Published in: ICTC 2017 - 2017 International Conference on Information and Communication Technology Convergence
DOI: 10.1109/ICTC.2017.8191064

Advanced Controller Resiliency in Software-Defined Networking Enabled Critical Infrastructure Communications

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

***Abstract*— Critical Infrastructures such as transportation or energy systems are central to the existence of modern societies. Due to the expected level of performance and developments like the shift towards renewable, fluctuating energy generation in Smart Grids, advanced monitoring and control systems are required for stable operation. This in turn increases the dependence on robust communication technology. However, due to the distributed nature of Critical Infrastructures, dedicated communication networks entail high costs. Hence a convergence of public and purpose built networks is pursued, so mission critical applications can be handled by a shared infrastructure. Software-Defined Networking facilitates this by separating network control from physical packet forwarding and centralizing it in a controller, which also creates a single point of failure. Although approaches capable of mitigating this threat to system robustness exist, the specific requirements of e.g. Smart Grids are not sufficiently addressed. Therefore a novel broker-based approach, allowing multiple Software-Defined Networking controllers to work in concert, is presented. The broker determines the correctness of network control decisions via majority votes. This novel concept for control plane fault tolerance, specifically designed to fulfil the challenging demands of Smart Grids, is detailed and contrasted with currently employed solutions. Also, resiliency against failures as well as malicious attacks is achieved, enabling reliable and highly available Critical Infrastructure communications.**

I. Introduction

Modern societies are built on the continuous availability of assets such as transportation, health services as well as water and energy supply. Due to their vital importance for day to day life, theses services are designated as Critical Infrastructures (CIs). To keep up with rising demands in terms of operational efficiency, these greatly distributed systems require increasingly complex monitoring and control strategies. These in turn are contingent on highly available, reliable communication networks. Due to the distributed nature of CIs, the deployment of dedicated communcation infrastructures entails considerable costs and long lead times. Therefore the convergence of public, shared Information and Communication Technology (ICT) and purpose built networks is pursued. Here Software-Defined Networking (SDN), also considered central to 5G cellular networks, is widely seen as promising solution [1]. Traditionally switches and routers implement data traffic flow control, such as routing, along with the physical packet forwarding process. SDN however abstracts the control functionalities and concentrates them in the so called SDN controller. By centralizing the decision making process in this device, a global view of the network's state is obtained, enabling e.g. more sophisticated routing schemes. The ability to create virtual, dedicated communication architectures - known as slices - is also offered by SDN and crucial for network convergence. Yet, SDN's controller centric approach has drawbacks, as failures of this single device threaten not only the stability of communication services, but all CIs dependent on it. This paper addresses this single point of failure through a novel resiliency strategy, specifically designed to meet the criteria of CIs in general and Smart Grids (SGs) [2] in particular. Current approaches provide fault tolerance by using passive redundancy, transferring a failed component's functions to another. The proposed methodology meanwhile utilizes active redundancy by way of a lock-step voting cluster, referred to as broker. By comparing the output of multiple, concurrently running SDN controllers the correct solution is selected by majority vote. This concept is effective against failures as well as malicious attacks and enables self-healing control. Moreover the recovery delay is calculated to be within even the harshest limits of SG communication protocols.

Figure 1 shows the layered SDN for CI communication architecture. At the bottom data plane devices, e.g. SDN-enabled switches, forward packets physically. The left of this plane is split, illustrating the ability to instantiate logically separated, virtual network slices. This constitutes a crucial feature for the convergence of public and dedicated communication networks.

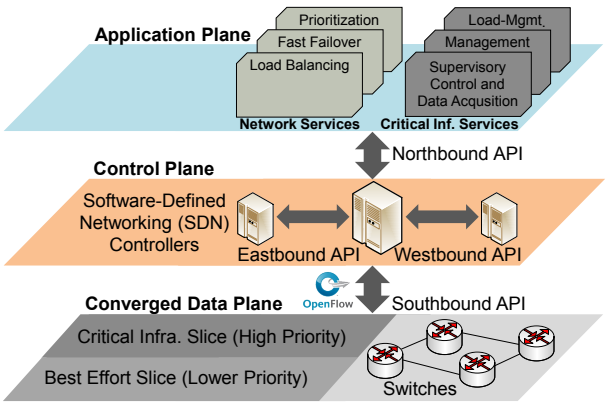

Figure 1: Software-Defined Networking for Converged Critical Infrastructure Communication

On top of the converged data plane resides the control plane. Here functionalities such as routing, traditionally inseparably linked to switches and routers distributed throughout the network, are centralized into the so called SDN controller. This device interfaces with the data plane via the southbound Application Programming Interface (API). OpenFlow [3], specified by the Open Networking Foundation, is the de-facto standard for this task. The east- respectively westbound API serves to interconnect multiple controllers. Hence multiple devices can exchange information for tasks such as controller failover or cross-domain traffic flows. Currently no standard exists to govern these interfaces. At the very top of the Figure the application plane is shown. In contrast to traditional communication networks services such as load balancing and prioritization can directly be exposed to applications via the northbound API. Thereby systems of CIs such as Supervisory Control and Data Acquisition (SCADA) devices, are enabled to transmit their communication requirements directly to the SDN controller. The latter then reconfigures the data plane to handle traffic according to the requested Quality-of-Service.

The structure of this paper is as follows: First an overview of related work is given in Section II. Afterwards Section III provides an introduction to communication for CI. There the general architecture as well as requirements of CI are outlined in Section III-A. Based on this Section III-B highlights the concept of SDN in context of the before mentioned use cases. Subsequently Section IV constitutes the main part of this work. After discussing currently existing control plane resiliency in Section IV-A, Section IV-B presents this paper's proposed, novel strategy. A detailed comparison of all approaches is given in Section IV-C. Finally Section V gives a conclusion and outlook on future work.

II. Related Work

Since its inception SDN has evolved into an approach widely considered as central for the next generation of communication networks, such as 5G. Considerable efforts by ICT vendors and operators as well as research institutions concentrate on its application to a large variety of use cases. As resiliency of the overall methodology is crucial for the integration in production networks, works addressing this aspect are readily available. However, the vast majority focuses on robustness of the data plane, as presented in [4], [5] or [6]. Papers on control plane, i.e. controller, fault tolerance meanwhile often center on performance aspects such as achievable data and packet forwarding rates [7] in degraded network states. Hence, these studies do not determine the influence of controller failures on traffic flow interruption, which is of utmost importance in CI communication.

In [8] and [9] failover is achieved by running two SDN controllers in parallel. If the primary device fails, the secondary takes over. While potentially a fast solution, failures caused by malicious attacks are not fully mitigated.

A variation of this strategy uses SDN itself for switching between controllers [10]. By rewriting the forwarding table of a switch connecting primary and secondary controllers with the data plane, a fast failover between both devices is achieved. Another approach is presented in [11] and [12]. There system states are synchronized across an arbitrary number of controllers via centralized databases. Distributed databases increase resiliency of this methodology and are employed by [13], [14] and [15]. Again, malicious attacks are not fully addressed.

Another resiliency strategy is presented by [16] and [17]. The authors introduce consensus finding algorithms, henceforth referred to as election systems. These either determine the correct answer on a case-by-case basis or elect a new primary controller in case the active one fails. Byzantine failures, in which a controller sends mixed results to different peers, are mitigated by this strategy. Yet, reaching a consensus can be too time consuming for use in CI communication networks.

A detailed breakdown of advantages and flaws of the individual, related resiliency strategies as well as a comparison with the proposed approach is given in Section IV.

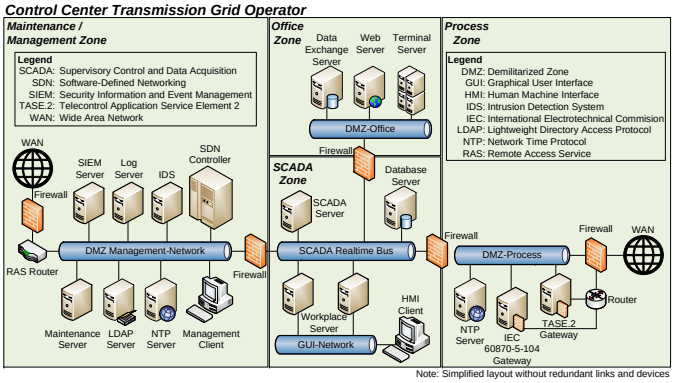

Figure 2: Control Center Communication Architecture of an Electrical Transmission Grid Operator

III. Converged Networks for Critical Infrastructure Communication

The following Section first presents the communication architecture and requirements of CIs on the example of SGs. Next the concept of SDN for Smart Grids and its ability to facilitate network convergence are introduced.

A. Architecture and Requirements of Communication for Critical Smart Grid Infrastructures

Smart Grids are a prominent example for CIs. Operating these complex, distributed infrastructures necessitates robust monitoring and control. This is achieved via SCADA systems. Located in operation centers of Transmission and Distribution System Operators (TSO / DSO), as shown by Figure 2, they serve as information aggregator and upmost control instance.

IEC 61850 and IEC 60870-5-104, defined by the International Electrotechnical Commission (IEC), are two of the most relevant and widely used SG communication protocols. In recent years the importance of IEC 61850, originally developed for primary substation automation, has grown considerably and now covers diverse applications like Electric Vehicles (EV),

Table I: IEC 61850 Smart Grid Traffic Characteristics [18]

	Typical Packet Size in [Byte]	Typical Inter-Transmisson-Time in [ms]	Max. End-to-End Latency in [ms]
Manufacturing Message Specification (MMS)	138 to 600	2	1000
Sampled Values (SV)	137 to 230	0.125	10
Generic Object Oriented Substation Event (GOOSE)	128	2 to 10	10

Distributed Energy Resources (DER) as well as wide area communication. The protocol supports the Ethernet standard, displacing legacy technologies previously complicating substation operation. Key communication services of IEC 61850 are Sampled Values (SV) and Generic Object Oriented Substation Events (GOOSE) [18]. Both encapsulate messages directly into Ethernet Media Access Control (MAC) frames. The former uses fixed time intervals (typically 250 μs) to transmit measurement streams from appliances such as Phasor Measurement Units (PMUs) to grid controllers or protection devices. GOOSE meanwhile is used to signal commands or status updates. Real-time supervisory reports, firmware updates and configuration data are handled by Manufacturing Message Specification (MMS). Table I gives an overview of the IEC 61850 communication protocols regarding packet sizes, latency and inter-transmission-time.

As 10 ms is, regardless of failures, the maximum acceptable end-to-end latency, error detection, mitigation and recovery need to be below this threshold, allowing for propagation and forwarding delays as well. Should the controller not be available, or recovery exceed the acceptable duration, new traffic flows can not be installed. Also the Quality-of-Service (QoS) of existing flows can not be reconfigured. In turn critical events in the electrical grid can not be signalled appropriately, threatening Smart Grid stability. Hence, any SDN controller failover approach aiming for deployment in CI communication networks needs to meet the outlined criteria.

According to the Council of European Energy Regulators (CEER) [19], in 2015 unplanned downtime of Germany's electrical grid was around 15 min. Hence, availability amounts to 99.997 %. As overall CI robustness is determined by its subsystems, the communication network needs to have even higher availability and reliability. Accordingly a fail-safe network design that enables ultra-low latencies is required.
Moreover, Smart Grid operators are faced with a variety of heterogeneous ICT. Typically fiber optic cables are included in stay cables of high-voltage transmission towers. While these links connect sites such as primary substations, renewable energy sources are often attached to widely distributed medium and low voltage grids. Thus the use of public communication networks is of high interest for TSOs and DSOs. This convergence of public and dedicated infrastructures is supported by SDN as outlined in the following Section.

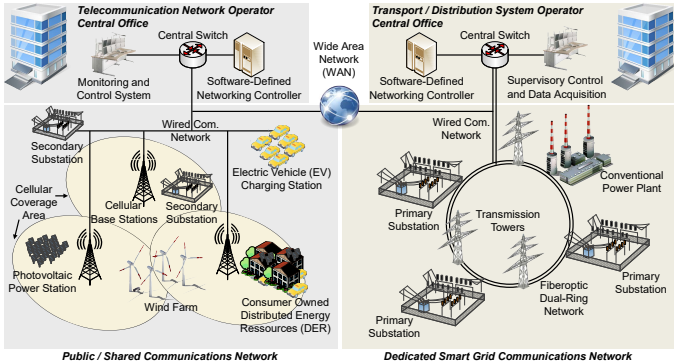

Figure 3: Convergence of Public Networks and Dedicated Critical Infrastructure Communication in Smart Grids

B. Software-Defined Networking Enabled Critical Infrastructure Communications

SDN promises to meet the requirements of CIs in general and SGs in particular [20]. Unlike traditional architectures, the controller centric paradigm allows improved algorithms (e.g. routing) to be implemented without the need to upgrade the packet forwarding hardware. Thus switches and routers only need to be replaced if their capabilities no longer satisfy the requested data rates and latencies. This fits to SG deployment cycles, which are on the order of decades and thus considerably longer than those of the ICT domain. Moreover, convergence of heterogeneous network technologies and architectures is enabled by slicing, in which virtual topologies are created on top of one physical data plane. A high-performance slice for CI services can be established within a telco's network, always taking precedence over less crucial traffic like real-time voice or bulk data transmissions. Hard QoS guarantees, fast data plane failover and good manageability contribute to the usefulness of SDN for CI communication.

Figure 3 illustrates the convergence of public and dedicated SG communication networks enabled by SDN [21]. On top of the right a high-level view of a TSO / DSO central office is given. Below a dedicated, fiber-based infrastructure is shown. A dual-ring topology deployed along transmission towers connects substations and power plants. The SCADA system transmits requirements of crucial information flows to a grid operator owned SDN controller via the northbound API. This enables the reconfiguration of the TSO-owned network according to needs of grid monitoring and control. The Figure's left side layout is similar at the central office level, here belonging to a telecommunication network operator (telco). Future technologies like 5G are set to use SDN, thus the corresponding controller is either planned or already in use. Cellular base stations cover many areas in which renewable energies such as Photovoltaics (PV), wind farms and DER are located. Other grid devices can be connected to the telco's pre-existing, wired uplinks. Cooperation of the telco's and power grid operator's SDN controllers can be realized via the westbound interface. Hence, both controllers configure their network according to the Smart Grid's requirements.

IV. Hardening the Control Plane

In this Section we provide a comprehensive overview of existing control plane resiliency concepts. Furthermore, our improved broker based strategy is introduced and compared. Table II contrasts the resiliency schemes' key attributes.

A. Common Control Plane Resiliency Concepts

Resiliency can be provided through redundancy of critical components, in this case the SDN controller. While concepts diverge in their failure detection and mitigation strategies, commonalities exist. For example warm, respectively hot standy modes are necessary for fast failover. A secondary device is either periodically (warm) or continually (hot) synchronized to the primary's state. The concepts are as follows:

1) Primary - Secondary Strategy: As indicated by its name, the strategy utilizes two controllers. In case the primary fails, the previously passively listening device takes over. This basic approach is supported by data planes such as Open vSwitch (OVS) [22]. Failover duration largely depends on error detection [23] and state synchronization time. The former is typically handled by heartbeats, for which devices check each other's availability by exchanging keep-alive packets with a high frequency. State updated can be included in these data frames as well. Other solutions use Virtual Router Redundancy Protocol (VRRP) or Common Address Redundancy Protocol (CARP) in which both devices share a virtual Internet Protocol (IP) address, which causes the network to observe only one, resilient controller. Although potentially a very fast solution, the primary-secondary setup is prone to errors and attacks. With two devices the system itself can only detect errors but not determine the source. Hence there is no validation of Open Flow messages which is necessary for self-healing networks. Also, state synchronization requires both controllers to be trustworthy, not signaling false information. Traffic routes calculated on faulty states could overload data plane links, severely impacting QoS and thus Smart Grid stability.

2) Database Centric Failover: Database centric failover can be seen as an extension of the primary-secondary case. Consequently its properties are similar. Network state is replicated either locally at multiple, distributed controllers or on external, redundant devices. Error detection and failover of the primary can be handled via heartbeats and VRRP or CARP. The switch between SDN controllers can be fast (largely dependent on heartbeat frequency), and databases can be replicated to distributed instances. However, neither the primary's actions nor entries to the database are validated.

3) Election Based Consensus Systems: Consensus finding systems, typically using ring or mesh topologies between controllers, are considerably more robust than the afore described approaches. Through different algorithms [24] multiple devices either elect a primary in intervals or vote on the correct answer on a case-by-case basis. Accordingly the population needs to be at least three. A minority of erroneous or malicious devices is overruled. In addition byzantine fault tolerance can be guaranteed, i.e. the correct solution is found even if entities communicate unequal results to different peers. Election based systems are robust against attacks and accidental failures, as it is not necessary to trust neighbors or the validity of databases. Yet, the election process is usually not as fast as required by CI. Furthermore, overall system complexity increases, which raises the probability of errors in the source code, i.e. bugs.

Other solutions to control plane resilience utilize cloud computing, or controllers taking over neighboring network domains if the corresponding SDN controller fails. However, they face the same challenges as the above described methodologies hence sharing similar advantages and flaws.

Table II: Comparison of SDN Controller Resiliency Schemes

	Broker	**Election Based**	**Database Centric**	**Primary / Secondary**
Failover Scheme	Proactive	Proactive	Reactive	Reactive
Fault Tolerance	High	High	Limited	Limited
Attack Resiliency	High	High	Low	Low
Additional Load on Controller	None	Medium	Low	Low
Controller Isolation	Yes	Limited	No	No
Voting	Yes	Yes	No	No

B. Broker Based Controller Resiliency

As consensus based systems provide the required robustness, our approach aims to retain voting capabilities while decreasing reaction times and complexity. In SDN the controller typically connects to the forwarding elements via redundant switches, which provide multiple links between control and data plane. At this point we integrate a stateless broker, so fast failover algorithms [5] already in place at these switches guarantee its resilience as well. Akin to [10] the destination addresses of all OpenFlow (OF) messages sent by data plane switches are rewritten. Hence, from every controller's point of view, all status and request packets seem to be intended solely for itself. Nevertheless an uneven number (minimum of three) of multiple other controllers run in parallel, isolated from one another thanks to the intermediary broker-switches. Yet, they are supplied with the same input, giving them an identical view of the network's status. Hence no error prone and time consuming state synchronization is required. When

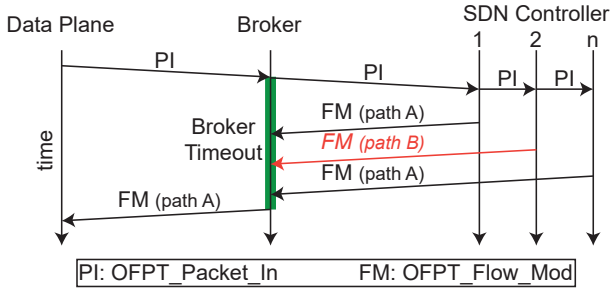

Figure 4: Illustration of the Broker-Based Voting Scheme

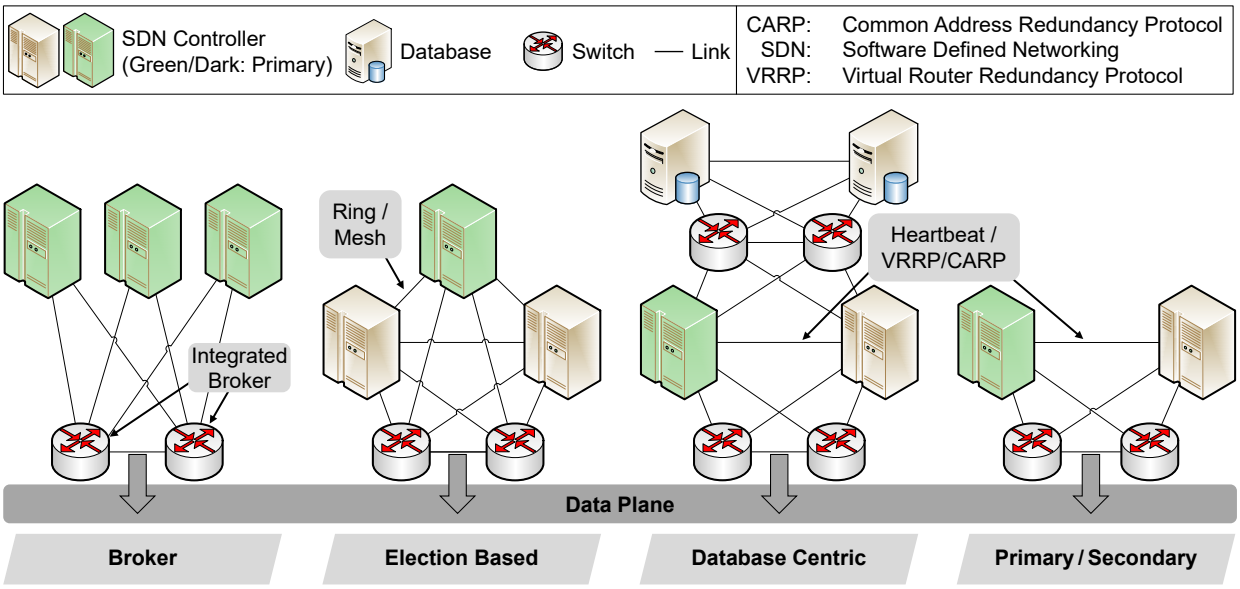

Figure 5: Overview of Broker-Based and Conventional SDN-Control Plane Resiliency Architectures

an instruction such as an OF FlowMod is sent by a controller the broker takes action. The packet is buffered and stored until the plurality has sent their reply, or until a timeout is reached. All gathered replies are compared, with the majority of equal answers being recognized as valid. The system operator is notified of diverging, i.e. faulty results so the corresponding source can be repaired or replaced with further votes being ignored. Afterwards a single OF packet, with rewritten source address (thus hiding the broker and other controllers), is sent out to configure the data plane. Figure 4 illustrates the process for a single OFPT_Packet_In message, sent by the data plane to signal the arrival of an unknown traffic flow. A prerequisite for the broker strategy is the use of controllers witch use defined algorithms, e.g. for load aware routing. Given the same input data (i.e. network state) they have to yield identical results within time limits acceptable for CI protocols. Thereby the functionality of the voting process is ensured. However, different vendors may be used to reduce the risk of bugs in individual implementations affecting overall system stability. If Transport Layer Security (TLS) is used, the broker requires the respective keys for OF packet access. Moreover controllers do not interact, accordingly security is raised as members affected by malicious attacks lack the capability to influence others. Therefore the approach is highly resilient against accidental and malicious (also byzantine) faults. An overview of the different strategies is shown by Figure 5. To improve readability, redundant links are excluded.

C. Performance Comparison of Fault Tolerant Control Planes

A low failover delay is of utmost importance for fault-tolerant SDN controllers deployed in Critical Infrastructures. Accordingly Figure 6 compares the performance of the presented strategies. Primary-Secondary configurations achieve recovery times of $4200\ ms$, $2443\ ms$ and $30\ ms$. Database centric solutions show a similar performance level with $2750\ ms$ to $3500\ ms$. Nevertheless, one implementation achieves a value of $50\ ms$, which is the threshold for carrier grade ICT [25]. Overall, the duration it takes for multiple keep-alive messages of the heartbeat-based monitoring systems to be lost - necessitated to avoid false positives - limits both strategies. Election based schemes surpass this with a duration of $14.5\ ms$ [16]. As election is performed for every decision of the controller cluster, this delay applies to all actions. Therefore this scheme is too slow for use in CI communication, even if no failures are present. Regarding the proposed broker-based voting scheme, the following extrapolation can be made. Our Software-Defined Universal Controller for Communication in Essential Systems (SUCCESS), specifically tailored to the requirements of CI communication, implements functionalities such as traffic prioritization and enhanced data plane failover [5]. The time from receiving an OF packet to replying with the corresponding configuration is $1.99\ ms$ (median), with $4.2\ ms$ assumed as worst case ($99th$ percentile) [23]. These delays include the calculation of optimal routes within the network and inefficiencies in the employed network stack (median controller processing delay amounts to just $0.15\ ms$). Assuming a delay of $3\ ms$ for the broker-switch to rewrite packet addresses, wait for and compare messages, a conservative estimate for broker based control plane resiliency can be made. Hence the best cast delay is around $5\ ms$ while controller jitter places the worst case at $7.2\ ms$. Both values are within the limits defined by IEC 61850, making broker based fault tolerance the only methodology suitable for CI communication networks. Since all available controllers are actively involved in finding a valid configuration, there is no reactive failover with slow error detection or election processes. By optimizing the network stack to obtain delays closer to actual computation time, recovery delay could be reduced further.

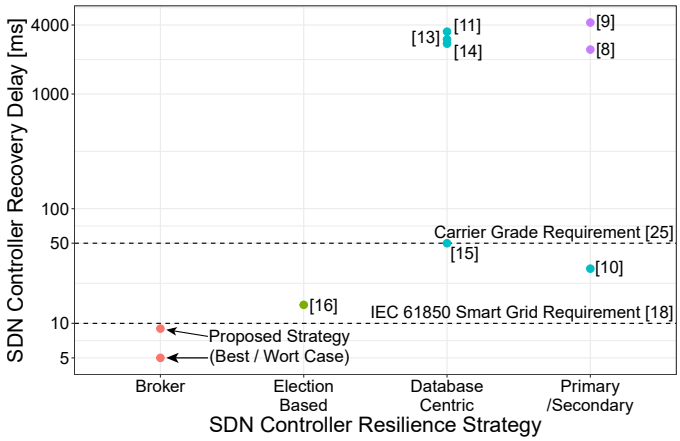

Figure 6: SDN Controller Recovery Delays Compared Against Communication Requirements (References in Brackets)

V. Conclusion and Outlook

In this work, a novel SDN control plane resiliency strategy designed specifically for converged CI communication networks is presented. By introducing a robust, redundant broker parallel operation of multiple controllers is made feasible. These work in isolation and each receive a copy of incoming requests and status updates. They in turn submit their corresponding configuration commands as votes to the broker. The majority of agreeing instructions is send on to the data plane. Source, respectively destination addresses of packets traversing the broker are rewritten. Thus switches, commonly only accepting responses from the device they send their updates and requests to, do not need to be adapted. Availability and reliability of the overall system is raised considerably, as faulty devices are isolated from their peers and can not influence the state of others. In contrast current fault tolerance mechanisms are shown to have weaknesses against attacks and to be too slow for the intended use case. However, the proposed strategy is shown to meet the challenging requirements of SGs.
Future work will focus on failover performance in a realistic 5G Smart Grid communication network demonstrator. Further, slicing will be pursued to provide virtual, dedicated infrastructures on top of a shared, i.e. converged data plane.

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] A. Cahn, J. Hoyos, M. Hulse and E. Keller, 'Software-Defined Energy Communication Networks: From Substation Automation to Future Smart Grids', in *IEEE International Conference on Smart Grid Communications*, 2013, pp. 558–563.

[2] 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.

[3] *OpenFlow Switch Specification Version 1.5.1*, Open Networking Foundation, Apr. 2015. [Online]. Available: https : / / www . opennetworking . org / images / stories / downloads / sdn - resources / onf - specifications/openflow/openflow-switch- v1.5.1.pdf.

[4] A. Aydeger, K. Akkaya, M. H. Cintuglu, A. S. Uluagac and O. Mohammed, 'Software defined networking for resilient communications in Smart Grid active distribution networks', in *IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–6.

[5] N. Dorsch, F. Kurtz, F. Girke and C. Wietfeld, 'Enhanced Fast Failover for Software-Defined Smart Grid Communication Networks', in *IEEE Global Communications Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[6] N. van Adrichem, B. Van Asten and F. Kuipers, 'Fast Recovery in Software-Defined Networks', in *European Workshop on Software Defined Networks (EWSDN)*, Sep. 2014, pp. 61–66.

[7] F. Botelho, T. A. Ribeiro, P. Ferreira, F. M. V. Ramos and A. Bessani, 'Design and Implementation of a Consistent Data Store for a Distributed SDN Control Plane', in *European Dependable Computing Conference (EDCC)*, Sep. 2016, pp. 169–180.

[8] K. C. Fang, K. Wang and J. H. Wang, 'A fast and load-aware controller failover mechanism for software-defined networks', in *International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Jul. 2016, pp. 1–6.

[9] M. A. S. Santos *et al.*, 'Decentralizing SDN's control plane', in *IEEE Conference on Local Computer Networks*, Sep. 2014, pp. 402–405.

[10] S. Yoon, J. Lee, Y. Kim, S. Kim and H. Lim, 'Fast Controller Switching for Fault-Tolerant Cyber-Physical Systems on Software-Defined Networks', in *IEEE Pacific Rim International Symposium on Dependable Computing (PRDC)*, Jan. 2017, pp. 211–212.

[11] L. Sidki, Y. Ben-Shimol and A. Sadovski, 'Fault tolerant mechanisms for SDN controllers', in *IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov. 2016, pp. 173–178.

[12] F. A. Botelho, F. M. V. Ramos, D. Kreutz and A. N. Bessani, 'On the Feasibility of a Consistent and Fault-Tolerant Data Store for SDNs', in *Euro. Workshop on Software Defined Networks*, Oct. 2013, pp. 38–43.

[13] Y.-C. Chan, K. Wang and Y.-H. Hsu, 'Fast Controller Failover for Multi-domain Software-Defined Networks', in *European Conference on Networks and Com. (EuCNC)*, Jun. 2015, pp. 370–374.

[14] D. Suh *et al.*, 'Toward Highly Available and Scalable Software Defined Networks for Service Providers', *IEEE Communications Magazine*, vol. 55, no. 4, pp. 100–107, Apr. 2017.

[15] V. Pashkov, A. Shalimov and R. Smeliansky, 'Controller failover for SDN enterprise networks', in *International Science and Technology Conf. (Modern Networking Technologies)*, Oct. 2014, pp. 1–6.

[16] K. ElDefrawy and T. Kaczmarek, 'Byzantine Fault Tolerant Software-Defined Networking (SDN) Controllers', in *IEEE Computer Software and Applications Conf. (COMPSAC)*, vol. 2, Jun. 2016, pp. 208–213.

[17] H. Li, P. Li, S. Guo and S. Yu, 'Byzantine-resilient secure software-defined networks with multiple controllers', in *IEEE International Conference on Communications (ICC)*, Jun. 2014, pp. 695–700.

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

[19] *6th CEER Benchmarking Report on the Quality of Electricity and Gas Supply 2016*, Council of European Energy Regulators (CEER), 2016. [Online]. Available: https://www.ceer.eu/.

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

[21] E. Molina, E. Jacob, J. Matias, N. Moreira and A. Astarloa, 'Using Software Defined Networking to manage and control IEC 61850-based systems', *Computers & Electrical Eng.*, vol. 43, pp. 142–154, 2015.

[22] *Open vSwitch Version 2.4.0/2.3.0*, 2015. [Online]. Available: http://openvswitch.org/.

[23] 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 (SmartGridComm)*, Nov. 2014, pp. 422–427.

[24] *High-performance Byzantine Fault-Tolerant State Machine Replication*, 2017. [Online]. Available: https://bft-smart.github.io/library/.

[25] B. Niven-Jenkins, D. Brugard, M. Betts, N. Sprecher and S. Ueno, *Requirements of an MPLS Transport Profile (RFC 5654)*, Internet Engineering Task Force, Sep. 2009.