vBump: Securing Ethernet-based Industrial Control System Networks with VLAN-based Traffic Aggregation

Nils Ole Tippenhauer
CISPA Helmholtz Center
for Information Security, Germany
tippenhauer@cispa.de

Binbin Chen
Singapore University of Technology and Design, Singapore
binbin_chen@sutd.edu.sg

Daisuke Mashima
Advanced Digital Sciences Center, Singapore
daisuke.m@adsc-create.edu.sg

David M. Nicol
University of Illinois at Urbana Champaign, USA
dmnicol@illinois.edu

ABSTRACT

Bump-in-the-wire (bump) devices can be used to protect critical endpoints in Industrial Control System (ICS) networks. However, bump devices cannot be used to authenticate incoming broadcast traffic, are complex to manage, and one bump is needed per host.

In this work, we propose a virtual bump-like solution called vBump, which allows to insert virtual bumps in front of Ethernet-based legacy ICS devices. The vBumps can be used to limit traffic to whitelisted destinations, inspect all traffic on or above Link-layer like a centralized intrusion detection systems (or monitoring systems), or even police the traffic like a centralized intrusion prevention systems. In particular, this also allows the network to apply fine-grained control on traffic between nodes that need to be in the same Link-layer broadcast domain. Compared to traditional bumps, vBumps do not require any changes in physical network topology, and the central server's global view allows for more informed decision, with less computational constraints. We implement the system in a high-fidelity ICS testbed, and demonstrate its capabilities to support even time-critical protection control traffic in smart grids. Our system can handle traffic rates of 150Mbps with one-way delay of ≈ 1ms.

CCS CONCEPTS

• **Security and privacy → Network security**.

KEYWORDS

ICS security; Industrial Control System; VLAN; Traffic Filtering

ACM Reference Format:
Nils Ole Tippenhauer, Binbin Chen, Daisuke Mashima, and David M. Nicol. 2021. vBump: Securing Ethernet-based Industrial Control System Networks with VLAN-based Traffic Aggregation . In *Proceedings of the 2nd Workshop on CPSIoT Security and Privacy (CPSIoTSec '21), November 15, 2021, Virtual Event, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3462633.3483983

1 INTRODUCTION

Industrial Control Systems (ICS) are vulnerable to cyber-attacks. In recent years there are reports of malware attacks on Middle Eastern energy industry involved in oil drilling, and reports of Russian penetration into the control room of U.S. electric utilities [1]. ICS are designed for safety and for operational efficiency—they are not typically designed with cyber-security in mind. As a result, many devices deployed today do not feature network-security related capabilities, e.g., to establish secure communication channels via TLS or VPN [2]. In contrast to common IT networks, ICS networks also rely on local broadcast traffic to exchange time-critical data (and only a fraction of traffic is sent through a central uplink to the control center), so local attack traffic does not pass through firewalls at central uplinks. As a result, each device's incoming and outgoing traffic cannot be authenticated, and integrity of received data cannot be verified at security checkpoints on the perimeter. Improving the industrial devices after they are deployed (e.g. by applying patches or newer firmware) is seldom done, as operational continuity is paramount in the industry. Thus, common mitigation of such threats to industrial systems is either central firewalls that aim to protect against outside attacks, or bump-in-the-wire (or for short: bump) devices that protect individual device's traffic. Bump devices [3–6] are usually dedicated hardware devices that (acting as a Man-in-the-middle) intercept traffic through a specific link, encrypt it, or filter out malicious traffic.

There are several disadvantages of both approaches: firewalls cannot protect against local (e.g., Link-layer broadcast) traffic, as they only observe traffic through common uplinks. Bump devices have constrained resources, need to be physically inserted into existing networking topologies, and do not scale well: One expensive bump device is required for each critical endpoint to counter local attackers that exploit broadcast/multicast communication, which is often used in the industrial control systems, for instance IEC 61850 GOOSE in smart grid systems. Furthermore, bump devices only have a local view of traffic, which prevents process-aware detection and consistency checks, and they often cannot authenticate incoming broadcast traffic.

In this work, we propose to leverage VLAN capabilities present in most of managed industrial switches to redirect each device's traffic to a central server which checks and (selectively) forwards traffic to the destination. This approach enables the introduction of *vBump*: virtual bump-like processing of traffic on the central server, called vBump Server. The processing could be passive – to observe exchanged traffic for detection of attacks and/or false data injection

CPSIoTSec '21, November 15, 2021, Virtual Event, Republic of Korea

ACM ISBN 978-1-4503-8487-2/21/11…$15.00
https://doi.org/10.1145/3462633.3483983

attempts, or active – like a traditional intrusion prevention system (IPS), by identifying and blocking malicious commands before they reach the attack targets, but even for Link-layer broadcast communication. In our implementation, we use a software-defined-networking (SDN) enabled vSwitch to perform selective, flow-based traffic forwarding on vBump Server (without the need for SDN-capable industrial switches). When SDN-capable industrial switches are introduced, our approach can leverage them for even better performance. We note that the general concept of vBumps can also be applied in different contexts (e.g., IT networks), in particular to police specified local traffic between hosts in the same Link-Layer broadcast domain (e.g. connected to the same switch). We focus on ICS networks, as there is a stronger incentive to push security to the network (instead of securing end hosts), owing to the difficulty to upgrade legacy devices.

Compared to traditional bumps, vBumps do not require any changes in physical network topology, and the vBump Server can make decision based on a more global view while subjecting to less computational constraints. In addition to traditional features of bumps (such as tunnels and firewalling), vBumps can be used for system-wide authentication and consistency checks of data values, detection of attacks launched in a distributed manner, and coordinated attack prevention (as if each bump would act as an IPS). While VLAN is an established technology, our use of VLANs is novel as we only have individual end host in each VLAN, but still enable Link-Layer communication between them with the vBump Server, which provides reliable mediation of all traffic. We summarize our contributions as follows:

- We propose the vBump architecture as alternative to dedicated bump devices, based on two ideas: 1) Automated distributed isolation of legacy ICS devices via individual per-port VLANs as a generic (and transparent) mechanism to isolate ICS devices in legacy networks, over multiple switches. 2) Introduction of *virtualized bump* functionality on the central vBump Server to allow policing, inspection, and Link-Layer forwarding of any traffic (for example using SDN on a vSwitch).
- We implemented and evaluated the proposed system in a high-fidelity ICS network testbed (a modernized power grid system with a power generation, transmission, and distribution system) and demonstrate that introduced delay is acceptable while security goals are met.
- We discuss advantages of vBump over traditional bump solutions for ICS systems, in terms of operational benefits (cost, scalability etc.) as well as advanced features including system-wide authentication and consistency checking, detection of attacks launched in a distributed manner, and coordinated attack prevention.

The remainder of this paper is organized as follows. Section 2 discusses the background on ICS security, including existing bump and VLAN solutions. Section 3 proposes the main idea of vBumps, its advantages are discussed in Section 4. Section 5 provides details on our implementation of vBump, with performance evaluation in Section 6. Section 7 provides supplementary discussion, Related work is discussed in Section 8. Finally we conclude in Section 9.

2 BACKGROUND

2.1 Ethernet-based ICS and Key Security Issues

In the following, we will use a modernized electrical substation in a power grid system based on IEC 61850 standards [7] as a concrete example to motivate the design of our vBump solution. We choose electrical substations because they present some of the most challenging requirements among the ICS we studied, in particular, with their strict latency requirements [8] and high bandwidth demands. In addition, IEC-61850-based substations are among the first standardized Ethernet-based ICS solutions, and have been widely used in many parts of the world. By showing our solution can work in even such challenging settings, it will provide strong evidence for its broader application in other (less-stringent) ICS environments.

As shown in Figure 1, a substation network typically consists of three levels: 1) *station level* where human-machine interface (HMI), servers, and gateway are located, 2) *bay level* where intelligent electronic devices (IEDs) and programmable logic controllers (PLCs) are located, and 3) *process level* where physical devices (e.g., circuit breakers and meters) are located [9, 10]. For communication within the station level and between the station level and bay level (also called *station bus* or *interbay LAN*), IEC 61850 MMS protocol over TCP is typically used. On the other hand, communication at the bay level and between the bay level and the process level (also called *process bus* or *process LAN*) utilizes IEC 61850 GOOSE protocols on Ethernet is typical for the sake of stringent latency requirement (e.g., less than 4ms for traffic that supports some critical protection functions) [8]. While the typical bandwidth of substation local area network is 100Mbps [11], use of Gigabit Ethernet is also becoming popular as is the case of EPIC testbed which will be elaborated in Section 6.1

2.2 IEC 61850 GOOSE Communication and Possible Attacks

IEC 61850 GOOSE protocol is utilized for exchanging power grid status information among IEDs in an electric substation local area network. GOOSE frames are sent by each IED to announce update of its status (e.g., change in a circuit breaker status). While GOOSE messaging is the key driver for crucial features in substations (such as automated protection), owing to the very stringent latency requirement, cryptographic protection is not usually implemented for ensuring authenticity and integrity of the messages.

Taking advantage of the lack of security, attackers can manipulate or inject malicious GOOSE messages. For example, an attacker may attempt to inject a GOOSE message with a maliciously crafted sequence number (*sqNum*) and/or status number (*stNum*) by impersonating a legitimate IED (i.e., with spoofed source MAC address). Such a simple attack will result in the intended recipient IED(s) discarding the future message sent by the legitimate IED [12]. This would cause impact similar to denial of service (DoS) attacks without significantly increasing network traffic volume, which therefore makes it difficult for network-based intrusion detection systems to counter. Furthermore, while suppressing the messages from the legitimate IED in this way, the attacker would be afterwards able to inject fake GOOSE messages to confuse the system by means of fake status update or to trigger unnecessary protection controls on other IEDs [13].

We also note that, because GOOSE utilizes publish-subscribe communication model using Link-Layer multicast, all the nodes

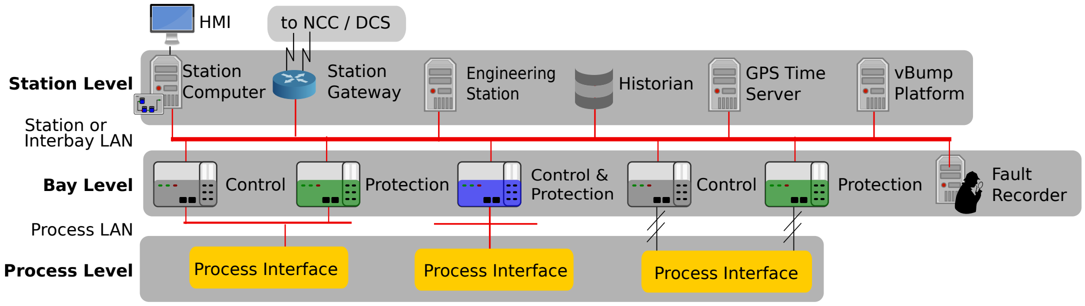


Figure 1: An example Ethernet-based substation. The vBump Server can be connected to the station or interbay LAN to provide the extra security.

in the same broadcast domain receive the message at the same time. This implies that network-based intrusion prevention systems (IPS) and other bump-in-the-wire security solutions, which will be elaborated next, cannot block the malicious messages before they reach the destionations, unless such systems are deployed in front of each invidivual IED in the network. Given that there can be hundreds of IEDs in a single substation, these approaches are prohibitive in terms of cost and thus are not often realistic.

2.3 Bump-in-the-wire (Bump) Security Devices

Although a security standard IEC 62351 [18] has been defined for smart grid systems based on IEC 61850, because of its heavy requirements such as use of TLS and digital signatures, the upgrades of ICS infrastructure for security reasons are not common or feasible in practice. Most SCADA/ICS protocols do not include security by design, and for this reason, complementary cybersecurity solutions are often introduced to legacy systems. The bump-in-the-wire (bump) devices are an example of such solutions, and can be introduced in a manner transparent to the existing ICS infrastructure. We summarize typical bump solutions in Table 1. In general, bump devices are classified by functionality as i) tunneling, ii) firewalls, iii) substation gateways, and iv) message authentication systems. The vBump solution discussed in this paper can provide these security features. In addition, it can provide more fine-grained and holistic network traffic control than traditional bumps.

2.4 VLANs

Virtual LAN (VLAN) technology provides a virtual way to create separate link layer broadcast domains (LLBDs) over a single physical network. IEEE 802.1Q standard defines the VLAN tags carried by network packets to provide the logical separation of network traffic. VLANs have been widely used to address issues such as scalability, security, and to enable easier network management. Typically, VLANs are used to isolate different groups of end hosts from each other, with hosts in one group potentially spread over multiple switches connected by VLAN trunks. VLANs are widely supported by industrial switches, as a number of industrial protocol require them. For example, VLAN can be used to separate different publish-subscribe groups for IEC 61850 GOOSE communication. Switch port-based membership in a VLAN always includes transmission and reception, so hosts cannot just receive traffic from a VLAN. As result, we cannot use VLANs directly to enable to hosts to only receive certain broadcast traffic without also being able to send traffic.

3 VBUMP: VLAN-BASED TRAFFIC AGGREGATION

In this section, we present our main contribution: vBump. The system allows to insert virtual bumps in front of Ethernet-based legacy ICS devices, without requiring change of physical topologies or device configuration. The vBumps can be used to limit traffic to whitelisted destinations, inspect all traffic on or above Link-layer like a centralized IDS (or more advanced monitoring systems), or even police the traffic like a centralized IPS. In particular, this also allows the network to apply fine-grained control on traffic between nodes that need to be in the same Link-layer broadcast domain.

3.1 System and Attacker Model

We consider an industrial system with legacy components such as IEDs and PLCs which do not support any security features (such as secure protocols based on IEC 62351 or TLS). In particular, we consider devices connected to a single Link-layer broadcast domain (such as a fieldbus network, or plant network), as required by protocols such as IEC 61850 GOOSE. We assume that the industrial switches that are used to span the Link-layer broadcast domain are managed ones, and support VLANs (e.g., the industrial Hirschmann RSP35 switch), but do not provide advanced networking features (SDN, firewall capabilities, deep packet inspection, etc.). Through the system specification, for example SCL (substation configuration language) files used in IEC-61850-compliant substations, it is known to the operator which device is supposed to communicate with which other devices, and the respective protocol. In particular, members of multicast groups are defined in SCL files, and intended receivers of broadcast traffic are also known [7]. In this work, we focus on discussing solutions to protect local traffic in each field substation, and not traffic forwarded over WAN connection to a remote control center, other substations, or similar.

We assume that the attacker has control over a local compromised device or host—e.g., by means of compromised firmware, SCADA station (e.g., via CrashOverride [19]), or via vulnerable VPN service often found at the station level. The attacker cannot physically alter the network topology, including switches. The goal of the attacker is to either gather information on the system, or to manipulate the operation of the physical process. To achieve that, the attacker could try to install herself as (Wo)Man-in-the-Middle (MitM, e.g., through ARP spoofing), or to send spoofed traffic (e.g., replay attacks). Attacks to local victim is forwarded via switches, and does not need to go through uplink (as a result, there is initially

Table 1: Existing bump Solutions for ICS. DPI= Deep Packet Inspection

Vendor	Device Model	Functionality	Approx. Price	Additional Comments
Hirschmann	Xenon Security Appliance	L2-4 filtering. DPI	3000 USD	128-416 microsecond latency[14]
MOXA	EDR series (e.g., G903)	VPN, firewall	900-1999 USD	84-514 microsecond latency[14]
Endian	4i edge series	VPN, firewall	570-1145 USD	Including IPS
eWON	eWON Cosy	VPN gateway	599-788 USD	Easy remote access customer sites
CISCO	Catalyst 6500 VPN module	VPN	3000-22000 USD	A range of backbone services
Virt. Access	GW2028 Industrial Router	Substation gateway	3200 USD	Including Protocol Conversion
Various	Link-Layer encryptor device	Encryption, tunneling	6k-30k EUR [15]	100Mbps to 10Gbps
Academia	A*CMD [6, 16]	Message auth.	150 USD	Context-aware validation of SCADA control commands (30+ messages per second)
Academia	F-Pro [17]	Message auth.	150 USD	Provenance-aware message auth. with very low latency (below 2ms in total)

no central location to intercept or even be aware of the attack). The actual attack steps performed are out of scope of this paper (we refer to [20] for an example).

To simplify the discussion, we assume that the central server (vBump Server) we introduce into each substation local network cannot be compromised by an attacker. We note that the server will not be addressable from the network on the Link Layer (and higher), and essentially acts as Link-layer bridge (or transparent Link-layer firewall). Therefore, exploitation over the network would require vulnerabilities in the code processing the network traffic (which we consider out of scope).

3.2 Problem Statement

Based on our system and attacker model, our problem statement is: *Without any modification of insecure legacy end hosts, how can we prevent a remote attacker with compromised local device from*

- *reading traffic (e.g. by eavesdropping using ARP spoofing),*
- *manipulating traffic (e.g., replaying or modifying traffic in real time as a man-in-the-middle), and*
- *injecting own traffic to any protected device?*

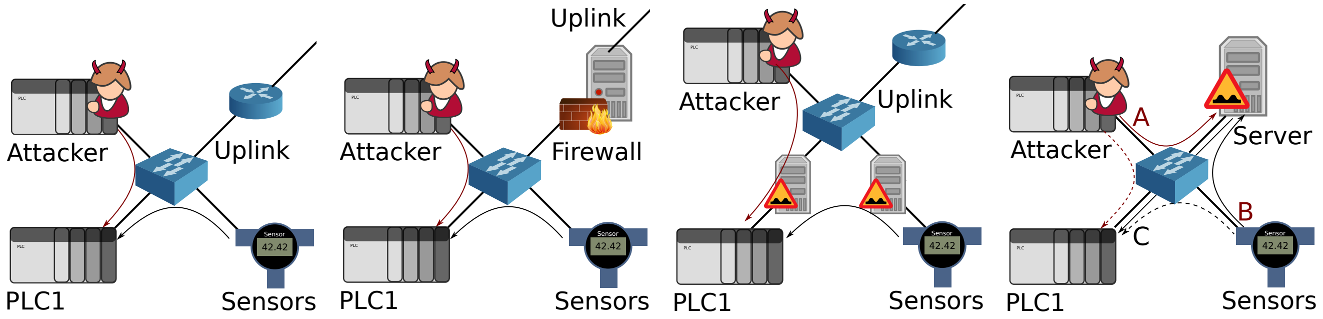
In particular, we consider attacks on devices placed in the same Link-layer broadcast domain (which as we argued earlier are not prevented by firewalls at uplinks, traditional use of VLANs, or traditional bump devices). Ideally, each protected device should be exactly able to only send/receive (specific types of) traffic to/from specific devices. The solution should be legacy-compliant (i.e., require no change in configuration of protected devices), economically affordable, secure, and should not impact normal system operations (i.e., introduced delay must be tolerable). We argue that traditional approaches using bump devices are not meeting those goals as they are expensive, provide coarse isolation (whole subnets are connected through VPN tunnels), and/or introduce non-negligible delays. Besides, traditional VLAN technology that groups end hosts into different broadcast domains will not prevent attacks within each VLAN.

3.3 vBump: VLAN-based Traffic Aggregation

The main idea of our scheme is to automatically configure VLANs on all network infrastructure to redirect *all traffic* through a central vBump Server in a substation. The server will have a chance to inspect, modify, or block it, and forward allowed traffic on the Link Layer afterwards. This central aggregation is enabled by the assignment of *individual VLANs* for each port of switches that faces ICS end devices, and forwarding of all VLANs through trunk connections between the switches. The vBump Server hosts virtual access ports for all VLANs (e.g., by leveraging VM guests), and runs appropriate code to inspect and modify traffic before forwarding it. The overall scheme will be completely transparent to ICS end devices, i.e., no configuration change is required on these end hosts. No Network-Layer subnets, gateways, or IP-addresses have to be changed. Link-layer sources, destinations, and general multi/broadcasting will remain the same from the perspective of end hosts. This use of VLANs is novel and very different from conventional use of VLANs (grouping multiple hosts in a VLAN to form a Link-layer broadcast domain or a Network-layer subnet, changing their Link-level environment and forcing them to be in different Network-Layer subnetworks). We show that configuration of the networking infrastructure can be automated (Section 5), and delays introduced by vBump are acceptable (Section 6).

Example. We demonstrate the concept in a minimal example in Figure 2. A PLC is connected to the same switch as a sensor and a compromised device, i.e., placed in the same LLBD (Link-layer broadcast domain), and is receiving Link-layer broadcast traffic from the sensor. In Figure 2a, the attacker is able to spoof broadcast traffic to manipulate the PLC. In Figure 2b, we show that a firewall placed at a gateway or uplink will not be able to filter the local broadcast traffic. In Figure 2c, physical bump devices are inserted into the connections between the end devices and the switch. We note that this setup requires *n* bumps for *n* ICS end devices, and physical changes to their network links. Even if bump devices are inserted, they cannot authenticate the broadcast traffic sources (to prevent the attack) since spoofing of MAC address is feasible. In Figure 2d, VLANs are configured on the switch to redirect traffic to vBump Server. On the server, a vBump bridges VLANs (enabling the inspection and manipulation of traffic). We note that the vBump setting requires an additional trunk connection and the vBump Server, but this setup supports a large number of vBumps.

Retaining High Reliability under vBump. One may be worried that the vBump Server could become a single point of failure for the whole system. Also, the switches between the vBump Server and the end hosts could become new failure points compared to the original setup without vBump. Fortunately, this risk can be effectively mitigated by using multiple, distributed physical vBump Servers to emulate a virtually single vBump Server in a fault-tolerant manner. This only incurs constant (e.g., 3 times) increase of the hardware cost. Finally, compared to traditional bump solutions, where multiple physical bumps could all be failure points and they may fail in



(a) Setting for GOOSE attack

(b) Firewall at uplink

(c) Dedicated bump devices

(d) vBump solution

Figure 2: Simplified example for vBump insertion. In (a), PLC1, Attacker, and sensor are all in same Link-layer broadcast domain. Attacker is directly sending malicious traffic to PLC1 via the switch. In (b), a firewall placed at the uplink does not observe or prevent the attack. In (c), dedicated bump devices are used to inspect and filter traffic to end devices, but are unable to authenticate the source of broadcast traffic. In (d), the vBump setup is used. Attacker, sensors, and attacker are in different VLANs (A, B, and C), all VLANs are forwarded to server via a trunk connection. On Server, a vBump bridges all VLANs (enabling inspection or manipulation of traffic).

an unnoticeable way (as they are at the edge of the network), we argue that it is in practice easier to monitor and maintain availability of a vBump-based system, thanks to the centralized design. This argument is similar to the reasoning that a centralized SDN approach does not necessarily sacrifice the reliability of a networked system.

3.4 Security Assessment of the Proposed Scheme

Traffic Policing. As result of introducing the vBump, any traffic sent by (compromised) end devices will be put into its own VLAN, and be forwarded via trunk connections to the vBump Server. That has two effects: a) even without any IDS running on the vBump Server, it is easy to limit traffic between *individual* end hosts to a set of allowed protocols or message types (even if source and destination are connected to the same switch), and b) if the server is running an IDS, more advanced manipulated messages can be detected because any attack traffic will first go through the vBump Server before it can harm the targeted victims on the network. We note that the switches themselves will not be addressable by remote attackers, and will not process content from the packets other than what is required for appropriate forwarding (VLAN tagging, CAM table building). As all ports to end devices will be access ports (which allow only untagged traffic from end devices, and always apply an 802.1q tag for forwarded traffic), attackers will not be able to manually attach VLAN tags to confuse switches. Even broadcast and multicast traffic can be selectively limited to specified sources and destinations.

VLAN Security. We note that VLANs are not designed as security feature. In particular, VLANs are not establishing secure channels (such as VPNs or TLS tunnels), and no encryption, authentication, or integrity protection is provided. Nevertheless, we argue that VLANs effectively prevent end devices that are connected to (correctly configured) access ports from exchanging traffic with other VLANs than the one configured for their port. As each end device will only have one other device on the same VLAN (i.e. the vBump Server), we argue that all malicious or leaked traffic would have to go through the server before being exchanged between an honest device and a compromised one. Therefore, additional encryption and authentication of the traffic is not required (given our attacker model, in particular a trusted server and switches).

VLAN hopping attacks have been widely studied to break the segregation enforced by VLAN [21]. However, those attacks commonly rely on the switches being configured incorrectly (e.g., allowing tagged traffic on access ports, untagged traffic on trunk ports). As our scheme uses automatic configuration of all switch ports, we are able to exclude such misconfigurations (a proof for a given implementation should be possible, but is out of scope here). Other attacks (e.g., CAM overflow attacks) were discussed in [22], but are also prevented by following security best practises [23].

3.5 Performance Impact

We now discuss the potential performance impact of introducing the vBump into an existing system. Later in Section 6 we demonstrate that the performance impact is still acceptable for practical substation network.

Impact on end-to-end delay. Introducing vBumps for end devices will incur additional delay for communication between hosts. The amount of introduced delay will depend on the following factors: processing delay at intermediate switches, transmission delay for additional links between switches and vBump Server, and processing delay by the vBump Server. In general, the delay will depend on the distance (in terms of the number of links) between vBump Server and any switch in the local network.

Impact on bandwidth consumption. We continue to use electric substation environment to discuss vBump's potential impact on the bandwidth requirement in a network. Specifically, the IEC-61850-standard based substations are among the most bandwidth-hungry ICS scenarios to our best knowledge.

In an electric substation environment, event and status update traffic can be transmitted in a high frequency and to a group of receivers. The IEC 61850 GOOSE messages transmit critical and time-sensitive substation events including alarms and fault notification. The GOOSE traffic is both periodic and event driven and hence has non-constant load. In [24], the authors present measurement data that lists the highest single-source traffic in their system is around 500kbps after an event, while without event, the GOOSE traffic is below 100kbps.

To examine the impact on the bandwidth usage to mediate multicast traffic, we look at the following four components:

1. Switch-to-host bandwidth usage
2. Switch-to-vBump Server bandwidth usage
3. A switch's backplane bandwidth usage
4. Switch-to-switch bandwidth usage. (if vBump Server is located at a different switch compared to the multicast group)

We first consider a simple case where there is only one switch, and there is a multicast group with one publisher and k subscribers, and vBump Server is located on the same switch. As such, only the first three components are relevant. In the original setting (without vBump), one packet is delivered to each of the k subscriber over their link with the switch, and the switch's backplane forwards these k packets. In the setting with vBump, still one packet is delivered to each of the k subscriber over their link with the switch. But before that, one additional packet is delivered to vBump Server, and the server will need to know all the k subscribers, and generate k packets for them respectively. Hence there are $k + 1$ packets transmitted over between vBump Server and the switch. The switch's backplane forwards a total of $k + 1$ packets. Comparing these two settings, we can see that the main bottleneck is at the link between the vBump Server and the switch.

For the case when the vBump Server is moved to a neighboring switch, there are an additional $k + 1$ packets being forwarded between the two switches, all the other traffic remain the same.

4 ADVANTAGES OVER TRADITIONAL BUMPS

Compared to traditional bumps, the proposed system has a number of advantages: i) security features (e.g., support of advanced IPS), ii) cost, iii) management, iv) flexibility, v) compatibility, vi) scalability, and vii) support of SDN. In addition, it allows for partial introduction of SDN, and other advanced features that we discuss in the following.

Security features supported by vBump. The main goal of vBumps is to enable fine-grained traffic policing and inspection for each end device, without requiring SDN capabilities on switches, and without requiring bump devices in front of each end host (or similar functionality on network appliances acting as switches). Compared to traditional bump solutions as summarized in Table 1, vBumps can be used for system-wide consistency checks of data values, distributed attack detection, and coordinated attack prevention (as if each vBump would act as an IPS). Some of the security features that vBump can offer will be discussed below, while implementation of them is part of our future work.

First off, attacks against IEC 61850 GOOSE protocol (discussed in Section 2.2) can be blocked. For instance, vBump Server can check the specific fields of GOOSE messages (e.g., stNum and sqNum) in a stateful manner. This way, vBump can detect abnormal changes in these values and block the message before forwarding it to the rest of the network.

As shown in Figure 1, a substation typically consists of process bus and station bus. While process bus may carry the IEC 61850 GOOSE and/or SV (Sample Value) traffic, which carries power grid measurements reported with high frequency, one can potentially configure vBump Server to mediate all traffic in the station bus while overhearing part of the process bus traffic. While IEC 61850 SV traffic can be of significant volume, most of the messages are repetition, and therefore sampled traffic monitoring will suffice for situation awareness. The vBump Server then can use the information from the process bus (assuming all devices in the process bus are trusted) to determine the traffic legitimacy in the station bus to enable system-wide consistency checking. For example, if a compromised device in the station bus sends out an IEC GOOSE message that is in conflict with the measurements reported by SV in the process bus, vBump Server can block such a message and prevent it from reaching the destinations.

It is also possible to detect misbehaving (or maliciously programmed) devices, such as PLCs, by correlating inputs (measurements sent by IEDs) and outputs (the control commands issued by PLCs). Similarly, network traffic aggregation by vBump allows vBump Server to perform anomaly or inconsistency detection in multi-hop communication. Because vBump Server can monitor both messages incoming to and outgoing from each device, vBump Server can perform such verification.

In [25], the authors proposed to use the IEC 61850 Substation Configuration Language (SCL) to model the configured multicast group and check against potential anomalies. Such detection or policy enforcement can be implemented when the traffic is mediated by vBump Server. Typical SCL files include network configurations, such as devices that are supposed to communicate with a certain protocol, along with IP addresses and/or MAC addresses.

Cost. In the traditional approach, one bump device is required per secured end host. That means the cost for additional hardware is around 2000 USD per device (average price of our reviewed bump devices, see Table 1). The vBump approach only introduces hardware cost for the vBump Server which does not need to be very powerful. In our experiments, we used a 2000 USD laptop without performance bottlenecks.

Management. In practise, managing several bump devices in the field can become a hassle, in particular if different model versions and vendors are used. A centralized vBump server presents a single point that needs to be managed and updated.

Flexibility. The vBump approach will allow easy reconfiguration, addition and removal of secured end devices. No physical changes will have to be performed, reducing downtime and maintenance cost. Additional connections can be allowed on demand, e.g. for maintenance, updating, and testing.

Compatibility. The proposed approach is agnostic of industrial protocols used, and can thus be used independently of the vendor and protocols used. It is transparent to end devices and does not require any configuration or network topology changes.

Scalability. The proposed solution is able to scale to cover all IEDs in typical electrical power grid substations. The only limitation to the size of supported network is related to bandwidth of the network (discussed in Section 3.5 and Section 6.4), and limits on VLAN IDs. A maximum of 4096 VLANs can be supported in a single network, which is translated into 4096 end devices. In the typical deployment scenario we envisioned, each substation will form one network (i.e. a router or industrial firewall is deployed at the entrance of the substation), and this number of VLANs should be sufficient as a substation (even a very large one) usually hosts at most a few hundred IEDs.

vBumps and SDN. The proposed system allows to redirect all traffic to one (or more) central locations. Among the primary uses introduced so far, we note that this also enables redirection of traffic to SDN-capable switches (such as the vSwitch we use in our implementation in Section 5). Effectively, this allows to introduce traffic policing using SDN controllers for networks that lack widespread SDN capabities. While our approach introduces higher delays and overheads (as traffic has to always be forwarded to central point), we note our approach could be used to test and prototype SDN solutions for industrial networks this way. When SDN switches are available in industrial settings, it will be possible to implement the security features of our approach in a distributed fashion over the switches, allowing for more fine-grained selection of what traffic to forward to the central bump, and application of simple filters directly on the switches. We leave exploration of further advantages for future work.

5 VBUMP FRAMEWORK AND IMPLEMENTATION

In the following, we discuss the required abstract components to realize the high-level vBump idea and the implementation of the components.

5.1 Abstract vBump Server Framework

The vBump Server has several components (see Figure 3): i) a module to inspect current network configuration, and determine the placement and configuration of the VLANs for the switches, ii) an access control analyzer to determine the intended rules for communication, iii) a module to automatically configure the switches based on the previously determined configurations, iv) a module to configure the networking settings on vBump Server and start appropriate vBump applications.

Network Analyzer. The network analyzer processes the network topology information (provided as .csv file), which includes the details (including IDs, number of interfaces) of all end devices and the vBump Server, as well as all switches. Their connectivity is expressed as a list of links with information about each link's two endpoints. An endpoint is described by the ID and interface index of an end device, vBump Server, or switch. Such network topology can also be systematically derived from IEC 61850 SCL files, which will be implemented in our future work. Based on that input, the analyzer then determines the VLANs that need to be created to isolate each end device, and the configuration for individual ports of the switches. Based on this VLAN mapping, VLAN associations of access ports on the switches are determined. In addition, suitable configuration for the trunk links between switches themselves, and switches and the vBump Server are determined. The resulting network configuration is provided in .csv format.

Network Configurator. The network configurator processes a .csv file with listings of all active switch ports in the network, mapped to VLAN IDs (as produced by the network analyzer). Based on that file, the network configurator automatically initiates SSH connections to all related switches, and configures them appropriately via shell commands. In particular, configuration considers a correct sequence to ensure that network connectivity to switches is not lost due to partially configured switches. During the overall configuration process, normal communications in the network between end devices are partially interrupted, so the configurator is optimized to run as quickly as possible. While the current implementation is designed for Hirschmann switches, given that many managed switches have similar remote configuration interface, the approach can be applied for other models with minor customization in configuration commands.

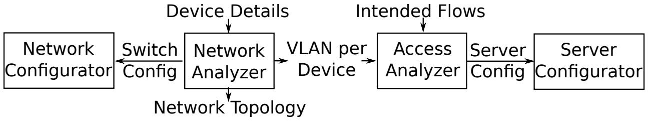


Figure 3: Components and input/output for the vBump framework configuration.

Access Analyzer. The access analyzer receives a mapping from VLAN ID to end device from the network analyzer, and requires a list of the intended flows (e.g., device a should send modbus traffic to device b). Based on that input, the access analyzer prepares a configuration for the vBump Server configurator. The mapping inputs need to translate between an abstract device identifier and a VLAN ID, the intended flow input will contain tuples with abstract ID of source and destination of a flow, together with a source and destination IP, MAC, and port where appropriate. Some of the information, including VLAN ID, can be derived from IEC 61850 SCL files, use of which is part of our future enhancement plan. If deep packet inspection or other advanced IPS features are intended to be performed on vBump Server in addition to basic SDN functionality, then additional rules to be applied to specific intended flows could also be provided.

vBump Server Configurator. The server configurator processes the output of the access analyzer to set up local networking to have a trunk connection to one of the configured switches, with the trunk carrying all VLANs that were created. That trunk interface is then locally connected to a component that allows to selectively inspect and modify traffic (e.g. move packets from one VLAN to another), e.g. by using a bridge with VLAN tagged ports or a virtual switch with SDN capabilities. Incoming packets can be forwarded based on access rules created from the initial information about the ICS components and the intended data flows, as provided in the Server Config by the access analyzer. The rules can be applied based on incoming packet's VLAN IDs, and/or MAC addresses, protocol types, IP addresses, etc. If a bridge is used, a dedicated server-based traffic control components (i.e. similar to an IPS) is required to enforce such rules. If an SDN-capable virtual switch is used, the related SDN controller is effectively able to play such a role if required.

Server-based Traffic Control. All traffic incoming and outgoing from the vBump Server over the trunk connection can be inspected by a component, which effectively decides to forward or drop the traffic. For example, a simple setup could use ebtables [26] to filter the traffic based on MAC addresses or VLAN tags. IPtables could be used to enforce network-layer policies, or more advanced network appliances could be used on the trunk connection to perform deep-packet inspection.

5.2 Implementation

We implemented the vBump framework as outlined above, using an SDN controller and Openvswitch [27] on the vBump Server for main traffic control. All components listed in Figure 3 have been implemented in Python programs, and are available at https://github.com/scy-phy/vbump. The SDN controller on vBump Server is Pox [28], using OpenFlow 1.0 as SDN protocol between the vswitch and SDN controller. As simple data format between the building blocks of the network analyzer, network configurator, access analyzer and server configurator we used a basic .csv format.

The network analyzer expects a topology description that specifies each port in the network with a tuple containing the unique switch name, the port number, a port type (access, trunk, server, other), and a comment. All industrial end devices that should be protected by the vBump Server should be connected to an access port. Links between switches need to be declared as trunk ports. Ports that should not be configured (and will be placed in a default VLAN) are declared as other. The server type ports are connected to the vBump Server. The network analyzer then produces a .csv file which assigned a unique VLAN to each access port, and the trunk and server ports are configured to carry those VLANs. From this data, the network configurator script is then producing shell scripts for Hirschmann switches that can be directly sent via SSH to automatically configure the switches.

For each VLAN, our access analyzer is expecting a dictionary with lists of allowed target VLANs (no MAC address or higher layer filtering is currently implemented). Based on this data structure, the pox controller is then processing each incoming flow request by the vSwitch. The flow for the switch is configured as follows (simplified). Only VLAN-tagged traffic is accepted. For each incoming packet, a copy is made for each allowed destination VLAN, and their VLAN IDs are updated. Then, the packets are sent by the switch to the respective VLANs on the network. Our program for the SDN controller (which generates these flows and sends them to the switch) is less than 100 lines of python code including comments.

6 EVALUATION

In this Section, we will evaluate vBump in the following aspects: 1) its basic security features; and 2) the extra delay it introduces to ICS traffic, especially those with real time requirements. We also evaluate its capability to support high-frequency traffic of greater than 100Mbps, which is a typical network bandwidth in a substation.

6.1 The EPIC testbed

While our proposed system is generic, the implementation and evaluation section of this work will use the EPIC testbed as platform.

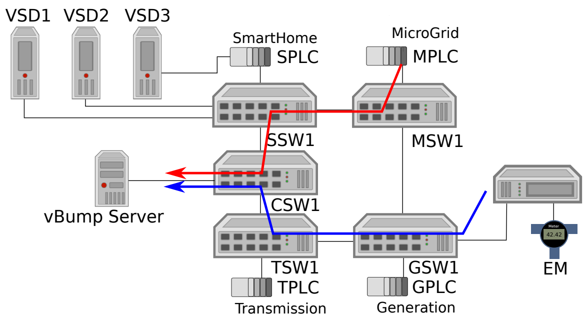


Figure 4: Subsystem of EPIC testbed used for the experimental evaluation. Industrial end devices such as the PLCs (SPLC, MPLC, GPLC, TPLC) are shown, while smaller devices are omitted. The red and blue arrows show the path of packets from two end hosts to the server, via 3 switches.

Instead of showing the full complexity of such a practical system, we concentrate on fundamental networking links and the general architecture. In particular, we consider the electric power generation process. In Figure 4, we provide an overview of the subsection of network architecture in EPIC testbed used for our experiments.

6.2 Experimental setup

We experimentally evaluated our implementation in the EPIC testbed (see Figure 4). We used 5 out of the 6 switches that connect the different subsections to the same LLBD in the original configuration. The switches used in that testbed are the aforementioned Hirschmann RSP35 and RSPL30 switches, which are capable of remote configuration (via SSH) and VLANs, but not SDN. The Hirschmann RSP35 Industrial Ethernet-Switch has eight 100Mbps Fast Ethernet (FE) ports and three Gigabit Ethernet (GE) Ports, while the Hirschmann RSPL-Lite (RSPL30) Industrial Ethernet-Switch has eight 100Mbps FE ports and 2 GE Combo Ports. Both types of switches handle the packets in a store-and-forward manner. We ran vBump Server on a PC with Intel Core i7-6600U CPU2.60GHz and 20GB RAM with 1Gbps Intel Ethernet Connection I219-LM NIC, running Ubuntu v 16.04.1. The goal of the experiments was a) to demonstrate the feasibility of the vBump approach, b) the functionality of our prototype implementation, and c) to investigate introduced delays (using a message exchanged between two devices connected to the same switch as baseline). For these purposes, the actual physical power grid system does not matter, and thus we will not provide further details here.

6.3 Evaluation of security features

We verified that the implemented system correctly applies our policies to the forwarded traffic in three ways: a) multicast traffic, b) malicious VLAN tagging, c) unicast traffic. With respect to multicast traffic, we recorded incoming traffic at end hosts (e.g., ARP multicast traffic) and verified that only traffic from white-listed sources was received (which was the case). Then, we tested if manually applied VLAN tags would lead to incorrect routing of the traffic (it did not, only allowed end hosts received the traffic, with the additional VLAN tags applied which lead them to be discarded). Last, we tested whether messages directed to the IP or MAC address of target devices that should not be reachable could be reached (e.g., as part of ARP spoofing). We confirmed that our implementation successfully blocked such messages (based on source and destination VLANs). As all of those actions are deterministic, we had 100% accuracy.

6.4 Delay and throughput performance of vBump

We started the delay measurement experiment by connecting two endpoints (host A and B) and the vBump Server to the same switch (switch CSW1 in the EPIC testbed). We measured the round trip time (RTT) of ICMP Ping messages between the two endpoints. To understand the impact of ICS traffic with different packet size, we tested two payload sizes (i.e. 140 bytes, which roughly corresponds to the average size of IEC 61850 GOOSE messages observed in the EPIC testbed, and 1400 bytes respectively) in our generated ICMP Ping messages. We measured the RTT with varying intensity of background traffic. We found that our RTT measurements remain stable under the loss-free settings (we found experimentally that our system can support 150Mbps of traffic without incurring any loss), hence we do not present the RTT performance under different traffic intensity here, but report only the RTT's overall distributions. For this "1-switch" topology, we measure the case when the two endpoints directly communicate with each other through a physical switch without going through vBump Server, which serves as the baseline to evaluate the additional overhead introduced by vBump. When we enable the vBump, the traffic from host A to B will be first forwarded from A to vBump Server via the physical switch, then from vBump Server to B via the physical switch again. There is also additional processing time at vBump Server. In particular, when we use the SDN-based implementation, the packets are forwarded by the virtual vSwitch in the vBump Server. We tested two implementations of the vBump Server. In the first implementation, incoming traffic from different VLANs are mapped to different VLAN interfaces. The vBump Server then uses ebtables to setup Link-layer bridges among different pairs of VLAN interfaces to allow the corresponding endpoints to communicate with each other. In the second implementation, we emulate a virtual SDN switch and use customized rules in the corresponding SDN controller to process the VLAN traffic.

In a large electric substation, there can be dozens or even hundreds of devices that are connected via multiple switches. If the vBump Server is deployed only at the main switch, there can be multiple hops of switches between an endpoint and the vBump Server. To study the impact of increasing packet traverse path introduced by vBump in such a setting, we keep the two endpoints A and B on the same switch, while increasing the number of switches between the endpoints and the vBump Server. Specifically, in addition to the 1-switch setting, we tested the case when there are 2, 3, and 4 switches between an endpoint and the vBump Server. Our goal is to determine if the round-trip time of communication in such settings can reliably be kept below $4ms$ (as the strongest constraint for critical power grid protection [8]).

Fig. 5 and Fig. 6 show the measurement results for the SDN setting with the two packet size settings respectively. Fig. 7 and Fig. 8 show the measurement results for the bridging setting with the two packet size settings respectively. As can be seen in these figures, for the 1-switch setting, enabling vBump roughly doubles the round-trip time between the two end-points. The additional RTT here includes one additional switch forwarding delay, and the vBump Server processing delay. For the bridging and 1-switch setting, processing the larger packet with 1400-byte payload takes about 5 times than processing the smaller packet with 140-byte of payload. The overall RTT is $< 2ms$ for one additional switch.

Further, for each additional switch between the vBump Server and the endpoints, the median RTT increases by roughly 0.1ms for the smaller packet and 0.5ms for the larger packet. In fact, by adding one additional switch, there are four additional switch forwarding operations conducted by this added switch, i.e., endpoint A to vBump Server, vBump Server to endpoint B, endpoint B to vBump Server, and vBump Server to endpoint A. As result, with 4 additional switches to forward over, the RTT for bigger packets approaches $4ms$ (but never reached $4ms$ in our experiments).

Finally, for the same packet size and topology setting, we observe that the SDN setting performs slightly better than the bridging setting, achieving both lower median value of RTT and lower variance. We assume this is due to efficient packet processing at the SDN switch after forwarding rules were installed, compared to the native bridging implementation.

As noted before, we found most critical time constraints on transmission delay in the context of critical safety equipment in electric power substations. Because the most stringent latency requirement among all the cases discussed by the IEEE PES's guideline on communication latency [8] is 2ms, we conclude that our scheme is not expected to have negative impact on other types of communication.

We note that in these figures we measure the round trip time, while the stringent delay requirements are posted on one-way delay of small-size packets (the critical traffic is similar to our 140 byte packet setting). As a result, even for 4 switches, the SDN setting can provide a median one-way delay of about 0.7ms/2=0.35ms, and almost 100% of packets incurs a one-way delay of less than 1.2ms/2 = 0.6ms, which is well below the 2ms threshold [8]. While our measurements do not include advanced access control logic as discussed in Section 4, the remaining time buffer can be taken advantage of to accommodate the enhanced verification and attack detection algorithms.

Last but not the least, our experimental results also show that our vBump Server implementation can process 150Mbps of aggregated traffic reliably without incurring any packet loss.

7 DISCUSSION

In this section, we provide further discussion of aspects related to the proposed vBump scheme.

Integration of Additional Security Measures. The implementation and measurements discussed in the previous section is basic network traffic filtering. However, we can deploy additional security measures on vBump Server. Here we briefly discuss practicality of intrusion detection. We deployed Zeek, an open-source intrusion detection framework, with IEC 61850 GOOSE protocol parser [29] on a VM with 4 CPU cores with 12GB RAM, which are equivalent to the spec of middle-class industrial PCs. Based on our prototype implementation, the latency to disect GOOSE messages is on average 17μs, and thus stateful checking of stNum and sqMum to counter attacks discussed in Section 2.2, as well as other detection rules that utilize message payload, can be performed within 20-30μs. Other security mechanisms discussed in Section 4 require handling of IEC 61850 SV messages. Since the message format is GOOSE and SV are very similar, the expected latency is within the same range. Moreover, calculation for consistency checking is basically solving one formula using the measurements of a certain time window, the latency for it is not significant. Adding this latency to the measurements discussed in Section 6, we see that the overall latency can still meet the requirement.

Compatibility with IEC 62351 Standard. There is a security standard, IEC 62351 [18], specifically defined for securing communication in a substation, i.e., IEC 61850 MMS, GOOSE, and SV protocols. The standard at the high-level, defines use of symmetric (mainly for low-latency communication such as GOOSE) and asymmetric cryptography for providing message authentication (for low-latency communication such as IEC 61850 GOOSE and SV) and/or confidentiality (for protocols over TCP/IP, such as MMS). Associated key management protocol, including use of PKI, is also defined. Regarding communication over TCP, which is applicable to IEC 60870-5-104 and IEC 61850 MMS, the security (authentication, confidentiality, and integrity) is added by means of TLS as defined in IEC 62351-3. IEC 62351-4 additionally defines application-layer security for MMS. Because our traffic aggregation using VLAN works at a layer below it, the proposed scheme does not interfere

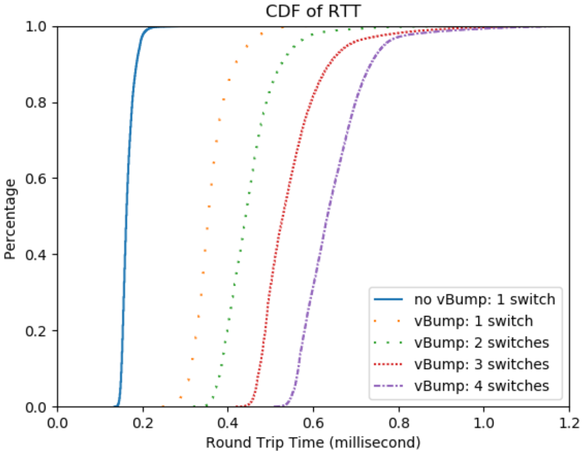


Figure 5: Delay measurement over increasing number of hops with 140 bytes of ping packets, using SDN setting.

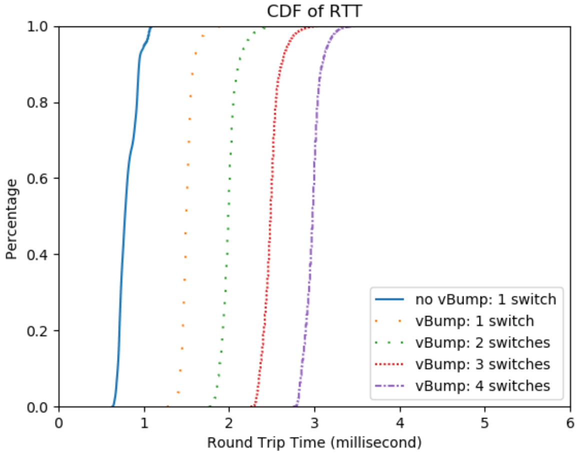


Figure 6: Delay measurement over increasing number of hops with 1400 bytes of ping packets using SDN setting.

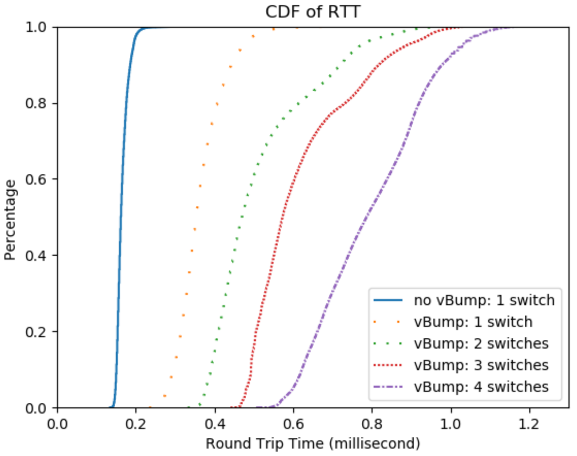


Figure 7: Delay measurement over increasing number of hops with 140 bytes of ping packets, using bridge setting.

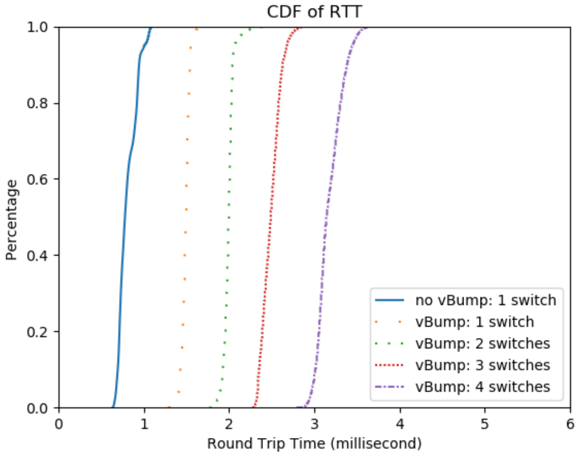


Figure 8: Delay measurement over increasing number of hops with 1400 bytes of ping packets using bridge setting.

with the IEC 62351 standard. Specification for IEC 61850 GOOSE and SV is defined in IEC 62351-6. The standard basically extends packet format of these protocols to convey additional information for security (digital signature) in such a way that the backward compatibility with devices that do not support the IEC 62351 standard is not broken (i.e., added security data can be simply ignored by legacy devices). Our scheme does not interfere with the transmission of GOOSE packets and is agnostic to the extension field used by IEC 62351, and thus vBump is compatible with the standard.

Compatibility with Existing VLAN. Substation implementations may utilize VLANs corresponding to the multicast groups, which would potentially conflict with vBump' VLAN definition. vBump scheme requires the system operator to replace such VLAN configuration corresponding to multicast group and instead to In such a case, the operator needs to remove the VLAN definitions and instead let the vBump Server handle the mapping between each multicast MAC address and VLANs, which correspond to end devices, so that the frame can be forwarded to appropriate devices that belong to the multicast group.

Tradeoffs between redundancy, performance, and cost. A potential concern for vBump's deployment is whether vBump introduces a single point of failure, as traffic is being aggregated at vBump Server. In this work, we focused on security and performance aspects of the solution. We see several options to address reliability concerns (e.g., triggered by watchdog timers). In particular, standard engineering solutions such redundant vBump servers (connected to alternative switches) could be used for additional cost. In electrical substations, it is a common adopted practice to introduce redundancy to all the mission-critical components. For example, a critical IED often has two interfaces that are connected to two different switches. This eliminates the single point of failure at the networking components. Our redundant vBump Server can leverage / expand such architecture. We also note that traditional bump devices also introduce additional points of failure, which are even harder to mitigate with duplication.

The VLAN configuration could also be automatically changed to fall back to solutions without the vBumps (temporarily disabling security features), or to forward traffic over alternative unaffected links. For larger substations with a large number of end points (e.g., a few hundreds of them), multiple switches need to be used to form a hierarchical topology to connect all these end points. In such a setting, the deployment of vBump Server needs to trade-off between the redundancy, the performance, and the cost. For example, a lower-cost deployment may only have two vBump Servers, each connected to one of the two top redundant switches. While they do not cause a single point of failure in this setup, the communication between two nearby endpoints now depend on more switches (i.e., all the switches along the path from the endpoints to the vBump Server) compared to the case without vBump Server. A more robust deployment solution is, e.g., to deploy one vBump Server per switch to deal with all locally generated traffic. This can also reduce the delay / bandwidth consumption, albeit at a higher deployment cost and higher management overhead for managing the multiple vBump Servers.

8 RELATED WORK

Similar to vBump, private VLANs [23, 30] assign unique VLAN ID for each individual end host. However, the main goal of private VLANs is to fully isolate non-gateway hosts connected to a single switch, and only enabling them to get their traffic forwarded towards a trunk port to be then routed on the network layer at the gateway to the Internet. Even if traffic would be routed between different local subnet at the gateway in Private VLANs (for which we could find no examples in the literature), two hosts can only communicate on Network layer or above, so Link-layer protocols such as the GOOSE protocol used in electrical substation networks cannot be used. In vBumps, hosts will still (from their perspective) be in the same Link layer broadcast domain as other hosts.

After completing this work, we found patents discussing approaches they call *service insertion* [31, 32]. While we admit the similarity of our approach to these, the patents are very generic and do not discuss the specific challenges of ICS networks, in particular smart grid systems (e.g., timing requirements, specific network topologies, etc). There does not appear to be any prior academic work on evaluating these aspects of service insertion in ICS.

There have been recent efforts that study the use of Network function virtualization (NFV) and SDN technologies for providing scalable and flexible cybersecurity solutions. For example, [33] proposed an vNIDS solution to virtualize the implementation and deployment of network intrusion detection systems (NIDS). Their focus is on how to partition the detection logic of NIDS so that they can be executed as separate microservices, and how to reduce the amount of states that need to be shared between different instances. In another work, [34] utilizes NFV to handle DDoS attack traffic and leverages SDN to redistribute the traffic that optimizes the bandwidth consumption. In contrast, vBump's focus is to use existing switches's VLAN capability to aggregate traffic to a central place, where extra virtualzed network functions may be implemented. On the other hand, similar to [33] and [34], vBump could benefit from SDN-enabled switches to further improve its flexibility and performance.

The use of SDN technologies have been explored for industrial control systems and smart grid context. For instance, Kumar et al. [35] applied SDN for ensuring end-to-end communication latency in industrial control systems. Specifically, SDN is used to allocate a dedicated queue for high-priority flow. Their focus is not enhancement of security for ICS, and therefore orthogonal to our approach. Dong et al. [36] studied the opportunities and challenges SDN bring to smart grids. Their focus is on use of SDN to quickly reconfigure networks in case of attacks. The authors did not explore the use of SDN in aggregating traffic for security purpose, or used SDN to provide port-based access control.

9 CONCLUSION

In this work, we proposed the vBump solution to protect end devices in ICS networks. Our solution is designed not only to reduce the implementation cost and management efforts that come with individual bump devices, but also to potentially introduce additional capabilities . We leveraged in-situ VLAN capabilities present in most managed industrial switches to redirect each device's traffic via individual VLANs to a central server which checks and (selectively) forwards traffic between the ICS devices. This use of VLANs is novel, as it allows end hosts to continue communication on the Link-layer (unlike private VLANs), and allows inspection of all traffic (unlike VLANs with multiple end hosts in them). Our approach enables the introduction of vBump: virtual bump-like processing of traffic on the central server. Compared to traditional bumps, vBumps do not require any changes in physical network topology, and the central server can make decision based on a more global ICS network view. In addition to security features of traditional bumps, vBumps can be used for system-wide authentication and consistency checks of data values, attack detection, and attack prevention. Our experimental results show that the extra network delays and bandwidth overheads introduced by vBump are acceptable even for time-critical smart grid contexts ($\approx$ 1ms when forwarding traffic over 3 additional switches) to the vBump server (and to the destination). These measurements demonstrate the practicality of our solution in a modernized electrical substation of realistic scale.

ACKNOWLEDGEMENTS

This research is supported in part by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme, and in part by the SUTD Start-up Research Grant (SRG Award No: SRG ISTD 2020 157).

REFERENCES

[1] "Russian hackers reach u.s. utility control rooms, homeland security officials say," 2017. [Online]. Available: https://www.wsj.com/articles/russian-hackers-reach-u-s-utility-control-rooms-homeland-security-officials-say-1532388110

[2] D. Quarta, M. Pogliani, M. Polino, F. Maggi, A. M. Zanchettin, and S. Zanero, "An experimental security analysis of an industrial robot controller," in *Proceedings of the IEEE Symposium on Security and Privacy (SP)*, vol. 00, May 2017, pp. 268–286. [Online]. Available: doi.ieeecomputersociety.org/10.1109/SP.2017.20

[3] S. Kent and K. Seo, "Rfc 4301: Security architecture for the internet protocol," 2005.

[4] D. Žagar, K. Grgić, and S. Rimac-Drlje, "Security aspects in ipv6 networks–implementation and testing," *Computers & Electrical Engineering*, vol. 33, no. 5-6, pp. 425–437, 2007.

[5] B. L. Chappell, D. T. Marlow, P. M. Irey, and K. O'Donoghue, "An approach for measuring ip security performance in a distributed environment," in *Proceedings of the Workshops Held in Conjunction with the Parallel Processing Symposium and Symposium on Parallel and Distributed Processing*. Springer Berlin Heidelberg, 1999, pp. 389–394.

[6] D. Mashima, P. Gunathilaka, and B. Chen, "Artificial command delaying for secure substation remote control: Design and implementation," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 471–482, Jan 2019.

[7] R. Mackiewicz, "Overview of iec 61850 and benefits," in *Power Systems Conference and Exposition, 2006. PSCE'06. 2006 IEEE PES.* IEEE, 2006, pp. 623–630.

[8] IEEE Power and Energy Society, "IEEE Standard Communication Delivery Time Performance Requirements for Electric Power Substation Automation," 2004.

[9] J. Hong, Y. Chen, C.-C. Liu, and M. Govindarasu, *Cyber-Physical Security Testbed for Substations in a Power Grid.* Springer Berlin Heidelberg, 2015, pp. 261–301.

[10] IEC TC57, "IEC 61850-90-2 TR: Communication networks and systems for power utility automation – part 90-2: Using iec 61850 for the communication between substations and control centres," *International Electro technical Commission Std*, 2015.

[11] D. Mashima, B. Chen, P. Gunathilaka, and E. L. Tjiong, "Towards a grid-wide, high-fidelity electrical substation honeynet," in *Proceedings of the Conference on Smart Grid Communications (SmartGridComm).* IEEE, 2017, pp. 89–95.

[12] M. El Hariri, T. Youssef, and O. Mohammed, "On the implementation of the iec 61850 standard: Will different manufacturer devices behave similarly under identical conditions?" *Electronics*, vol. 5, no. 4, p. 85, 2016.

[13] P. P. Biswas, H. C. Tan, Q. Zhu, Y. Li, D. Mashima, and B. Chen, "A synthesized dataset for cybersecurity study of iec 61850 based substation," in *Proceedings of Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm).* IEEE, 2019.

[14] M. Cheminod, L. Durante, M. Maggiora, A. Valenzano, and C. Zunino, "Performance of firewalls for industrial applications," in *Proceedings of the Symposium for ICS SCADA Cyber Security Research (ICS-CSR)*, Aug. 2016.

[15] C. Jaggi, "Layer 2 encryptors for metro and carrier ethernet wans and mans," 2017.

[16] D. Mashima, B. Chen, T. Zhou, R. Rajendran, and B. Sikdar, "Securing substations through command authentication using on-the-fly simulation of power system dynamics," in *Proceedings of the Conference on Smart Grid Communications (SmartGridComm).* IEEE, 2018.

[17] E. Esiner, D. Mashima, B. Chen, Z. Kalbarczyk, and D. Nicol, "F-pro: a fast and flexible provenance-aware message authentication scheme for smart grid," in *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm).* IEEE, 2019, pp. 1–7.

[18] F. Cleveland, "IEC TC57 WG15: IEC 62351 security standards for the power system information infrastructure," *White Paper*, 2012.

[19] "Crashoverride malware," [Online]. Available: https://www.us-cert.gov/ncas/alerts/TA17-163A, 2017, (Date last accessed on Feb. 4, 2019).

[20] D. Urbina, J. Giraldo, N. O. Tippenhauer, and A. Cárdenas, "Attacking fieldbus communications in ICS: Applications to the SWaT testbed," in *Proceedings of Singapore Cyber Security Conference (SG-CRC)*, January 2016.

[21] R. Farrow, "VLAN insecurity," Mar. 2003. [Online]. Available: http://rikfarrow.com/Network/net0103.html

[22] S. Convery, "Hacking layer 2: Fun with ethernet switches," *Blackhat [Online Document]*, 2002.

[23] T. Kiravuo, M. Sarela, and J. Manner, "A survey of ethernet lan security," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1477–1491, 2013.

[24] U. Carmo, D. H. Sadok, and J. Kelner, "Iec 61850 traffic analysis in electrical automation networks," in *Smart Grid Communications (SmartGridComm), 2015 IEEE International Conference on.* IEEE, 2015, pp. 466–471.

[25] J. Zhang and C. A. Gunter, "Application-aware secure multicast for power grid communications," *International Journal of Security and Networks*, vol. 6, no. 1, pp. 40–52, 2011.

[26] Netfilter Coreteam, "Ebtables: a filtering tool for a linux-based bridging firewall," 2018. [Online]. Available: http://ebtables.netfilter.org/

[27] B. Pfaff, J. Pettit, T. Koponen, E. J. Jackson, A. Zhou, J. Rajahalme, J. Gross, A. Wang, J. Stringer, P. Shelar *et al.*, "The design and implementation of open vswitch," in *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2015.

[28] J. McCauley, "The pox network software platform," 2015. [Online]. Available: https://github.com/noxrepo/pox

[29] "GOOSE protocol parser for Zeek IDS," 2020. [Online]. Available: https://github.com/smartgridadsc/Goose-protocol-parser-for-Zeek-IDS

[30] S. HomChaudhuri and M. Foschiano, "Cisco systems' private vlans: Scalable security in a multi-client environment (rfc 5517)," Tech. Rep., 2010.

[31] T. M. Breslin, D. Kucharczyk, and J. A. Hinshaw, "Method, apparatus and system for inserting a vlan tag into a captured data packet," Sep. 9 2014, uS Patent 8,832,222.

[32] S. A. Naiksatam, K. Jiang, G. M. Maier, S. Ramasubramanian, S. D. Modi, R. W. Sherwood, M. S. Dhami, and M. Cohen, "Systems and methods for performing network service insertion," Jan. 17 2017, uS Patent 9,548,896.

[33] H. Li, H. Hu, G. Gu, G.-J. Ahn, and F. Zhang, "vnids: Towards elastic security with safe and efficient virtualization of network intrusion detection systems," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security.* ACM, 2018, pp. 17–34.

[34] S. K. Fayaz, Y. Tobioka, V. Sekar, and M. Bailey, "Bohatei: Flexible and elastic ddos defense." in *USENIX Security Symposium*, 2015, pp. 817–832.

[35] R. Kumar, M. Hasan, S. Padhy, K. Evchenko, L. Piramanayagam, S. Mohan, and R. B. Bobba, "End-to-end network delay guarantees for real-time systems using sdn," in *Proceedings of the Real-Time Systems Symposium (RTSS).* IEEE, 2017, pp. 231–242.

[36] X. Dong, H. Lin, R. Tan, R. K. Iyer, and Z. Kalbarczyk, "Software-defined networking for smart grid resilience: Opportunities and challenges," in *Proceedings of the ACM Workshop on Cyber-Physical System Security (CPSS).* ACM, 2015, pp. 61–68.