De-Synchronisation Attack Modelling in Real-Time Protocols Using Queue Networks: Attacking the ISO/IEC 61850 Substation Automation Protocol

James G. Wright[1] and Stephen D. Wolthusen[1,2]

[1] School of Mathematics and Information Security,
Royal Holloway, University of London,
Egham TW20 0EX,
United Kingdom
[2] Norwegian Information Security Laboratory
Norwegian University of Science and Technology
Norway james.wright.2015@live.rhul.ac.uk &
stephen.wolthusen@rhul.ac.uk

Abstract. Applications for developed Supervisory Control And Data Acquisition (SCADA) protocols in several domains, particularly the energy sector, must satisfy hard real-time constraints to ensure the safety of the systems they are deployed on. These systems are highly sensitive to Quality of Service (QoS) violations, but it is not always clear whether a compliant implementation will satisfy the stated QoS in the standard. This paper proposes a framework for studying a protocol's QoS properties based on a *queuing network* approach that offers a number of advantages over state machine or model-checking approaches.
The authors describe the framework as an instance of a network of M/M/1/K of queues with the block-after-service discipline to allow for the analysis of probabilistic packet flows in valid protocol runs. This framework allows for the study of denial of service (DoS), performance degradation, and de-synchronisation attacks. The model is validated by a tool allowing automation of queue network analysis and is used to demonstrate a possible breach of the QoS guarantees of the ISO/IEC 61850-7-2 substation automation standard with a de-synchronisation attack.

Keywords: Queue Networks, ISO/IEC 61850, Quality of Service, Protocol Analysis, De-synchronisation Attack

1 Introduction

The methods that are used to secure SCADA technologies are being stretched due to their incorporation into Smart Grid (SG) communication standards. To maintain the safety of the physical distribution infrastructure, stringent QoS requirements that govern how instructions and data are transmitted and processed across the network must be maintained by the implementer of the SG's communications network. Any delays in transmission could potentially lead to the disruption and damage of the physical distribution network. The imbalance of maintaining the QoS requirements over the security

promises laid out in the communications standards in a SG cyber-physical system could lead to the safety of physical infrastructure still being compromised. This is due to malicious actors being able to force the communications network into either an undesirable state, or being allowed to send malicious commands. Whilst both the academic and industrial research communities are now focusing on solving these new security challenges posed by SG technologies, there is very little focus dedicated to checking if the security and QoS promises made by SCADA protocols hold true. It is important that the communications standards have rigid security definitions as the networks they will be deployed on will be made up of devices supplied by different manufactures, who may have different interpretations of the standard, as well as the infrastructure itself being required to run uninterrupted on a time scale of decades. Alongside this the communications network will be distributed across a wide area network, which provides an adversary with ample attack vectors to undermine the security of the cyber infrastructure. A benefit of securing the protocols is that it could prevent some of the attacks that have already been implemented against SG technologies using these standards[2].

The focus of this work is to describe and demonstrate the effectiveness of a framework developed by the authors to analyse distributed communications models that are being incorperated into SCADA standards. The authors have pursued a new framework because the standards that govern communications networks provide a large number of communications models, whose permutations of use would make the searchable space of potential attacks close to boundless[13]. Also SG standards are more flexible on the acceptable behaviour within a communication models than their cryptographic counterparts, so traditional model checking frameworks would find it difficult to detect undesirable behaviours in their communications models.
The framework uses queuing networks to model the flow of both regular and malicious packets between the stages of the state machine representing a possible protocol run. The model can analyse how an adversary interacts with the semantic steps of either the client or server machines in a run that involves, theoretically, as many communication models as the user of the framework needs, in attempt to drive it into a undesirable state. Each queue in the network is truncated to allow for a more realistic modelling of buffer sizes of the state machine, so that packets can be prevented from travelling through the network if all of its recipient queues are full. This inclusion allows DoS attacks against a run to be encapsulated in any model computed by the framework. The framework also allows the user to model attacks against the semantics of synchronization of state machines in a protocol run. In its current form the framework provides a global view of the run, showing the user the mean total number of packets in each stage of the run. The mean total number of packets can be used to derive a range of performance metrics that allow the user analyse the affects that an attack will have on a specific part of the protocol run. The framework can also be reconfigured to look at attacks at the packet level of the network.

The author's are primarily using the framework to investigate the promiseses of IEC 61850 substation automation standard, which has stringent QoS requirements governing communications network. These stringent requirments provide a fertile development

ground for the discovery of bespoke attacks that only arise in networks with distributed communication topologies. As well as this, the generation of any potential solutions for these undesired behaviours must be efficient enough meet the QoS of the protocol. The real-time nature of the protocol means that the communications networks of the SG are easily susceptible performance degradation and attacks against the security promise of availability. However, the way in which framework has been developed means that any attacks discovered, and their potnetial solutions, can be applied to other SCADA and distributed communications standards. Using the framework the authors have discovered an attack against IEC 61850's control communication model[12]. In this attack the adversary alters the rate of processing of one of the queues in the protocol run, which increases the probability that the server's state machine will go down the timeout path of the run. As the server's state machine doesn't announce that it has timed out, the client continues sending request for the server to follow. This will eventually lead to a de-synchronisation of state between the client and server, and will force both machines to reset the run.

The remainder of this paper proceeds as follows: Section 2 describes the current research on how DoS and timing attacks affect SGs, along with how queuing theory has been used to model DoS attacks. Section 3 goes onto to describe the mathematical formalism used in the framework. Then section 4 describes the de-synchronisation attack demonstrated by in framework. Before giving a conclusions and a direction for future work in section 5.

2 Related Works

Many different frameworks have been developed over the years to model the security promises of cryptograpic standards. The underlying principle of each varies; the main ones being belief logics, theorem proving and state exploration, along with an array of different model checkers[17].

The main use of the queing theory formalism in the security domain has been to describe DoS attacks. It is a suitable formalism for this type of attack, as DoS scenarios can be modeled without much abstraction. This is due to how queuing theory calculates the efficency of a series of objects being processed in a queue according to a specific set of rules. These rules can easily be translated to represent a processor with a specific memory allocation. Despite this natural affinity, seemingly little research has been persued in the modelling DoS attacks using queuing theory. Most of the research describes various packet level scenarios with either a single $M/M/1$ queue or an open Jackson network, which is a network of $M/M/1$ queues. Relying on $M/M/1$ limits what the user can discern from the framework, as a queue of this type can hold an infinite number of objects. Given this underlying assumption, all that can be discerned from these models is the degradation of the queues performance. It doesn't tell the user at what point the system being modelled will fail to meet its promise of avaliability. However, Xiao-Yu *et al.*[22] does use a $M/M/c/K$ queue to investigate SIP INVITE request flooding scenario. Their solution is to create a queue that deals only with INVITE re-

quests. Kammas *et al.*[8] created an open Jackson network of $M/M/1$ queues to model virus propagation across a network. Their state space included the internal transitions of state of each node, as well as the global state of the network. Wang *et al.*[23] developed a mathematical model, using embedded two dimensional Markov chains, to allow the user to generate different probability distributions for any distribution of acceptance rates. Their model also allows for the separate analysis of the malicious packets properties from the normal traffic's.

There has been some research into how DoS attacks will affect SG communication networks. Hurst *et al.*[6] developed a mathematical framework to help security practitioners to evaluate the scale of the damage their communication network would suffer if they were attacked by various types of distributed DoS attacks. Liu *et al.*[10] demonstrated that a DoS attack against load frequency controls can affect the stability of the power grid. Li *et al.*[9] studied the time delay suffered by critical communication packets on an IEC 61850 communications network, when either the physical or application layer is flooded with malicious messages. There seems to have been no other research into how a DoS attack could be used against the IEC 61850 standard. One potential solution to this problem is to use flock based behavioural transitions rules to make sure the packets avoid denying a node availability to the network[24]. Ansilla *et al.*[1] developed a hardware based algorithm to deal with SYN flooding on SG networks. Srikantha & Kundur[18] developed a game theoretic approach to deal with DoS attacks. Their solution is a system that provides each node in the communication network with a reputation score, which, if a node is compromised, automates the rerouteing process to exclude that node from the communications topology.

Clock synchronisation protocols are a fertile group of standards investigation of desynchronisation attacks. There have been various analyse of the security vulnerabilities of the Network Time Protocol (NTP)/Precision Time Protocol (PTP) protocols, where a taxonomy of the various attack vectors against the network show how an andversary would be able to either manipulate or control the network[7, 4, 11]. Each taxonomy has proposed various countermeasures, such as introducing the Confidentiality Integrity Authentication triad into this domain or basing the protocol on the peer-to-peer network paradigm. Ullmann & Vogler[21] performed an analysis on the consequence of a delay attack against both the NTP and PTP protocols. They proved that a delay in a sync message would affect all the client clocks, and a delayed request message would only affect the client that sent the message. They proposed implementing a Secure Hash Algorithm on the protocol to mitigate these attacks. Tsang & Beznosov[20] created a qualitative taxonomy of attacks against the PTP protocol. They laid out how an adversary could potentially misuse certain messages in the protocol to create undesirable effects, while suggesting countermeasures for most of them. Mizrahi[14] developed some game theoretic strategies to prevent delay attacks against NTP. Moussa *et al.*[15] who produced a detailed analysis of the consequences of a delay attack in a SG substation environment. They also provided a mathematical model to counter delay attacks.

3 Mathematical Theory of the Framework

The framework uses a network of truncated $M/M/1/K$ queues to create a probabilistic state exploration methodology for checking the promises of a protocol. The $M/M/1/K$ provides a more realistic model than the $M/M/1$ queues because it imposes a limit, K, on the length of the queue, so when the queue is full it will no longer accept any packets. This allows the user to model attacks against the availability of a step in a protocol run. Also, using a network of $M/M/1/K$ provides the user with the versatility of being able to model different layers of abstraction of the system, as it can be set up to represent the flow of packets over the communications network as well as the semantics of a protocol run. When the formalism is used to describe the semantic flow of run each queue represents an action that a state machine can perform. The framework adapts the work of Osorio & Bierlaire[16], which describes the state of an individual queue in a $M/M/c/K$ network, to describe the global state of the network.
The assumptions made by the framework are;

- Each queue obeys the first-in-first-out (FIFO) discipline for processing packets.
- If the queue is blocked, it uses the blocked-at-service discipline.
- The process time and time between successive unblocking are each assumed to follow an exponential distribution. However, the effective probability distribution describing the rate at which packets are complete its time with a queue isn't an exponential distribution.
- That the transition between states is memoryless.

The rest of the this section is dedicated to describing the mathematical formalism used by the framework to create models. It first gives a brief overview of continuous time Markov Chains (CTMC) that are used to find the probabilities of certain events occurring within the network. The next section describes how the physical properties of each queue in the network are calculated. The state space that is used to generate the transition rate matrix is described, before, finally, going over the various performance metrics the framework provides. The below description focuses on the assumptions and set up for the semantic flow protocol runs.

3.1 Probability for Queuing Theory

As the state of the queuing network is memoryless, the probability of a transition happening between states can be described using CTMC[5]. To find the vector of the probabilities of the state that the network will be in, the global balance equation must be solved. The assumptions used in solving this equation are that the system:

- It is independent of time.
- It is independent of the initial state vector.

If the CTMC are ergodic, then an unique steady state probability vector, $\boldsymbol{\pi}$, exists that is independent of any initial probability vector. This means the global balance for each state can be described by the conservation of probability flux in and out of the state:

$$\sum_{j \in \mathscr{I}} \pi_j q_{ji} = \pi_j \sum_{j \in \mathscr{I}} q_{ij} \tag{1}$$

where π_j is the probability of being in a state, q_{ij} is the rate of transmission between state i and j, and $\mathscr{I}$ is the state space. Rearranging equation (1) it can be represented in the matrix form:

$$\mathbf{0} = \boldsymbol{\pi} \boldsymbol{Q} \tag{2}$$

where Q is the transition matrix. The steady state vector can be found by solving the system of linear equations described in (2), using the boundary condition:

$$\sum_{i \in \mathscr{I}} \pi_i = 1. \tag{3}$$

3.2 The Mathematical Description of the Topological Space

The first step that the framework must do is to calculate the parameters that govern how each queue in the network performs. For this framework that is being developed there are a few built in assumptions that govern how the calculations are performed. Packets in a queue can be in one of three states, being processed, a, waiting to be processed, w, or blocked b. For queues in the model's network $a+b \leq c$, where the number of servers of a queue $c = 1$. The next assumption of the framework is $a+b+w \leq K$, where K is the maximum capacity of packets in each queue in the network. Another assumption when using the framework to model a semantic protocol run is that a packet cannot return to a queue behind it in the network topology. This guarantees that the model doesn't break causality as protocols runs are assumed to have unidirectional flow of time built into them.

Before calculating the endogenous parameters of each queue in the network, the user must set up a series of exogenous parameters for each node. These are:

- K_i: The maximum capacity of each queue.
- μ_i: The service rate of each queue
- γ_i: The external arrival rate to a queue, if it is at the starting edge of the network.
- $\phi(i,1)$: The average number of distinct target queues that are blocking a packet at each queue. If a queue is at the concluding edge of the network this term is not required. A method for approximating this value is given in Osorio & Bierlaire[16].
- p_{ij}: The probability of packet transitioning from queue i to queue j once processed.

Once these have been set, the rest of the parameters describing the queues behaviour can be solved by using the following set of non-linear simultaneous equations. The first equation calculates the probability that a queue is full. It is a standard result for $M/M/1/K$ queues. The probability that a queue is full is given by:

$$P(N_i = K_i) = \frac{(1-\rho_i)\rho_i^{K_i}}{1-\rho_i^{K_i+1}} \tag{4}$$

where $\rho_i = \frac{\lambda_i}{\mu_i^{eff}}$ is the traffic intensity[5]. In the steady state approximation $\rho < 1$. This equation calculates the total arrival rate into the queue, including those that are lost:

$$\lambda_i = \frac{\lambda_i^{eff}}{1 - P(N_i = K_i)} \tag{5}$$

This is the effective arrival rate of only the packets that are processed by the queue:

$$\lambda_i^{eff} = \gamma_i(1 - P(N_i = K_i)) + \sum_j p_{ji}\lambda_j^{eff} \tag{6}$$

$\mathscr{P}_i$ is the probability that of being blocked at a queue:

$$\mathscr{P}_i = \sum_j p_{ij} P(N_j = K_j) \tag{7}$$

The following equation is the approximation that presents the common acceptance rate of all the queues that a queue can send a packet to:

$$\frac{1}{\widetilde{\mu_i^a}} = \sum_{j \in \mathscr{I}^+} \frac{\lambda_j^{eff}}{\lambda_i^{eff}\mu_j^{eff}} \tag{8}$$

The effective service rate of a queue, which includes time being blocked, is given by:

$$\frac{1}{\mu_i^{eff}} = \frac{1}{\mu_i} + \frac{\mathscr{P}_i}{\widetilde{\mu_i^a}\phi(i,1)} \tag{9}$$

3.3 The Description Probabilistic View of the Protocol Run

Once all of the queue's parameters have been discerned, a transition rate matrix, Q, can be generated for the transitions between all the possible states in the state space. Once that is done $\boldsymbol{\pi}$ can be calculated and the marginal probability of an event happening can be calculated.
Currently the framework allows the user to calculate the most likely path across a protocol run. The state space for this view is given by.

$$\mathscr{I} = \{(k_1, ..., k_N) \in \mathbb{N}^N\} \tag{10}$$

There are three types transitions that can occur in this state space.

Initial state s	New state t	Rate q_{st}	Conditions
$(i, ...)$	$(i+1, ...)$	λ_i	$p_{0i} \neq 0$ & $N_i \leq k_i - 1$
$(..., i)$	$(..., i-1)$	μ_i^{eff}	$p_{0i} \neq 0$ & $N_i \geq 1$
$(..., i, ..., j, ...)$	$(..., i-1, ..., j+1, ...)$	μ_i^{eff}	$p_{ij} \neq 0$ & $N_i \geq 1$ & $N_j \leq k_i - 1$

1. Flow into the network.
2. Flow out of the network.
3. Packet transmission between states.

Q is then used to find the steady state vector for this CTMC.

The marginal probabilities that can be calculated in this view is the likelihood that a specific queue has k packets. It is calculated by summing the probability of all the states that have k packets in the queue of interest:

$$\pi_i(k) = \sum_{k_i = k \in \mathscr{I}} \pi(k_1, ..., k_N) \tag{11}$$

3.4 Performance Metrics

Below are various performance metrics that can be applied to the marginal probabilities produced by this mathematical formalism.

Performance Metric	Equation
Traffic Intensity	$\rho_i = \sum_{k=1}^{k} \pi_i(k)$
Throughput	$\lambda_i = \sum_{k=1}^{k} \pi_i(k)\mu_i^{eff}$
Total Throughput	$\lambda = \sum_{i=1}^{N} \lambda_{0i}$
Mean Number of Packets	$\bar{k_i} = \sum_{k=1}^{k} k\pi_i(k)$
Mean Queue Length	$\bar{q_i} = \sum_{k=c_i}^{k} (k-c_i)\pi_i(k)$
Mean Response Time	$\bar{T_i} = \frac{\bar{k_i}}{\lambda_i}$
Mean Wait Time	$\bar{W_i} = \bar{T_i} - \frac{1}{\mu_i^{eff}}$
Mean Number of visits	$e_i = \frac{\lambda_i}{\lambda}$
Relative Utilisation	$x_i = \frac{e_i}{\mu_i^{eff}}$

4 Results

This section describes de-synchronisation attack against the IEC 61850's control model that has been discovered using the framework described in section 3.

4.1 De-synchronisation attack

The attack developed using the framework shows that an adversary can cause the client's and server's state machines to become de-synchronised. The adversary achieves this by either increasing or decreasing the rate at which the server receives the $oper-req[TestOK]$ in the Select Before Operate (SBO) control model, described in section 19.2.2 of IEC 61850-7-2 (shown in a truncated form in fig. 1). The adversary can cause this disruption of state because the standard can be interepreted as not requiring the server to send out a *timeout* message to the client. Whilst this is happening the client still thinks the run is following the operation request branch of the protocol run, and will expect an update that will never arrive. This is a legitimate interpretation of the standard that devices will have to be prepared for, due to the QoS promise of interoperability of all devices regardless of manufacturer. This attack vector provides the adversary the ability to create doubt over the state of any logical node that has "*data object instances of a Controllable common data class and whose ctlModel DataAttribute that is not set to status-only*"[12]. This includes safety equipment, such as circuit breakers, whose response to an emergency situation have to be completed within $5ms$[19]. If the disruption caused by the de-synchronisation of states causes a control command to violate the QoS requirements, the adversary can cause physical damage to the distribution network.

The adversary in this result is the same the symbolic one described by Dolev-Yao[3]. The adversary in this model is an omnipotent and omnipresent party on the communications network who "*can intercept messages before they reach their intended destination,*

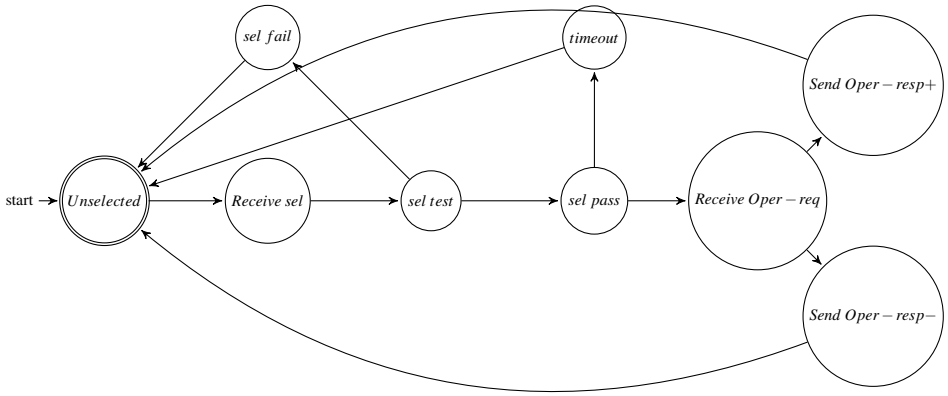

Fig. 1: A truncated version of the server side of the SBO version of the control communication protocol run described in section 19.2.2 of IEC61850-7-2.

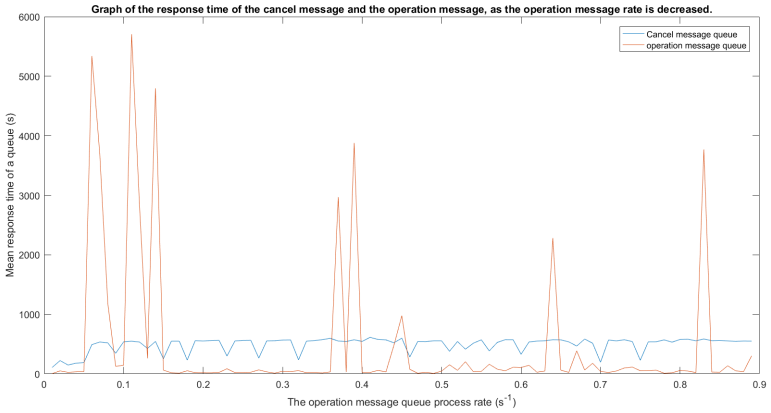

Fig. 2: The graph shows the mean response time a queue dealing with *timeout* and receiving the $oper-req[TestOK]$ message as the adversary changes the processing rate of the $oper-req[TestOK]$.

it can modify and reroute them, possibly with invalid sender fields".
Fig. 2 shows that adversary can increase the mean response time of the protocol run receiving the $oper-req[TestOK]$ message by several orders magnitude by altering the processing rate of queue accepting those packets. The dramatic increase in the response time to receiving the packets increases the probability that the protocol run will take the $timeout$ path, which would cause the de-synchronisation. In this analysis no other variables were altered. The analysis also used the truncated form of the protocol run depicted in Fig. 1.

5 Conclusion

The above analysis has shown that the explicit QoS requirements of IEC 61850 are not upheld throughout specification. This has been shown using a framework that allows the user to look at how specific adversarial interactions with a protocol run can undermine certain performance metrics in the data stream. The de-synchronisation attack allows an adversary to cause confusion as to the state of a physical device on the SG's communication network. This could cause physical damage to the cyber-physical system if the client's communications are delayed due to having to reset its state machine, and begin the protocol again when it is dealing with safety equipment. This attack could be prevented by having the standard declare that the server broadcast that the session has timed out, so the client can revert back to the previous known state without having to query the server to find out what state it is in.
Future versions of the framework will allow for greater resolution in describing this attack, as it will give the user the ability to calculate the probability that the protocol run will go down a certain path.

Progressing onwards, the authors plan to further develop the framework by allowing the user to see the possible internal states of each queue. The two possible views are a module that allows for the probability of the specific packet ordering to be calculated, and a module to see whether the packets are blocked, processing, or waiting. These modules will be added so adversaries with less knowledge and capabilities can be modelled. They will also add a calculation parameter to the non-linear equations that calculates the probability that a blocked packet will drop out of the network after a certain period of time. After this the aim is to develop more performance criteria for other types of undesirable metrics in the communication network so that more attacks against the security and QoS promises of IEC 61850 can be discovered in the other communication models in the standard. Once the development of the framework is complete, the development of new attacks will be undertaken. The ultimate aim of the authors is to produce attacks that have a more realistic adversary, one that isn't omnipresent in the communications network, and develop a framework in such a way that it can warn SG managers if they are under attack.

6 Acknowledgment

This work is supported by an EPSRC Academic Centres of Excellence in Cyber Security Research PhD grant. The authors would like to thank Joshua Robinson and Ela Kasprsky for their help in understanding some of the mathematical concepts used in this paper.

References

1. J. D. Ansilla, N. Vasudevan, J. JayachandraBensam, and J. D. Anunciya. Data security in Smart Grid with hardware implementation against DoS attacks. In *International Conference on Circuits, Power and Computing Technologies, ICCPCT 2015*, pages 1–7, March 2015.
2. A. Cherepanov. WIN32/INDUSTROYER: A New Threat for Industrial Control Systems. Technical report, ESET, 12-06-2017.
3. D. Dolev and A. Yao. On the Security of Public Key Protocols. *IEEE Transactions on Information Theory*, 29(2):198–208, Mar 1983.
4. G. Gaderer, A. Treytl, and T. Sauter. Security aspects for IEEE 1588 based clock synchronization protocols. In *IEEE International Workshop on Factory Communication Systems, WFCS 2006, Torino, Italy*, pages 247–250. Citeseer, 2006.
5. D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris. *Fundamentals of Queueing Theory*. Wiley-Interscience, New York, NY, USA, 4th edition, 2008.
6. W. Hurst, N. Shone, and Q. Monnet. Predicting the Effects of DDoS Attacks on a Network of Critical Infrastructures. In *IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, 2015*, pages 1697–1702, Oct 2015.
7. E. Itkin and A. Wool. A security analysis and revised security extension for the precision time protocol. In *IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication, ISPCS 2016*, pages 1–6, Sept 2016.
8. P. Kammas, T. Komninos, and Y. C. Stamatiou. A Queuing Theory Based Model for Studying Intrusion Evolution and Elimination in Computer Networks. In *The Fourth International Conference on Information Assurance and Security*, pages 167–171, Sept 2008.
9. Q. Li, C. Ross, J. Yang, J. Di, J. C. Balda, and H. A. Mantooth. The effects of flooding attacks on time-critical communications in the smart grid. In *2015 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5, Feb 2015.
10. S. Liu, X. P. Liu, and A. E. Saddik. Denial-of-Service (dos) attacks on load frequency control in smart grids. In *IEEE PES Innovative Smart Grid Technologies Conference, ISGT 2013*, pages 1–6, Feb 2013.
11. A. Malhotra and S. Goldberg. Attacking NTP's Authenticated Broadcast Mode. *SIGCOMM Comput. Commun. Rev.*, 46(2):12–17, May 2016.
12. TC 57 Power Systems Management and Associated Information Exchange. Communication Networks and Systems for Power Utility Automation - Part 7-2: Basic Information and Communication Structure - Abstract Communication Service Interface. IEC standard 61850-7-2. Technical report, International Electrotechnical Commission, 2010.
13. D. L. Mitchell, N. A. Durgin, P. D. Lincoln, J. C. Mitchell, and A. Scedrov. Undecidability of bounded security protocols. In *In Workshop on Formal Methods and Security Protocols*, 1999.
14. T. Mizrahi. A game theoretic analysis of delay attacks against time synchronization protocols. In *IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication, 2012*, pages 1–6, Sept 2012.

15. B. Moussa, M. Debbabi, and C. Assi. A detection and mitigation model for PTP delay attack in a smart grid substation. In *IEEE International Conference on Smart Grid Communications, SmartGridComm 2015*, pages 497–502, Nov 2015.
16. C. Osorio and M. Bierlaire. An analytic finite capacity queueing network model capturing the propagation of congestion and blocking . *European Journal of Operational Research*, 196(3):996 – 1007, 2009.
17. R. Patel, B. Borisaniya, V. Patel, D. Patel, M. Rajarajan, and A. Zisman. *Comparative Analysis of Formal Model Checking Tools for Security Protocol Verification*, pages 152–163. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
18. P. Srikantha and D. Kundur. Denial of service attacks and mitigation for stability in cyber-enabled power grid. In *2015 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5, Feb 2015.
19. TC 57 Power systems management and associated information exchange. Communication networks and systems for power utility automation - Part 5: Communication requirements for functions and device models. IEC standard 61850-5. Technical report, International Electrotechnical Commission, 2013.
20. J. Tsang and K. Beznosov. *A Security Analysis of the Precise Time Protocol (Short Paper)*, pages 50–59. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
21. M. Ullmann and M. Vgeler. Delay attacks - Implication on NTP and PTP time synchronization. In *International Symposium on Precision Clock Synchronization for Measurement, Control and Communication, 2009*, pages 1–6, Oct 2009.
22. X. Y. Wan, Z. Li, and Z. F. Fan. A SIP DoS flooding attack defense mechanism based on priority class queue. In *2010 IEEE International Conference on Wireless Communications, Networking and Information Security*, pages 428–431, June 2010.
23. Y. Wang, C. Lin, Q. Li, and Y. Fang. A Queueing Analysis for the Denial of Service (DoS) Attacks in Computer Networks. *Comput. Netw.*, 51(12):3564–3573, August 2007.
24. J. Wei and D. Kundur. A flocking-based model for dos-resilient communication routing in smart grid. In *IEEE Global Communications Conference, GLOBECOM 2012*, pages 3519–3524, Dec 2012.