# Security Evaluation of Two Intrusion Detection Systems in Smart Grid SCADA Environment

Vivek Kumar Singh, Haythem Ebrahem, Manimaran Govindarasu
Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011
Email:vsingh@iastate.edu, hebrahem@iastate.edu, gmani@iastate.edu

***Abstract*—The increased complexity and interconnectivity of SCADA infrastructure in the power system have exposed it to the multitude of vulnerabilities. There is a growing emphasis towards developing an efficient intrusion detection system (IDS) to strengthen the security of the SCADA control system. This is a research-in-progress paper which presents the application of two anomaly-based intrusion detection systems (AbIDS) in detecting the stealthy cyber-attack on the SCADA control system. We have applied the IDS tools Snort and Bro, in designing the IDS and later, compared their performances in terms of detection rate and latency in the alert packets with a motive of selecting better IDS for the SCADA security. Specifically, the timing-based rule is applied to identify the malicious packets based on the high temporal frequency in the network traffic. For the case study, we have implemented the SCADA based protection scheme which performs an autonomous protection to mitigate the system disturbances. We first implemented the stealthy cyber-attack which compromised the SCADA controller followed by data integrity attack on the system generator. Next, we perform the impact analysis during the attack followed by performance evaluation of IDS tools. Our experimental results show that the IDS tools are efficient in detecting cyber-attacks within an acceptable time frame for different sizes of network packets.**

## I. Introduction

The electric power grid is evolving into complex and interconnected cyber physical system which is operated through the state-of-the-art information and communication technology based SCADA system. The SCADA system works as the brain of the smart grid, which consists of multiple sensors and actuators talking to the control center through the remote terminal units (RTUs) over the wide-area communication [1]. The recent advancements in high-speed communication and data sharing devices have rendered SCADA systems increasingly vulnerable to the multitude of attack surfaces which can be exploited by threat actors, enabling them to design severe sophisticated attacks. Several literature and government documents have highlighted the fact that the critical infrastructure like the power grid is increasingly becoming a constant target of cyber related attacks [2]. In recent years, several malicious cybersecurity incidents have been reported by the Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) targeting the industrial control systems [3]. The paper in [4] provides the detailed documentation related to the several cyber-security incidents related to SCADA critical infrastructures. Based on the analysis, it highlights the observations that the attacks are happening frequently, the majority of the attacks have disrupted normal operation and the attackers are operating in the stealth mode through malware attacks, social engineering, etc. Stuxnet worm, the complex, sophisticated malware, has directly affected more than 100,000 industrial controllers worldwide [5]. The recent hack of Ukraine's power grid is considered a sophisticated malware based coordinated attacks in the SCADA environment which caused shutdown of 7 110 kv and 23 35 kv substations for three hours [6]. The incident is the first known and officially reported cyber-attack causing the power outages.

The SCADA/ EMS system provides essential functions as necessary through wide-area monitoring, protection and control for maintaining the stability and reliability of the power system. In general, the architecture of the SCADA system consists of three layers: Supervisor Control layer, Automatic Control layer and Physical layer [7]. The supervisory control layer operates at the control centers and responsible for the data monitoring and sending control commands. Automatic control layer transmits the control signals to the field devices through the remote terminal units (RTUs), etc. The physical layer is integrated with the sensors and actuators which sense the data and perform necessary control actions based on the control signals. The SCADA control system relies on the communication network for the information exchange and timely operation of geographically distributed devices. Considering the essence of its applications, the existing vulnerabilities in the communication network and computers/devices can be exploited by attackers to launch simple or elaborated classes of attacks like denial-of-service (DoS), data integrity, etc. Moreover, the system cannot prevent themselves from legitimate users who misuse their privileges to perform malicious cyber-attacks like malware attacks, hacking, phishing, insider threats etc. The papers in [7], [8], [9] have demonstrated the communication vulnerabilities, design limitations in SCADA system and necessitated the urgency to strengthen the cyber security applications. Since the SCADA operates non-stop in real time, which cannot be patched or modified frequently, the traditional IDS is not reliable. Therefore, there is a compelling urge for the in-depth analysis of security threats based on the behavior of SCADA system for the development of efficient IDS in the face of advanced, persistent adversaries.

The Intrusion Detection System (IDS) is based on the notion

[1] Acknowledgement: This research is funded in part by NSF CPS and DOE CEDS programs.

that system behavior during the attack would be different from the legitimate behavior. The anomaly-based IDS (AbIDS) identifies the malicious usage or deviations based on the defined threshold during the normal behavior of the system. Therefore, based on the known trails of system behavior and malicious network packets captured during the attack, intrusion detection system can be developed for the SCADA control system. In this paper, we develop an anomaly-based intrusion detection system (AbIDS) using IDS tools, Snort and Bro, in detecting stealthy generation-altering attacks in the context of SCADA based protection scheme. The proposed multi-stage approach involves packet monitoring and filtering, learning the function codes of network packets, defining the rule based on the communication pattern and finally implementing for real-time testing. Specifically, based on the temporal behavior in the network traffic, timing based rule is developed for the DNP3 protocol. We have implemented the proposed IDS using IDS tools in real-time and later evaluated their performances based on experimental results. We have leveraged the testbed resources available in Iowa State's PowerCyber CPS security lab for implementing the attacks, validating and testing the proposed IDS in a SCADA environment.

The organization of the paper is as follows: Section II discusses about the previous works related to anomaly based intrusion detection system. Section III provides a brief overview of protection scheme and Snort, Bro IDS. Section IV talks about creating stealthy cyber- attacks and discusses the approach for developing IDS and related implementation. Section V discusses about the experimental testbed setup, provides impact analysis during the attack and performance evaluation of IDS tools. Finally, the conclusion is provided in the Section VI.

## II. Related Work

There exists a plenty of research work related to developing IDS pertinent to the SCADA system. Cheung et al. presents the model based IDS based on the system behavior for the Modbus TCP networks [10]. The papers presented in [11], [12] have proposed the neural-network based anomaly detection using the SCADA network and system information to detect bad packets, however, they have not considered the internal/insider threats where attackers can inject malicious control logics to the infected critical devices. Yang et al. talks about the hybrid IDS which includes access-control whitelisting, protocol-based whitelisting and behavior-based rules in detecting the external and internal threats [13]. This paper shows the significance of behavior based rules in the deep packet inspection and further achieving 100% accuracy. In a similar work, Sayegh et al. shows how the anomaly based IDS can detect injection attacks using the network packets correlation and system behavior [14]. The paper in [15] talks about the behavior-rule based IDS which detects the attack through the detailed correlation analysis of communication and data payload pattern based on the defined rules. It is analogous to the scenario and outbound based IDS with high detection rate against the malware related attacks. The work presented in [16] shows how the existing platform (Bro) can be applied to detect bad traffic using packet whitelisting, timing characteristics and protocol based validation policies. Valli et al. leverages the Snort tool to develop the intrusion detection system against the network threats for Modbus and DNP3 protocol [17]. The papers in [18], [19] describe the different IDS tools including Snort and Bro, which can be applied in developing real-time detection engine. The papers provide qualitative analysis of the IDS tools in the IT environment. Although very useful, none of these works explicitly focuses on the wide area controllers, including SCADA based protection in the smart grid environment. In this paper, we provide the quantitative evaluation of anomaly based IDS using Snort and Bro against the insider threats in the SCADA environment.

## III. Background

### A. SCADA based Protection Scheme

In this paper, we have considered the SCADA based protection scheme, also known as remedial action scheme, which is an automatic protection scheme which performs corrective actions during disturbances to maintain the power system's stability and reliability [20], [21]. In this work, the remedial action scheme controller (RASc) is operating at the control center, which collects the data from the sensors at regular intervals in terms of relays status, line flows and power output of the generator. During a line outage, the controller is triggered/activated, it checks the operational transfer capability (OTC) of the other adjacent lines directly connected to the generator. If the current line flows exceed its maximum operational transfer capability limit, it performs corrective action by shedding the generation to prevent the thermal overloading in other adjacent lines. Apart from the generation shedding, it is also allowed to restore the generation once the fault/contingency is cleared. More detailed information is provided in [22].

### B. Bro and Snort IDS

Snort and Bro are the most popular network based IDS tools widely used for traffic analysis. Snort IDS is a single-threaded detection engine for real time traffic analysis. Fig. 1 (upper layer) shows the major components involved in this process. It initiated with the packet decoder which collects packets from the network and send them to preprocessor for the required arrangement modifications. The detection engine detects any anomaly based on the predefined snort rules, generate alerts and log messages to the users. BRO IDS is a Unix-based intrusion detection system which provides real time extensive network traffic analysis by deep scanning all the network packets [18], [19]. Fig. 1 (lower layer) shows the major components involved in the process. It initiates with capturing and filtering the packets from the network, and sending the remaining packets to the event engine. The event engine performs various integrity checks by verifying the IP headers checksum and handles parsing of the specific protocol such as DNP3. The generated events are sent to the policy layer which analyses the packets for detecting anomalies and generates alerts and actions based on the scripts/rules.

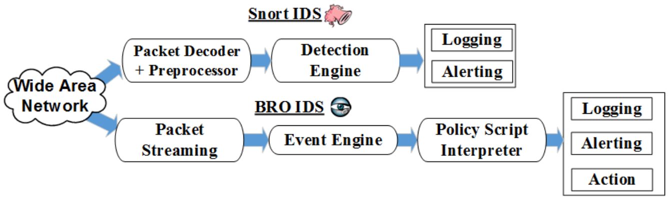
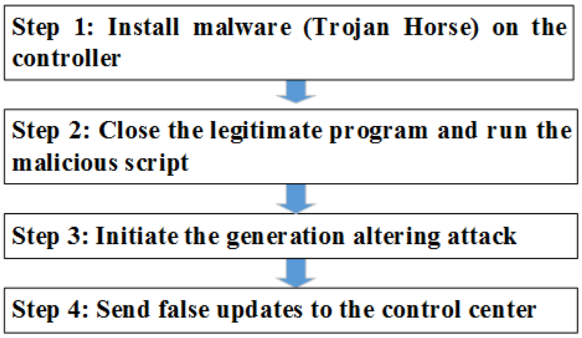


Fig. 1: Major Components of Snort IDS and Bro IDS.

## IV. PROPOSED APPROACH AND IMPLEMENTATION

### A. *Cyber Attack Vector*


Step 1: Install malware (Trojan Horse) on the controller

Step 2: Close the legitimate program and run the malicious script

Step 3: Initiate the generation altering attack

Step 4: Send false updates to the control center

Fig. 2: Steps involved in creating stealthy cyber-attack on RASc.

Fig. 2 shows the different steps involved in creating the stealthy generation-altering attacks through the remedial action controller. In our case, we are assuming that the malware, Trojan Horse, is installed in the RASc using any opening like USB drive, emails or other social engineering skills. Once the malware is installed, it provides backdoor access to the attacker from any network. The affected controller turns into an attacker's boat which blindly follows every command from the attacker. The attacker can read, modify or delete any running program/script as well as transfer any program/script to the affected controller. In this case, once the controller is compromised, attacker transfers the malicious scripts and disables the legitimate RASc program. The malicious script performs two functions. First, it initiates the generation-altering attacks on the generator. Specifically, for creating the generation altering attacks, we have considered the ramp attack model as part of the experiment and mathematically represented in equation 1. Second, it sends the false update to the operator to disguise the attack from getting detected.

The main motivation of considering the ramp attack is that it is difficult to detect in the short time frame as it is slowly altering the generation as compared to other attacks like pulse, scaling attacks, discussed in [17], which causes sharp deviations in the system measurements.

*1) Ramp Attack:* This attack vector involves adding a time varying ramp signal to the input control signal ($P_i$) based on a ramp signal parameter $\lambda_{ramp}$.

$$P_{ramp} = P_i + \lambda_{ramp} * t \tag{1}$$

It is obvious to note that the implemented attack models may not be detected using access control and protocol whitelisting IDS. Therefore, we have developed a rule based IDS which can observe the behavior of network traffic using in-depth protocol analysis to identify the malicious packets which will be discussed in the next sub-section.

### B. *Proposed Approach for IDS*

Fig. 3 shows the generic architecture of the proposed network based intrusion detection system (NIDS) where intrusion detection engine (IDE) is monitoring the network traffic whenever the controller is sending control signals to the actuators in the power system. It is based on the notion that the RASc performs corrective actions only during line contingencies/ faults and such events do not happen very frequently as compared to the continuous attacks as described in the previous section. Therefore, we can detect the attacks based on the threshold values computed by capturing the normal packet during the physical disturbances (faults/contingencies). It is important to note that the proposed detection is tested in the real-time for the DNP3 protocol. The detailed description of DNP3 packet fragments and function codes is beyond the scope of the paper.

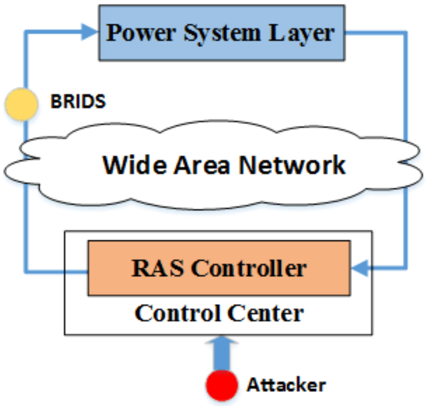


Fig. 3: Generic architecture of intrusion detection system for the modeled attack vectors.

Fig. 4 shows the proposed approach which can be divided into 5 stages:

1) Network-packet sniffing
2) Protocol packet filtering
3) Learning phaseg
4) Rules defining phase
5) Real-time detection

The first stage monitors the network traffic whenever the controller is sending control signals to the actuators in the power system. The second stage filters the normal DNP3 packets based on the IP addresses and port numbers. The third stage learns about the DNP3 packet function codes. Since the controller sends critical commands during the specific scenarios (faults/ contingencies), it has certain time related constraints. It was observed during the attack that the large number of write/operate conditions can be observed in DNP3 packet as compared to the normal operation of the controller. Therefore, we can count the selected write/operate condition function code in a DNP3 application layer over TCP. Based on the packet learning, timing based rule is defined as shown in fig. 4, the fourth stage. $T_n$ and $T_{n-1}$, represents the time of the $n^{\text{th}}$ and $(n-1)^{\text{th}}$ packets where n is the positive integer (n $>$ 0). $T_t$ is the inter-arrival time between the two consecutive packets. The time threshold, $T_{thres}$, is defined based on statistical analysis of the network traffic during the normal disturbances. If the time difference between the two packets, $T_t$, is less than the defined threshold, $T_{thres}$, the alert messages are sent to the operator to take appropriate corrective actions.

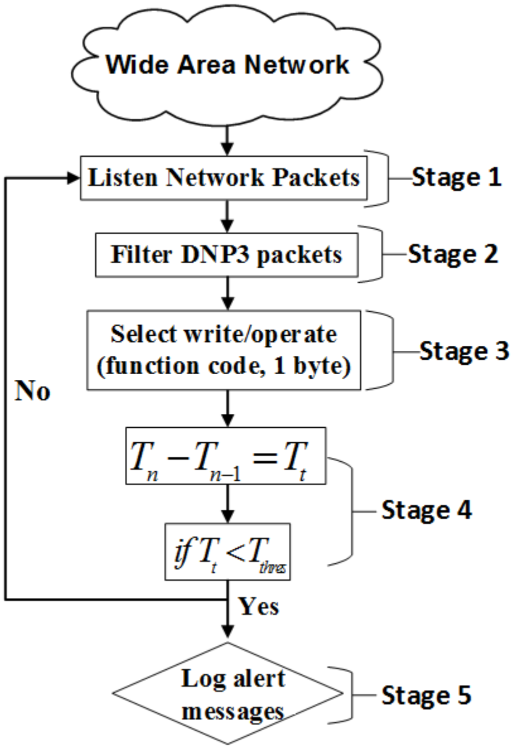


Fig. 4: Proposed intrusion detection engine for the ramp attack.

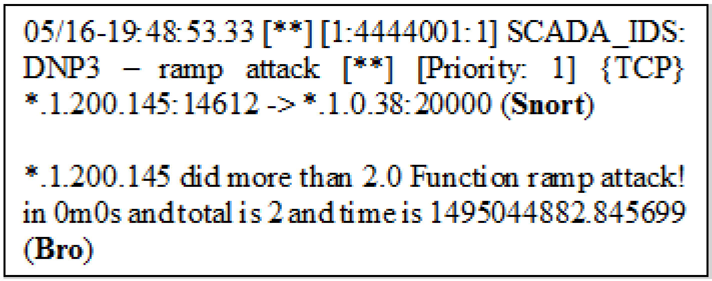
Generally, the RASc sends the first control signal to reduce the generation when the line is out and the second signal to restore the generation back to the initial values after the line is reclosed (auto-reclosing). In this work, based on the paper [3], we have considered the high-speed type auto-reclosing where circuit breaker recloses the line in 0.3 sec which will eventually trigger the RASc to send the second signal. Therefore, the value of $T_{thres}$ is assigned to 0.3 sec for two consecutive normal DNP3 packet.

### C. IDS Implementation

We have developed and implemented the proposed intrusion detection system by utilizing the network based IDS tools, Bro and Snort. In Snort IDS, we have implemented the rule that can detect the event which happens more than C times in time T, where, C is the count threshold for the write request. In this case, the count would be updated every time the RAS controller sends the write request to the simulator. Digital bond has already provided the rule for identifying the specific function code [22]. Hence, we have combined that rule with our defined threshold rule for identifying the malicious packet.

The Bro IDS creates an observer for the specific function code that calculates the number of times the specific event has happened. Once it crosses the threshold value, C, it will send the alert messages to the control center.

In both IDS, we have set the parameters, C=2 and T=$T_{thres}$=0.3 sec to detect the anomalies. If the IDE (Bro, Snort) receives more than 2 packets within 0.3 s, it will send an alert to the control center, otherwise, it will keep monitoring the network traffic. Fig. 5 shows the alerts examples in the log files of Snort and Bro IDS.



```
05/16-19:48:53.33 [**] [1:4444001:1] SCADA_IDS:
DNP3 – ramp attack [**] [Priority: 1] {TCP}
*.1.200.145:14612 -> *.1.0.38:20000 (Snort)

*.1.200.145 did more than 2.0 Function ramp attack!
in 0m0s and total is 2 and time is 1495044882.845699
(Bro)
```

Fig. 5: Alert examples in the log files of Snort and Bro IDS.

## V. Experimental Evaluation

### A. Experimental Setup

Fig. 6 shows the experimental setup for the attack-detection experiment using the testbed. We have modeled the modified IEEE 9 bus system on the real time digital simulator (RTDS). The distributed remedial action scheme is implemented in the system where each RASc is operating for a single generator. The controller, RASc2, operating for the generator 2, is communicating through the DNP3 protocols to the simulator. It collects data in terms of relay status, line flows and power generation at every 0.125 second and takes corrective actions by shedding different level of generations to avoid thermal overload during the contingency. For simplicity, we have the considered the overhead limit to be 1.5 times of the initial line flows. In the attack scenario, as shown in blue dashed arrows, we have installed the malware (Trojan Horse), written in python script for Windows hosts, in RASc2 which provides unauthorized access to the attacker. Once malware is installed, the attacker transfers the fake RAS script to the affected controller using Cryptcat [23]. The Cryptcat is a Unix utility which allows data/file transferring in encrypted form. In the next step, the attacker closed the original RAS script and malicious script is executed. The malicious script initiates the ramp attack on the generator while sending fake updates to the control center operator. For attack detection, IDS tools Snort and Bro are running in Kali Linux VMware which are listening the ongoing traffic between the controller and RTDS.

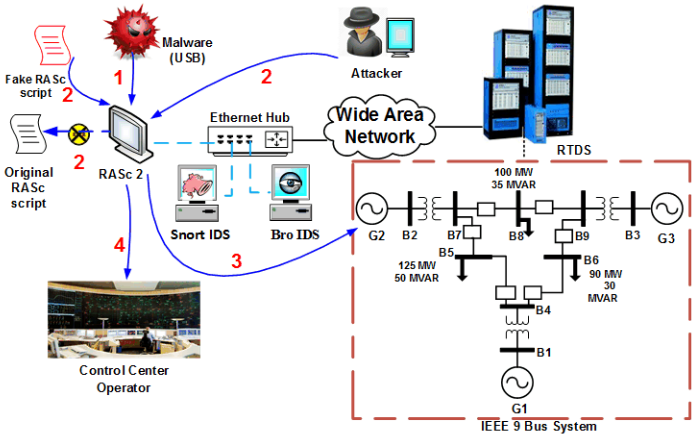


Fig. 6: Experimental setup for attack implementation and detection on the PowerCyber Testbed.

### B. Results and Discussions

*1) Impact Analysis:* Fig. 7 shows the system behavior during ramp attack which is the final attack vector. Due to the successful stealthy attack, the ramp attack starts at 12.4 s, when attacker slowly ramps down the generator 2 output (Pg2). The continuous ramping forces the generation lower than allowable limit during the line contingency. The continuous decrease in the generation level drives the frequency to decrease below the 60 Hz and eventually triggering the underfrequency load shedding (UFLS) at bus 8. We have modeled the UFLS which sheds the loads at two stages, first at 59.5 Hz and second at 59.3 Hz [20]. In this case, the frequency reaches to 59.5 Hz at around 25.4 s, triggering the UFLS stage 1 which sheds 40 MW of load. During stage 1 load shedding, frequency improves slightly, but keeps on decreasing as the attacker keeps on pushing the generation away from the safe operating point. The additional 40 MW of load shedding occurs when the frequency decreases to 59.4 at around 34 s, as shown in fig. 7 (c). During the attack period of 21.6 s, the system has lost a major portion of the load (80%) while causing a significant impact on system stability.

*2) Performance Evaluation of IDS:* We have evaluated and compared the performance of Bro and Snort IDS in terms of detection rate and latency in the alert messages. We have also computed the number of alert packets dropped for different sizes of packets for both the cases. Higher number of alert packets dropped signify more false negatives as the false negative represents the cases the detection engine fails to detect an attack. Fig. 8 shows the number of alert messages dropped with respect to the total number of packets sent from the controller to the simulator. It can be observed that Snort and Bro have similar performances for small size of packets, however, the gap the between the two graphs increases with the number of packets and Bro IDS exhibits better performance than Snort especially during the huge chunks of alert messages. Fig. 9 shows the detection rate for different sizes of alerts packets. It is the ratio of captured alert messages to the total alert messages. We have varied the alert packets from 21 to 2000, and it can be observed that the Bro IDS has performed consistently with a detection rate greater than 90%, while the Snort IDS's detection rate varies sporadically from 93.5% to 75%. Fig. 10 shows the computed average latency in the alert messages for different number of packets. We observe the constant delay of 0.6 s in the most cases of Snort IDS. The Bro IDS tends to perform slightly faster with maximum average delay of 0.58 sec and minimum of 0.534 sec.

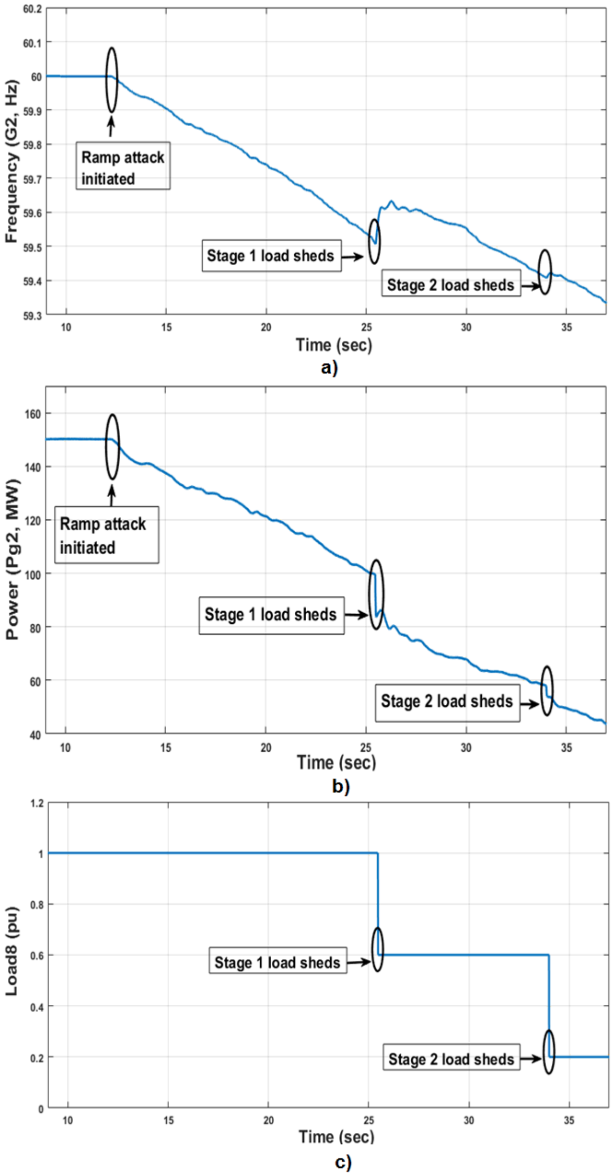
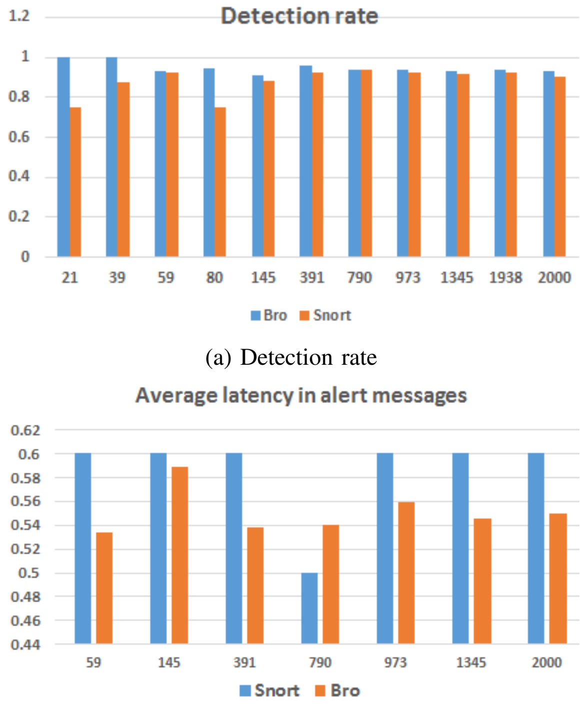


Fig. 7: Power output (Pg2), Frequency and Load (Bus8) during Ramp attack.



(a) Detection rate

(b) Average latency

Fig. 8: Detection rate and Average latency of alert messages for Bro and Snort IDS for different sizes of packets.

## VI. CONCLUSION

In this paper, we have showcased the application of IDS tools, Snort and Bro, in developing the anomaly-based intrusion detection system (AbIDS) for detecting the generation altering attack on the SCADA based protection scheme, also known as remedial action scheme. The proposed multistage IDS approach involves listening and filtering the network packets between the controller and power system simulator, learning the function codes (write/ operate conditions) and eventually detects the anomaly packets based on the timing based rule between the two consecutive packets. In this work, the detection approach is developed for DNP3 protocol, however, it can also be applied to other SCADA based protocols. We have applied the IDS tools for developing the real-time intrusion detection engine in the SCADA environment. We also described several steps involved in creating the ramp attack on the generator and discussed the experimental setup for the attack implementation and detection on ISU's PowerCyber testbed. We then performed the impact analysis in terms of system frequency and forced load shedding during the ramp attack. Furthermore, we evaluated the performances of IDS tools in terms of detection rate and latency in the alert packets. It can be inferred from our experiments that the Bro IDS has performed better in terms of detection rate and latency. Our experimental results also showed that the deployed IDS tools were able to detect the attacks in a small-time frame (0.5-0.6 s) with the average detection rate of around 88% for Snort IDS and 94.7% in the case of Bro IDS. For future studies, we are planning to develop multiple rule sets based on the system behavior in multi-dimensions to detect different classes of attacks.

## REFERENCES

[1] Y. Zhang and J. L. Chen, ”Wide-area SCADA system with distributed security framework,” in Journal of Communications and Networks, vol. 14, no. 6, pp. 597-605, Dec. 2012.

[2] NERC Critical Infrastructure Protection Committee (CIPC) Cyber Attack Task Force (CATF) Update, North American Electric Reliability Corporation (NERC), Dec. 2011.

[3] ICS-CERT,”Monitor (ICS-MM201212)”, january 2012 [Online].

[4] B. Miller and D. Rowe, A survey of SCADA and critical infrastructure incidents, Proceedings of the First Annual Conference on Research in Information Technology, pp. 51–56, 2012.

[5] N. Falliere, L. O'Murchu, and E. Chien. W32.stuxnet dossier. Technical report,Symantec, Feb. 2011.

[6] ICS-CERT, “Cyber-Attack Against Ukranian Critical Insfrastructure”[Internet]; 2016.

[7] V. L. Do, L. Fillatre, I. Nikiforov and P. Willett, ”Feature article: security of SCADA systems against cyber–physical attacks,” in IEEE Aerospace and Electronic Systems Magazine, vol. 32, no. 5, pp. 28-45, May 2017.

[8] A. Hahn, A. Ashok, S. Sridhar, and M. Govindarasu, “Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid,” Smart Grid, IEEE Transactions on, vol. 4, no. 2, pp. 847–855, 2013.

[9] C. W. Ten, C. C. Liu and M. Govindarasu, ”Vulnerability Assessment of Cybersecurity for SCADA Systems Using Attack Trees,” 2007 IEEE Power Engineering Society General Meeting, Tampa, FL, 2007, pp. 1-8..

[10] S. Cheung, B. Dutertre, M. Fong, U. Lindqvist, K. Skinner, and A. Valdes, “Using model-based intrusion detection for SCADA networks,” In Proceedings of the SCADA Security Scientific Symposium, Miami Beach, FL, USA, 2007.

[11] O. Linda, T. Vollmer, and M. Manic, “Neural network based intrusion detection system for critical infrastructures,” In International Joint Conference on Neural Networks, 2009., pages 1827 –1834, June 2009.

[12] W. Gao, T. Morris, B. Reaves, and D. Richey, “On SCADA Control System Command and Response Injection and Intrusion Detection,” in the Proceedings of 2010 IEEE eCrime Researchers Summit. Dallas, TX. Oct 18-20, 2010.

[13] Y. Yang et al., ”Multiattribute SCADA-Specific Intrusion Detection System for Power Networks,” in IEEE Transactions on Power Delivery, vol. 29, no. 3, pp. 1092-1102, June 2014.

[14] N. Sayegh, I. H. Elhajj, A. Kayssi and A. Chehab, ”SCADA Intrusion Detection System based on temporal behavior of frequent patterns,” MELECON 2014 - 2014 17th IEEE Mediterranean Electrotechnical Conference, Beirut, 2014, pp. 432-438.

[15] Masayoshi Mizutani, et al, “Behavior Rule based Intrusion Detection”, CoNEXT Student Workshop'09, 2009.

[16] Robert Udd, Mikael Asplund, Simin Nadjm-Tehrani, Mehrdad Kazemtabrizi, and Mathias Ekstedt. 2016. Exploiting Bro for Intrusion Detection in a SCADA System. In Proceedings of the 2nd ACM International Workshop on Cyber-Physical System Security (CPSS '16). ACM, New York, NY, USA, 44-51.

[17] C. Valli, “SCADA Forensics with Snort IDS,” in Proceedings of WORLDCOMP2009, Security and Management, pp. 618-621, USA, 2009.

[18] George Khalil, “Open Source IDS High Performance Shootout” SANS, Infosec reading room, Feb 2, 2015.

[19] D. A. Bhosale and V. M. Mane, ”Comparative study and analysis of network intrusion detection tools,” 2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Davangere, 2015, pp. 312-315.

[20] NERC,”Remedial Action Development” Definition Development project 2010-05.2 –Special Protection System.

[21] WECC remedial action scheme catalog summary [Internet]; 2008

[22] V. Kumar Singh, A. Ozen and M. Govindarasu, ”Stealthy cyber attacks and impact analysis on wide-area protection of smart grid,” 2016 North American Power Symposium (NAPS), Denver, CO, 2016, pp. 1-6.

[23] Digital Bond For Secure and Robust ICS.