Indian Journal of Science and Technology, *Vol 9(32), DOI: 10.17485/ijst/2016/v9i32/100220, August 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

SDN in SCADA Based System for Power Utilities: A Case Study of Himachal Pradesh State Electricity Board Limited SCADA System

Abhinav Sharma Manu Sood and S.K. Sharma

Department of Computer Science, Himachal Pradesh University, Shimla; aasvi2006@gmail.com, soodm_67@yahoo.com, sukal.sharma@yahoo.com

Abstract

Objectives: Supervisory Control And Data Acquisition (SCADA) systems are used to manage and monitor the flow of powers through transmission lines, substation and control points. In this paper, we have considered the use of Software Defined Networking (SDN) as an approach to assist in the modernization of existing traditional SCADA systems taking Himachal Pradesh State Electricity Board Limited SCADA system as a case. **Methods/Statistical Analysis:** The use of Information and Communication Technology has enhanced the transforming process of the electric system and facilitated the efficient and reliable end to end intelligent bidirectional delivery system through integration of both energy sources renewable and non-renewable. The Software Defined Network Programming Languages Pyretic, Kinetic were used by the authors to study and compare the easiness of network programming languages. Moreover, a detailed study of SCADA system of Himachal Pradesh State Electricity Board Limited was done in which inflexibility and insecure issues were found out. **Findings:** The traditional network of Himachal Pradesh State Electricity Board Limited (HPSEBL) lacks flexibility and security mechanism. SDN-based SCADA system can facilitate robustness and flexible nature to the Power Grid Network Applications. SDN provides flexibility in form of vendor-free approach and various programming languages to configure the network as per one's need. Moreover, SDN with its distributed nature, keeping control plane away from the data plane has made it easy to monitor the network. Most of the programming languages for SDN use programming approach of standard computer language Python, thereby also making it easier for general programmers to make changes in the computer network as and when desired, which as of now is vendor specific and uses low level languages. **Application/ Improvements:** SDN based SCADA system will provide six fold benefits to the existing system viz. Flexible System, Centralized System, Open Standards, Network Programming, Green Networking and Security.

Keywords: Green Networking, OpenFlow, Power-Utilities, SCADA, Software Defined Networking.

1. Introduction

The Power Utilities are facing challenges for the constant maintenance and improvement of communication system networks in order to get the accurate and reliable data for optimization and control of power flow in a power network. Earlier, the SCADA networks were built as closed systems. Nowadays due to complex networks and intranets, Internet Access in the SCADA network is more vulnerable to security issues.

Some of the issues in traditional SCADA system are:

1. Insecure System thus susceptible to malware attacks or eves-dropping thereby compromising the security in power grids.
2. Present SCADA system is inflexible in nature as it is difficult to modify, interface and integrate new systems and Remote Terminal Units (RTUs).
3. High Power Consumption and Heat dissipation due to number of switches and hardware in use.

**Author for correspondence*

2. Supervisory Control And Data Acquisition (SCADA)

SCADA is a category of software application program for gathering of data in real time from remote locations in order to control equipment and conditions.

SCADA systems consist of multiple software and hardware elements that allow:

1. Collect, process and Monitor data
2. Interact with control hardware and devices which are connected through Human-Machine Interface (HMI) software
3. Record real time events into a log file and run real time energy management system application.

Generic architecture of SCADA is shown in Figure 1, where control centres, local sites and remote sites are connected to each other using Internet, Leased Lines or Satellites.

3. Traditional Himachal Pradesh State Electricity Board Limited (HPSEBL) SCADA System

Himachal Pradesh State Electricity Board Limited (HPSEBL) has a large network of high voltage/EHV transmission lines in whole HP. Transmission lines transfer power from power houses to substations and from one substation to many other substations and is bi-directional. Power is generated at low Voltage (of the order of 0.4kV, 3.3kV to 11kV) and is stepped-up to high voltage (220kV, 132kV, 66kV, 33kV and 22kV) for evacuating power into the grid network through transmission lines.

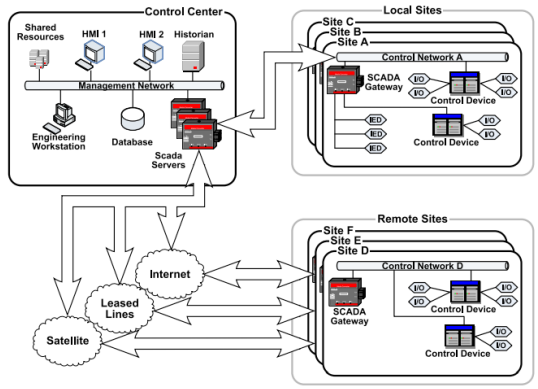

Figure 1. Generic SCADA architecture[1]

33/11kV and 66/22kV Substations of distribution draw power from transmission substations through 33kV lines and distribute that to consumers (at 0.4kV, 11kV, 22kV, 33kV and 66kV).Distribution contains industrial, commercial and domestic load. In some cases, huge variations in load and demand cause over/under loading of transmission lines, sub-stations or generators. Such abnormalities beyond limits cause fluctuations in voltages and grid disturbance due to abnormal frequency of the network.

Power Control Dispatch Centers, in hierarchical form, are set up for smooth operation of the grid. Each power substation has its own Monitoring and Control Centre. These centres report to State Area Load Dispatch Station (SALDS). SALDS report to State Central Load Dispatch Station (SCLDS). SCLDS reports to Regional System Coordination and Control Centre and finally on top is National Load Dispatch Centre (NLDC)[22].

Figure 2 shows communication between switches and RTUs in the SCADA Network of HPSEBL. HPSEBL has used Optical Fibre Ground Wire (OPGW) on lattice tower as communication media between controls centres switches and RTUs and in some cases PLCC (Power Line Carrier Communication) is also being used.

4. Software Defined Networking

Software Defined Networking is an emerging architecture for managing networks. SDN supplements traditional

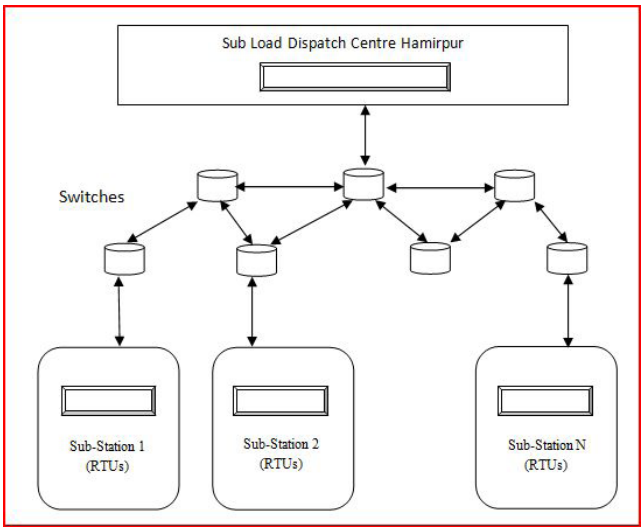

Figure 2. Communications between Switches and RTUs (Remote Terminal Units) in SCADA Network

networking by offering much flexibility and software centric control thus providing more opportunity to create policy based processes for adding intelligence into today's networks. In essence, SDN separates control plane from the forwarding plane and provides a centralized view of networking thereby making SDN controller the brain of the network. Control Plane decides how to handle the traffic while data plane is responsible for forwarding traffic according to control plane decisions. SDN controller relays information to switch and routers

Table 1. Programming languages for SDN

	Language	Programming Paradigm	Policy Composition	Dynamic Policies	Support for Modular Programming	Language Expression
FRENETIC[10, 13]	Python	Functional Reactive (declarative)	Yes Sequential Composition (>>)	Yes	Yes	SQL like query language
PYRETIC[11]	Python	imperative	Yes 1.Parallel Composition(+)2. Sequential Composition (>>)	Yes	Yes	Boolean Predicates
NETTLE[12]	Haskell	Functional Reactive Programming (declarative)			Yes (but hard to implement)	Electric circuit (streams of control messages are expressed as signals)
PROCERA[14]	Haskell	Functional Reactive Programming (declarative)		Yes		uses set of high level abstractions to make it easier to describe reactive and temporal behaviours.
HIERARCHICAL FLOW TABLES[15]			Yes uses high-level conflict resolution operator (+)	No		policies organized as trees
CORYBANTIC[16]	Python				intermediate level support (inter-module security is not present)	uses different modules for different functions
NETEGG[17]	automatically generates a controller program using scenarios	based on Example behaviours				uses timing diagrams and topologies, network polices are specified using scenarios
MERLIN[18]		declarative (logic)		Yes (using small run time components known as negotiations)		uses logical predicates and regular expressions
KINETIC[19]	Python	Functional reactive	Yes Provides better composition using Located Packet Equivalence Classes (LPECs) with parallel and sequential composition	Yes	Yes	expresses policies in terms of Finite state machines

via Southbound APIs and to the applications with Northbound APIs [6-9,23].

In SDN, the Northbound and Southbound interface allows a particular network component to communicate with higher-or-lower level network component, respectively. With the Northbound-APIs, the SDN controller can inform the network about its properties and states[23].

SDN has also made it easier to program the network. Table 1 gives a detailed comparison of various programming languages for SDN.

The benefits of SDN can be reaped to assist the management of SCADA system of Himachal Pradesh. SDN can provide flexibility to SCADA system, as addition of new policies and services will require updating of controller only. Moreover, SDN provides flexibility and also frees networking from its traditional hard-wired nature.

Figure 3 shows the proposed system for HPSEBL SCADA Network.

SCADA systems can benefit from SDN as:

1. Flexible System: HPSEBL Power System network is going to be more complex with the integrated operation of future Hydro Projects being constructed by HP Power Corporation Ltd. (HPPCL), EHV(Extra High Voltage i.e. > 66 kV) Substation by HP transmission corporation and Small Hydro Projects and Solar Projects being executed by the Independent Power Producers (IPPs). Thus future power flow on the network requires secured SCADA applications. SDN provides flexibility thus will allow easily adding of new field devices in SCADA network.

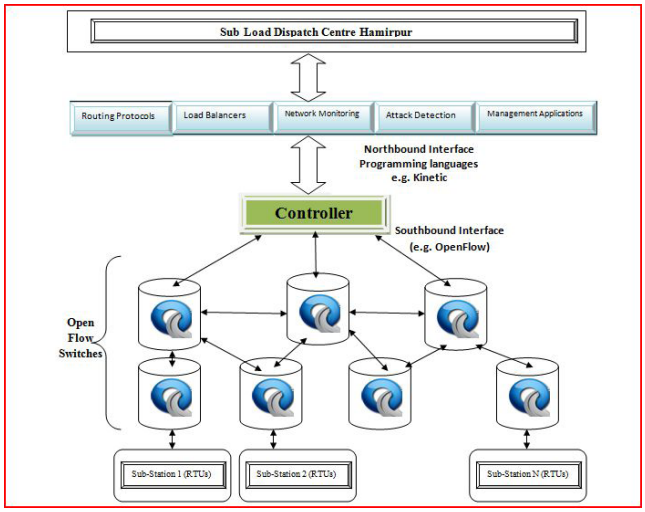

Figure 3. Proposed system for HPSEBL SCADA system

2. Centralized System: As SDN provides a centralized controller it can view the whole SCADA network thereby making it easier to manage the network.
3. Open Standards: SDN focuses on open to all approach e.g. commonly used protocol by SDN network is OpenFlow which is vendor free[21].
4. Programming the Network: SDN provides a vast opportunity to program the network according to ones' need. In SCADA network, this will open whole new opportunities like auto management of power-flow as per loading conditions e.g. in case of breakdown due to weather conditions, the load has to be reduced according to demand.
5. Green Networking: Present SCADA system is very complex and contains number of hardware components like switches which generate too much heat. 11% of the heat in switches is because of the control plane. Thus, as SDN separates the control plane from switches thereby giving direct reduction of 11% in total power consumption by switches/routers. Moreover, SDN also increases link utilization of the network to almost 100%[3,4].
6. Security: SCADA system largely uses insecure and unencrypted communication networks. Moreover, most of the routing algorithm for SCADA systems uses single-path routing thus making the network more vulnerable to security threats. SDN with its flexibility can give more security to SCADA network as one can program the controller to allow multipath routing as well as enforce security policies easily[5,20].

5. Conclusion

With increasing complexity of SCADA system, managing the SCADA network and adding policies on need basis has become need of the hour. Also, securing the Power Flow Network and reducing malware attacks, eavesdropping and thereby cases of black out of power flow etc. has raised an issue which calls for implementation of software centric approach in SCADA. SDN has successfully managed to pave the way for next generation networks. Authors have studied the traditional network of HPSEBL and found that traditional network lacks flexibility and security mechanism.

SDN-based SCADA system can facilitate robustness and flexible nature to the Power Grid Network Applications. Moreover, simplification and flexibility is also required as new Hydro Projects as well as Solar Based Power Projects

are integrating with the existing Power system. Further, SDN based network will reduce the hardware component resulting in enhancement in green networking. Thus, Software Defined Networking provides an ample opportunity to make SCADA network more secure, flexible as well as robust.

6. References

1. Igure Vinay M, Sean A Laughter, Ronald D. Williams. Security issues in SCADA networks. *Computers and Security.* 2006; 25:498–506.
2. Sharma SK, Modi PK, Sharma MP, Singh SP. Loss reduction in Indian Rural Distribution System by Network Reconfiguration. *International Journal of water and energy.* 2003; 60(1):30–45.
3. Raghavan Barath, Ma J. The energy and energy of the internet. In Proceedings of the 10th ACM *Workshop on Hot Topics in Networks, ACM*, 2011, 9.
4. Jain Sushant, Kumar A, Mandal S, Ong J, Poutievski L, Singh A, Venkata S. B4: Experience with globally-deployed software defined WAN. In: *ACM SIGCOMM Computer Communication Review*, 2013; 43(4):3–14.
5. Germano da Silva, Eduardo, Luis Augusto Dias Knob, Juliano Araujo Wickboldt, Luciano Paschoal Gaspary, Lisandro Zambenedetti Granville, Alberto Schaeffer-Filho. Capitalizing on SDN-based SCADA systems: An anti-eavesdropping case-study. In: *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium*. 2015; 165–173.
6. Goransson P, Black C. Software Defined Networks: A comprehensive Approach, Morgan Kaufmann, June, 2014.
7. Shenker Scott, Casado M, Koponen T, McKeown N. The future of networking, and the past of protocols. Open Networking Summit 20, 2011.
8. Nunes Bruno, Mendonca M, Nguyen X, Obraczka K, Turletti T. A survey of software-defined networking: Past, present, and future of programmable networks. *Communications Surveys & Tutorials, IEEE.* 2014; 16(3):1617–1634.
9. Feamster N. Online course on SDN. www.Coursera.org
10. Foster Nate, Guha A, Reitblatt M, Story A, Freedman MJ, Katta NP, Monsanto C. Languages for software-defined networks. *Communications Magazine, IEEE.* 2013; 51(2):128–134.
11. Reich, Joshua, C. Monsanto, N. Foster, J. Rexford, and D. Walker. Modular SDN programming with pyretic. Technical Report of USENIX, 2013.
12. Voellmy Andreas, Agarwal A, Hudak P. Nettle: Functional reactive programming for openflow networks. 2010, YALEU/DCS/RR–1431.
13. Foster Nate, Harrison R, Freedman MJ, Monsanto C, Rexford J, Story A, Walker D. Frenetic: A network programming language. In: *ACM SIGPLAN Notices*. 2011; 46(9):279–291.
14. Voellmy Andreas, Kim H, Feamster N. Procera: a language for high-level reactive network control. In: *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*, ACM, 2012.
15. Ferguson D. Andrew, Guha A, Liang C, Fonseca R, Krishnamurthi S. Hierarchical policies for software defined networks. In: *Proceedings of the first workshop on Hot topics in software defined networks,* ACM, 2012, 37–42.
16. Mogul C. Jeffrey, Young AA, Banerjee S, Popa L, Lee J, Mudigonda J, Sharma P, Turner Y. Corybantic: Towards the modular composition of SDN control programs. In: *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, 2013, 1.
17. Yuan Yifei, Alur R, Loo BT. NetEgg: Programming Network Policies by Examples. In: *Proceedings of the 13th ACM Workshop on Hot Topics in Networks*- ACM, 2014.
18. Soulé Robert, Basu S, Marandi PJ, Pedone F, Kleinberg R, Sirer EG, Foster N. Merlin: A language for provisioning network resources. In: *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, ACM, 2014, 213–226.
19. Kim Hyojoon, Reich J, Gupta A, Shahbaz M, Feamster N, Clark R. Kinetic: Verifiable Dynamic Network Control.
20. Chandia Rodrigo, Jesus Gonzalez, Tim Kilpatrick, Mauricio Papa, Sujeet Shenoi. Security strategies for SCADA networks. In: *Critical Infrastructure Protection*, Springer US, 2007, 117–131.
21. Open Networking Foundation. SDN Architecture Overview, Version 1.0 December 12, 2013.
22. Power System Communication and Supervision Control and Data Acquisition System of UPPTCL http://www.upptcl.org/tech_info/power_system.htm. Date Accessed: 20/04/2016.
23. Abhinav Sharma, Manu Sood. Network Flexibility and Policy Making in Software Defined Networks. *Transactions on Networks and Communications.* 2015; 3(5):63-78. ISSN: 2054–7420.