Published in: NetSoft 2016 - 2016 IEEE NetSoft Conference and Workshops
DOI: 10.1109/NETSOFT.2016.7502484

Empirical Comparison of Virtualized and Bare-Metal Switching for SDN-based 5G Communication in Critical Infrastructures

Fabian Kurtz, Nils Dorsch, Christian Wietfeld
Communication Networks Institute, TU Dortmund, Otto-Hahn-Strasse 6, 44227 Dortmund
Email: {fabian.kurtz, nils.dorsch, christian.wietfeld}@tu-dortmund.de

***Abstract*—Advancements in the field of Critical Infrastructures (CI), such as the advent of Smart Grids or automated transportation systems, offer new services and functionalities. However these novel Cyber Physical Systems (CPS) necessitate increasingly complex monitoring and control schemes, which are commonly orchestrated by industrial Supervisory Control and Data Acquisition (SCADA) systems. As thousands of distributed field devices are involved in the operation of Critical Infrastructures, the enabling communication technologies need to fulfil exacting performance requirements as well as provide high levels of robustness, Quality of Service, flexibility and scalability. The ongoing evolution of cellular LTE (Long Term Evolution) towards 5G networks promises to meet all these specifications. In this context Software-Defined Networking (SDN) is set to play a crucial part in achieving those goals and is thus included in many proposed 5G architectures. However, basic performance characteristics allowing a comparison of the two prevalent approaches to SDN, fully virtualised and hardware based Bare-Metal switching, are currently not widely available. Therefore this paper proposes a test-platform for the benchmarking of SDN as well as a prototypical architecture. The latter serves to facilitate the evolutionary development of LTE towards 5G for use in Critical Infrastructure communication in general and Smart Grids in particular. A comparative evaluation of switching performance, realized via the described testbed, is presented and an assessment is given. Based on the introduced architecture and testbed, future work will develop new mechanisms for end-to-end slicing as well as application aware scheduling.**

I. Introduction

Critical Infrastructures, such as the electrical grid or transportation systems, are undergoing fundamental changes as they evolve towards *Cyber Physical Systems*. *Smart Grids* for example involve the coordination of potentially millions of entities like so called *Customer Energy Management Systems* (CEMS) or *Automated Meter Reading* (AMR) devices, which facilitate the advanced features provided by the electrical system [1]. Here advancements in functionality are driven by the need to adapt to the fluctuating power generation of renewable energy sources, such as wind turbines or photovoltaics. These developments challenge the stability of the electrical grid and require a drastic increase in monitoring and control to guarantee reliable and safe operation. This in turn necessitates robust and flexible communication networks for handling the new information flows [2]. Additionally, the deployment of communication infrastructures for exclusive use by critical applications is associated with high costs. Therefore the shared use of public, cellular infrastructure is highly desirable from a financial view. Yet, existing solutions like LTE in its current form are unlikely to meet all requirements of emerging Critical Infrastructures such as Smart Grids. The next generation of cellular networks, i.e. 5G, however is expected to support these applications. Hence this paper proposes an architecture for extending and evolving LTE towards 5G. The authors use this design as a basis for researching advanced technologies, such as Software-Defined Networking, within the area of critical communications. As key performance indicators need to be thoroughly assessed to judge the suitability of SDN for Critical Infrastructure communication, this paper focuses on a performance evaluation of two crucial paradigms in switching: Bare-Metal and fully virtualized switching. This document is structured as follows. First an overview of related work is given in Section II. Section III describes the concept and architecture for the evolution of LTE towards 5G thereby enabling CI communication. Software-Defined Networking and its application in CI is also introduced in this Section. The SDN testbed, which is part of the proposed architecture, is introduced in Section IV, after which benchmark results of virtual and Bare-Metal switching are discussed. Section V provides a conclusion and gives an outlook on future work.

II. Related Work

Although Software-Defined Networking is an active topic in communication research, there is only limited work available on its use in communication architectures for Critical Infrastructures. The authors of [3] performed a scalability analysis of SDN by benchmarking off-the-shelf computer components with tasks such as the calculation of prime numbers. While the results yield an estimation of performance boundaries applicable to virtual switches, no real world applications or actual SDN protocols were used. A benchmark for evaluating SDN-Controllers is presented in [4]. By setting up multiple virtual switches the authors generate different traffic flows for evaluation of SDN-Controllers. However, evaluations of both new approaches to the data plane, i.e. Bare-Metal and virtualized switches (c.f. *Network Function Virtualization* (NFV)) are not fully developed. Hence this paper aims to provide empirical data based on actual Smart Grid protocols to uncover potential performance issues impacting the suitability of SDN for Critical Infrastructures communications.

978-1-4673-9486-4/16/$31.00

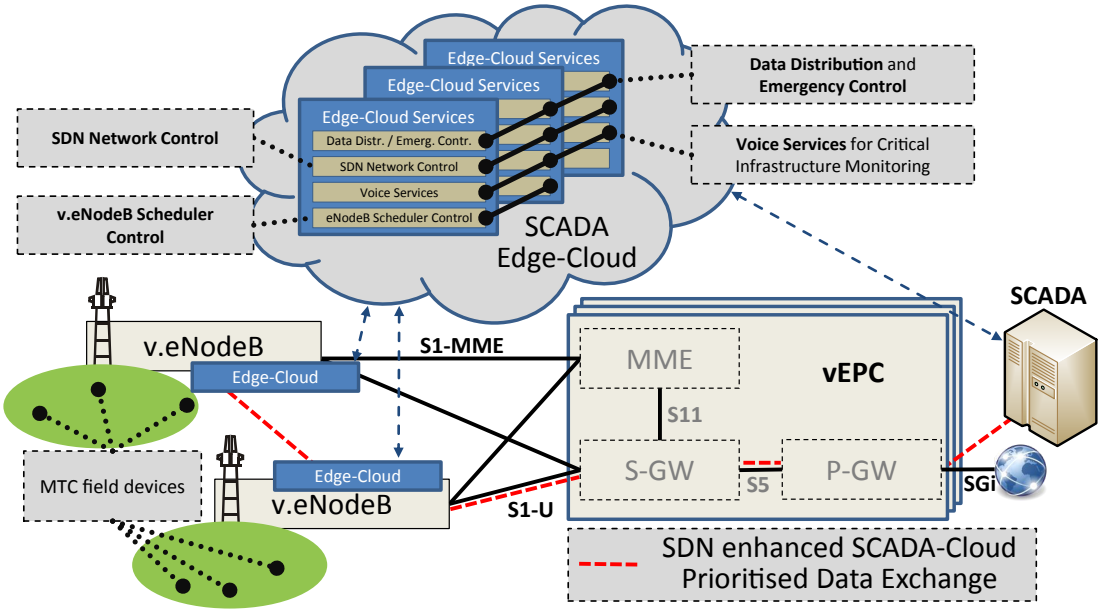

Fig. 1: Architecture for Development of 5G Solutions on the Basis of LTE for Use in Critical Infrastructure Communications

III. Concepts for the Evolution of LTE towards 5G for Critical Infrastructures

This section outlines the architectural concept for evolving LTE towards 5G for Critical Infrastructure communication. Central to this is the SDN testbed, suitable for prototypical 5G development, as discussed in section IV-A.

A. Architectural Concept and Use Case

Critical Infrastructures become increasingly dominated by *Information and Communication Technology* (ICT) as they evolve to meet ever increasing demands. Requirements of Smart Grids for example include highly reliable real-time monitoring and control services of large numbers of devices with a sophisticated *Quality-of-Service* (QoS) level. A solution to attain the desired performance levels more cost efficiently than with the creation of dedicated networks, is the shared used of public communication networks. However, critical traffic must be prioritized to provide the required performance. Therefore Figure 1 gives an overview of the architectural concept developed for the evolution of LTE towards 5G communication in Critical Infrastructures. Starting from the left it can be seen how *Machine-Type-Communication* (MTC) devices are distributed throughout the field. This serves to reproduce the high level of Machine-to-Machine traffic that is being generated e.g. via *Intelligent Electrical Devices* (IED) as they occur in Smart Grids. Measurement values and control data is either exchanged between these nodes, or is transmitted to higher layers of the architecture, such as SCADA systems, for processing. However, latency caused by the number of network devices traversed from endpoint to endpoint of such a communication chain, as well as resulting from propagation delay, can be too high to be acceptable for the application. An example in this context is the *IEC 61850* [5] protocol. Originally developed for use in Smart Grid sub-station automation its scope has continuously widened and is set to become a grid-wide standard, including the SCADA domain [6]. *Manufacturing Message Specification* (MMS) [7], *Sampled Value* (SV) [8] and *Generic Object Oriented Substation Event* (GOOSE) [5] are the three messaging mechanisms defined within IEC 61850. The latter two are directly encapsulated into the Ethernet *Media Access Control* (MAC) frame, carry raw measurement values respectively event data and require End-to-End latencies of below 10 ms. Since this requirement is to be fulfilled regardless of communication disturbances or outages, our approach uses SDN to guarantee timely message delivery. Furthermore, due to the high criticality of latencies, an edge-cloud based solution is proposed. By virtualizing the LTE eNodeBs (v.eNodeB), SCADA system functions can be moved into the newly created *edge-cloud*, thus shortening the physical distance to field devices hence reducing latencies. Associated services of the SCADA edge-cloud are data distribution and emergency control, voice services for critical infrastructure monitoring as well as SDN network control and v.eNodeB scheduler control. This adds flexibility to the communication network and Critical Infrastructure services alike. Specifically SDN reduces configuration complexity while facilitating the prioritization of critical traffic flows even during high network utilization. By combining Software-Defined Radios at the air interfaces with SDN and NFV towards a dynamic, *virtualized Evolved Packet Core* (vEPC), a fully *Software-Defined Infrastructure* (SDI) is achievable.

B. Software-Defined Networking for Critical Infrastructures

Traditionally Critical Infrastructures are conservative in their adoption of new strategies and technologies. With the ongoing, rapid transition to smart architectures however the need for new approaches to the associated communication infrastructures arises [2]. Here Software-Defined Networking with its separation of data and control plane, as well as

the introduction of a central controller instance, promises a high level of adaptability while reducing complexity. With its *North-Bound Interface* CI applications can convey their requirements in terms of communication directly to the SDN-Controller, which in turn configures the network appropriately. Meanwhile the *East/West-Bound Interfaces* allow multiple SDN controllers to cooperate, making the transition between different networks more seamlessly. Combined with the planned use of public communication infrastructures for services such as Smart Grid data transmission, this interaction is a crucial feature as it allows the coupling of exclusive and non-exclusive networks while retaining the required QoS. Ahead of widespread deployment however, the suitability of Software-Defined Networking for use in Critical Infrastructure communication has to be proven conclusively. In order to achieve this, the different approaches to packet switching have to be considered first. The arrival of SDN triggered a shift from hitherto vertically integrated, closed devices to open platforms. This shift is parallel to a movement within the IT industry to provide standardized platforms capable of running open source software [9]. From this development so called *Bare-Metal* switches emerged, which are based on traditional *Application Specific Integrated Circuits* (ASIC) for physical switching, but also run open source operating systems (e.g. Linux) to fully expose the hardware's capabilities. Network Function Virtualization also impacts the development of SDN, as it can be based on virtualized switching devices and controllers. In this regard Open vSwitch [10] is a software which allows off-the-shelf computers to perform switching tasks. As these approaches to switching allow for greater flexibility compared to more traditional, closed devices, they lend themselves better to the concept of SDN and thus are the focal point of this paper. In previous work [11] the authors of this paper identified the detection of physical disconnections to be a critical factor in overall communication restoration time. Therefore the subsequent Section dissects the performance of Bare-Metal and virtualized switching in this context.

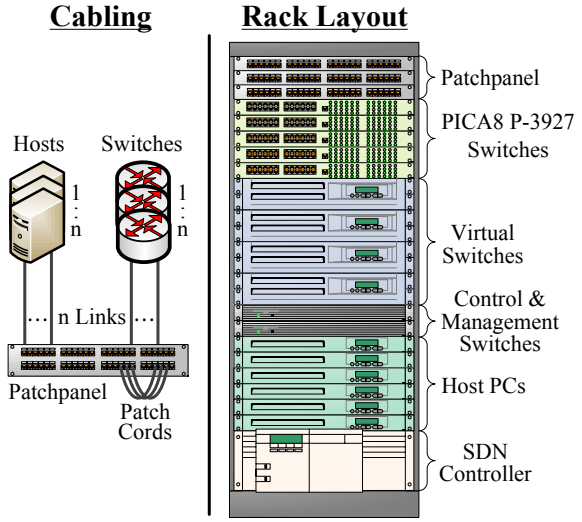

Fig. 2: Setup of the *SDN4CriticalInfrastructures* Testbed with Cabling (left) and 19-Inch Rack Equipment (right)

IV. Testbed and Results

This section presents the testbed created for the evaluation and development of Software-Defined Networking for Critical Infrastructure applications, with a strong focus on Smart Grid Communications. Results concerning the performance of virtual and Bare-Metal switching hardware in Packet Forwarding and Fast Failover scenarios are discussed and a comparative analysis of both approaches is given.

A. The CNI SDN for Critical Infrastructures Testbed

The active components of the CNI *SDN for Critical Infrastructures* (SDN4CI) testbed comprise eleven Ethernet (IEEE 802.3ab 1000Base-T) switches, six hosts for traffic generation and one SDN-Controller. These devices are distributed across three separate networks for data, control and maintenance to avoid any interference between the architecture under test and the measurement equipment. As shown in Figure 2 nine switches form the data network, with four of them being workstations configured as virtual switches. These run Ubuntu Server 14.04.3 64Bit (3.13.0-32-generic Kernel) with Open vSwitch 2.4.1 and are equipped with a third generation quad core Intel Core i7 CPU, an onboard Intel I217-LM as well as a 4 Port Intel I350 Ethernet *Network Interface Card* (NIC). In addition, five Bare-Metal Pica8 P-3297 switches running PicOS 2.6.32 (based on Debian Linux) with Open vSwitch 2.3.0 are part of the testbed. Traffic can be injected into the data network via six identical hosts operating either as server or client. They run Ubuntu Mate 14.04.3 (4.0.9-generic Kernel), connect to the data network trough onboard Intel I210-AT NICs and are Intel Celeron J1900 based. As can be seen from figure 2, the testbed utilizes patch panels which link at the back to the hosts and switches of the data network. This way network architectures can be configured freely just by connecting ports on the panels instead of at the switches, as indicated by the left side of Figure 2. Furthermore this implementation shifts the burden of connector wear-out from costly equipment to relatively inexpensive patch panels. Switches and SDN controller communicate via an out-of-band network (i.e. separate links using the Intel I210-AT and I217-LM interfaces) distributed via a Zyxel GS1900-24E switch. The *SDN for Critical Infrastructures* Controller runs on Ubuntu Mate 14.04.03 (4.0.0 low latency Kernel) and features several enhancements over its Floodlight [12] basis. It links to the control network through an Intel I217-V NIC and utilizes the OpenFlow protocol [13], the de facto standard for interaction with switching devices [14].

The third network is for maintenance and is designed to allow for configuration and management of the SDN network, independently of ongoing measurements in the control or data plane. Here remote access to the hosts is provided through *Secure Shell* (SSH) and an *Intelligent Platform Management Interface* (IPMI), which provides a full *Graphical User Interface* (GUI) via a second, identical Zyxel switch. Performance characteristics of this auxiliary switch were measured with the help of the setup given in Figure 4. Packet Forwarding requires on average $48.346\,\mathrm{ms}$ with low jitter, while port mirroring incurs a delay of $1.47\,\mu\mathrm{s}$ at the mirrored port.

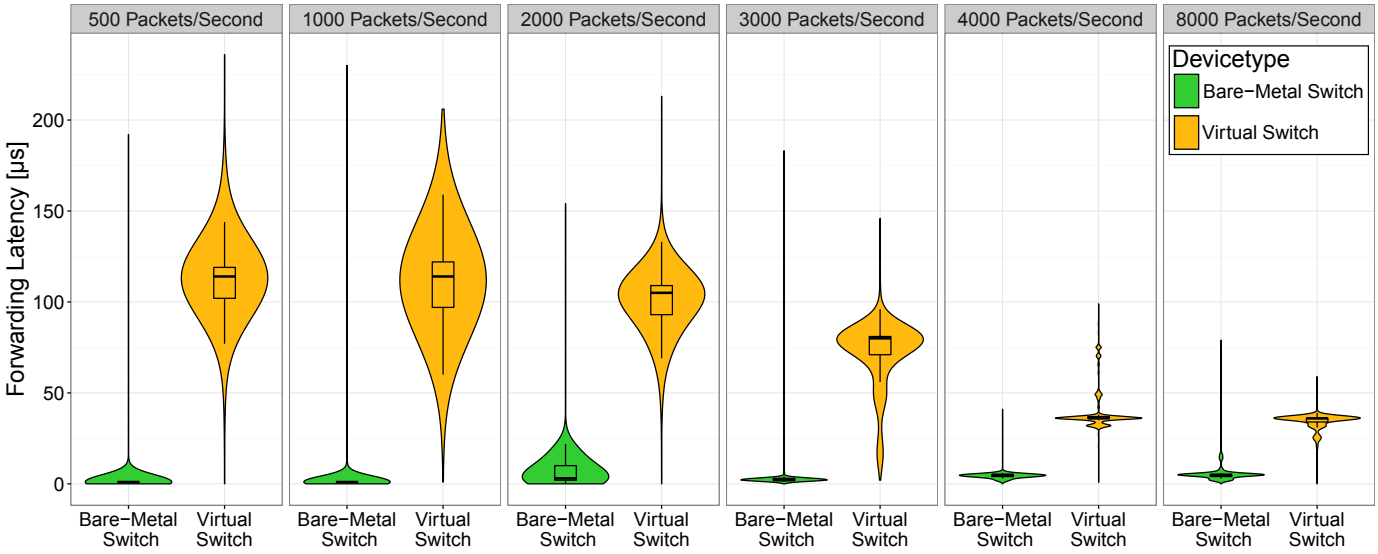

Fig. 3: Packet Forwarding Latency for Typical Inter-Transmission-Times in Critical Infrastructure Communications

B. Scenario 1: Packet Forwarding Performance

Packet Forwarding Delay is a key performance indicator of switched communication networks. It is crucial in achieving low End-to-End latencies with minimal jitter, as required by critical communication services. Hence this section discusses the forwarding performance of virtual and Bare-Metal switch devices. Figure 4 depicts the setup used for methodology validation, as well as measurement of packet forwarding latencies. A Host PC configured as traffic generator sends *User Datagram Protocol* (UDP) packets with a size of 60 Byte to the measurement computer. *Inter-Transmission-Times* (ITT) from 2 ms down to 125 µs (i.e. packet-rates of 500 to 8000 packets per second) are used to recreate typical message intervals employed within IEC 61850 based Smart Grid communication [5]. A Zyxel GS1900-24E switch acts as port mirror and mirrors all outgoing packets to a second link which connects the measurement computer's NIC 1. Meanwhile the same data enters the switch under test via the forwarded link and is subsequently switched to NIC 2. Packet Forwarding Delay at the switch under test thus calculates as the time delta between the arrival of packets at both NICs. An advantage of this approach is that synchronization of multiple clocks is not necessary as only one computer with one clock is required for data collection. Propagation delay can be neglected as link

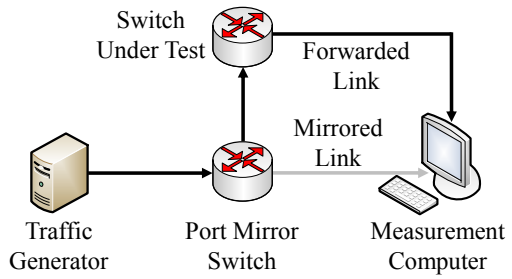

Fig. 4: Packet Forwarding Latency Measurement Setup

lengths from mirror and switch under test to the measurement PC are equal and thus do not affect the results. Also, the link between both switches has a length of 2 m, resulting in about 10 ns of propagation delay [15]. This is too low to have a meaningful impact on observed performance. Additionally, forwarding latency of the mirror switch does not impact measurements, as results show it to affect both output ports equally. Measurements for each parameter set were performed for at least one minute in steady state to gain the sample volume necessary for reliable results. The forwarding delay of virtual and Bare-Metal switching, as observed with the setup described above, is given by Figure 3. It can be seen, that the Bare-Metal device consistently provides significantly lower latencies, regardless of packet-rate. Moreover the distribution of values indicates far lower jitter and thus more stable performance. Averages range from 5.785 µs at 125 µs ITT to 1.366 µs at an ITT of 2 ms. A peak of 6.475 µs exists at 500 µs ITT. This performance level is in line with non-SDN-capable switches [16] and closes in on the precision achievable with the test setup (c.f. 1.47 µs port mirroring delay, here not taken into account). Noteworthy is the non-linear relationship of latencies to ITT and the broader distribution of values at 2000 packets per second. Individual outliers spike above the values of the virtual switch, but can be attributed to rare variances in packet generation and recording precision. In comparison latencies of the virtual switch average between a maximum of 113.290 µs at 2 ms ITT and a minimum of 34.295 µs at an ITT of 125 µs. These values are at least one order of magnitude higher than those of the Bare-Metal devices. However, contrary to the latter, latencies decrease with an increase in packet rate. The same applies to the distribution of latencies, which tends to decrease with shorter ITTs. This behaviour is counter-intuitive as higher ITTs put more stress on the equipment. This observation could be caused by an adaptive buffering-scheme of the NIC or Linux. However, as evident from Figure 3, performance is never on a par with Bare-Metal switches.

C. Scenario 2: Fast-Failover in Critical Infrastructures

Critical Infrastructures such as Smart Grids are dependent on reliable communication networks. Therefore End-to-End connectivity needs to be ensured in all cases. As device and link outages, the latter being the focus of the following discussion, occur due to several causes (e.g. failures or attacks) and can have a high impact on the systems availability [17] a mechanism for fast recovery of communication needs to be established. This *Fast-Failover* functionality is realized by pre-computing alternative routes for all active traffic flows at the SDN-Controller. In case a link failure is detected by the switch it notifies the controller. Here alternative paths are determined by a look-up procedure, after which the required re-routing commands are send to the affected switches. As shown in previous work [11], the SDN-Controller performs this operation on average within $2.01\,\mathrm{ms}$ from reception of an OFPortStatus message to the transmission of OFFlowMods. For virtual switches the delay for re-establishment of End-to-End connectivity however was assessed to be $360.64\,\mathrm{ms}$. The interruption of message reception is evaluated at the client in order to arrive at the values presented here, which are crucial for communication in Critical Infrastructures. In order to determine if Bare-Metal switches are able to reduce this delay without additional means for link failure detection the following measurements are presented.

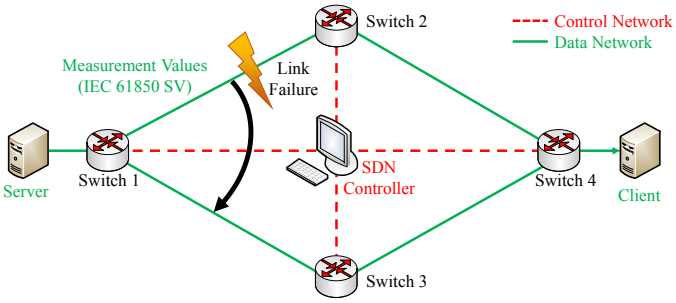

Fig. 5: Network Topology for the Fast-Failover Scenario

Figure 5 presents the topology used for measurement of End-to-End delay in controller driven failover scenarios. Four Bare-Metal respectively four virtual switches are connected into a diamond shaped data network, with one server attached to switch 1 and a client at switch 4. Thus two routes from server to client, namely 1-2-4 and 1-3-4, are available for the transmission of SV messages. All switches connect to the SDN-Controller via a dedicated out-of-band control network as described in Section IV. Realistic Smart Grid traffic conditions are provided by a custom SV message service developed at the Communications Networks Institute. For the purposes of this paper the Inter-Transmission-Time is set to $1\,\mathrm{ms}$ with packet sizes of $122\,\mathrm{Byte}$, conforming to the transmission of SV measurement samples. In order to achieve a high level of confidence in the results, every measurement is repeated over 100 times. Link failures are simulated by physically disconnecting the cable connecting switch 1 with switch 2 or 3, depending on the route used by the injected traffic. The failover delay, as observed at the client device,

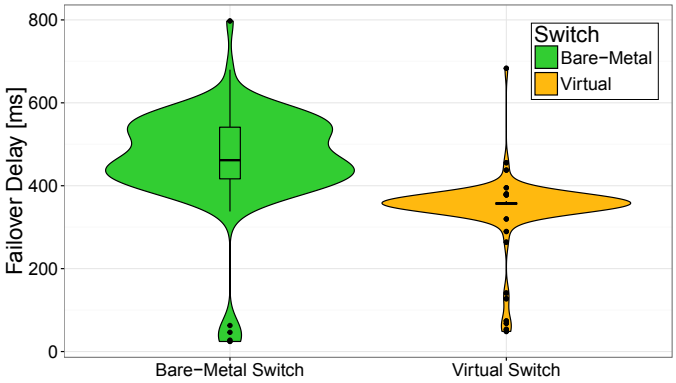

Fig. 6: Failover Delay at the Receiver for IEC 61850 Sampled Value Based Measurements

is shown by Figure 6. A Bare-Metal switch based network shows a comparatively broad distribution of failover durations and achieves an average failure mitigation time of $467.119\,\mathrm{ms}$ with a peak of $797.494\,\mathrm{ms}$. Contrary to the previous packet forwarding scenario virtual switches yield higher performance compared to their counterparts, with an average delay of $346.024\,\mathrm{ms}$ and a significantly narrower distribution of values. This result confirms previous work (i.e. $360.64\,\mathrm{ms}$) [11] and indicates stable performance of Open vSwitch and the SDN Controller. It can be concluded, that the failover delay in its current state is too large to be suitable for the use of controller driven failover in Critical Infrastructure communication in general and in IEC 61850 based Smart Grids specifically.

D. Comparison of Bare-Metal and Virtual Switching

Results from the previous sections allow an insight into the differences in performance between Bare-Metal and fully virtualized switching. While packet forwarding was shown to be a point of weakness for virtual devices, they outperformed their counterparts in restoring communication in cases of link failures. Starting with the latter scenario, delay caused by the SDN-Controller factors into the observed values, but this applies for both device types. Furthermore pure switching was shown to be slower on virtual devices and thus does not contribute to the performance advantage. Therefore the lower latency in failover can be attributed to the mechanisms employed for physical link failure detection. An explanation for the vastly faster packet forwarding of Bare-Metal switching seen in Section IV-B, can be found in the use of the *cut-through* switching paradigm typically employed by high-performance devices for use in latency sensitive applications [18]. To decrease latencies, *cut-through* capable devices forward frames directly after the reception of the first $12\,\mathrm{Byte}$ of the Ethernet header. The more common, slower switching method *store-and-forward* however, stores the entire incoming frame, calculates the checksum and then forwards the data. While latency is significantly reduced, *cut-through* devices will forward damaged frames and can thus cause problems in large broadcast domains. *Ternary Content-Addressable Memory* (TCAM) integrated in hardware switches serves to improve forwarding address lookup times and also contributes

TABLE I: Comparison of Switching Device Paradigms

	Virtual Switches	Bare-Metal Switches
Forwarding Latency	-	+
Forwarding Jitter	+	+
Failover Delay	+	-
Throughput	-	+
Port Density	-	+
Feature Set Flexibility	+	-
Computing Resources	+	-
Forwarding Table Size	+	-

to the observed performance. A qualitative comparison of both switching paradigms is given in Table I.

Additionally to the latency and delay observations of the previous sections, further differences between the two discussed paradigms have to be noted. In the domain of backbone and core-networks, port density and the associated throughput per rack unit is an important performance metric. Here virtual switches, built of commodity parts are limited in their capabilities because of their inherently larger size and the available NICs. Given enough space it would be feasible to attach multiple NICs with high port counts, but the expansion slots and required bandwidth between them put a low limit on this strategy. Nevertheless virtual switches deployed at the edges of a network, as proposed in Section III-A, are not limited by port counts. Moreover this switching paradigm supports NFV via the integration of edge-cloud computing tasks directly on the same underlying hardware as the virtualized switch, thereby severely reducing latencies to the end-user or field device. Another differentiator is the flexibility in the available feature sets. Although the Bare-Metal design increases flexibility compared to traditional switches (e.g. reducing vendor lock-in), ASIC manufacturers need to provide open-source *Software Development Kits* (SDK) to allow software developers to fully exploit the capabilities of the hardware. Here open-source based, fully virtualized devices have greater adaptability but at the cost of performance in some areas, as shown in this paper. Compared to hardware based switches, virtual devices typically have far greater computational resources (c.f. IV-A), supporting their flexibility.

V. Conclusion and Future Work

An architecture for the development of LTE towards prototypical 5G networks for use in Critical Infrastructure communication is presented and discussed. A centerpiece of this topology is the Software-Defined Networking testbed in which baseline performance indicators for Bare-Metal and fully virtualized switching are measured. The measurements given in this paper provide baseline performance data which will serve as a basis for subsequent work. While Bare-Metal switching has significant advantages over traditional, closed and vertically integrated devices, performance is not always as high as achievable with fully virtualized switches. Specifically it is shown that in both cases the failover delay is not sufficient to meet the stringent requirements of Critical Infrastructure communications. From this it is inferred, that failure detection and controller driven solutions, as the one employed in this paper, are too limited in performance for the use cases at hand. Hence failure detection mechanisms such as *Bidirectional Forwarding Detection* (BFD) and *heart-beat messages* will be the focus of future work. The use of local *Failover Groups* is a potential solution to achieve failover delays comparable to *Multi Protocol Label Switching* (MPLS) [19] and will be analysed in subsequent studies.

Acknowledgement

This work has been carried out in the course of research unit 1511 *'Protection and control systems for reliable and secure operations of electrical transmission systems'*, funded by the German Research Foundation (DFG) and the Franco-German Project *BERCOM* (FKZ: 13N13741) co-funded by the German Federal Ministry of Education and Research (BMBF).

References

[1] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart Grid - The New and Improved Power Grid: A Survey," *IEEE COMMUNICATIONS SURVEYS AND TUTORIALS*, vol. 14, no. 4, pp. 944–980, 2012.

[2] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, "A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 5–20, 2013.

[3] A. Kondel and A. Ganpati, "Evaluating System Performance for handling scalability challenge in SDN," in *Green Computing and Internet of Things, International Conference on*, Oct. 2015, pp. 594–597.

[4] M. Jarschel, F. Lehrieder, Z. Magyari, and R. Pries, "A Flexible OpenFlow-Controller Benchmark," in *Software Defined Networking (EWSDN), 2012 European Workshop on*, Oct. 2012, pp. 48–53.

[5] IEC TC57, *IEC 61850: Communication Networks and Systems for Power Utility Automation*, International Electrotechnical Commission.

[6] H. Falk, *SCADA GOOSE Messaging*, 2010. [Online]. Available: http://data.utc.org/meetings/presentations/canada_2010/Falk.pdf.

[7] IEC TC57, *IEC 61850-8-1: Specific Communication Service Mapping (SCSM) - Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3*, International Electrotechnical Commission.

[8] IEC TC57, *IEC 61850-9-1: Specific Communication Service Mapping (SCSM) - Sampled values over serial unidirectional multidrop point to point link*, International Electrotechnical Commission.

[9] *Open Compute Project*. [Online]. Available: www.opencompute.org/.

[10] *Open vSwitch*. [Online]. Available: openvswitch.org.

[11] N. Dorsch, F. Kurtz, H. Georg, C. Hägerling, and C. Wietfeld, "Software-Defined Networking for Smart Grid Communications: Applications, Challenges and Advantages," in *IEEE International Conference on Smart Grid Communications*, Nov. 2014, pp. 422–427.

[12] *Floodlight*, http://www.projectfloodlight.org/floodlight.

[13] Open Networking Foundation, *OpenFlow Switch Specification Version 1.3.0*, 2012. [Online]. Available: https://www.opennetworking.org/.

[14] N. McKeown *et al.*, "OpenFlow: Enabling Innovation in Campus Networks," *SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[15] Siemens, "Latency on a Switched Ethernet Network," Ontario Canada, App. Note 8. [Online]. Available: w3.siemens.com/mcms/industrial-communication/en/rugged-communication/Documents/AN8.pdf.

[16] Y. Yang, *Understanding Switch Latency*, http://www.cisco.com/c/en/us/products/collateral/switches/nexus-3000-series-switches/white_paper_c11-661939.pdf, Jun. 2012.

[17] G. Andersson *et al.*, "Causes of the 2003 Major Grid Blackouts in North America and Europe, and Recommended Means to Improve System Dynamic Performance," *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1922–1928, 2005.

[18] David Newman, Network World, *Latency and jitter: Cut-through design pays off for Arista, Blade*, Jan. 2010. [Online]. Available: http://www.networkworld.com/article/2241573/virtualization/latency-and-jitter--cut-through-design-pays-off-for-arista--blade.html.

[19] Y. Lei, C.-H. Lung, and A. Srinivasan, "A Cost-Effective Protection and Restoration Mechanism for Ethernet Based Networks: an Experiment Report," in *Workshop on High Performance Switching and Routing*, 2004, pp. 350–354.