SDN Based Architecture for Clustered WSN

Olivier Flauzac, Carlos Gonzalez, Florent Nolot

SDN Based Architecture for Clustered WSN

FLAUZAC Olivier, GONZALEZ Carlos, NOLOT Florent
University of Reims Champagne-Ardenne, Laboratory CReSTIC
Telephone: (+33) 3 26 91 32 15, Fax: (+33) 3 26 91 33 97
REIMS, FRANCE
Email: olivier.flauzac@univ-reims.fr, carlos.gonzalez-santamaria@etudiant.univ-reims.fr, florent.nolot@univ-reims.fr

***Abstract*—In this paper, we propose to the concept of software-defined networking (SDN) in wireless sensor networks (WSNs) with a structured and hierarchical management. We argue that structured and hierarchical management using SDN promises a solution to some inherent problems in WSN management. Furthermore, we propose a cluster based architecture with multiple base stations as hosts for the SDN control functions and as cluster heads in the same time. We also suggest a general architecture for a software-defined wireless sensor network where the controller exchanges information with other SDN domains. We call this new architecture: software-defined clustered sensor networks (SDCSN). In addition we highlight some future directions and raise some important questions that need to be investigated in future research in software-defined networks for ad-hoc and sensor networks.**

***Keywords*-software defined network; SDN; wireless sensor network; WSN; ad-hoc network**

I. Introduction

Software-Defined Networking (SDN) has emerged as a new paradigm for enabling innovation in networking research and development. SDN [1], [2] is actually attracting significant attention from both academia and industry, obtaining support from a large number of main industries such as, Microsoft, Google, Facebook, HP, Deutsche Telekom, Verizon, Cisco, IBM, and Samsung. A central software program, called SDN controller, manages the overall network behavior. With SDN principles the control and data planes are decoupled and network intelligence is logically centralized. SDN controller can add, update, and delete flow entries, both reactively in response to packets and proactively with predefined rules. Moreover, SDN enables fast reaction to security threats, granular traffic filtering, and dynamic security policies deployment. The SDN architecture provides a programmatic interface inside the controller. SDN allows network control operations such as:

- run on top of one or multiple server platforms with higher performance,
- use vendor independent hardware and an open operating system,
- are able to communicate with other operating systems or control platforms using standard protocols.

SDN controller can use OpenFlow protocol to establish connection with the switches. OpenFlow is one of the protocols that can be used to control the network device. In this context, the SDN controller builds a global network view based on the information received via OpenFlow. Furthermore, the SDN controller can:

- perform network discovery, using the Link Layer Discovery Protocol (LLDP),
- collect statistics about network traffic using a special field in the flow rules earlier installed by the controller.

A Wireless Sensor Network (WSN) is composed of several wireless nodes (also called motes) able to collect data information from scenarios such as military and civilian surveillance, vital body functions, structural monitoring, weather station, pollutant level, among others. Sensor nodes can be deployed in the area of interest to sense and detect events and communicate these to the Base Station. This area of study perhaps is appropriate in inaccessible zones where human intervention is impossible. Therefore the trustworthiness of WSN in this scenario becomes of utmost importance. The components of a sensor node are a power unit, processing unit, sensing unit and communication unit. The processing unit is responsible for communicating with the other sensor nodes to collect signals captured and transmit them to the network. But generally the sensors nodes have limited resources (energy), low processing power, low storage and low communication bandwidth [6], [26]. Hence, they can only transmit information within a limited transmission range and with limited onboard processing capacity. In clustering terms, the whole sensor network is divided into groups of sensor nodes called clusters and in each cluster, a sensor node is elected as cluster head (CH). There is an important method for prolonging the lifetime of sensors networks (WSNs). Several strategies of sensing modalities have been studied to optimize the performance of sensor nodes [25]. The authors describe the benefits and limitation of sensing modalities that must be considered to facilitate deployment of WSN. They also discuss the fact that the sensor nodes are usually battery powered. To increase battery life, communication protocols to improve the power performance of sensor nodes will be considered. When two or more interfering nodes transmit at the same time during data collection, collision occurs. Because of this more energy is consumed during the process and latency is increased. In [26] authors provide a comparative study of all aggregation techniques to opti-

mize energy efficiency and redundancy in WSN. They have described the advantages and disadvantages of each data aggregation method, using soft computing techniques. An important characteristic of this work is the security measures used by data aggregation techniques. If a cluster-head or the aggregator node is compromised by a malicious attacker, the data sent to the base station cannot be guaranteed. To solve potential problems concerning the security of cluster-head, we take into account the Grid of Security concept proposed by Flauzac et al. in [27]. Previous studies [3], [7], [8], [9] in Software-Defined Networking implementations with wireless sensor nodes provide a better view of this paradigm. However, nobody explains where the controller could be placed in a WSN using Clustering mechanism. From this situation, we can ask the following question: How to design the Software Defined Sensor Networks in a WSN? For this reason, we propose architecture for the WSN and Ad-Hoc network based in clusters. We call this architecture Software Defined for Clustered Sensor Networks (SDCSN). This architecture can be deployed on a large scale for monitoring and security applications. In this paper, we argue a concept for software-defined wireless sensors networks, providing a perspective of the field by different research and also describing in detail the SDN WSN paradigm. The paper is organized as follows: in Section II, we begins by discussing the research on software defined networking and WSN. Section III provides an overview of SDCSN and describes our implementation. Section IV describes the domain interconnections with SDCSN. Finally, Section V discusses research challenges and future directions.

II. State of the Art

An idea for managing WSN with SDN is proposed by Luo et al. [3], (Fig. 1), with the concept of Software-Defined Wireless Sensor Networks SD-WSN. This study is based on the idea that each sensor node supports OpenFlow, and sensors should be able to recognize the flow table's entries. This architecture proposes a separation between data and control plane. The Sensor OpenFlow (SOF) is a communication protocol between the control plane and data plane. The data plane contains sensor flow packet forwarding, and the control plane is a controller for performing routing and QoS network control.

Another work by Zeng et al. in [7] has studied the Evolution of Software-Defined Sensor Networks, integrating sensors nodes into cloud computing using a SaaS. This model is named Sensing-as-a-service (Fig. 2) and combines the sensing data with existing cloud services such as mashup services. The controller is a sensor controller with SDN functionalities and it is provided with a local database to store sensed data. In this work the authors assume that every sensor node is able to deliver packets to a sensor control server.

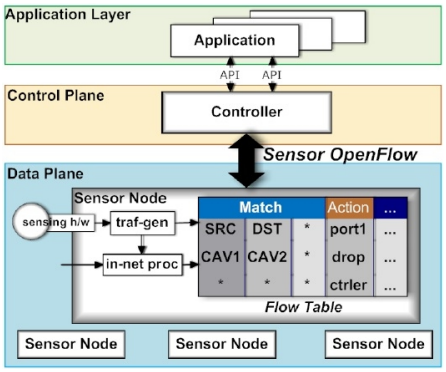

Figure 1. Architecture for Software-defined wireless sensor networks [1]

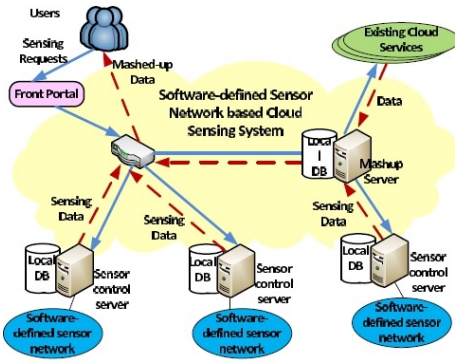

Figure 2. SDN based Sensing Architecture [7]

Gante et al. [8] presented base station architecture for WSN based on SDN with a review of benefits of this technology. The authors mentioned the use of an SDN controller as a base station in WSNs, but did not present any detail about communication between sensor nodes. The controller can determine the best routing, forwarding decisions and inserting these decisions into sensor nodes flow tables. In others words, the sensor nodes do not make routing decisions, they only forward and drop packets according to the rules set by the controller (base station).

In, [9] Contanzo et al. analyze the opportunities and challenges of SDN in IEEE 802.15.4 networks, implementing a scenario called SDWN. The controller is executed into sink nodes, and in order to communicate, each node must learn a path (as convenient as possible) to reach the controller. The controller periodically generates a beacon packet to send it to the nodes. Also, the nodes store the list of nodes from which they received a beacon packet. All discovered neighbors will be linked frequently to the sink node using a packet called a report packet.

Many solutions have been proposed by different authors in the SDN and WSN field, but to the best of our knowledge

there is no work about WSN in the SDN context with cluster architecture. In this paper we propose Clustered Software Defined Sensor Networks. Clustering consists in organizing the network into groups of nodes, in a hierarchical structure. Each cluster is managed by a cluster head. To develop this architecture we place the SDN controller in the cluster head. Different clustering solutions have been proposed in the literature. Some solutions proposed building 1-hop clusters [11], [12], [13], [14]. In those solutions, each node is at a most a distance of 1 from the cluster head, and the maximum diameter of each cluster is 2. Other solutions build k-hops clusters [15], [16], [24]. In k-hop cluster solutions, each node can be located at a distance at most of k from the cluster head and the maximum diameter of clusters is 2k.

III. The Software Defined Clustered Sensor Networks Architecture

In this section, we introduce the concept of SDCSN and explain the different technologies for the realization of SDCSN. The WSN may contain hundreds or thousands of sensor nodes. Normally, a large network cannot operate efficiently without some organized structure. For this reason, we propose to cluster the network and assume that each cluster head is a controller. Each node can be in one of the following states [5], [6], [24]: Simple Node (SN), Gateway Node (GN) or Cluster Head (CH). In our approach, Cluster Head (CH) in SDCSN architecture are called SDN Cluster Head (SDNCH). Each cluster is called an SDN Domain. In each SDN domain, the SDNCH [7] is in charge of managing the operation of the sensor nodes (Fig. 3). With this clustering approach the collected information about the environment is processed on the domain by nodes and will be routed to the SDNCH. Moreover, the controller is the most powerful node on the cluster. By using the equal interaction between SDNCH, it will have a full access to the switch and the same flows rules. Based in this approach, we can set configuration parameters, store data and aggregate the collected information in the domain or sending information to the sink or some other SDNCH. If a SDNCH is disabled, the entire domain becomes inaccessible temporarily, based on self-stabilizing clustering for WSNs [24] a new SDNCH can be selected. Each sensor node exchanges information with its neighbors or 2-hop neighbors. Under the assumption of graph connectivity, information generated at one sensor can reach the SDNCH by routing it through the network. The gateway is engaged in aggregating and transmitting the data from entire sensor node domains to the other domains. Thus, when a gateway goes down, the communication between domains will be disconnect and the associated sensors to SDNCH are isolated.

The SDNCH plays the role of coordinator in every domain. Every SDNCH acts as a temporary base station within its domain, and it is capable of communicating with other SDNCH. A Domain is therefore composed of a SDNCH, gateways and sensor nodes.

- SDN Cluster Head (SDNCH): it is the coordinator of the domain.
- Gateway: is a bridge node between Sensor Nodes and SDNCH.
- Sensor Nodes: are groups of nodes in domain together with their gateway nodes.

In the base station architecture proposed by Gante et al. [8], an SDN controller is combined with WSN and the controller needs to know the topology of the entire network. SDN has a higher potential to develop forwarding decisions of SN based on the rules set generated by the Controller, permitting a better cooperation among SDNCH and SNs [3], [8], [9], [10]. Certainly, QoS could be an efficient way for sensor nodes to function in this architecture using the concept of meters tables. This consists in meter entries, defining per-flow meters [21]. The per-flow meters permit to Openflow to implement different QoS operations, such as rate-limiting, and can be combined with per-port queues to implement complex QoS frameworks (ex. Diff-Serv). Moreover, SDN controllers [8], can reduce the energy consumption by different sensor nodes, making the best routing decision and injecting these in the nodes flow tables. With the network management controlled by the SDN, the routing decisions and policies have low convergence time in comparison with routing protocols. To deploy this architecture, SDNCH not only has to manage the domain network. But it also have to monitor and efficiently secure the domain to prevent outside and inside attacks. The emergence of the next generation Internet requires high level security. Many works have studied network security using the SDN controller or installing security policies into OpenFlow switches, either by implementing firewalls, IPS, or IDS [17], [18], [19]. Our purpose is to achieve maximum synchronization of SDNCH in a security perimeter enabling a granular control over network access and continuous monitoring of SNs. Moreover, this approach acts as a security guard on the edge of domain to ensure the domain safety. For example, if the service is forest fire detection, SNs can identify threats by smoke, ultraviolet and temperature sensors and achieve optimal transmission to the SDNCH.

IV. Domain interconnection in SDCSN

By interconnecting all SDSN domains via border controllers (see Fig. 4) we intend to extend the concept of SDSN. Each domain has its SDN controller which controls all traffic in its domain. In every domain SDNCH has its own security policies and management strategy to distribute routing functions and security rules to each border controller. We adopt this architecture to guarantee the security with the concept of a grid of security [27] embedded in each controller to prevent attacks. When a sensor node need to transmit the collected data to another domain, the flow has

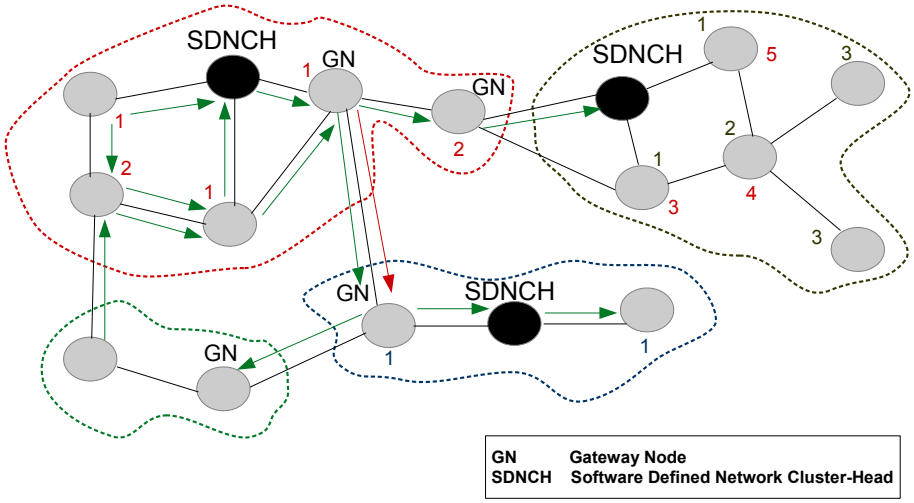

Figure 3. Data Communication Software-defined wireless sensor networks

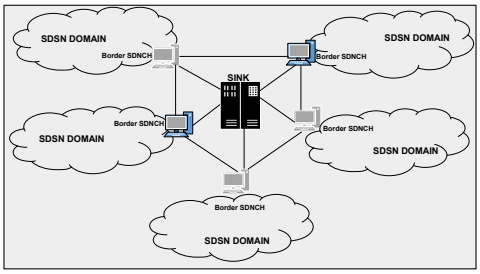

Figure 4. Domain Software-defined wireless sensor networks

to be forward to the security controller on the border of domain SDNCH.

When a SDNCH fails, another controller can take control to avoid system failures [20], increasing trustworthiness and fault tolerance. The Openflow specifications in version 1.3 identify two modes of operations when multiple controllers exist in the SDN: Equal interaction and Master/Slave interaction. These options are available in the Open Daylight project [20], permitting to support a Cluster based High Availability Model. Each SDNCH has a partial view of the entire network, which can help to collaborate and exchange information.

We analyze three options for connecting the SDN Domains: Equal and Master/Slave interaction, SDN Hierarchy and Routing between SDNCH. For equal interaction [20], all of the SDNCH will have a global view of the entire network. The controllers exchange information among all domains. In Master/Slave interaction one master controller collects the data from multiple slaves. In this case each and every SDNCH sends the information to one master controller. In a hierarchical SDN Domain, the network is divided into several domains, where domain is managed by a cluster head SDNCH. The sensor nodes transmit data to their SDNCH, which transmits the aggregated data to the base station. Sensor nodes can receive policies and routing decisions from one or more SDNCH. The SDNCH acts as the local base station for the particular domain. SDN inter-domain routing is required to exchange data among controllers. Each SDNCH builds a local network view, then it may exchanges its local domain view with other SDNCH through a WE-Bridge [22]. This WE-Bridge provides a mechanism for different SDN administrative domains to peer and collaborate with each other. In [23], the authors propose a distributed SDN control plane, DISCO for WAN and overlay networks. It relies on the domains, and each controller is in charge of an SDN domain. Every neighbor domain is capable of exchanging aggregated network-wide information.

V. Experimentation

We are working to build a testbed in order to experiment with our approach. We using many LXC containers connected to openvswitch. This virtual switch is controlled by opendaylight.

VI. Conclusion

This paper introduces a first attempt to analyze the opportunities and challenges of applying the SDN Clustered in WSN. We provide an overview of the SDN and WSN approaches with some of the most important design and implementation details of SDWN architecture. Existing research has focused on connectivity for sensor nodes and the SDN controller. However, no one had proposed how to place the controller in cluster architecture. A specific solution for

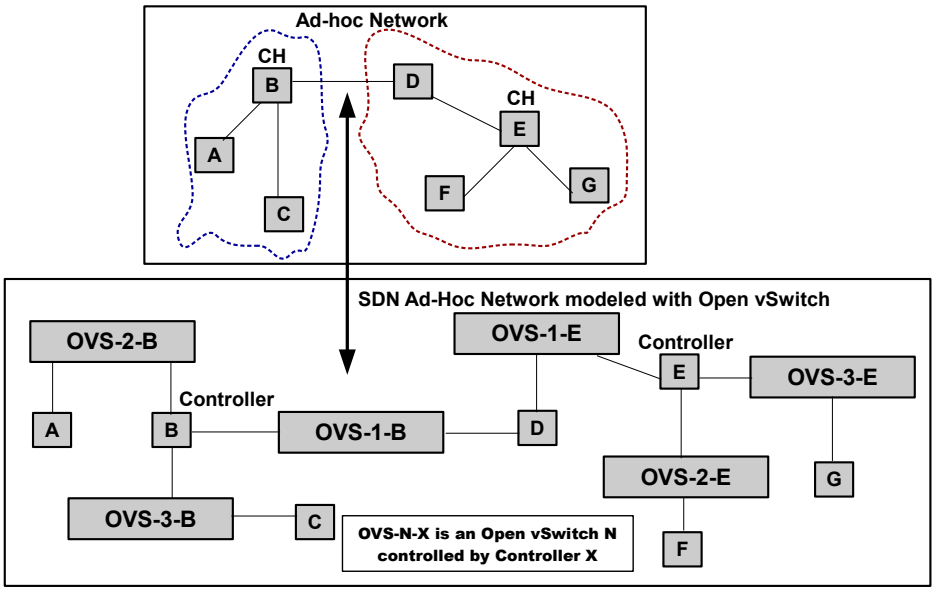

Figure 5. Snapshot of the proposed architecture (SDCSN)

such scenario is an SDN cluster which we call SDCSN. The benefits of our approach is that the controller can not only manage the network, also it can manage the security of one SDN domain. We extend this solution to include multiple controllers by interconnecting SDN domains via border controllers, which leads us close to a secure model for the WSN ad-hoc networks. The architecture proposed in this article present a new method to deploy a distributed security solution where the flow traffic between each sensor nodes can be controlled in a collaborative manner by the SDNCH.

ACKNOWLEDGMENT

This work was developed with financial support from the Secretara Nacional de Ciencia, Tecnologa e Innovacin (SENACYT) Panama.

REFERENCES

[1] B.A.A. Nunes, M. Mendonca, N. Xuan-Nam et al.: *A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks*, Communications Surveys and Tutorials, IEEE, 2014, 16, (3), pp 1617-1634

[2] Open Networking Foundation, Software-defined networking: the new norm for networks, white paper, Apr. 2012

[3] T. Luo, H. Tan, T.Q.S. Quek, *Sensor OpenFlow: Enabling Software-Defined Wireless Sensor Networks*, Communications Letters, IEEE, 2012, 16, (11), pp.1896-1899

[4] B. Nunes, M. Santos, B. de Oliveira et al., *Software defined networking enabled capacity sharing in usercentric network*, IEEE Communications Magazine, 2014, 52, (9), pp 28-36

[5] P. Azad and V. Sharma, *Cluster Head Selection in Wireless Sensor Networks under Fuzzy Environment*, ISRN Sensor Networks, 2013

[6] M. Ba, O. Flauzac, R. Makhloufi et al.: *A Novel Aggregation Approach Based on Cooperative Agents and Self-Stabilizing Clustering for WSNs*, The Seventh International Conference on Communication Theory, Reliability, and Quality of Service, CTRQ 2014

[7] D. Zeng, T. Miyazaki, Song Guo et al., *Evolution of Software-Defined Sensor Networks*, Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference, 2013, pp 410-413

[8] A. De Gante, M. Aslan and A. Matrawy, *Smart wireless sensor network management based on software-defined networking*, Communications (QBSC), 2014 27th Biennial Symposium 2014, pp. 71-75

[9] S. Costanzo, L. Galluccio, G. Morabito et al., *Software Defined Wireless Networks: Unbridling SDNs*, Software Defined Networking (EWSDN), European Workshop 2012, pp.1-6

[10] Y. Fan , V. Gondi, J.O. Hallstrom et al.: *OpenFlow-based load balancing for wireless mesh infrastructure*, Consumer Communications and Networking Conference (CCNC), 2014 pp 444-449

[11] N. Mitton, A. Busson, E. Fleury, *Self-organization in large scale ad hoc networks*, in MED-HOC-NET, 2004

[12] O. Flauzac, B.S. Haggar, F. Nolot, *Self-stabilizing clustering algorithm for ad hoc networks*, ICWMC, 2009, pp 24-29

[13] C. Johnen and L. Nguyen, *Self-stabilizing construction of bounded size clusters*, ISPA, 2008, pp 4350

[14] C. Johnen and L. H. Nguyen, *Robust self-stabilizing weight-based clustering algorithm*, TCS, 2009, pp 581594

[15] N. Mitton, E. Fleury, I. Guerin Lassous et al., *Selfstabilization in self-organized multihop wireless networks*, in ICDCSW, 2005, pp. 909915

[16] E. Caron, A. K. Datta, B. Depardon et al., *A selfstabilizing k-clustering algorithm for weighted graphs*, JPDC, 2010, pp 11591173

[17] S. Son , S. Shin, V. Yegneswaran et al., *Model checkinginvariant security properties in openflow*, in Proceedings of the IEEE International Conference on Communications, 2013, pp 1974-1979

[18] H. Hu, W. Han, G.J. Ahn et al., *Building robust firewalls for software-defined networks*, in Proceedings of the Third Workshop on Hot Topics in Software Defined Networking, 2014, pp 97102

[19] R. Jin and B. Wang, *Malware detection for mobile devices using softwaredefined networking*, in Proceedings of the Workshop of Research and Educational. pp. 81-88, 2013.

[20] Network Functions Virtualization (NFV), OpenDaylight, http://www.opendaylight.org/, accessed November 2014

[21] OpenFlow Switch Specification, Open Networking Foundation, http://www.opennetworking.org/, accessed November 2014

[22] P. Lin, B. Jun, Ch. Ze et al., *WE-bridge: West-east bridge for SDN inter-domain network peering*, Computer Communications Workshops (INFOCOM WKSHPS), IEEE Conference 2014, pp 111-112

[23] K. Phemius, M. Bouet, J. Leguay, *DISCO: Distributed multi-domain SDN controllers*, Network Operations and Management Symposium (NOMS), 2014, pp 1-4

[24] M. Ba, O. Flauzac, B. S. Haggar et al, *Self-stabilizing k-hops clustering algorithm for wireless ad hoc networks*, in: 7th ACM ICUIMC (IMCOM), 2013, pp 138

[25] G. Owojaiye and Y. Sun, *Focal design issues affecting the deployment of wireless sensor networks for intelligent transport systems*, Intelligent Transport Systems, IET, 2012, 6, (4), pp 432-432

[26] H.R. Dhasian and P. Balasubramanian, *Survey of data aggregation techniques using soft computing in wireless sensor networks*, Information Security, IET, 2013, 7, (4), pp 336-342

[27] O. Flauzac , F. Nolot, C. Rabat et al., *Grid of security: A new approach of the network security*, in Proceedings Third International Conference on Network and System Security, 2009 pp 6772