support TSN, then the other devices are marked as non-TSN compatible devices in the OT security center. In another embodiment, the asset inventory system identifies and marks a device as TSN flow mandatory by comparing the identified devices through profiling (active and passive) and referencing a device schema file obtained from the device manufacturer. Such a schema file could be represented as a digital data sheet (as defined by IEC/IEEE 60802) or in the form of the Netconf/Yang model for the devices, which additionally classifies the end device as a TSN-mandatory end device and contains the TSN flow information.

In a fifth step, once the asset inventory system filters TSN-compatible devices and the network switches, it exchanges this information with the TSN CUC along with the network topology of the network switches and the end devices, which the asset inventory system has learned. This may be done automatically through an application programming interface (API) connection. Thus, the CUC is programmed automatically as devices supporting TSN are discovered in the network – in a sense, automatically on-boarding TSN automation requirements.

In a sixth step, the TSN controller (the CNC) determines the routes, computes the schedule, and programs the network switches with the TSN flows for each of the devices. The TSN controller assigns a unique identifier for each TSN flow. A typical TSN flow consists of unique identifiers for each TSN flow (e.g., destination Media Access Control (MAC) address, Virtual Local Area Network (VLAN), class of service), a start and end of a transmit window at each hop, a start and end of a receive window at each hop (listeners and bridges), end-to-end latency as computed, and additional credit based shaper.

In a seventh step, network sensors continue to monitor the activity of the TSN flows and report the metadata (e.g., unique identifier, latency at each hop, etc.) related to TSN flows to the asset inventory system to monitor whether the devices are receiving the TSN flows. For this, activity tags, as reported by the sensor, are enhanced to include additional TSN flow information (e.g., TSN unique identifier, latency at each hop, etc.). The asset inventory system may be enhanced to offer digital twin capability where the TSN controller programs the switches with TSN flows and triggers the asset inventory system to validate whether the flows are correctly installed.

In summary, techniques described herein provide for an asset inventory system to discover TSN-capable devices. After confirming a device is TSN capable and learning its requirements, a TSN tag is added to the asset inventory system to help describe the function of the device. The asset inventory system then communicates to the TSN controller (the CNC) to on-board the device to a network as TSN-mandatory and sets the latency bounds for the network based on the known or discovered device characteristics.