Traffic Analysis in Support of Hybrid SDN Campus Architectures for Enhanced Cybersecurity

William Brockelsby
NC State University Department of Computer Science
Duke University Office of Information Technology
Email: wjbrocke@ncsu.edu

Rudra Dutta
NC State University
Department of Computer Science
Email: rdutta@ncsu.edu

***Abstract*—The scale and complexity of campus networks continues to accelerate due to recent paradigms such as the Internet of Things (IoT) resulting in a heightened awareness of the need for enhanced cybersecurity. Traditional cybersecurity approaches such as the placement of firewalls and other policy enforcement mechanisms at strategic choke points effectively divide the network into zones and are unable to regulate intrazone host-to-host communication. This traditional approach introduces significant risk as there is little in place to prevent the horizontal propagation of malware or other unwanted traffic within a given zone. In this paper we explore approaches for improving cybersecurity in campus networks by analyzing contemporary campus traffic patterns and propose several architectural enhancements in light of these patterns which introduce strategically placed hardware or hardware-accelerated software data planes which are evaluated from performance and effectiveness perspectives.**

***Index Terms*—cybersecurity, campus network architecture**

I. Introduction

Data networks have exploded in terms of scale and complexity to meet modern demands for connectivity and this trend is expected to increase as additional cyberphysical devices are developed and the Internet of Things (IoT) begins to take shape. While networks are growing significantly in response, so too has the awareness of the need for improved cybersecurity, as expectations of privacy, confidentiality, and predictability grow apace. The remainder of this section introduces the challenges of providing pervasive cybersecurity in traditional campus networks by briefly reviewing traditional approaches.

Traditional campus networks were designed to support the client/server computing architecture [1] and over time a three-tier network architecture typically referred to as core, distribution, access became the defacto standard for deploying campus networks [2]. This architecture, shown in Fig. 1, is scalable to permit evolutionary growth as the needs of the organization change and can provide highly available network connectivity to critical resources [2].

The network forwarding elements within this architecture typically consist of switches performing layer 2 forwarding at the access layer, multi-layer switches at the distribution layer forwarding based on layers 2 and 3 followed by routers at the core forwarding based on layer 3 [2]. Policy enforcement mechanisms such as firewalls, intrusion detection/prevention systems and other devices that perform specialized functions are often deployed at the network perimeter or at strategic "choke" points within the network [3]. These specialized devices are often deployed to partition the network into zones and enforce policies on inter-zone traffic [3]. The resulting unchecked intra-zone host-to-host communication sometimes referred to as east/west traffic is natively supported by fundamental protocols such as Ethernet and TCP/IP but introduces significant cybersecurity risk: unchecked host-to-host communication could be leveraged for the horizontal propagation of malware or other malicious use cases.

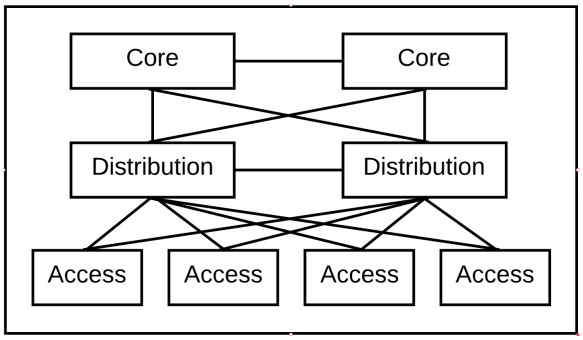

Fig. 1. Core, Distribution, Access Architecture

In our prior work on network forensics [4] we made a serendipitous discovery that naturally occurring east/west host-to-host traffic was less prevalent within a variety of buildings serving diverse use cases at NC State University. This paper further explores our initial discovery by analyzing traffic characteristics in buildings serving a variety of use cases at Duke University to see if our initial finding was unique to NC State or more widespread. After discussing the traffic patterns observed in both campus networks, we propose a hybrid SDN campus architecture in support of enhanced cybersecurity that leverages opportunities offered by the observed traffic characteristics.

In §II and §III, we provide the aforementioned traffic analysis followed by the architectural approaches in §IV. Within §V, we evaluate the performance of the proposed architecture by analyzing data collected while evaluating equipment from multiple vendors within a network laboratory and evaluate effectiveness, from a cybersecurity perspective, by examining worm propagation under various scenarios.

978-1-7281-7705-2/21/$31.00 ©2021 IEEE

II. Prior Work

We developed a series of ratios to explore the volume of traffic ingressing and egressing a network forwarding element through the analysis of interface octet counters recorded by most network management systems [5], [6] in our prior work on network forensics [4]. These ratios use the phrase "uplink" to refer to interfaces on the device that are closer to the core of the network and "downlink" to refer to interfaces on the device that are closer to hosts at the access layer of the network as shown below:

Ratio-1: $\sum UplinkIngress : \sum DownlinkEgress$ [4]
Ratio-2: $\sum UplinkEgress : \sum DownlinkIngress$ [4]

When Ratio-1 ≈ 1 and Ratio-2 ≈ 1: The volume of traffic ingressing the uplinks is approximately equal to the volume egressing the downlinks while simultaneously the volume of traffic egressing the uplinks is approximately equal to the volume of traffic ingressing the downlinks [4]. In practice, Ratio-1 $\lesssim 1$ and Ratio-2 ≈ 1 is similar to the previous scenario due to the arrival of broadcast or multicast traffic on the uplink interfaces that must be replicated for egress on the downlink interfaces [4]. These scenarios are consistent with a predominantly north/south traffic pattern with little to no east/west traffic [4]. At NC State, buildings associated with use cases such as academic, residential, clinical, dining and administrative showed little to no east/west traffic [4].

III. Further Analysis

To further understand the prevalence of east/west traffic within campus networks, and to check whether it might be peculiar to the NC State campus network only, we extend our prior work [4] by exploring architectural similarities and differences within building networks at both universities and explore the ratios presented previously for a number of diverse buildings at Duke. While both university networks leverage the standard core, distribution and access layer architecture described in in Fig. 1 they instantiate the architecture in different ways as highlighted by Figs. 2-3.

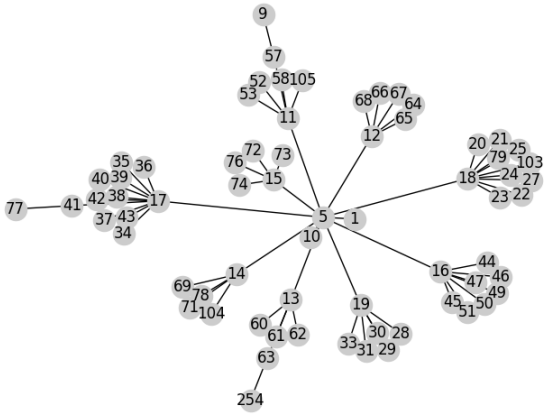

Fig. 2. NC State - Example Building Single-Line Topology

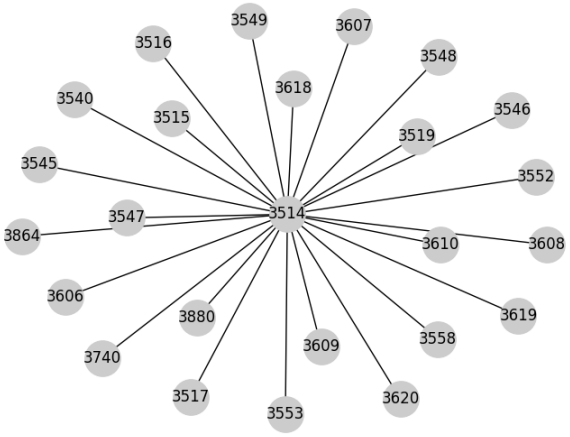

Fig. 3. Duke - Example Building Single-Line Topology

At NC State, buildings are typically layer 2 and leverage a topology consisting of hierarchical aggregation switches: access layer aggregation switches are located within each intermediate distribution frame to aggregate traffic from local access layer switches while a building aggregation switch within the building distribution frame aggregates traffic from the access layer aggregation switches and uplinks the building to a router in a main distribution frame within the same geographic region of campus. As shown in Fig. 2, node 5 represents a building aggregation switch and nodes 11-19 represent access layer aggregation switches while the remaining nodes are access layer switches.

In contrast, many buildings at Duke leverage a resilient router within the building distribution frame to attach the building to geographically diverse core routers and to aggregate and route traffic from access layer switch stacks within intermediate distribution frames. As shown in Fig. 3, node 3514 represents the resilient building distribution router and the other nodes represent dual-attached switch stacks containing multiple access layer switches.

While the topologies in Figs. 2-3 look significantly different, they are fairly similar from a layer 2 perspective in that the functionality provided by the access layer aggregation switches at NC State is natively provided by the access layer switch stacking technology deployed at Duke. In our earlier work at NC State and in the new analysis at Duke, we obtain Ratios 1-2 by querying network management systems to obtain network topology and octet counter information for layer 2 switches in a variety of diverse building types. Note that at Duke, for consistency, this does not include the layer 3 router used within many buildings that is not present within most buildings at NC State. The node-level ratio values for an example building at Duke, that aligns with Fig. 3, are presented within Table I. Ratio data for Node 3864 is not provided within Table I as valid data was unavailable at the time of the study. Further summary statistics for Ratios 1-2 across buildings supporting a variety of use cases at Duke are presented within Tables II-III.

An examination of the mean and median values for the summary statistics across academic, administrative, library and

TABLE I
DUKE EXAMPLE BUILDING - RATIO ANALYSIS

Node	Ratio-1	Ratio-2	Node	Ratio-1	Ratio-2
3515	0.96	1.02	3558	0.94	1.00
3516	0.98	1.02	3606	0.98	1.01
3517	0.99	1.02	3607	0.99	1.02
3519	0.98	1.01	3608	0.95	1.02
3540	0.97	1.02	3609	0.99	1.01
3545	0.98	1.01	3610	0.98	1.01
3546	0.97	1.02	3618	0.93	1.02
3547	0.98	1.02	3619	0.90	1.01
3548	0.96	1.01	3620	0.98	1.01
3549	0.98	1.01	3740	0.97	1.02
3552	0.98	1.01	3880	0.98	1.01
3553	0.98	1.02	3864	N/A	N/A

TABLE II
DUKE RATIO 1 - SUMMARY STATISTICS

Facility	Min	Max	Mean	Median
Academic	0.71	1.05	0.91	0.88
Administrative	0.68	1.01	0.94	0.96
Library	0.92	0.96	0.95	0.95
Residential	0.90	0.99	0.97	0.98

residential facilities are consistent with a network consisting of primarily north/south traffic with some broadcast and/or multicast replication of frames/packets from the distribution layer of the network towards the access layer. More interestingly, the data from NC State [4] aligns with the data collected from a variety of buildings at Duke. This encourages the view that prevalence of north-south traffic (and rarity of east-west traffic) is typical of campus networks, and arises from the use case applications faced by those networks, rather than the architecture or implementation details of a particular campus network. This view is all the more persuasive since the two campus networks we examined have somewhat different architectures as described above.

With these traffic insights from contemporary campus networks in mind, we now explore architectural options in support of enhanced cybersecurity.

IV. ARCHITECTURAL OPTIONS

While current developments make it paramount to improve cybersecurity in campus networks, and to provide a significantly expanded and flexible set of cybersecurity options, the reality of campus networks also imposes serious constraints upon architectural approaches that can be attempted. First of all, as with most enterprise networks, the fundamental protocols used within modern packet switched data networks such as Ethernet and TCP/IP, although not designed to natively support modern cybersecurity objectives such as microsegmentation and zero-trust networking, are too deeply entrenched for any serious consideration of replacement; and as witnessed by the slow deployment of IPv6 over two decades, it is unlikely even that significant updates can be made to these pervasive protocols. Although solutions such as Virtual LANs (VLANs) at layer 2 and Virtual Routing and Forwarding (VRF) instances at layer 3 have worked well for decades, these approaches can only be leveraged to provide coarse segmentation and have limited scalability. Campus networks are often budget constrained, so that significant addition of costly hardware resources is not an option: nor can a blanket replacement of existing

TABLE III
DUKE RATIO 2 - SUMMARY STATISTICS

Facility	Min	Max	Mean	Median
Academic	0.89	1.03	0.99	1.00
Administrative	1.00	1.08	1.02	1.01
Library	1.00	1.02	1.01	1.01
Residential	1.00	1.02	1.01	1.01

equipment with SDN counterparts be countenanced. The tried and tested nature of equipment from incumbent vendors is also likely to be preferred over a changeover to extensive use of SDN equipment instead. Thus, any practical solution has to involve minimum touch-points, leverage a maximum (ideally all) existing equipment, and introduce either a few SDN boxes in very few places, or very lightweight SDN boxes, or appliances, at a larger number of places. Lastly, each campus network has its own specific architecture within the general CDA architecture, as we have seen above in the contrast between NC State and Duke University networks, and to be generally useful, an approach has to be applicable across these variations.

In this section we review some existing solutions such as Private VLANs which can be used to minimize or restrict host-to-host communication but point out some challenges with these approaches which may prevent pervasive real-world deployment. With this background in place we then propose a hybrid SDN architecture to improve cybersecurity within campus networks which aligns with our traffic analysis findings in the previous section, but remains practical.

A. Existing Approaches

This subsection briefly reviews some existing approaches:

1) Private VLANs: Private VLANs were originally developed by Cisco as a value-add feature to allow administrators to control host-to-host communication within a layer 2 network. Years later Cisco released additional implementation details for private VLANs via non-standards track informational RFC 5517 in 2010 [7] although by that time many other vendors had developed similar proprietary technology. The implementation of Private VLANs does not provide isolation directly within a single VLAN but instead leverages a hierarchical relationship between VLANs: A "primary" VLAN is associated with "promiscuous ports" while a second "isolated" VLAN and optionally one or more "community" VLANs are in turn associated with the primary VLAN [7]. Although private VLAN technology has been available for over a decade there are still a number of restrictions and caveats associated with the technology which may vary across vendors and even across platforms from the same vendor - some examples from a recent Cisco access platform include but are not limited to [8]:

- VoIP voice VLANs are not supported
- Basic 802.1X authentication is supported but modern features such as "802.1X with port security, voice VLAN, or per-user ACL" are not supported
- Link Aggregation Control Protocol (LACP) is not supported

The proprietary nature of private VLANs, varying support, associated restrictions and caveats can make real world implementation difficult to achieve.

2) 802.1X Extensions: 802.1X is an IEEE standard for port-based network access control first published in 2001 and most recently updated in January, 2020 [9]. Early versions of the standard focused on providing an authentication mechanism for devices attaching to the network and vendors implementing 802.1X often added proprietary features or extensions not specified within the IEEE standard [9] some of which were introduced in later revisions. Some early examples of vendor proprietary extensions included support for non-802.1X compatible client systems through MAC address bypass or the specification of dynamic access control lists to filter host traffic entering the network. Over time, some of these optional vendor-specific capabilities were described in other standards such as RFC4849 - RADIUS Filter Rule Attribute which describes an approach to specify dynamically delivered access control lists at the time of authentication [10]. The proliferation of vendor proprietary extensions makes deployment in a multi-vendor or even single-vendor multi-platform environment challenging as highlighted within RFC4849 "[...] a RADIUS server cannot automatically discover whether a NAS supports the NAS-Filter-Rule attribute. A legacy NAS not compliant with this specification may silently discard the NAS-Filter-Rule attribute while permitting the user to access the network [8]."

B. Hybrid SDN Architecture

While there are many approaches to realize Software Defined Networking (SDN), the separation of the control and forwarding planes, as popularized by OpenFlow [11] and later P4 [12], permits developers to deploy novel forwarding policies through forwarding abstractions. When the OpenFlow specification was first released over a decade ago [13], many features were optional and a number of vendors added an OpenFlow agent onto existing forwarding elements resulting in feature and scale limited implementations. As an example, some switches supporting OpenFlow were limited to ∼100-2000 forwarding entries (flows) [14] requiring SDN capable switches to be positioned close to or at the access layer [15], [16]. As highlighted in Figs. 1-3, the access layer within typical campus networks comprises the largest quantity of forwarding elements thereby introducing high cost and potentially risk when introducing significant paradigm shifts. More recently, advanced purpose-built SDN forwarding elements have come to market which can scale up to a million wildcard forwarding entries [17] under certain pipeline deployment scenarios or support high-density policy-rich 100Gb/s forwarding with OpenFlow or P4 [18]. Interestingly, new intelligent Network Interface Cards (NICs), informally referred to as SmartNICs, have come to market from a variety of vendors that are capable of offloading and accelerating Open Virtual Switch (OVS) [19] SDN deployments on commodity servers some of which are capable of offloading nearly half a million of flows in hardware [20].

By leveraging the high flow table capacity and high throughput interfaces of these next-generation SDN forwarding elements, we propose a hybrid campus architecture as illustrated within Fig. 4.

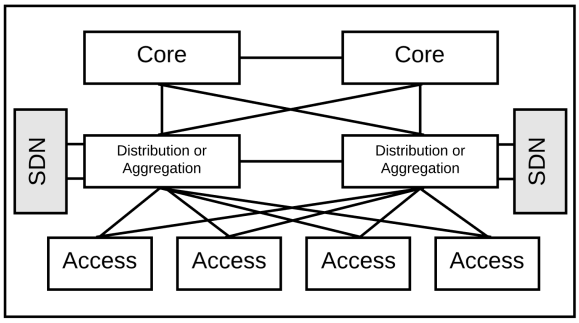

Fig. 4. Hybrid SDN Architecture

The hybrid architecture augments existing traditional networking infrastructure with high capacity SDN forwarding elements at the distribution (or aggregation) layer of the network to provide a mechanism upon which rich policies can be deployed to enhance campus cybersecurity. The SDN forwarding elements are ideally located within each building to facilitate horizontal scalability and attach to the layer 2 aggregation switch (in the case NC State) or distribution router (in the case at Duke) but, depending on the campus architecture, policy density, and throughput requirements, could also be deployed at regional distribution routers serving a number of buildings. In either case, the SDN forwarding elements should be inserted at layer 2 either at or just before the layer 2-3 boundary (distribution router). Insertion of the SDN appliances at layer 2 permits rich policies to be deployed in support of enhanced cybersecurity as all supported protocol fields can be leveraged as match conditions. These approaches are illustrated further in Fig. 5.

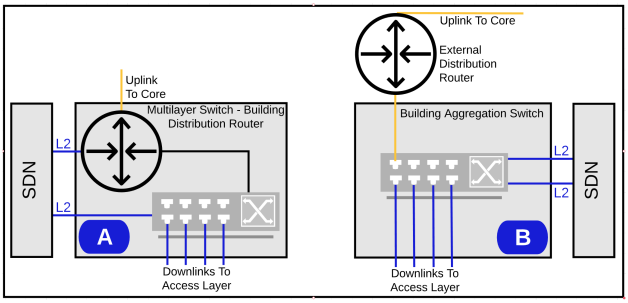

Fig. 5. SDN Node Insertion Strategies

In scenario (A), within Fig. 5, the building leverages a multi-layer switch as the distribution router and the SDN forwarding element attaches directly via layer 2 interfaces. In typical building distribution router scenarios, router interfaces are often virtual constructs that are "connected" to VLANs by way of integrated routing and bridging configuration. To plumb traffic associated with downstream access layer switches through the SDN forwarding element, we review two options, as below.

Option (1): Remove the virtual router interfaces from the associated VLAN and instead leverage a physical router interface with 802.1Q layer 3 sub-interfaces per VLAN which

is then connected to a layer 2 interface on the SDN forwarding element. In addition, a layer 2 802.1Q trunk containing multiple VLANs is also connected to the SDN forwarding element as shown. This option is applicable only to Scenario (A) above.

Option (2): This option is applicable to both Scenarios (A) and (B), within Fig. 5 where a layer 2 aggregation switch is used within the building instead of a local distribution router. In Option (2), assuming we want to work with VLANs {X,Y,Z}, we ensure no router interface is attached and extend these VLANs to the SDN forwarding element via an 802.1Q trunk. Next, we create a new set of VLANs {X', Y', Z'} and attach them to the SDN forwarding element via an 802.1Q trunk. In this way, the SDN forwarding element can perform VLAN translation by mapping $X \leftrightarrow X'$, $Y \leftrightarrow Y'$ and $Z \leftrightarrow Z'$ where a router interface is located within {X', Y', Z'} and not {X, Y, Z}. An additional benefit of these insertion strategies is that the SDN forwarding element can be inserted on a per-VLAN basis facilitating insertion over time in support of specific use cases or to minimize risk by inserting the SDN forwarding elements gradually. This insertion strategy, as described so far, provides the ability to instantiate rich policies to regulate traffic inter-subnet and inter-VLAN traffic but does not address the regulation of intra-VLAN flows within the same subnet.

In the rest of this section, we discuss intra-vlan regulation and the various strengths of this architectural approach.

Generality: As described within §IV-A, proprietary approaches introduce significant complexity in multi-vendor environments or even across product lines from single vendor. We explored scenarios that would restrict host-to-host east/west traffic within the access layer with the goal of identifying a strategy that avoids the use of vendor-specific proprietary features. In our prior work on network automation [21], we found that unnecessary configuration complexity within the access layer often led to operational problems and wanted to ensure that our solution would lend itself towards automation to eliminate these concerns. After reviewing the capabilities of access layer devices from multiple vendors, we realized that a universal solution applicable to the broadest set of installed access layer devices was to assign each access layer port on each access layer switch to a unique VLAN. By assigning each port on the switch to a unique VLAN, we can be assured that no host-to-host east/west flows will be permitted as each VLAN provides an isolated broadcast domain. All of the unique VLANs on a switch are then added to the 802.1Q uplink interface towards the distribution/aggregation layer where the frames will flow through the SDN forwarding element as described previously. The SDN forwarding element can then map the 802.1Q tags from the access layer to an 802.1Q VLAN with the appropriate router interface if the traffic should be forwarded to the default gateway or to the correct access layer 802.1Q VLAN tag if host-to-host east/west flows are expected and approved by policy. The approach essentially eliminates local switching within access layer switches and forces the decision to be made within the SDN forwarding element at the distribution (or aggregation) layer thereby allowing fine-grained policies to be implemented. The elimination of local switching within the access layer may improve cybersecurity but sounds as if it could introduce a performance penalty in that traffic switching must take place at one level higher within the topology. Interestingly, by referring back to the trends identified within §II-III, we noted that natural east/west flows are not prevalent across a variety of buildings located within two different campus networks. This observation allows our restriction of east/west traffic within campus networks to improve cybersecurity without introducing a significant performance impact as the predominant flows within contemporary campus networks remain unaffected.

Scalability: Although the solution is elegant, depending on the characteristics of the deployment environment, a potential scalability limit may be encountered depending on the quantity of access layer interfaces associated with the SDN forwarding element. Given that 802.1Q VLAN tags are 12-bits, we have 2^{12} = 4096 VLANs available. Many access layer switches and distribution routers support up to approximately 48 interfaces per 1RU device. As an example, consider a hypothetical deployment scenario consisting of quantity 48 48-port access layer switches attached to a distribution router: 48*48 = 2,304 VLANs required which is well below the 4096 limit and leaves sufficient VLANs available for mapping to multiple subnets as needed. Note that stackable and chassis-based switches can quickly add density. In particularly large buildings where the number of switching ports would require more than 4096 VLANs, a divide-and-conquer approach can be adopted where the building would leverage multiple distribution (or aggregation) devices each associated with dedicated SDN forwarding elements. In a very large facility, there may in fact be some benefit to this approach, in that it minimizes risk in terms of operational maintenance and equipment failure. Nonetheless, we present an additional option that could be leveraged in this scenario which does not require additional distribution nodes.

It is possible to leverage a stack of 802.1Q VLAN tags often referred to as QinQ [22] for the case when two VLAN tags are applied. This technique is common in service provider networks in metro Ethernet deployments where different customers specify potentially overlapping VLAN tags on their circuits which must be forwarded uniquely within the provider network. By leveraging QinQ, the service provider can use a unique outer VLAN tag to represent distinct customer circuits which may have overlapping inner VLAN tags [22]. Although originally envisaged as a service provider technology, many enterprise switches, even some dating back over 15 years, have support for QinQ [23] so it remains a viable option for our proposed architecture as long as the selected SDN forwarding element also supports QinQ. In this scenario, ports on each access layer switch will be set to a unique VLAN except now the need for unique VLANs across access layer switches connected to the same distribution router is no longer a requirement. This scenario simplifies the situation in that the same port on each access layer switch can be set to the same VLAN. Uniqueness is achieved by pushing a second unique

VLAN on single-tagged frames egressing the uplink towards the distribution layer. With this approach, the outer tag must be unique to each access layer switch, stack or chassis. From the perspective of the SDN forwarding element, the outer VLAN tag can serve as an index pointing to the switch while the inner VLAN tag can serve as an index to the port on the selected switch. This dual indexing scheme permits the development of rich policies that can be granular to the host, switch or port level if the appropriate mapping is maintained.

Configuration Action Elimination: Both of the previous approaches facilitate streamlined network automation at the access layer as VLANs become static once set on each access layer switch. As mentioned in our prior work on network automation [21], we leveraged two processes to deliver a zero-touch installation experience of access layer switches: The first process prepared new switches by installing device-specific image and configuration and the second process set ports to specific VLANs when new patch cables were connected by consulting a historical mac-to-subnet binding database [21]. This work could enhance our prior strategy by ensuring that once an access layer switch is imaged and configured, there would no longer be any reason to deploy any further configuration changes as all host-to-vlan (and implicitly host-to-subnet) mapping can now take place within the SDN forwarding element itself. By eliminating the need to routinely update access layer switch configuration significant development savings can be achieved as it would no longer be required to develop complex device-specific modules for multiple vendors and models of access layer switches in support of full automation. We feel that this is a significant contribution as many groups within the community are currently working towards solving these device specific automation challenges [24].

V. Evaluation

Within this section we evaluate the performance of the proposed architecture by analyzing latency and loss while qualifying equipment from multiple vendors across multiple traffic patterns and throughput levels in a network laboratory. We then evaluate effectiveness, from a cybersecurity perspective, by examining synthetic worm propagation under various scenarios to see if the proposed architecture enhances cybersecurity through the introduction of policy.

A. Performance Evaluation

To assess the performance of the overall solution, we constructed an evaluation topology as illustrated within Fig. 6 consisting of a typical building access and distribution hierarchy leveraging the Cisco Catalyst 9300 [25] and Catalyst 9500 [26] respectively. A Spirent TestCenter [27] was used to generate traffic according to different profiles and throughput rates to measure latency and loss in both a traditional (non-SDN) topology as well as the proposed hybrid topology. For the hybrid topology several next-generation SDN data planes were evaluated including the NS2122 [17] hardware data plane from NoviFlow which leverages a Mellanox NP-5 network processor, an Edge-Core Wedge 100BF-32X [18] which leverages a Barefoot Networks Tofino SDN ASIC with NoviFlow OS and Open vSwitch (OVS) [19] version 2.11.0 on CentOS 7.7 Linux hosts utilizing Intel Xeon Silver 4110 (8-Core 2.1GHz, 11M Cache) processors with 64GB of RAM with a 25Gb/s Agilio SmartNIC from Netronome [28] which accelerates OVS in hardware by leveraging an on-NIC network processor.

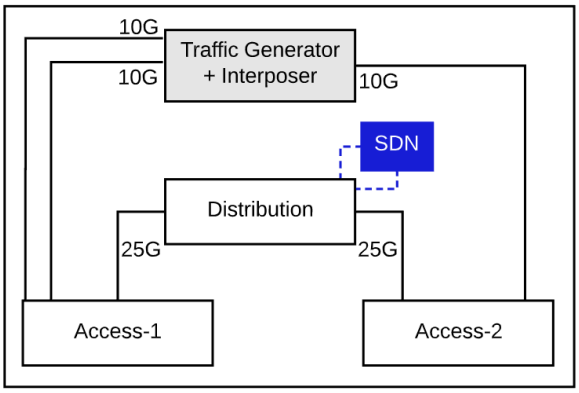

Fig. 6. Performance Evaluation Topology

We leveraged the Spirent TestCenter to generate emulated host-to-host traffic between access-layer switch ports with unidirectional rates of 0.25, 0.5, 1, 2.5, 5 and 9 Gb/s leveraging three different traffic patterns consisting of (1) fixed 128-byte frames, (2) fixed 9000-byte jumbo frames and (3) Spirent Internet Mix (IMIX-IPSEC) consisting of 58.67% 90-Byte frames, 2% 92-Byte frames, 23.66% 594-Byte frames and 15.67% 1418-Byte frames [27]. The columns within result Tables IV-VII consist of the traffic pattern and rate mentioned above while columns prefixed with "L:" refer to the latency in microseconds as follows: L:T traditional non-SDN latency, L:O OVS+Netronome latency, L:N NoviFlow NS2122 latency, L:E Edge-Core/Tofino latency and Loss indicates % frame loss.

TABLE IV
Single Switch Port-to-Port - Same Subnet

Pattern	Gb/s	L:T	L:O	L:N	L:E	Loss %
128	0.25	12.6	28.72	29.39	25.34	0.00
128	0.5	12.6	28.73	33.37	25.34	0.00
128	1	12.6	28.73	32.31	25.34	0.00
128	2.5	12.58	28.72	35.72	25.32	0.00
128	5	12.92	28.8	32.58	25.34	0.00
128	9	12.72	29.43	31.4	25.43	0.00
9K	0.25	27.7	56.93	66.51	59.98	0.00
9K	0.5	27.7	56.93	66.5	59.98	0.00
9K	1	27.7	56.93	66.5	59.98	0.00
9K	2.5	27.7	57.11	66.5	59.97	0.00
9K	5	27.7	56.99	66.51	59.98	0.00
9K	9	27.7	56.8	66.38	60.19	0.00
IMIX	0.25	13.06	29.67	30.37	26.34	0.00
IMIX	0.5	13.06	29.67	30.58	26.34	0.00
IMIX	1	13.06	29.74	32.17	26.44	0.00
IMIX	2.51	13.12	30.09	33.93	26.82	0.00
IMIX	5.02	13.45	30.93	35.11	27.74	0.00
IMIX	9.03	16.96	35.2	39.48	32.34	0.00

Table IV results include port-to-port latency and loss between emulated hosts within the same subnet attached to

TABLE V
SINGLE SWITCH PORT-TO-PORT - DIFFERENT SUBNET

Pattern	Gb/s	L:T	L:O	L:N	L:E	Loss %
128	0.25	20.81	36.65	37.94	29.87	0.00
128	0.5	20.81	36.65	45.82	29.87	0.00
128	1	20.81	36.71	44.07	29.88	0.00
128	2.5	20.78	36.71	49.45	29.85	0.00
128	5	20.81	36.79	44.19	29.87	0.00
128	9	20.88	42.82	42.04	29.97	0.00
9K	0.25	41.43	72.29	91.57	78.5	0.00
9K	0.5	41.43	72.29	91.57	78.51	0.00
9K	1	41.43	72.29	91.57	78.5	0.00
9K	2.5	41.43	72.28	91.58	78.51	0.00
9K	5	41.44	72.32	91.58	78.51	0.00
9K	9	41.44	72.95	91.71	78.51	0.00
IMIX	0.25	21.43	37.89	39.34	31.25	0.00
IMIX	0.5	21.43	37.92	39.89	31.25	0.00
IMIX	1	21.44	38.11	43.16	31.49	0.00
IMIX	2.51	21.57	38.72	46.35	32.2	0.00
IMIX	5.02	22.08	39.99	48.1	33.6	0.00
IMIX	9.03	25.87	44.99	52.97	38.69	0.00

TABLE VI
INTER-SWITCH - SAME SUBNET

Pattern	Gb/s	L:T	L:O	L:N	L:E	Loss %
128	0.25	19.75	27.67	28.34	24.28	0.00
128	0.5	19.75	27.68	32.33	24.29	0.00
128	1	19.75	27.67	31.27	24.28	0.00
128	2.5	19.75	27.69	34.69	24.29	0.00
128	5	19.75	27.76	31.49	24.3	0.00
128	9	19.75	28.43	30.27	24.3	0.00
9K	0.25	39.8	55.44	65.02	58.48	0.00
9K	0.5	39.8	55.44	65.02	58.48	0.00
9K	1	39.81	55.44	65.01	58.48	0.00
9K	2.5	39.8	55.44	65.02	58.48	0.00
9K	5	39.81	57.26	65.03	58.48	0.00
9K	9	39.81	55.31	66.75	58.73	0.00
IMIX	0.25	20.32	28.57	29.26	25.24	0.00
IMIX	0.5	20.33	28.58	29.48	25.24	0.00
IMIX	1	20.33	28.64	31.09	25.33	0.00
IMIX	2.51	20.44	28.96	32.87	25.71	0.00
IMIX	5.02	20.92	29.86	34.03	26.63	0.00
IMIX	9.03	24.62	34.23	38.44	31.31	0.00

Access-1 within Fig. 6. In this scenario, traffic is switched locally in the traditional topology whereas in the hybrid architecture traffic must flow to the SDN data plane as each port is on a unique VLAN. As shown within Table IV, there is no loss for any scenario and only a slight (and expected) increase in latency for all traffic patterns and rates when leveraging the hybrid architecture. Table V covers the same scenario as above except that the emulated hosts are in different subnets and inter-subnet routing takes place at the distribution layer. In this scenario, local switching is not possible in the traditional topology so latency is higher for that scenario as expected.

Tables VI-VII results include inter-switch latency and loss between emulated hosts attached to switches Access-1 and Access-2 in Fig. 6 within the same or different subnets respectively. In these scenarios, traffic must always flow through the distribution layer whether switched or routed. As shown within Tables VI-VII, there is no loss for any scenario and only a slight (and expected) increase in latency for the hybrid architecture.

Note that the latency is higher in the inter-subnet routing scenarios described in Tables V, VII as the traffic must pass through the SDN forwarding element twice due to inter-subnet routing. From a performance perspective, we can see that the introduction of the SDN forwarding elements only introduces a small amount of additional latency (less than 51 µs in all scenarios). Next we will evaluate the cybersecurity effectiveness of the proposed policy-driven architecture.

B. Cybersecurity Effectiveness

To assess the cybersecurity effectiveness of the proposed architecture we developed a synthetic worm in Python [29] that emulates a process listening on a TCP socket with a hypothetical security vulnerability. When the synthetic worm receives a specially crafted message to trigger the worm functionality, the process scans for vulnerable processes on hosts within the same and adjacent subnets in an attempt to trigger the worm functionality on the discovered hosts. The synthetic worm was attached to Hosts 11-13 (same subnet)

TABLE VII
INTER-SWITCH - DIFFERENT SUBNET

Pattern	Gb/s	L:T	L:O	L:N	L:E	Loss %
128	0.25	19.75	35.59	36.88	28.81	0.00
128	0.5	19.76	35.59	43.77	28.81	0.00
128	1	19.75	35.6	43.01	28.82	0.00
128	2.5	19.75	35.68	48.42	28.82	0.00
128	5	19.75	35.77	43.09	28.82	0.00
128	9	19.76	41.86	40.9	28.85	0.00
9K	0.25	39.95	70.8	90.08	77.01	0.00
9K	0.5	39.95	70.8	90.08	77.01	0.00
9K	1	39.95	70.8	90.08	77.01	0.00
9K	2.5	39.95	71.69	90.08	77.01	0.00
9K	5	39.95	71.69	90.13	77.01	0.00
9K	9	39.95	73.04	90.8	77.01	0.00
IMIX	0.25	20.34	36.79	38.22	30.15	0.00
IMIX	0.5	20.34	36.83	38.78	30.15	0.00
IMIX	1	20.34	37.02	42.04	30.39	0.00
IMIX	2.51	20.45	37.62	45.22	31.11	0.00
IMIX	5.02	20.92	38.93	46.94	32.5	0.00
IMIX	9.03	24.62	44.02	51.76	37.66	0.00

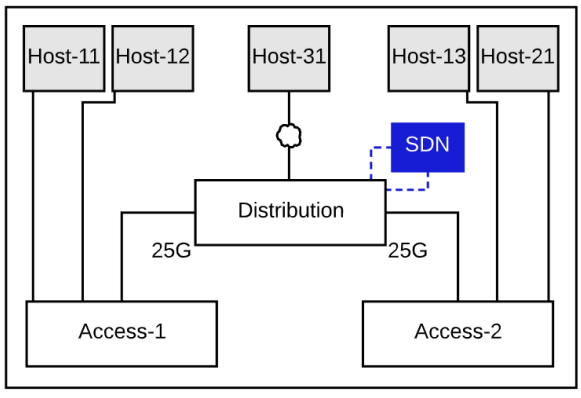

Fig. 7. Cybersecurity Effectiveness Topology

and 21 (different subnet) within Fig. 7. Our cybersecurity effectiveness results are summarized within Table VIII. As shown, with the traditional non-SDN approach, all hosts are compromised by intra and inter-subnet reachability. In the hybrid architecture, all traffic is dropped by default unless permitted by policy. In this test scenario, we permit all

TABLE VIII
CYBERSECURITY EFFECTIVENESS

Scenario	Attack Source	Attack Target	Compromises
Traditional	Host-31	Host-11	Host-11-13, Host 21
SDN-L3	Host-31	Host-11-13, 21	None
SDN-L3	Host-11	Host-12	Host-12-13
SDN-L3	Host-21	Host-11	None
SDN-L2+L3	Host-31	Host-11	None
SDN-L2+L3	Host-11	Host-12	None
SDN-L2+L3	Host-11	Host-21	None
SDN-L2+L3	Host-21	Host-11	None

traffic but implement an ACL blocking traffic destined to the affected TCP port. With the SDN-L3 approach that still permits local switching within the access layer, horizontal propagation within the same subnet is possible and Hosts 12-13 were compromised when Host-11 was used to launch the attack which represents a common scenario whereby someone brings a machine from outside onto a trusted network. The final SDN-L2+L3 scenario that forces all traffic through the SDN appliance thereby eliminating unchecked local switching dropped all malicious traffic resulting in zero additional compromises.

VI. CONCLUSION AND ACKNOWLEDGEMENT

This work builds upon an observation made in our prior study of traffic characteristics at NC State [4] by extending the analysis to a variety of buildings at Duke. The observation of minimal naturally occurring host-to-host east/west traffic flows within contemporary campus networks across a variety of buildings supported the development of a hybrid architecture that blends traditional (existing) cyberinfrastructure components with a smaller number of next generation SDN forwarding elements in support of rich policies that can be used to improve cybersecurity by eliminating unchecked host-to-host communication often used for the horizontal propagation of malware and has been shown to introduce minimal performance overhead under a variety of traffic patterns and throughput rates. Moreover, the melding of network and computing services within the campus architecture delivers a foundation upon which further innovation can be delivered.

A portion of this research is based upon work supported by the National Science Foundation under Grant No. 1925550 at Duke University [30] and a deployment of the proposed architecture is currently underway in support of existing and future research use cases at Duke.

REFERENCES

[1] W. Stallings, *Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud*. Addison-Wesley Professional, 2015.

[2] C. Systems, "Campus Network for High Availability Design Guide," 2008.

[3] R. Cameron, B. Woodberg, P. Giecco, T. Eberhard, and J. Quinn, *Junos Security: A Guide to Junos for the SRX Services Gateways and Security Certification*. " O'Reilly Media, Inc.", 2010.

[4] W. Brockelsby and R. Dutta, "A Graded Approach to Network Forensics with Privacy Concerns," in *2019 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2019, pp. 292–297.

[5] K. McCloghrie and F. Kastenholz, "RFC 2863-"The Interfaces Group MIB"," *IETF, June*, 2000.

[6] K. McCloghrie, "RFC 1229-"Extensions to the generic-interface MIB"," *IETF*, 1991.

[7] S. HomChaudhuri and M. Foschiano, "Cisco Systems' Private VLANs: Scalable Security in a Multi-Client Environment," 2010.

[8] C. Systems, *VLAN Configuration Guide, Cisco IOS XE Gibraltar 16.12.x (Catalyst 9300 Switches)*, 2019.

[9] "IEEE Standard for Local and Metropolitan Area Networks–Port-Based Network Access Control," *IEEE Std 802.1X-2020 (Revision of IEEE Std 802.1X-2010 Incorporating IEEE Std 802.1Xbx-2014 and IEEE Std 802.1Xck-2018)*, pp. 1–289, 2020.

[10] P. Congdon, M. Sanchez, and B. Aboba, "RADIUS Filter Rule Attribute," *RFC 4849 (Proposed Standard)*, pp. 1–9, 2007.

[11] N. McKweown, "OpenFlow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, 2008.

[12] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese *et al.*, "P4: Programming protocol-independent packet processors," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 87–95, 2014.

[13] *OpenFlow Switch Specification Version 1.0.0*, Dec 2009. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.0.0.pdf

[14] *Cisco Plug-in for OpenFlow Configuration Guide for Catalyst 3850 and 3650 Series Switches*, Jan 2018. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3850/software/release/37e/ consolidated_guide/b-openflow-37e-3850-and -3650/b-openflow-37e-3850_chapter_01.html

[15] M. Hoit, R. Dutta, W. Brockelsby, and G. Sparks, "CC-NIE Networking Infrastructure: Data Intensive e-Science and SDN at NCSU," Aug 2013. [Online]. Available: https://www.nsf.gov/awardsearch/showAward?AWD_ID=1340609

[16] T. Futhey and J. Board, "CC-NIE Network Infrastructure: Using Software-Defined Networking to Facilitate Data Transfer," Aug 2012. [Online]. Available: https://www.nsf.gov/awardsearch/showAward?AWD_ID=1246042

[17] *NoviFlow NS2122 Data Sheet*, Nov 2019. [Online]. Available: https://noviflow.com/wp-content/uploads/2019/11/NoviSwitch-2122-Datasheet-400_V5.pdf

[18] *Edge-Core Wedge 100BF-32X Data Sheet*, Dec 2019. [Online]. Available: https://www.edge-core.com/_upload/images/Wedge100BF-32X_65X_DS_R05_20191210.pdf

[19] *Open Virtual Switch (OVS)*. [Online]. Available: http://openvswitch.org/

[20] *Agilio Open vSwitch TC User Guide*. [Online]. Available: https://help.netronome.com/support/solutions/articles/36000081172-agilio-open-vswitch-tc-user-guide

[21] W. Brockelsby and S. Dilda, "Tactical Network Automation with NetZTP and One Shot," in *2019 IEEE 40th Sarnoff Symposium*. IEEE, 2019, pp. 1–3.

[22] J. Tony, "IEEE 802.1 AD," 2006.

[23] *Catalyst 3750 Switch Software Configuration Guide, 12.2(52)SE*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3750/software/release/12-2_52_se/configuration/guide/3750scg/swtunnel.html

[24] Internet2, "Network Automation and Cloud Networking & Security Workshops." [Online]. Available: https://sites.google.com/iu.edu/2020network-cloud-sec-workshop/

[25] *Cisco Catalyst 9300 Data Sheet*, Feb 2020. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-9300-series-switches/nb-06-cat9300-ser-data-sheet-cte-en.html

[26] *Cisco Catalyst 9500 Data Sheet*, Sep 2020. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-9500-series-switches/nb-06-cat9500-ser-data-sheet-cte-en.html

[27] *Spirent Test Center Data Sheet*, Feb 2020. [Online]. Available: https://assets.ctfassets.net/wcxs9ap8i19s/6ymR9bySVmV2KRvRZBv7zr/c83fc72c4ce2cda485af15404701c34a/Spirent_Platform_Brochure.pdf

[28] *Netronome Agilio CX 2x25G Data Sheet*, Jul 2018. [Online]. Available: https://www.netronome.com/m/documents/PB_Agilio_CX_2x25GbE.pdf

[29] G. Van Rossum *et al.*, "Python Programming Language," in *USENIX annual technical conference*, vol. 41, 2007, p. 36.

[30] T. Futhey, R. Biever, M. Gorlatova, and W. Brockelsby, "CC* Integration: Archipelago: Linking Researchers On-Campuses and in the Cloud through SDN-Enabled Microsegmentation," Nov 2019. [Online]. Available: https://www.nsf.gov/awardsearch/showAward?AWD_ID=1925550