June 2020

INDUSTRIAL ASSET INVENTORY AND VULNERABILITY DETECTION IN OPERATION TECHNOLOGY NETWORKS

Mariusz Kaźmierski

INDUSTRIAL ASSET INVENTORY AND VULNERABILITY DETECTION IN OPERATION TECHNOLOGY NETWORKS

AUTHOR:
Mariusz Kaźmierski

ABSTRACT

The techniques presented herein provide efficient, scalable, and comprehensive industrial asset inventory and vulnerability detection in operation technology (OT) networks. The techniques achieve this by adaptively and dynamically chaining traffic-monitoring methods through an OT network. Additionally, the techniques automatically and efficiently deploy sensors in an OT network to support the traffic-monitoring chaining and efficiently utilize resources in the OT network. These techniques may resolve vulnerability detection issues and inventory tracking and classification issues that are commonly encountered in OT networks, which that often utilize network elements without traffic sensing features.

DETAILED DESCRIPTION

The trend of incorporating smart technology into traditional manufacturing and industrial platforms to is often referred to as the fourth industrial revolution (i.e., “Industry 4.0”). Industry 4.0 will, at least in theory, result in ubiquitous smart factories and a wide ranging industrial Internet of Things (IoT). However, in operation technology (OT) networks, assets are rarely fully visible and properly classified/inventoried, which presents a challenge that must be overcome to fully realize Industry 4.0.

In standard information technology (IT) networks, classification and vulnerability detection tasks may be relatively straightforward; however, the techniques used in IT networks are often not applicable to OT networks. The main reason for this is because IT networks typically have: (1) endpoints, such as computers, smartphones, printers, etc., that are connected to a switch that can host advanced feature-sets; (2) traffic that mostly flows to or from resources/servers located in remote data centers, the Cloud, etc. (e.g., traffic

flowing "north-south"); and (3) continuous network monitoring deployed for various assurance purposes. The general north-south nature of traffic (e.g., item (2)) allows traffic classification to be run at a distribution/core layer and, collectively, the three aforementioned features may make classification and vulnerability detection tasks relatively straightforward for an IT network.

By comparison, OT networks typically have: (1) endpoints, such as Programmable Logic Controllers (PLCs), Remote Terminal Units (RTUs), Human Machine Interfaces (HMIs), etc., that are connected to low-end switches (including unmanaged devices); (2) traffic flowing between endpoint controllers (e.g., flowing "east-west"); and (3) too much traffic to allow for continuous copying of traffic between all endpoints. These three characteristics may render classification and vulnerability detection tasks more difficult on OT networks than IT networks. In particular, since the endpoints are typically connected to low-end switches, the switches are unlikely to provide features that allow for full endpoint sensing and traffic monitoring. Moreover, the general east-west flow of traffic may prevent traffic classification from being run natively on a distribution/core layer (as traffic never gets there). Finally, without continuously copying traffic, it may not be possible to continuously capture all of the traffic flowing between endpoints.

In view of the foregoing, passive network traffic sensing features (e.g., SPAN and/or NetFlow features) are sometimes enabled on IoT endpoints (e.g., on PLCs, RTUs, HMIs, etc.) to try to classify OT network assets and/or to try to detect OT network vulnerabilities. For example, this may be achieved with hardware sensors and/or software sensors/apps deployed on endpoints. However, each of these solutions has downsides. For example, hardware sensors must be physically inserted into the network (usually as pass-through devices or as a device towards redirected traffic), which is labor/cost intensive, especially for large OT networks. Moreover, a large number of hardware sensors may be required to holistically cover an OT network and to capture all communication patterns. Meanwhile, software sensors require compute and storage components that may not be readily available on an existing OT network. Additionally, many industrial switches do not allow software switches to leverage existing network infrastructure and some cannot support hardware or software sensors. For example, only some switches have traffic sensing features, such as Encapsulated Remote SPAN (ERSPAN) and Flexible NetFlow.

In view of these issues, the techniques presented provide efficient industrial asset inventory and vulnerability detection in an OT network. The techniques adaptively and dynamically chain traffic-monitoring features while automatically deploying sensors in strategic locations within the OT network. More specifically, the proposed techniques chain various traffic monitoring techniques to eliminate hardware and software constraints in the network. For example, the techniques may create one or more of the following chains:

RSPAN --> ERSPAN --> Sensor
RSPAN --> ERSPAN --> Flexible Netflow --> Sensor
ERSPAN --> SD-AVC --> Sensor

This configuration could be deployed automatically by a network orchestrator.

The techniques are dynamic because they can take into account events occurring in the network (e.g. link up/down events where endpoints are connected to, results from a reasoner about the state of asset classification, etc.). Observed events can be used to start and stop monitoring of a given traffic flow, thereby reducing an overall impact on the network (network traffic) and sensor utilization (compute, storage resources). Meanwhile, the techniques are adaptive because the techniques may determine and deploy sensors in on selected network elements to optimize the OT network and minimize costs.

As a more specific example, at least a portion of the techniques are now described in connection with the relatively simple OT network illustrated in Figure 1 below:

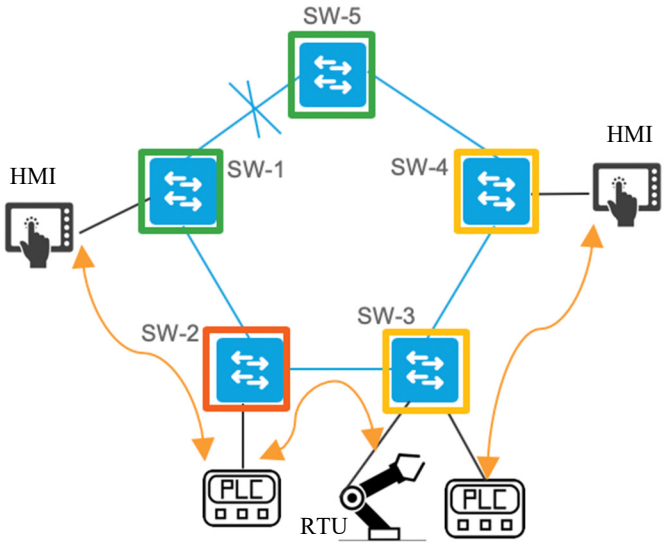

Figure 1: Example Traffic flows in an OT network

In this OT network, there are a number of endpoints (e.g., PLCs, HMIs, and an RTU) connected to network elements (SW-1, SW-2, SW-3, SW-4, and SW-5). SW-1 and SW-5 (marked in green) are high-end switches and fully support sensor capabilities (e.g., switches with a software sensor). Meanwhile, SW-2 (red) is an unmanaged switch and SW-3 and SW-4 (yellow) are low-end switches, and none of SW-2, SW-3, and SW-4 support sensor capabilities.

In the Figure 1 OT network, the techniques presented herein can dynamically chain various traffic monitoring techniques as follows. First, a network orchestrator can build a topology of the network and monitor link utilization on OT devices, as well as events coming on these devices (e.g., in any manner now known or developed hereafter). Then, for each endpoint, the network orchestrator can:

I) calculate the closest sensor to the endpoint that can monitor the endpoint, insofar as "closest" may be determined based on a compound metric that may include sensor utilization, link utilization, path length to the sensor, etc.;

II) dynamically configure and deploy a chain of traffic monitoring features; and

III) monitor the state of the endpoint (e.g., based on replicated traffic or a captured traffic summary) for industrial asset inventory and vulnerability detection.

As an example of a chain of traffic monitoring features, a network element to which the endpoint is connected may be configured to implement remote SPAN (RSPAN), a path towards the sensor may be configured to implement RSPAN on a virtual local area network (VLAN), and a network element hosting a sensor (e.g., software or hardware) can be configured to implement RSPAN, either alone or in combination with additional traffic sensing features (e.g., Flexible NetFlow). As soon as a classification and/or detection process is completed, the network elements can be reconfigured to remove the added features (e.g., RSPAN) from the network and reduce overall resource consumption in the network and/or on the sensor.

If, after an initial configuration of endpoints, a new endpoint joins the session, the network orchestrator can detect the endpoint (e.g. via Simple Network Management Protocol (SNMP) interface-up trap message or other known techniques) and iterate the techniques laid out above.

Figure 2 illustrates the OT network of Figure 1 with traffic sensing features (e.g., RSPAN sessions) deployed for industrial asset inventory and vulnerability detection. As is shown, in this example, SW-3 enables traffic sensing features (e.g., RSPAN) for two endpoints and sends data through RSPAN VLAN100 to SW-1, which has software sensor capabilities.

However, Figure 2 is just an example and, in other instances, the deployment of traffic sensing features could differ. For example, if network elements are unmanaged and/or do not support traffic sensing features, monitoring can be enabled on the uplinks of unmanaged switches. Figure 3 illustrates an example use-case where both uplinks of unmanaged SW-2 are monitored by remote sensors by mirroring traffic from unmanaged switch uplinks.

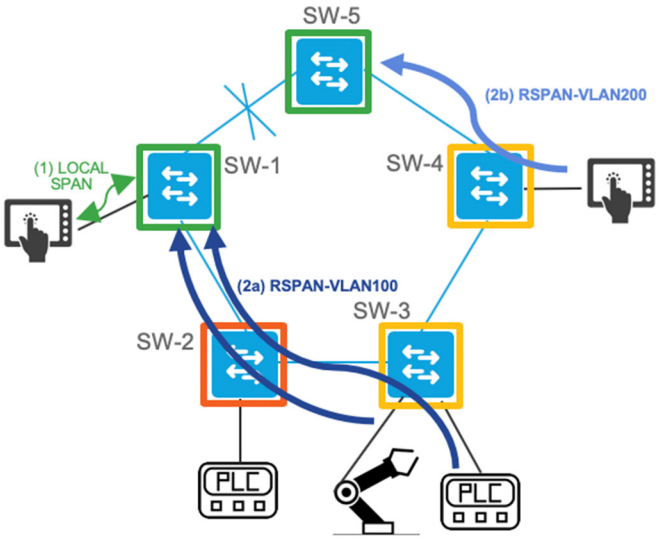

Figure 2: Sensor placement and traffic redirection sessions (use-case: low-end switches)

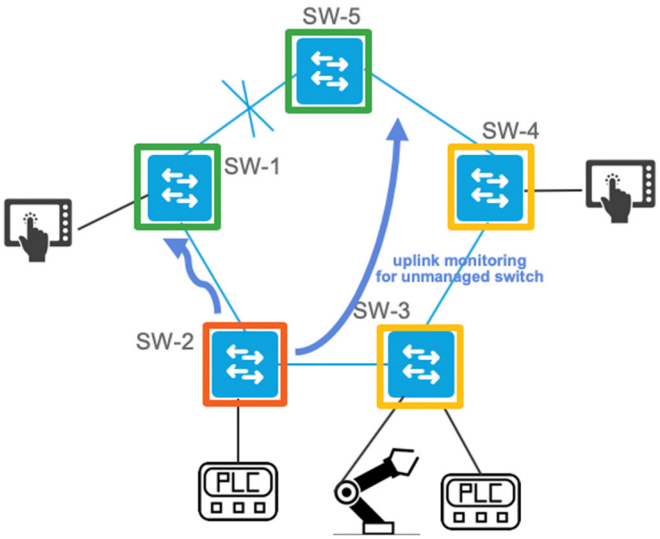

Figure 3: Sensor placement and traffic redirection sessions (use-case: unmanaged switches)

Next, as mentioned above, the techniques presented herein can automatically deploy sensors on selected network elements. Specifically, based on the OT network topology, a network orchestrator can dynamically scale-out/scale-in sensors to limit amount of traffic redirection from the endpoints in the OT network. This automatic deployment will optimize the usage of compute, network, and/or storage resources in the OT network. That is, automatic deployment of sensors can be achieved with an adaptive mechanism that can insert sensors on (or remove sensors from) network elements in an OT

network to provide industrial asset inventory and vulnerability detection while optimizing required resources needed for this task.

According to one example, the mechanism may be achieved in accordance with an algorithm that, for a given graph network topology, determines a list of vertices in the graph for sensor deployment that will optimize resource functionality. The determination is made based on knowledge of where sensors are positioned in the topology, knowledge of which network elements can support sensor functionality, and knowledge of which network elements in the topology need to be monitored.

Put another way, a list of vertices in a graph T, represented as L'={l1', l2', ..., lj'}, can be calculated to optimize resource given function: f(T, L, L', S, M): L' = min f(T, L, L', S, M), where the aforementioned variables are:

- T represents the network topology and may represented by T = (V, E), where V represents all vertices (e.g., network elements) in the network and E represents edges in the network (e.g., weighted links based on link bandwidth / averaged link utilization;
- L represents a list of sensors in graph T which are working at given time and may be represented as L={l1, l2, ..., ln}, where l1,l2,...ln belong to vertices V in graph T;
- S represents a list of network elements that are capable of sensor functionality and may be represented as S={(s1,c1),(s2, c2),..., (sm, cm)}, where s1,s2,...sm belong to vertices V in graph T and c1,c2,...,cm are network-element capabilities and/or constraints (e.g. memory/central processing unit/ storage / available licenses, etc.); and
- M represents a list of network-elements that needs to be monitored is represented by M={(m1, c1'),(m2, c2'), ..., (mi, ci'))}, where m1,m2,..,mi belong to vertices V in graph T that needs to be monitored with under constraints c1', c2',..., ci'.

As an example, the algorithm could use shortest-path first, where overall distance between set M (network-elements that needs to be monitored) and set L' (network-elements where sensor can be deployed) is minimal. As is shown in Figure 4, with such an algorithm, initially for: T={}, L={all network-elements marked in red, such as M1}, the optimization function will give as a result: L'={SW-4, SW-5, SW-6} that can be used by the network orchestrator to deploy sensors at network elements SW-4, SW-5, and SW-6.

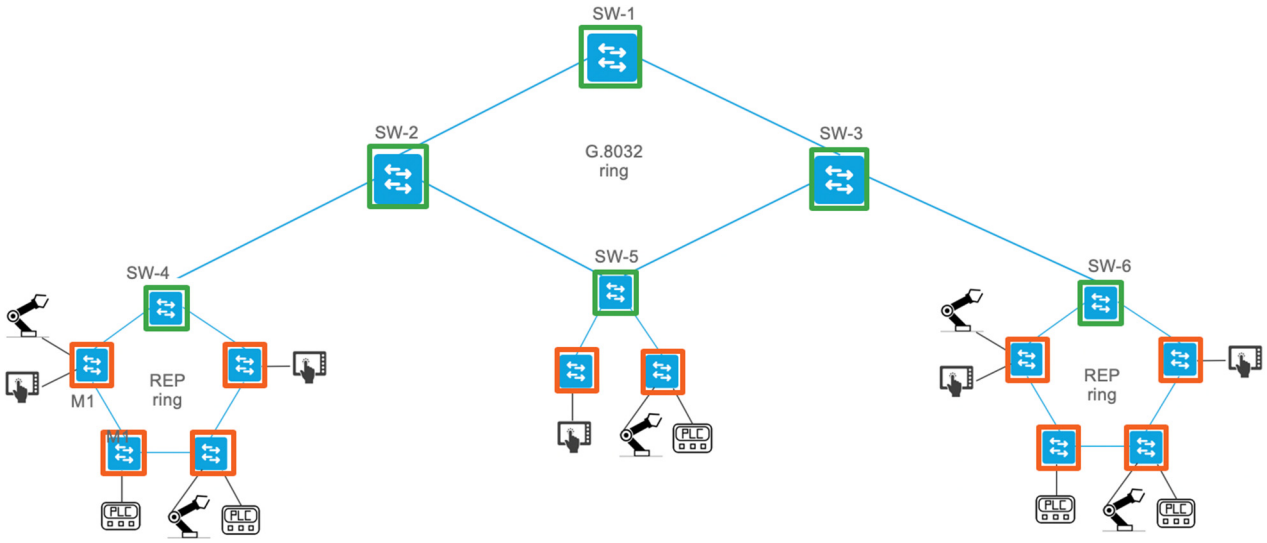

Figure 4: OT network / Automatic sensor deployment on selected network elements

Additionally or alternatively, the algorithm can could attempt minimize sensor deployment (i.e., instead of or in addition to determining a deployment based on shortest path-first). In such a case, the final sensor placement could be: L'={SW-1} or L'={SW-5}, assuming equal cost link bandwidth.

In sum, the techniques described herein provide efficient, scalable, and comprehensive industrial asset inventory and vulnerability detection in operation technology (OT) networks. The techniques achieve this by adaptively and dynamically chaining traffic-monitoring methods through an OT network. Additionally, the techniques automatically and efficiently deploy sensors in an OT network to support the traffic-monitoring chaining and efficiently utilize resources in the OT network. These techniques may resolve vulnerability detection issues and inventory tracking and classification issues that are commonly encountered in OT networks, which that often utilize network elements without traffic sensing features.