Cyber-security in Smart Grid Communication and Control

OGNJEN VUKOVIĆ

KTH Electrical Engineering

Doctoral Thesis
Stockholm, Sweden 2014

Cyber-security in Smart Grid Communication and Control

OGNJEN VUKOVIĆ

Doctoral Thesis
Stockholm, Sweden, 2014

TRITA-EE 2014:039
ISSN 1653-5146
ISBN 978-91-7595-250-5

School of Electrical Engineering
KTH, Stockholm, Sweden

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av doktorsexamen tisdag den 7 Oktober 2014 i Hörsal F3, KTH, Stockholm.

Tryck: Universitetsservice US AB

Abstract

Society is increasingly dependent on the reliable operation of power systems. Power systems, at the same time, heavily rely on information technologies to achieve efficient and reliable operation. Recent initiatives to upgrade power systems into smart grids target an even tighter integration with information technologies to enable the integration of renewable energy sources, local and bulk generation and demand response. Thus for the reliable operation of smart grids, it is essential that its information infrastructure is secure and reliable in the face of both failures and attacks. This thesis contributes to improving the security of power systems against attacks on their information infrastructures. The contributions lie in three areas: data integrity, data confidentiality, and data availability of power system applications.

We analyze how characteristics of power system applications can be leveraged for detection and mitigation of data integrity attacks. We consider single and multi-area power system state estimation. For single-area state estimation, we look at the integrity of measurement data delivered over a wide area communication network. We define security metrics that quantify the importance of particular components of the communication network, and that allow us to optimize the deployment of network, transport and application layer security solutions. For multi-area state estimation, we look at the integrity of data exchanged between the control centers of neighboring areas in face of a targeted trojan that compromises an endpoint of the secure communication tunnel. We define multiple attack strategies and show that they can significantly disturb the state estimation. Moreover, we propose schemes that could be used for detection, localization, and mitigation of data integrity attacks.

We investigate how to provide data confidentiality for power system applications when they utilize cloud computing. We focus on contingency analysis and propose an approach to obfuscate information regarding power flows and the presence of a contingency violation while allowing the operator to analyze contingencies with the needed accuracy in the cloud. Our empirical evaluation shows that the errors introduced into power flows due to the proposed obfuscation are small, and that the RMS errors introduced grow linearly with the magnitude of obfuscation.

We study how to improve data availability in face of gray hole attacks combined with traffic analysis. We consider two cases: SCADA substation to control center communication using DNP3, and inter-control center communication. In the first case, we propose a support vector machine-based traffic analysis algorithm that uses only the information on timing and direction of three consecutive messages, and show that a gray hole attack can be effectively performed even if the traffic is sent through an encrypted tunnel. We discuss possible mitigation schemes, and show that a minor modification of message timing could help mitigate the attack. In the second case, we study how anonymity networks can be used to improve availability at the price of increased communication overhead and delay. We show that surprisingly availability is not always improved with more overhead and delay. Moreover, we show that it is better to overestimate than to underestimate the attacker's capabilities when configuring anonymity networks.

The progressive development of man is vitally dependent on invention. It is the most important product of his creative brain. Its ultimate purpose is the complete mastery of mind over the material world, the harnessing of the forces of nature to human needs.

Nikola Tesla

Acknowledgments

I would like to thank my main advisor Assoc. Professor György Dán for his guidance, and for his very helpful and continuous feedback. I am deeply grateful for all our insightful discussions that helped me in enhancing my knowledge, and in identifying and addressing exciting research problems. I am proud to have had him as my advisor. I would also like to thank my second, but originally main advisor, Professor Gunnar Karlsson, for giving me an opportunity to join this lab, and for introducing me into the world of scientific research. I am thankful for his support and for his comments on my work. Furthermore, I am grateful to all colleagues in the LCN for providing a friendly and stimulating work atmosphere.

I am thankful to all my friends in Stockholm, back home, and abroad, who were always there for me, and whose presence was priceless to me. They were always an endless source of energy and inspiration. I want to personally thank: Zargham, for his invaluable friendship, and for always being a source of cheerfulness, motivation and support. Stavros and Sylvia, for being great friends and very supportive. Vladimir, Ljubica, Marin, and Vedran, my Serbian connection at KTH in Stockholm, for going through this journey together. I am happy to thank Elena for her precious support, understanding, patience, for always knowing how to cheer me up, and for always being there for me.

I am greatly thankful to my family: želeo bih da se zahvalim svojoj mnogobrojnoj porodici, čiju sam podršku svakodnevno osećao i koja mi je izuzetno značila. Zahvalan sam na našim okupljanjima u domovini kao i u Stockholmu, koja su mi uvek bila izvor radosti i davala dodatnu snagu. Ceo ovaj period bili su mi neiscrpan izvor energije i pružali osećaj da smo tako blizu, iako u stvarnosti prilično daleko. Posebno bih izdvojio svoju majku Milku i sestru Jovanu i zahvalio im se za njihovu neizmernu podršku, ljubav i razumevanje. Njima dvema posvećujem ovu tezu.

Contents

Chapter 1

Introduction

The electric power system is a network of components that generate, deliver, and consume electrical energy. The power produced by electric generators is delivered to loads through power system transmission and distribution networks. Transmission networks transfer the energy over long distances, and they may contain a large number of substations interconnected by transmission lines. In order to minimize the energy loses, the electrical energy is transmitted at high voltages, typically ranging from 100 kV to 500 kV [62]. When close to consumers, step-down transformers are used to decrease the voltage levels before connecting to the distribution networks that transmit the energy at lower voltage levels, typically under 70 kV [62]. Distribution networks transfer the energy between the transmission network and the consumers, and they typically operate in a radial configuration: feeders emanate from substations and form a tree structure with their roots at the substation and branches spreading over the distribution area [62].

Traditionally, power systems have been unidirectional hierarchical systems, where the generators ensure energy supply through the transmission and distribution networks to the loads often without any real-time information about the service parameters of the loads [25]. Consequently, generators are dimensioned to withstand anticipated peaks in demand by the loads, and as the peaks rarely occur, the system is inherently inefficient [25]. Furthermore, to meet the rapid increase in demand for the electrical energy, the system will operate closer to its capacity limits, which calls for more intelligent monitoring and control. To address these shortcomings, the new concept of smart grid has emerged with the idea to provide the system operators with remote real-time monitoring and control, and to allow smooth integration of renewable sources of energy, such as wind, solar, and biomass, so that the system is more efficient, stable, and resilient to anomalies [4, 39, 25, 46]. However, due to the size of the existing systems, one can easily see that smart grids cannot be an immediate replacement; instead, they will coexist with the existing power systems, adding more functionalities and capabilities with new technologies, but keeping full backward compatibility with the existing legacy systems.

A key factor in keeping the power system stable and efficient is its information infrastructure. The information infrastructure includes a system for remote monitoring and control, called Supervisory Control And Data Acquisition (SCADA) system, a suit of applications used to operate the power system, called Energy Management System (EMS), power system communication infrastructure, and computational and storage resources. The SCADA system acquires telemetry data and provides control of remote equipment, and therefore, it relies on the power system communication infrastructure to deliver messages over wide area networks [53]. The EMS includes applications such as state estimation, used to estimate the state of the system based on imperfect measurements [55], and contingency analysis, used to evaluate how an outage would affect the system, and it requires reliable and on-time computational and storage resources. The information infrastructure is essential for realization of the smart grid [31]; it is required to enable real-time monitoring and control as well as forecast and planing. The requirements of the smart grid put higher demands on communication and computation resources as a significantly larger amount of data will be generated, e.g., due to increased number of sensors and more frequent reporting, and that data need to be communicated, stored, and further analyzed within a short time frame [31]. Thus, it is important to find suitable communication and computation infrastructure to handle the demands. Some advanced technologies and applications, such as cloud computing, might be adopted [15].

As proper functioning of information infrastructure is crucial for power systems, the information infrastructure should be secure and reliable both in the face of failures and in the face of attacks. Security of information infrastructures has three aspects: data integrity, data confidentiality, and data availability [54]. Data integrity protects the data against unauthorized generation and modification, and it can be achieved by message authentication codes. Data confidentiality protects the privacy (readability) of the data against unauthorized users, and it can be achieved by data encryption or obfuscation. Finally, data availability ensures data accessibility without excessive delay.

Traditionally, security and reliability of power systems have been achieved by isolating the information infrastructure, and by protecting the system design and implementation. However, the power system information infrastructure is becoming more and more integrated with other information infrastructures, such as the public Internet and potentially cloud computing. Moreover, some parts of the system design, e.g., the communication protocols and application algorithms, have been standardized, and are therefore known. Due to concerns about the cyber security of their systems, power system operators have started applying commercial security solutions, such as cryptographic protection, in their information infrastructures. However, due to the size of the systems, it may be economically and practically unfeasible to protect the entire system. Furthermore, the integration with other information infrastructures may leave the system open to unforeseen threats. Therefore, it is important to evaluate the security of both the existing power system and the future smart grid.

This thesis addresses a number of problems related to integrity, confidentiality and availability of power system information technologies. The objectives are described as follows.

- Integrity: we investigate how violations of data integrity in the power system communication infrastructure can affect power system applications, in particular power system state estimation.
- Confidentiality: we investigate how to provide data confidentiality for power system applications when they utilize cloud computing, in particular the contingency analysis.
- Availability: we analyze how data availability can be improved using anonymity networks. Furthermore, we analyze susceptibility of encrypted SCADA communications to gray hole attacks, and consider various mitigation schemes.

The structure of this thesis is as follows. In Chapter 2, we discuss power system communication and computation technologies, and elaborate on data integrity, data confidentiality, and data availability provided by the technologies. In Chapter 3, we discuss power system applications and describe in details power system state estimation and contingency analysis. Furthermore, we discuss how a violation of data integrity can affect the state estimation, and how to provide data confidentiality for contingency analysis when the computation is performed in the cloud. Chapter 4 provides a summary of the papers included in this thesis along with the contributions of the author of this thesis to the each paper. Chapter 5 summarizes the main findings and conclusions, and outlines potential directions for future research.

Chapter 2

Power System Communication and Computation Technologies

Power systems rely heavily on their communication and computational infrastructures to achieve a secure and reliable operation [56]. The communication infrastructure connects the control center with field devices so that measurements can be acquired and remote control can be performed. This is the basis for Supervisory Control and Data Acquisition (SCADA) systems, used by an operator to monitor and to control the system [6], and a core component of Phasor Networks, where Phasor Data Concentrators aggregate measurements from Phasor Measurement Units (PMUs). Furthermore, the communication infrastructure connects the control centers of interconnected power systems in order to improve operational efficiency and system stability. The connection between control centers enables the secure operation of large and highly inter-connected systems such as Western Interconnect (WECC) in the U.S. and ENTSO-E in Europe.

The computational infrastructure enables Energy Management System (EMS), a suit of applications used to securely and to efficiently operate the power system. Examples of such applications are power system state estimation and contingency analysis. Traditionally, the EMS operates centrally within the control center of a power system operator and utilizes the local computational infrastructure in the control center. However, when large amount of acquired data has to be promptly processed for online operation decision support, e.g., on-line contingency analysis, computing resources provided by the computational infrastructure can become the limiting factor, and could impede the execution of computationally heavy algorithms [33]. Furthermore, as the smart grid is expected to increase both the size and the complexity of power systems and to impose stricter latency requirements on EMS applications, the centralized operation and computation will no longer be scalable [42]. Therefore, the computational infrastructure may need to adopt a distributed architecture and may have to embrace new technologies in order to meet the demands [42]. An example of such technologies is cloud computing, which could provide the ability to occasionally scale computation as needed as well as to make the storage, management and the exchange of data much easier [15].

2.1 SCADA Systems

The SCADA system delivers information from sensors and relays through Remote Terminal Units (RTUs) to SCADA severs, and delivers control messages from SCADA servers through RTUs to relays. Sensors provide measurements of power flows, voltages and currents. Relays control breakers in order to open or to close a line if a fault is detected (protective relays), or to reconfigure a circuit on demand by remote control (control relays). RTUs collect measurements from the sensors, monitor the status of protective relays, and deliver commands to the control relays. RTUs deliver the measurements and the status information to a SCADA server over a Wide-Area Network (WAN), and receive commands for the control relays from the SCADA server over the WAN. The SCADA server is the central processor of the SCADA system located at the control center, and usually provides a human interface for monitoring and control.

SCADA WAN

The types of WANs used for the communication between RTUs and SCADA servers can include point-to-point connections over dedicated or shared lines. In the case of dedicated lines, such as serial links, there is a separate line for every RTU to a SCADA server connection. The advantage of this solution is that it can provide the best quality of service, but the main disadvantage is the cost, since one line per RTU needs to be built or leased. In the case of shared lines, there is a number of RTU to SCADA server connections that utilize the same line. In order to avoid collision between the connections, a telecommunication network based on virtual circuit, packet or cell switching is implemented. Circuit switched networks provide dedicated communication channels (circuits) between RTUs and SCADA servers. Unlike for the case of dedicated lines, communication channels in circuit switched networks are not always active, they are established and used when needed so the network resources can be shared among many pairs of end points. Examples of technologies used are Frequency Division Multiplexing (FDM), where each communication channel gets a non-overlapping frequency range, and Time Division Multiplexing (TDM), where each communication channel gets recurrent fixed-length time slot. In packet switched networks, one communication channel may be shared by many participants, who communicate by exchanging variable-length packets. Examples of such technologies are X.25, Frame relay, GPRS, and Ethernet. Finally, cell switched networks are similar to packet switched networks, but they use fixed, instead of variable, length packets (cells). Prior transporting, data is divided into fixed-length cells. An example of such technology is Asynchronous Transfer Mode (ATM).

In principle, the communication infrastructure used for the WAN can be owned by the operator, e.g., optical ground wires (OPGW) that run between the tops of high-voltage transmission towers, or leased, e.g., Public Switched Telephone Network (PSTN), Public Land Mobile Networks (PLMN), and satellite networks. In practice, the infrastructure is mostly owned by the power system operator for reliability reasons. However, as smart grid technologies, with a growing number of interconnected devices used for monitoring and controlling, impose increasing demands in capacity and in reachability from the communication infrastructure, it may become more economically efficient for the operators to lease commercial networks than to deploy their own.

SCADA/RTU communication protocols

Historically, the SCADA communication protocols were independently designed by different SCADA equipment manufacturers. Each manufacturer developed the protocols to be a part of its proprietary system, and to meet its specific needs [13]. These proprietary protocols had disadvantages for the user, the user could not combine equipment produced by different manufacturers. With the increasing use of SCADA systems, these disadvantages were becoming more prominent, and the need for open standards was recognized [13]. To address the issues, standards organizations were working on defining open protocols that would provide interoperability between systems. One of the arising standards was the IEC 60870-5 standard, created and progressively published from 1990 by the International Electro-technical Commission (IEC) Technical Committee (TC) 57 [53]. IEC 60870-5 is the foundation for today's most commonly used protocols for the communication between RTUs and SCADA servers: IEC 60870-5-104 (including its predecessor IEC 60870-5-101), and Distributed Network Protocol 3 (DNP3). IEC 60870-5-101 and IEC 60870-5-104 are predominantly used in Europe, while DNP3 is predominantly used in the Americas, South Africa, Asia, and Australia [13].

IEC 60870-5

IEC 60870-5 is a part of the IEC 60870 standard, that defines operating conditions, electrical interfaces, performance requirements, and data transmission protocols. IEC 60870-5 defines communication protocols used for sending basic telecontrol messages between two systems. IEC 60870-5 is based on the Enhanced Performance Architecture (EPA) model, which is a simplified version of the International Standards Organization (ISO) Open Systems Interconnection (OSI) model [53]. EPA is designed to provide optimum performance for telecontrol applications, and it defines only three layers: physical layer, link layer, and application layer. The physical layer is defined by IEC 60870-5-1, in particular, coding, formatting, bit error check, and synchronization of data frames of variable and fixed lengths. It includes the specification of four frame formats. IEC 60870-5-2 defines the link layer: link transmission procedures using a control field and address field. IEC 60870-5-3 defines how the application data units are structured in transmission frames. IEC 60870-5-4 provides rules for defining information data elements, such as process variables that are frequently used by the applications. Finally, IEC 60870-5-5 specifies standard services (functions) of the application layer which serve as basic guidelines when creating application profiles for specific tasks. Each application profile uses a specific set of functions. If there is a function needed by the application but not specified in the standards, it should be specified within the profile.

IEC 60870-5-101 (IEC 101)

IEC 101, published in 1995, was the first IEC complete working SCADA protocol under IEC 60870-5 [53]. It was designed to provide all necessary application level functions for telecontrol applications that operate over large geographical areas, using low bandwidth point-to-point links.

Transmission Modes

IEC 101 supports unbalanced and balanced transmission modes. In the unbalanced mode, only the server can initiate a message exchange. The server polls a remote station, and the station responds with data. In the balanced mode, both the server and the remote stations can initiate data exchange. The remote station can initiate the exchange if, e.g., a measured value has significantly changed since the last reported value.

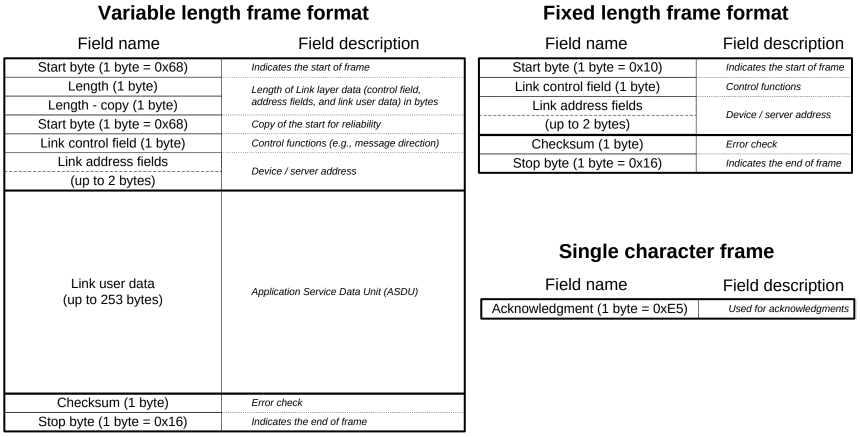
Addressing

IEC 101 uses the FT1.2 frame format defined in IEC 60870-5-1 [13]. The FT1.2 frame format has three forms: variable-length frame format for bidirectional data transmission, fixed-length frame format for commands or acknowledgments, and a single character frame only for acknowledgments. The structures of the three forms of the FT1.2 frame are given in Figure 2.1 (based on [13]). IEC 101 provides addressing on the data link layer through the link address field in the FT1.2 frame format [13]. The link address field can be from 0 to 2 bytes for the balanced transmission mode, or from 1 to 2 for the unbalanced transmission mode. Since the balanced transmission mode may go through a point-to-point link, the link address is redundant. In that case the link address can be omitted.

Reliability

The detection of frame losses or duplication is achieved through a Frame count bit that alternates between 0 and 1 for sequential frames, and it is a part of the Link control field. The frame count bit is used only for the direction from the server to remote stations.

IEC 101 provides detection of bit transmission errors through a checksum provided by FT1.2 [13]. FT1.2 uses an 8-bit checksum calculated as the modulo 256 sum of the link layer data [13], which is the data that starts after the second start field and ends before the checksum field (Figure 2.1). By recalculating the checksum on the receiver side, bit errors due to transmission can be detected but not corrected. If the checksum indicates a transmission error, the data are discarded and a retransmission is requested. However, it may happen that many bit errors occur so



Variable length frame format

Field name	Field description
Start byte (1 byte = 0x68)	*Indicates the start of frame*
Length (1 byte)	*Length of Link layer data (control field, address fields, and link user data) in bytes*
Length - copy (1 byte)	
Start byte (1 byte = 0x68)	*Copy of the start for reliability*
Link control field (1 byte)	*Control functions (e.g., message direction)*
Link address fields (up to 2 bytes)	*Device / server address*
Link user data (up to 253 bytes)	*Application Service Data Unit (ASDU)*
Checksum (1 byte)	*Error check*
Stop byte (1 byte = 0x16)	*Indicates the end of frame*

Fixed length frame format

Field name	Field description
Start byte (1 byte = 0x10)	*Indicates the start of frame*
Link control field (1 byte)	*Control functions*
Link address fields (up to 2 bytes)	*Device / server address*
Checksum (1 byte)	*Error check*
Stop byte (1 byte = 0x16)	*Indicates the end of frame*

Single character frame

Field name	Field description
Acknowledgment (1 byte = 0xE5)	*Used for acknowledgments*

Figure 2.1: Three FT1.2 frame forms used by IEC 101. The figure is based on [13].

that the 8-bit checksum calculation results in the same 8 bits as in the case without errors. The strength of a checksum can be evaluated by the maximum number of single bit errors that will be always detected, which is called the Hamming distance. If the number of single bit errors is larger than the Hamming distance of a checksum, the checksum may not detect the errors. The Hamming distance of the checksum used by the IEC 101 frames is equal to 4 [13].

IEC 60870-5-104 (IEC 104)

With the increasing usage of packet switched networks instead of circuit switching networks, IEC 101 needed to be changed to support packet switching. The modification came in the form of the IEC 104 standard, published in 2000 [13, 53]. The application layer of IEC 104 is based on IEC 101, but some data types and functions are no longer used and supported. Consequently, IEC 104 supports the same transmission modes as IEC 101.

Addressing

IEC 104 relies on TCP [35] and IP [34] as transport and network protocols, and it does not impose any limitations on the data link layer and the physical layer protocols. Therefore, IEC 104 does not provide any addressing under the application layer.

Reliability

IEC 104 relies on underlying protocols for detection of bit transmission errors. TCP uses a 16-bit checksum (the bitwise complement of the sum of 16-bit words added using one's complement arithmetic [35]) to verify the TCP header together with the IEC 104 data. Moreover, some other underlying protocols (e.g., Ethernet) may have verification algorithms that consider the IEC 104 data (Ethernet uses a 32-bit cyclic redundancy check).

Distributed Network Protocol 3 (DNP3)

The DNP3 protocol was developed in the early 1990s by Harrison Controls Division based on some early versions of the IEC 60870-5 standard [13, 53]. Initially, it was developed as a proprietary protocol for use in the electrical utility industry. However, in 1993, DNP3 was taken over by the DNP Users Group, and it became an open standard that has been used by other industries as well (oil and gas, water supply, etc.). Later on, IEEE adopted DNP3 as standard in [21].

Transmission Modes

DNP3 supports only balanced transmission mode (both server and client can initiate the exchange). The server sends polling messages and the client replies immediately with all data. The client can initiate the exchange in case of some sudden changes, e.g., some measured values get significantly changed since the last report. Between the data link layer and the application layer, DPN3 defines the pseudo-transport layer to allow transmission of larger blocks of application data by fragmenting [13].

Addressing

The DNP3 frame format is based on the FT3 frame format defined in IEC 60870-5-1 [13]. FT3 frame format has variable length, and its structure is shown in Figure 2.2 (based on [13]). DNP3 provides addressing on the data link layer through the destination and source address fields in the frame header. The address fields are two bytes each.

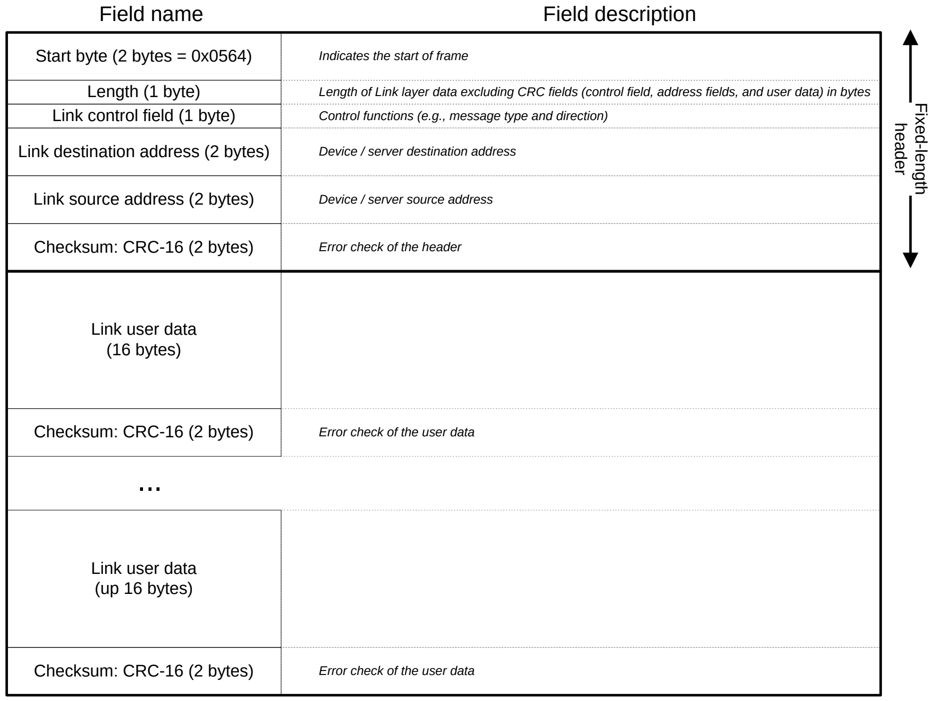
Reliability

DNP3 controls the communication flow, and is able to detect lost frames and duplicates through a sequence number located in the control header of link user data. The sequence number can have a value from 0 to 15 for requests, outstation responses, and from 16 to 31 for unsolicited responses and confirmations. Confirmations have the same sequence number as the request or the response.

DNP3 can detect bit transmission errors using 16-bit cyclic redundancy check (CRC-16) checksum [13]. There is one CRC-16 checksum for the frame header, and thereafter one for every block (max 16 bytes) of user data [13] (Figure 2.2). By recalculating all CRC-16 checksums on the receiver side, bit errors due to transmission can be detected. In the case of DNP3 frames and the CRC-16 checksum, the Hamming distance is equal to 6 [13], which is higher than in the case of IEC 101. However, DNP3 has also a higher transmission overhead in terms of the checksum bits: the ratio of checksum bits to the message bits is higher since it includes a CRC-16 checksum per every block of 16 bytes of user data.

Secure extensions of IEC 101, IEC 104, and DNP3

IEC 101, IEC 104, and DNP3 do not provide any of the three security aspects: data confidentiality, data integrity, and data availability. With increasing cyber security



Field name	Field description
Start byte (2 bytes = 0x0564)	*Indicates the start of frame*
Length (1 byte)	*Length of Link layer data excluding CRC fields (control field, address fields, and user data) in bytes*
Link control field (1 byte)	*Control functions (e.g., message type and direction)*
Link destination address (2 bytes)	*Device / server destination address*
Link source address (2 bytes)	*Device / server source address*
Checksum: CRC-16 (2 bytes)	*Error check of the header*
Link user data (16 bytes)	
Checksum: CRC-16 (2 bytes)	*Error check of the user data*
...	
Link user data (up 16 bytes)	
Checksum: CRC-16 (2 bytes)	*Error check of the user data*

Figure 2.2: DNP3 frame format. The figure is based on [13].

concerns in SCADA systems, IEC 101, IEC 104, and DNP3 needed to be upgraded to address the security concerns. The highest priority was put on data integrity and availability, since it may be more harmful for the power system if control actions and measurements are incorrect or undelivered than if they are disclosed [27, 29]. Researchers and the industry have been proposing different solutions to upgrade the protocols. The most distinguished results are the standard IEC 62351-5 [38] by IEC TC 57 and the standard DNP3 Secure Authentication (DNP3 SA) [21] by the DNP Users Group. IEC 62351-5 and DNP3 SA have been developed in parallel, and IEC TC 57 and DNP Users Group worked together closely so that IEC 62351-5 and DNP3 SA are compatible [29]. Both IEC 62351-5 and DNP3 SA focus on data integrity, while data confidentiality is provided only for the key-exchange messages.

IEC 62351-5 [38] defines the security standards for IEC 60870-5, including IEC 101 and IEC 104, and for IEC 60870-5 derivatives, such as DNP3. The security standards can be divided into two categories: one for the protocols that utilize low bandwidth point-to-point links (IEC 101), and the other for the protocols that can rely on the TCP/IP protocol stack (IEC 104 and DNP3). The protocols in the first category, e.g., IEC 101, are supplemented with additional security measures, which involve cryptographic algorithms, to primarily protect the data integrity. The protocols in the second category, e.g., IEC 104 and DNP3, rely on a challenge-response mechanism combined with a Message Authentication Code (MAC) to protect data integrity, and utilize Transport Layer Security (TLS) version 1.0 [19] to provide data confidentiality.

DNP3 SA [59] has been developed in parallel with IEC 62351-5 by the DNP User Group, as a secure extension of DNP. DNP3 SA is compliant with IEC 62351-5, and is a part of the IEEE standard [21]. To protect data integrity, DNP3 SA uses the challenge-response mechanism described in the IEC 62351-5 standard [38], and utilizes TLS version 1.0 [19] to protect data confidentiality.

Challenge-response mechanism used by IEC 62351-5 and DNP3 SA

The challenge-response mechanism is applied at the application layer, assuming that the underlying layers do not provide any security. The main motivation behind this approach is that it permits that some data exchange can be left unprotected, if desired, which reduces bandwidth and processing requirements [59]. The challenge-response mechanism can be described as follows [59]. Upon receiving a message, the recipient (challenger) decides whether the data in the message are of critical importance. If not, the message is processed without any verification. However, if the data are of critical importance, the challenger initiates the verification of data integrity by sending a challenge message to the sender (responder). The challenge message contains information about the MAC algorithm that the responder should use in the reply, and some randomly generated number to be sent back in the reply (used as a protection against replay attacks). The challenge message also specifies if the data from the received message should be contained in the reply: if not, the challenger only verifies the identity of the responder, if yes, the challenger also verifies the data. The responder generates the reply message that includes the responder identification, the randomly generated number sent by the challenger, and, if requested, the data to be verified. Before sending the reply message, the responder performs the specified MAC algorithm on the message using a pre-shared session key, and adds the resulting MAC value to the reply message. Upon receiving the reply, the challenger performs the same MAC algorithm, and if the resulting MAC values match, the verification of the data integrity is successful. Examples of the challenge-response mechanism are shown in Figure 2.3.

The MAC algorithms that can be used for the challenge-response mechanism are specified in IEC 62351-5 and DNP3 SA. The keys for the MAC algorithms are pre-shared by default. However, the need for more sophisticated management of the keys is recognized by IEC and the DNP User Group, and is a subject of future standard releases. Some recent releases, e.g., DNP SA version 5, provide methods to remotely change the keys [59].

TLS, used by IEC 62351-5 and DNP3 SA to protect data integrity through encryption, relies on digital certificates, encryption, and MAC. IEC 62351-5 and DNP3 SA specify the requirements for the digital certificates, such as application of the certificates, and the procedures for their revocation based on Certificate Revo-

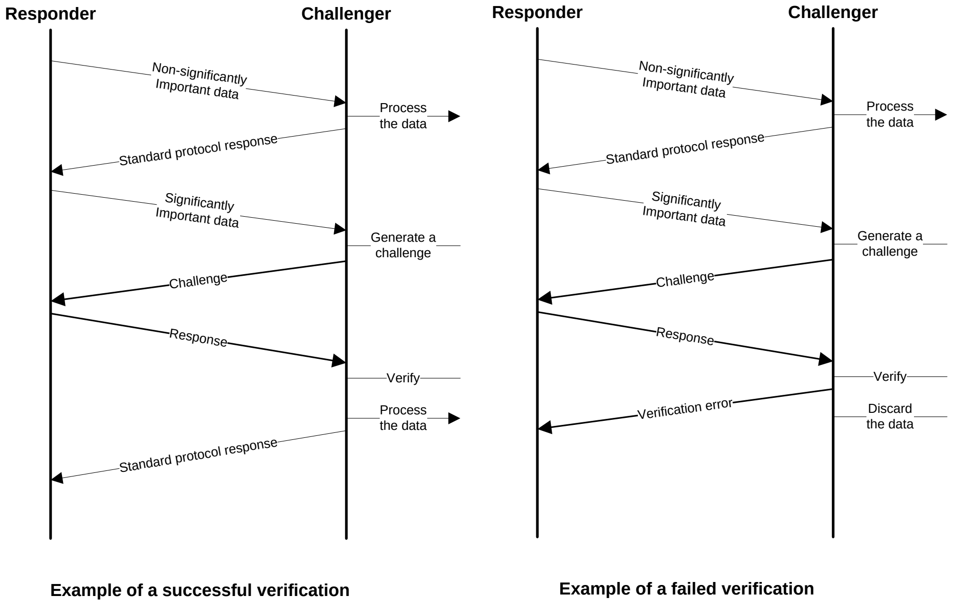

Responder
Challenger
Non-significantly Important data
Process the data
Standard protocol response
Significantly Important data
Generate a challenge
Challenge
Response
Verify
Process the data
Standard protocol response
Example of a successful verification
Responder
Challenger
Non-significantly Important data
Process the data
Standard protocol response
Significantly Important data
Generate a challenge
Challenge
Response
Verify
Verification error
Discard the data
Example of a failed verification

Figure 2.3: Examples of the challenge-response mechanism. The figure is based on [59].

cation Lists (CRL). However, the generation, and provisioning (including the initial distribution) of the certificates remain underspecified [27]. They are acknowledged by IEC as important, and could be a part of future standard extensions [27]. IEC 62351-5 and DNP3 SA manage the keys used by TLS similarly as the keys used by the challenge-response mechanism.

Data Integrity and Availability Issues and Proposed Solutions

The SCADA infrastructure has been traditionally designed to operate in an isolated environment in order to achieve secure and reliable operation. Cyber security has been provided through isolation: it was assumed that no attacker had detailed knowledge of the system design and implementation, including the used proprietary protocols [22]. This security principle is called security through obscurity, and it has been widely criticized as it provides a very fragile security: the system is secure as long as the details remain secret, but quickly breaks once the details are released [54]. Moreover, SCADA infrastructures are becoming more and more integrated with the other corporate infrastructures, and components and protocols have been standardized and are available to practically anyone. This may leave the SCADA systems vulnerable to cyber attacks [22].

A cyber attack on the SCADA communication infrastructure may result in manipulation of the data exchanged between RTUs and the SCADA server. If the protocols IEC 101, IEC 104 over TCP/IP, and DNP3 are utilized without any additional cryptographic protection, the attack could remain undetected if the checksums are recalculated after the modification. The attack could result in intentionally wrong control signals and modified (incorrect) measurements, and it could significantly disturb the power system applications that rely on these signals and measurements [49].

Clearly, the communication needs to be cryptographically protected in order to protect the SCADA system against data integrity attacks on the messages exchanged between RTUs and the SCADA server. Cryptographic protection can be provided by encapsulating (or tunneling) the protocols (IEC 101, IEC 104, and DNP3) into a protocol that provides cryptographic protection [22], e.g., IPsec [41] or TLS [19], or by using the recent protocol extensions that provide message authentication: IEC 62351-5 [38] and DNP3 SAv5 [21]. The most important difference between the two is that, unlike IEC 62351-5 and DNP3 SAv5, tunneling appends a MAC to each message and thereby protects the integrity of every message, but at the cost of increased bandwidth and processing requirements. The cryptographic protection requires an upgrade of all RTUs in the system so they can support the computationally intensive cryptographic operations, and the key management. Some RTUs could be reprogrammed, while other legacy RTUs, which do not have sufficient processing power, would need to be replaced or supplemented by bump-in-the-wire (BITW) devices [60]. BITW is an approach where a network security mechanism is transparently implemented outside the devices whose communication is being protected. In the case of SCADA system, one hardware module (BITW device) is positioned next to a legacy RTU and it tunnels the communication between the RTU and the SCADA server. The communication between BITWs and SCADA servers is protected while the communication between BITWs and RTUs remains vulnerable. Due to the size of power systems, it may be practically and economically unfeasible to perform the upgrades in a short amount of time, and therefore, the upgrade is expected to go in stages. In every stage of the upgrade, it is challenging to evaluate the system security and to optimally select RTUs that will maximally improve the security by upgrading. On the other hand, the complexity of key management increases with the number of upgraded RTUs. Therefore, it is important to keep the number of upgraded RTUs low while achieving a desirable level of system security.

In this thesis, we propose a framework that captures the characteristics of the SCADA communication infrastructure in order to help in evaluating and improving data integrity protection. The framework can be used in every stage of the upgrade to prioritize the RTUs to be cryptographically protected. The framework is described in Paper A, which extends our earlier work [65]. We use the framework to evaluate and to improve the security of power systems considering power system state estimation. Our results show that power system state estimation could be secured by upgrading only a small subset of all RTUs in the system.

Once cryptographic protection is applied to protect data integrity and confidentiality, one might expect that it would also make it impossible for an attacker to identify and to drop mission critical measurement and/or control messages without dropping all messages, and thus remain undetected or difficult to be detected. However, the strict timing rules used in the SCADA communication protocols, such as immediate client responses to master station's polling messages, might facilitate traffic analysis attacks and consequently allow the attacker to perform gray hole attacks.

In this thesis, we address the vulnerability of SCADA communication to a gray hole attack when cryptographic protection is applied. The vulnerability to a gray hole attack is investigated in Paper F, where we show through the example of DNP3 that targeted gray hole attacks may be feasible despite sending messages through an encrypted tunnel. We propose a support vector machine based traffic analysis attack, which is computationally simple and is based on the inter-arrival times and directions of consecutive encrypted messages, and show that an attacker would not need exact knowledge of system parameters for a successful attack. We also discuss potential mitigation schemes, and show that the attack can be mitigated by relaxing the strict timing rules, e.g., by introducing a random delay before answering to DNP3 poll messages.

2.2 Inter-Control Center Communication

Modern power systems have become increasingly inter-connected in order to improve operational efficiency, e.g., the Western Interconnect (WECC) in the U.S. and the ENTSO-E in Europe. The proper operation of an inter-connected system depends on the proper operation of its constituent control regions. Therefore, neighboring control regions need to exchange some information about their systems in real-time, so that they can detect disturbances and quickly restore the system to a secure state in case of outages [66]. The exchange of real-time data between control centers is expected to be even more frequent in future power systems [66].

Historically, power system operators relied on proprietary protocols for inter-control center communication [17]. However, with the increasing interconnectivity between independent operators, the inability of proprietary protocols to provide interoperability has become a problem. To address the problem, the power industry jointly developed the international IEC 60870-6 standard based on the OSI model, and submitted it to the IEC for standardization [66]. IEC 60870-6 is a part of the IEC 60870, and it defines protocols for data exchange between control centers over a WAN. There are two protocol versions used for the data exchange: Tele-control Application Service Element-1 (TASE.1) and TASE.2. One of the differences between the two versions is in the specification of mechanisms for message control and interpretation. TASE.2 uses the Manufacturing Message Specification (MMS) for the specification, and it appears to be the prevalent version used. TASE.2 is usually referred to as the Inter-control Center Communication Protocol (ICCP) [56].

ICCP (IEC 60870-6/TASE.2)

ICCP specifies only the application layer of the OSI model, and it relies on other protocols for the underlying layers. ICCP specifies the use of MMS for the message control and interpretation, and it specifies the data object formats and the methods for data request and reporting. ICCP also specifies how the data can be shared among applications at different control centers.

ICCP is realized through bilateral logical connections, called associations. A control center may establish associations with more than one control center. Moreover, it may establish more than one association with the same control center that could be used to separate data transfers by priority.

ICCP defines data access control through bilateral tables. Bilateral tables specify for every association which data elements can be accessed. However, ICCP does not provide any security of the data during transport.

Secure ICCP

Since ICCP does not protect the data during transport, IEC Technical Committee 57 specified in the standards IEC 62351-3 [36] and IEC 62351-4 [37] how lower layer protocols can protect the data. IEC 62351-3 specifies security measures for end-to-end security for protocols that go over TCP/IP. In particular, it describes the parameters and settings for the TLS protocol [20] that should be configured by the operators. It also considers IPsec [41], but TLS is preferred [36]. IEC 62351-4 specifies security measures for protocols that use MMS, and provides application layer security: prevents unauthorized access to information through authentication [37]. The authentication is achieved through the use of TLS.

Applied together, IEC 62351-3 and IEC 62351-4 protect the data integrity and confidentiality while transported over ICCP, thanks to TLS. However, TLS does not protect against denial-of-service attacks, and such protection should be applied through implementation-specific measures [36, 37].

The end-to-end security provided by IEC 62351-3 and IEC 62351-4 protects ICCP data transfer between two ICCP hosts, one per control center. These hosts, including databases that contain the data shared over ICCP, should be separated from the Master Local Area Network (LAN), also referred to as the control LAN, where all critical applications (e.g., SCADA server and EMS) coexist [51]. ICCP hosts should be in a LAN which is separated by a firewall from the Master LAN on one side, and on the other side separated by another firewall from the WAN used to transfer the ICCP data, as shown in Figure 2.4 (based on [51]). Such separation is a common security practice when some network services should be accessible from outside of the network but connections or hosts cannot be fully trusted. The separated segment of the network that contains the services accessible from outside, is commonly referred to as the demilitarized zone (DMZ). In the case of ICCP, the lack of trust typically comes from the fact that the WAN may be insecure and that the other end may be compromised [51].

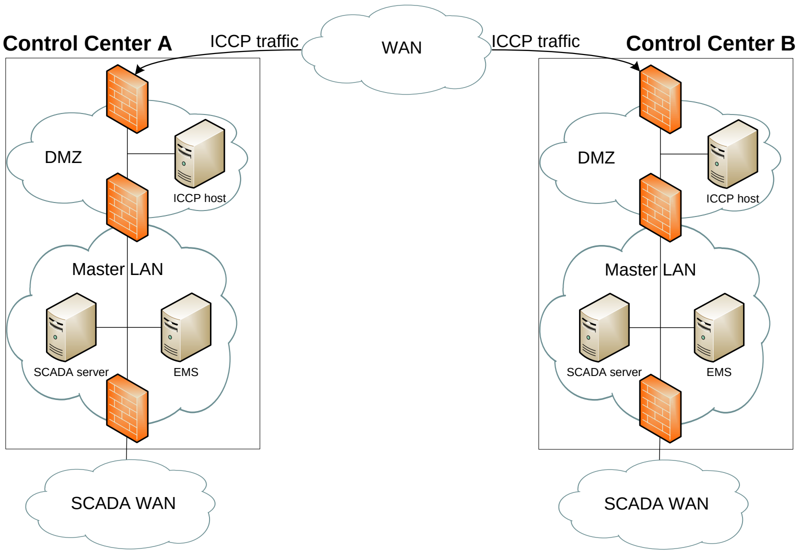


Figure 2.4: Inter-control center communications. The figure is based on [51].

Data Integrity and Availability Issues and Proposed Solutions

By following the standards IEC 62351-3 and IEC 62351-4, the integrity of ICCP data can be protected when transfered between two ICCP hosts in DMZs. However, the ICCP data integrity may not be always protected, and IEC 62351-3 and IEC 62351-4 may not always provide high communication availability, as explained in the following.

First, within an ICCP host, the ICCP data might be unprotected after the TLS protection is removed and before the data are stored in a database (and the other way around), which leaves a potential security threat. Moreover, the threat is aggravated by the fact that the ICCP hosts are in DMZs. They could be victims of sophisticated targeted trojans, whose goal is to manipulate the ICCP data. Examples of recent sophisticated targeted trojans that were targeting industrial control systems are Stuxnet and Duqu [57]. The manipulation of ICCP data could disturb the power system applications that rely on the data exchanged by ICCP.

In this thesis, we address this issue. In Paper B, we study how an attack against the integrity of ICCP data can affect fully distributed multi-area power system state estimation, which requires timely data exchange between control centers of neighboring regions. We define attack strategies for sophisticated manipulation of the exchanged data and show on a well established fully distributed multi-area state estimator, that they can disable the state estimation. We also show a possible way to detect the attacks.

In Paper C, which extends our earlier work [63], we show that the attacks can even disable a state of the art fully distributed state estimator. We propose an attack detection algorithm based on the properties of the state estimator algorithm and based on the exchanged data. Furthermore, we propose an attack localization and mitigation algorithm based on the consensus of the beliefs of the individual regions about the attack location, and show that strong attacks can often be localized and mitigated faster than weak attacks.

Second, TLS protects data integrity and provides confidentiality for the transmitted data, but it does not protect against denial of service attacks [36, 37]. An attacker that obtains access to the WAN may identify some critical low latency data exchange by observing the size, and the sender and the receiver addresses of every message, and it may perform a targeted denial-of-service attack, i.e., a gray hole attack, against such data exchange. Such an attack might be misinterpreted as packet loss due to a congestion, and therefore be undetected. As a consequence, the attack may disturb power system applications that rely on timely delivery of exchanged data.

In this thesis, in Paper E which extends our earlier work [64], we study how anonymity networks could be used to improve the data availability if face of gray hole attacks. Anonymity networks disguise the sender and the receiver of every message through message relaying, which increases the communication overhead and delay. However, the delay may be a concern for some power system applications, such as distributed state estimation. Furthermore, increased traffic overhead may result in additional costs. Therefore, we analyze how much the availability can be improved for a given delay. We quantify the availability by the provided anonymity, i.e., the difficulty of the attacker to correctly identify the origin and the destination of the data. We quantify the delay by the number of times the data are relayed before reaching the destination, and the traffic overhead by the number of times the data are relayed in total. Our results show that, surprisingly, the availability does not always get improved with additional delay or traffic overhead. Moreover, we show that it is better to overestimate than to underestimate the attacker's capabilities when dimensioning anonymity networks.

2.3 Cloud Computing in Power Systems

Cloud computing is a new paradigm for computing technology that provides on-demand network access to shared metered computing resources [32, 5]. It provides a flexible mechanism for offering end users a variety of services, from hardware to application level, so that the users can utilize the computing resources in a completely customizable execution environment [61]. There are three common deployment models of Cloud computing: private cloud, public cloud, and hybrid cloud. Private cloud is a cloud infrastructure exclusively operated and utilized by a single user, and it can provide a high level of data security and privacy but at the price of high initial and unpredictable operating costs. Public cloud is, on the other hand, a cloud infrastructure owned and operated by a third party, such as Amazon AWS, Google, and Microsoft, that provides many users with access to computing resources via Internet. Consequently, users face little to none initial costs and predictable operating costs, but at the price of no guaranteed data security and privacy. Hybrid cloud is a composition of a number of clouds that can include both public and private clouds in order to offer the benefits of both deployment models.

Power systems could greatly benefit from cloud technology, which can provide reliable data storage and meet the computational demands by applications with time-varying computational needs [15]. Power system operators maintain huge databases of past system states in order to enable reconstructions of events in case of system failures, and to improve operational efficiency through data mining. Traditional SCADA systems generate a few thousands data points a few times per minute which results in around 100TB of data per year [15]. With recent implementations of PMUs that can provide data points 30 times per second, the amount of data to be stored increases drastically. Furthermore for reliability reasons, the stored data are replicated at various locations. Cloud-based data storage could be a cost-efficient solution for storing such large quantities of data.

Many EMS applications used in planing and operation have time-varying computational needs [15]. They are either used periodically/occasionally with high computational demand, or they are used continuously but the computational demand depends on the actual system state that changes with time. An example of such applications is Contingency Analysis (CA) used to identify whether one or more contingencies (failures of system components) from a set of considered contingencies would render the system unstable. A set of considered contingencies depends on the instantaneous load of the power system, the higher the load the more contingencies might need to be considered, and is in practice limited by the capacity of the computational infrastructure in the control center. CA involves solving a non-linear weighted least squares (WLS) estimation problem using an iterative algorithm, and is performed every time the system state is recalculated, which can be as often as once a minute. CA that utilizes cloud computing could allow a power system operator to freely scale the number of considered contingencies based on the system state.

Data Security Issues and Proposed Solutions

Perhaps the most significant issue for utilizing cloud computing in power systems is the fact that for a certain amount of time the control over data and data processing leaves the physical and the electronic security perimeter of the power system operator [3]. To overcome the issue, all three aspects of data security (availability, integrity, and confidentiality) must be preserved while the data is out of the security perimeter.

The security aspects must be preserved while data are being communicated to the cloud infrastructure as well as while data are being stored and processed in the cloud infrastructure. Data availability of real-time applications might be altered by the communication network connecting the security perimeter and the cloud infrastructure if the network is unreliable or introduces large delay. Furthermore, the response time of the cloud infrastructure must fit in an acceptable time span so that the functionality of real-time applications is not hampered. While it may be acceptable to leave data in the clear when they are stored within the security perimeter, the data have to be cryptographically protected while being communicated to and stored at the cloud infrastructure so that data integrity and confidentiality are guaranteed. However, if the data need to be processed within the cloud infrastructure, e.g., by a power system application that utilizes cloud computing, cryptography is typically not applicable without affecting the outcome of the processing. In such a case, data integrity and confidentiality might be at risk to get compromised by other users utilizing the same cloud computing infrastructure.

One potential approach to protect data confidentiality is to use homomorphic encryption [11, 10], which is a form of encryption that allows specific types of computations to be carried out on encrypted data and generate an encrypted result which, when decrypted, matches the result of operations performed on the original data. However, finding an encryption algorithm that would support the required computations is far from trivial for many power system applications as they require solving non-linear optimization problems. Another approach could be to obfuscate the data enough that a potential adversary cannot infer any sensitive system information while keeping any introduced computational errors to the minimum [9].

In this thesis, we address the issue of providing data confidentiality for power system applications that utilize cloud computing. In Paper E, we consider cloud-based contingency analysis and propose an approach to obfuscate system information, including the presence of a contingency violation, while allowing the operator to analyze contingencies with the needed accuracy in the cloud.

Chapter 3

Power System Applications

A power system operates in one of three possible operating states: normal, emergency and restorative [47]. Normal operating state means that all the loads, i.e., power demanded by the consumers, can be supplied by the active generators through the transmission and distribution network without violating any operating constraints, such as bounds on the transmission line power flows. Normal operating state can be secure or insecure. The normal operating state is secure if the system can reside in the normal operating state after experiencing a contingency from a list of critical contingencies. Typically considered contingencies are outages of transmission lines and generators. Contrary, the normal operating state is insecure if the system may not preserve the normal operating state after the occurrence of some contingency from the list. In this case, some actions must be taken so that the system is moved to the normal operating secure state, and therefore the emergency operating state is avoided. However, the system may still move to the emergency operating state, e.g., in the event of a non-considered contingency. Emergency operating state means that some of the operating constraints may be violated. In this state, instant actions are required to avoid the system collapse and to return the system to the normal operating state. The actions may result in disconnecting some parts of the system, such as loads and generators. This may stabilize the system, so that all operating constraints are satisfied again. However, the balance between the generated and consumed power may have to be restored. The system is then in the restorative operating state.

The state of the power system can be described by a network model and the voltage phasors at power system buses [1]. The voltage phasors are called state variables, and the set of voltage phasors is called the static state of the system [1]. If the collected measurements are the voltage phasors of all buses, then the static state of the system can be directly obtained. However, typically collected measurements are power injections and power flows. Such measurements need to be processed so that the static state of the system can be determined. Moreover, the measurements are prone to errors, and it may not be economically or technically feasible to provide the measurements of every power flow and power injection in the system. Therefore, the idea of estimating the system state based on the network model and the collected imperfect measurements was proposed in [55]. The ability to estimate the state of the system provides the foundation for the establishment of Energy Management Systems (EMS). EMS is a suit of applications used to operate the power system, and includes applications such as state estimator, used to estimate the state of the system, contingency analysis, used to evaluate how an outage would affect the system, and optimal power flow, used to estimate the optimal power flows based on particular criteria, e.g., minimization of the cost of generation or minimization of transmission line losses.

3.1 Transmission Network Model

Let us consider a transmission network that consists of buses that are interconnected by branches. The term bus is derived from the Latin omnibus, which means "for all", and it is a bar of metal to which all incoming and outgoing conductors, i.e., wires through which the electric current can flow, are connected [62]. Branches include transmission lines, transformers and phase shifters [1].

The admittance matrix $\mathbf{Y}$ of the entire transmission network can be built from scratch by introducing components one at a time (their models) of the system, and updating the corresponding entries in $\mathbf{Y}$ [1]. The components include transmission lines, loads, generators, transformers, shunt capacitors and reactors. The matrix $\mathbf{Y}$ is complex in general, and can be written as $\mathbf{G} + j\mathbf{B}$, where $\mathbf{G}$ is the conductance matrix and $\mathbf{B}$ is susceptance matrix. For more information about the components and their models, and how the matrix $\mathbf{Y}$ is built, we refer to [1].

A transmission network model can be built by deriving a set of nodal equations by using the Kirchhoff's current law at every bus in the transmission network [1, 52]. Let us denote the vector of bus voltage phasors by $\mathbf{V}$, and the vector of bus current injections by $\mathbf{I}$. Then, in a network of n buses, the nodal equations can be expressed with the following matrix equation,

$$\mathbf{I} = \mathbf{Y} \cdot \mathbf{V}; \quad \begin{bmatrix} I_1 \\ I_2 \\ \cdots \\ I_n \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ y_{n1} & y_{n2} & \cdots & y_{nn} \end{bmatrix} \cdot \begin{bmatrix} \mathcal{V}_1 \\ \mathcal{V}_2 \\ \cdots \\ \mathcal{V}_n \end{bmatrix}. \tag{3.1}$$

Power injections at any bus can be derived by multiplying the vector $\mathbf{V}$ with the conjugate of the vector $\mathbf{I}$ from (3.1) [62]. Active and reactive power injections can be further derived by considering the real and the imaginary part of equation $\mathbf{V} \cdot \mathbf{I}^*$. The active power injection P_{b_i} and reactive power injection Q_{b_i} at bus b_i

can expressed as

$$
\begin{aligned}
P_{b_i} &= V_{b_i} \sum_{b_j \in \mathcal{N}(b_i)} V_{b_j}(g_{ij} cos(\theta_{ij}) + b_{ij} sin(\theta_{ij})), \\
Q_{b_i} &= V_{b_i} \sum_{b_j \in \mathcal{N}(b_i)} V_{b_j}(g_{ij} sin(\theta_{ij}) - b_{ij} cos(\theta_{ij})),
\end{aligned}
\tag{3.2}
$$

where V_{b_i} is the voltage amplitude at bus b_i, θ_{ij} is the difference of phase angles between bus b_i and bus b_j, g_{ij} and b_{ij} are the corresponding entries in matrices $\mathbf{G}$ and $\mathbf{B}$, respectively, and $\mathcal{N}(b_i)$ is the set of adjacent buses to bus b_i [1, 62].

Power flows from bus b_i to bus b_j can be derived similarly to (3.2), and expressed as

$$
\begin{aligned}
P_{b_i b_j} &= V_{b_i}^2(g_{si} + g_{ij}) - V_{b_i} V_{b_j}(g_{ij} cos(\theta_{ij}) + b_{ij} sin(\theta_{ij})), \\
Q_{b_i b_j} &= -V_{b_i}^2(b_{si} + b_{ij}) - V_{b_i} V_{b_j}(g_{ij} sin(\theta_{ij}) - b_{ij} cos(\theta_{ij})),
\end{aligned}
\tag{3.3}
$$

where $g_{si} + jb_{si}$ is the admittance of the shunt branch connected at bus b_i [1].

3.2 Measurement Model

Based on (3.2) and (3.3), all current and power injections or flows can be determined once we know the voltage phasors. However, we can use the same model to compute the voltage phasors based on the measurements. The most commonly used measurements are power flows, power injections, bus voltage magnitudes and current flow magnitudes [1]. Unfortunately, we cannot just directly use the measured values in (3.2) and (3.3) to get the voltage phasors. The measurements are prone to errors, and typically not all flows and injections are measured in the system. Therefore, we need to estimate the voltage phasors based on the obtained measurements. In order to perform the estimation, we need a model of measurements, which is described as follows.

Let us consider M measurements that are given by the vector

$$
\mathbf{Z} = \begin{bmatrix} z_1 \\ z_2 \\ \cdots \\ z_M \end{bmatrix} = \begin{bmatrix} f_{z_1}(x) \\ f_{z_2}(x) \\ \cdots \\ f_{z_M}(x) \end{bmatrix} + \begin{bmatrix} e_{z_1} \\ e_{z_2} \\ \cdots \\ e_{z_M} \end{bmatrix} = F(x) + e,
\tag{3.4}
$$

where x is the state vector constructed from the vector $\mathbf{V}$ by considering the phase angles and the voltage amplitudes separately, $f_{z_i}(x)$ is a function relating measurement z_i to the state vector x, and e is the vector of measurement errors. If the measurement z_i is an injection or a flow, then the function $f_{z_i}(x)$ can be expressed based on (3.1), (3.2), or (3.3). However, if the measurement z_i is a voltage amplitude or a phase angle, then the function $f_{z_i}(x)$ equals to the corresponding entry in the vector x. Measurement errors are typically assumed to be independent random noise with Gaussian distribution of zero mean, and consequently the covariance matrix $\mathbf{W} = E(ee^T)$ is diagonal [1, 52, 62].

3.3 State Estimation

State estimation can be centralized (single-area) or distributed (multi-area). Single-area state estimation obtains the estimate of an entire power system, or a single-area power system, performed by a single computing entity. An example of single-area state estimation is the state estimation of a power system controlled by an independent power system operator, where the estimation is performed in the operator's control center. Multi-area state estimation obtains the estimate of a power system that consists of multiple interconnected areas, where the estimation of each area is performed by an independent computing entity. To obtain a consistent state estimate of the entire multi-area power system, the computing entities need to cooperate and exchange some data used as input to the state estimator in every computing entity. An example of multi-area state estimation is the state estimation of an interconnected power system that consists of a multiple areas controlled by independent operators. The state estimation of an area is performed in the control center of the operator that controls the area.

Single-area State Estimation

In the case of single-area state estimation, all the measurements and the entire transmission network model are passed to a computing entity that performs the state estimation.

Maximum Likelihood Estimation
Maximum Likelihood Estimation (MLE), a method widely used in statistics, can be used to determine the most likely state of the system based on the measurements. The measurement errors are assumed to have a known probability distribution, but with unknown parameters. Let us denote by $l(z_i)$ the probability density function which represents the probability of measuring z_i. Assuming that the measurement errors are independent, we can express the joint probability density function of all measurements as the product of individual probability density functions [1]

$$l_M(\mathbf{Z}) = l(z_1)l(z_2)\cdots l(z_M). \tag{3.5}$$

The function $l_M(\mathbf{Z})$ is referred to as the likelihood function, and it represents the probability of measuring the measurements in $\mathbf{Z}$. It will obtain its peak value when the unknown parameters are chosen to be the closest to the actual values [1]. Therefore, by maximizing (3.5) we will reach the maximum likelihood estimates for the parameters of interest. Typically, the measurement error probability distributions are assumed to be Gaussian distributions, as described in Section 3.2. In that case, the parameters of interest are the mean values and the variances. In order to simplify the maximization problem, the likelihood function is replaced by its logarithm, the so called Log-Likelihood function, and it can be expressed as

$$\mathcal{L} = \log(l_M(\mathbf{Z})) = \sum_{i=1}^{M} \log(l(z_i)) = -\frac{1}{2}\sum_{i=1}^{M}(\frac{z_i - E(z_i)}{\sigma_i})^2 - \frac{M}{2}\log(2\pi) - \sum_{i=1}^{M}\log(\sigma_i), \tag{3.6}$$

where the measurement error probability distributions are assumed to be Gaussian distributions with the mean value $E(z_i)$ and standard deviation σ_i for the measurement z_i [1]. The expected value $E(z_i)$ can be expressed as $f_{z_i}(x)$, and σ_i is assumed to be known (it equals to the square root of diagonal entry w_{ii} of the covariance matrix $\mathbf{W}$) [1]. Finally, the state vector x can be found by solving the MLE problem defined as

$$\max_{x} \quad \log(l_M(\mathbf{Z})), \tag{3.7}$$

which is equivalent to

$$J(x) = \min_{x} \quad \sum_{i=1}^{M}(\frac{z_i - E(z_i)}{w_{ii}})^2 = \min_{x} \quad [\mathbf{Z} - F(x)]^T\mathbf{W}^{-1}[\mathbf{Z} - F(x)]. \tag{3.8}$$

Weighted Least Squares Estimator (WLSE)

The optimization problem (3.8) can be solved by using the weighted least squares estimator (WLSE), which can be formulated as follows. At the minimum of (3.8), the first-order optimality conditions have to be satisfied:

$$g(x) = \frac{\partial J(x)}{\partial x} = -H^T(x)\mathbf{W}^{-1}[\mathbf{Z} - F(x)] = 0, \tag{3.9}$$

where $H = [\partial F(x)/\partial x]$ is the Jacobian of $F(x)$ [1]. By expanding the function $g(x)$ into its Taylor series around $x^{(k)}$, where k is the iteration index, and by considering the first two terms of the series we yield an iterative scheme,

$$x^{(k+1)} = x^{(k)} + [H^T(x^{(k)})\mathbf{W}^{-1}H(x^{(k)})]^{(-1)}H^T(x)\mathbf{W}^{-1}[\mathbf{Z} - F(x)], \tag{3.10}$$

known as the Gauss-Newton method [1]. Therefore, at each iteration k, the update vector $\Delta x^{(k)} = x^{(k+1)} - x^{(k)}$ can be calculated by solving the set of equations

$$\Delta x^{(k)} = [H^T(x^{(k)})\mathbf{W}^{-1}H(x^{(k)})]^{(-1)}H^T(x)\mathbf{W}^{-1}[\mathbf{Z} - F(x)], \tag{3.11}$$

also known as the Normal Equations.

WLSE includes the iterative solution to (3.11) and it can be outlined as follows.

1. Set $k = 0$, and assume the starting vector $x^{(0)}$.

2. Calculate the update vector $\Delta x^{(k)}$ using (3.11).

3. If $|\Delta x^{(k)}|_\infty \nleq \epsilon$, update $x^{(k+1)} = x^{(k)} + \Delta x^{(k)}$ and $k = k+1$, and go to Step 2. Else, stop the estimation: the estimator found the solution vector $k^* = x^{(k)}$, after $k^* = k$ iterations (*convergence time*). ϵ is the convergence threshold and $|\cdot|_\infty$ denotes the maximum norm of a vector.

Bad Data Detection (BDD)

Large measurement errors may cause the state estimator to find an incorrect solution (a state vector that is far from the actual one), and therefore, should be detected, identified, and eliminated. Such errors may occur when the meters have bias, drift, and wrong physical connections [1]. Some of the errors are obvious, e.g., negative voltage amplitudes, and can be detected and eliminated a-priori state estimation. Unfortunately, some other errors may not be so easily detectable, and therefore the state estimator needs to be complemented with features that are able to detect and identify any type of bad data. These features depend on the state estimation method, and are referred to as Bad Data Detection (BDD) [1].

After the WLSE obtains a solution, the BDD is done by processing the resulting measurement residuals, i.e., $\Delta\mathbf{Z}^{(k^*)} = \mathbf{Z} - F(x^*)$. The most commonly used BDD algorithm is the Largest Normalized Residual Test (LNRT) [1, 52]. LNRT identifies the largest element in the normalized residual vector $(\Delta\mathbf{Z}^{(k^*)}/||\Delta\mathbf{Z}^{(k^*)}||_2)$, and if that element is larger than a statistical threshold, then the corresponding measurement as assumed as bad data. The threshold can be chosen based on the desired detection sensitivity. After the bad data is identified, the measurement is discarded and the WLSE is performed again.

Data Integrity Issues and Proposed Solutions for Single-area State Estimation

Measurements used as input to the WLSE are provided by the SCADA infrastructure. The integrity of measurements in face of bit errors is typically provided by an error detection code, e.g., cyclic redundancy check or a cryptographic has function, calculated at the RTUs, which is sent along with the data. All communication protocols used for the communication with RTUs implement such error detection, as described in Chapter 2, Section 2.1. However, the integrity of measurements in face of malicious manipulation of the data may not be ensured (Section 2.1), which leaves the measurements vulnerable to cyber attacks [28].

An attacker that gains access to the SCADA infrastructure could manipulate the measurements sent from the RTUs to the control center. The BDD is supposed to detect inconsistent measurements, but it turns out that the measurements could be manipulated in a way that the BDD does not detect the manipulation [8, 16, 50]. Such manipulations are usually referred to as *stealth attacks* on the state estimator.

The manipulation of measurements can be described by an attack vector a added to the actual measurement vector $\mathbf{Z}$, i.e.,

$$\mathbf{Z}_a = \mathbf{Z} + a, \tag{3.12}$$

where $\mathbf{Z}_a$ denotes the measurements after the manipulation. If the attack vector satisfies

$$a = Hc, \quad \text{for some } c \in \mathbb{R}^n, \tag{3.13}$$

then BDD will not detect the manipulation, and the vector a is a stealth attack. Hence, if an attacker wants to change a particular measurement z_i, it might have to change several other measurements to avoid the BDD.

The difficulty of performing stealth attacks against some measurements has been investigated in [50, 8, 58, 45, 16, 43]. However, a common assumption was that the measurements are delivered directly to the control center, ignoring the actual communication network topology. The characteristics of the SCADA communication infrastructure were considered in [16], where the authors assumed that the measurements are first multiplexed in the substations, and then sent directly to the control center. However, often the measurements visit other substations before they get delivered to the control center due to the topology of the SCADA wide area network, described in Section 2.1.

In this thesis, we propose a framework that captures the power system characteristics and the characteristics of the SCADA communication infrastructure in order to estimate the vulnerability of a given system to stealth attacks, and to understand how the stealth attacks can be mitigated using various mitigations schemes. In Paper A which extends [65], we develop quantitative metrics to assess the importance of substations and communication equipment with respect to stealth attacks against the state estimation. We use the metrics to evaluate the potential of various mitigation schemes, such as single-path routing, multi-path routing, and data authentication. We consider data authentication achieved either by encapsulating (or tunneling) the communication through bump-in-the-wire (BITW) devices adjacent to legacy RTUs [60], or by replacing the legacy RTUs with modern RTUs that support message authentication and secure extensions of SCADA/RTU communication protocols (Section 2.1). SCADA system designers and operators can use the framework to evaluate the vulnerability of their systems to stealth attacks, and to evaluate the efficiency of different mitigation schemes to protect their systems against the attacks.

Multi-area State Estimation

In the case of multi-area state estimation, the power system consists of a number of areas and the state estimation of each area is performed by an independent computing entity. Each entity receives only a subset of all measurements and the part of the transmission network model that correspond to its area. Areas can share buses and transmission lines, so the entities need to coordinate to obtain a consistent state estimate.

There have been many proposed algorithms for multi-area state estimation, e.g., [14, 44, 24, 23, 2, 55, 12, 48, 56, 40]. Typically, the algorithms use the normal equations (3.11), or their modifications, to perform updates within the areas before the coordination [14, 44, 23, 2, 55, 12, 48, 56, 40]. The algorithms can be categorized based on a number of criteria [30]. First, they may differ in the way the coordination is done: in a hierarchical manner, e.g., in [14, 44, 24, 23, 2], or in a distributed manner, e.g., in [55, 12, 48, 56, 40]. Second, they may differ in terms of the time when the coordination is done with the respect to the iterations of the areas' local state estimators. The coordination can be done after each iteration, e.g., in [44, 24, 12, 48, 56, 40], or after a number of iterations, e.g., in [14, 23, 2]. Third, they may differ in the assumption on the shared buses and transmission lines between areas. Some assume that areas share only transmission lines [44, 24, 2, 12, 56, 40], while others assume that the areas share only buses [14, 23, 55, 48], or both transmission lines and buses. For a detailed overview of multi-area state estimation algorithms and their categorization, we refer to [30].

Hierarchical Multi-Area State Estimation

In a hierarchical architecture, there exists a central unit that supervises the entities, and subsequently, coordinates the estimates performed by the entities. The entities communicate only with the central unit. The estimation can be considered as a two step process. In the first step, areas perform independent local calculations using their best knowledge of the state estimates of the other areas. In the second step, the central processor coordinates the solutions obtained by areas until a consistent state estimate is found. The steps may be cyclically repeated a number of times before a solution is found.

Fully Distributed Multi-Area State Estimation

In a fully distributed architecture, the areas directly communicate among each other in order to obtain a consistent state estimate. The estimation can be considered as a two step process, similarly to the hierarchical architecture. The only difference is in the second step: the areas coordinate among themselves. They exchange their most recent estimates of the state variables that correspond to the shared buses [56, 40]. The exchanged values are later used when the first step is repeated [56, 40]. The exchange may be synchronous, in which case the steps are synchronized among the areas, or asynchronous [56]. In the asynchronous case, it might be hard to guarantee that a solution will be found [56].

Data Integrity Issues and Proposed Solutions for Multi-Area State Estimation

Measurements used as input to a multi-area state estimator are provided by the SCADA infrastructure, similar to the case of single-area state estimator. An attacker that is able to manipulate the measurements sent from RTUs to the control center could perform attacks similar to the stealth attacks described in Section 3.3 so that the BDD of a multi-area state estimator does not detect the manipulation [26]. Moreover, by denying the delivery of a set of particular measurements, the attacker could make a multi-area state estimator unable to estimate some entries in the state vector x [26].

It is expected that the integrity of the data exchanged between the computing entities is protected. However, in the case of an interconnected power system operated by independent system operators, the integrity of data exchanged between the operators may get violated, as described in Chapter 2, Section 2.2.

In this thesis, in Paper B, we study how a violation of the integrity of data exchanged between independent computing entities can affect fully distributed multi-area state estimation. We consider an attacker that compromises a single computing entity and manipulates with the data sent from and to the entity. We define various attack strategies that differ in the attacker's knowledge of the system, and show on the example of a well-established fully distributed state estimator [56] that they can significantly disturb the state estimation: they can prevent the state estimator to find a solution, or they can lead the state estimator to an erroneous solution. Moreover, our results emphasize the importance of protecting the confidentiality of the measurements: the attacker can perform significantly stronger attacks if it knows the measurements. We also show a possible way to detect the convergence problems, e.g., caused by the attacks, and a simple mitigation scheme where each area performs independent estimation upon detecting the attacks. Note that such independent estimates can result in high estimation errors on any line connecting two different areas, regardless of whether these areas are compromised or not.

In Paper C which extends [63], we show that the attacks can even disable a state of the art state estimator [40]. We propose an attack detection algorithm based on the convergence properties of the state estimator algorithm and based on the evolution of the exchanged state variables. Furthermore, we propose an attack mitigation algorithm based on the consensus of the beliefs of the individual regions about the attack location, formulated as the stationary distribution of a random walk on a graph. We establish existence, uniqueness, and convergence of the stationary distribution. Upon localizing the compromised area, other areas can neglect the data received only from this area and continue performing fully distributed state estimation among non-compromised areas. Our simulation results on an IEEE benchmark power system show that strong attacks can often be localized and mitigated faster than weak attacks.

3.4 Contingency Analysis

Contingency analysis provides the operator of a power system with an indication of the system operating state in case one or more contingency occur, i.e., it determines whether the system is normal secure operating state or normal insecure operating state [7]. Typical considered contingencies are outages such as disconnection of generators or transmission lines. Therefore, the contingency analysis informs the operator of a dangerous contingency that would move the system to the emergency state. Given the information, the operator should take certain actions to avoid a possible system collapse if the contingency occurs, and thus, to move the system to the normal secure operating state. Contingency analysis is performed every time a new state estimate becomes available as a result of state estimation, and it can happen as often as every few minutes.

Contingency analysis uses a model of the transmission network, described in Section 3.1, and a list of considered contingencies to calculate the output that consists of estimated voltage phasors at power system buses and power flows on transmission lines. In the following we outline AC load-flow based contingency analysis, which is widely used.

AC Load-flow based Contingency Analysis

Let us denote by P_I the vector of power injections, by c a contingency, and by f^c the function that describes the power flows under contingency c as a function of the system state, i.e., $P^c = f^c(x)$. If a contingency concerns a disconnection of a transmission line, then the system topology is changed and thus $f^c(.) \neq f(.)$. Similarly, if a contingency concerns the disconnection of a generator, then the vector of power injections $P_I^c \neq P_I$. To capture the relationship between the power injections before and after the contingency we introduce the matrix F^c such that $P_I^c = F_I^c P_I$. If contingency c does not affect the power injections then F_I^c is the identity matrix.

Given the vector of power injections P_I^c under contingency c, contingency analysis requires the solution of the load-flow problem, i.e., finding the state vector x^c that solves $P_I^c = f_I^c(x^c)$. The state vector is obtained through solving the power balance equations,

$$\Delta P_b \overset{d}{=} -P_b + \sum_m P_{bm} = 0. \tag{3.14}$$

Since the sum of the injections over all buses is zero, there are in total $n-1$ power balance equations and $N-1$ unknowns, as the phase angle of the reference bus is set to zero.

The equations (3.3) are non-linear, thus the solution to (3.14) is obtained using an iterative numerical method, typically the Newton-Raphson method. Starting from an initial guess $x^c(0)$, the Newton-Raphson method obtains an updated estimate at iteration k by computing

$$\Delta x^c(k+1) = -J_k^{-1} \Delta P_I(k), \tag{3.15}$$

where $J_k = \frac{\partial P_I}{\partial x}|_{x=x^c(k)}$ is the Jacobian evaluated at the most recent guess $x^c(k)$, and then letting $x^c(k+1) = x^c(k) + \Delta x^c(k+1)$. Observe that the Jacobian is a non-singular square matrix of size $(n-1) \times (n-1)$. The algorithm terminates when the power mismatch ΔP_I is below a certain threshold. Let x^c be the computed system state under contingency c.

Given the system state x^c under the contingency, the power flows can be calculated as $P^c = f^c(x^c)$. If any of the power flows exceeds the capacity limit (e.g., thermal capacity) of the transmission line then the system is said to be in an insecure state, and a corrective action must be taken by the operator to move the system to a state in which no contingency results in a capacity violation.

Data Confidentiality Issues and Proposed Solutions for Cloud-based Contingency Analysis

The number of contingencies that needs to be considered depends on the instantaneous load of the power system, the higher the load the more contingencies might have to be considered. The number of contingencies considered in practice is limited by the computational power available in the control center, and is often constrained to considering the loss of a single components known as N-1 security. Cloud-based contingency analysis could allow an operator to scale the number of considered contingencies freely as a function of the instantaneous system state and enable N-x security that is considered desirable, but it could expose the current system state and possible critical contingencies, thereby facilitating targeted attacks.

In this thesis we address this issue; in Paper D, we propose an algorithm to obfuscate information regarding power flows and the presence of a contingency violation while allowing the operator to analyze contingencies with the needed accuracy in the cloud. Our empirical evaluation shows that the error introduced by the approach when using an AC model is quite small and that the RMS error grows linearly with the magnitude of obfuscation applied.

Chapter 4

Summary of original work

Paper A: Network-aware Mitigation of Data Integrity Attacks on Power System State Estimation

Ognjen Vuković, Kin Cheong Sou, György Dán, Henrik Sandberg.
In IEEE Journal on Selected Areas in Communications (JSAC), vol. 30, no. 6, July 2012.

Summary: In this paper we investigate the vulnerability of single-area power system state estimation to attacks performed against the communication infrastructure used to collect measurement data from the substations. We propose a framework that captures the power system characteristics and the SCADA communication infrastructure, and define security metrics that quantify the importance of individual substations and the cost of attacking individual measurements. We also propose approximations of these metrics, that are based on the communication network topology only, and we compare them to the exact metrics. We provide efficient algorithms to calculate the security metrics. We use the metrics to show how various network layer and application layer mitigation strategies, like single and multi-path routing and data authentication, can be used to decrease the vulnerability of the state estimation. We illustrate the efficiency of the algorithms on the IEEE 118 and 300 bus benchmark power systems.

Contribution: The author of this thesis developed the framework in collaboration with the third co-author, defined the metrics, implemented and carried out the simulations, and analyzed the resulting data. The article was written in collaboration with the co-authors.

Paper B: On the Security of Distributed Power System State Estimation under Targeted Attacks

Ognjen Vuković and György Dán.
In Proceedings of ACM Symposium on Applied Computing (SAC), March 2013.

Summary: In this paper we investigate the vulnerability of fully distributed multi-area power system state estimation to attacks against data exchange between independent computing entities, e.g., control centers of an interconnected power system. We consider an attacker that compromises a single control center and manipulates the data exchanged between the control center and its neighbors. We describe five attack strategies, and evaluate their impact on the IEEE 118 benchmark power system. We show that even if the state estimation converges despite the attack, the estimate can have up to 30% of error, and bad data detection cannot locate the attack. We also show that if powerful enough, the attack can impede the convergence of the state estimation, and thus it can blind the system operators. Our results show that it is important to provide confidentiality for the measurement data in order to prevent the most powerful attacks. Finally, we discuss a possible way to detect and to mitigate these attacks.

Contribution: The author of this thesis defined the attack strategies and the detection method in collaboration with the second co-author, implemented and carried out the simulations, and analyzed the resulting data. The article was written in collaboration with the second co-author.

Paper C: Security of Fully Distributed Power System State Estimation: Detection and Mitigation of Data Integrity Attacks

Ognjen Vuković and György Dán.
In IEEE Journal on Selected Areas in Communications (JSAC), vol. 32, no. 7, July 2014.

Summary: In this paper we address the vulnerability of fully distributed state estimation to data integrity attacks. We consider an attacker that compromises the communication infrastructure of a single control center and can manipulate the state variables exchanged between the control center and its neighbors. We show that a denial of service attack can be launched against a state of the art state estimator this way. We propose an attack detection algorithm based on the convergence properties of the distributed state estimation algorithm and based on the evolution of the exchanged state variables. Furthermore, we propose an attack mitigation algorithm based on the consensus of the beliefs of the individual regions about the attack location, formulated as the stationary distribution of a random walk on a graph. We establish existence, uniqueness, and convergence of the stationary distribution. We show the efficiency of the attack detection and mitigation algorithms via simulations on an IEEE benchmark power system, and we use the simulations to illustrate the trade-off between localization speed and localization accuracy. Our numerical results also show that strong attacks can often be localized and mitigated faster than weak attacks.

Contribution: The author of this thesis defined the detection algorithm and the mitigation algorithm as well as provided the corresponding analytical results in collaboration with the second co-author, implemented and carried out the simulations, and analyzed the resulting data. The article was written in collaboration with the second co-author.

Paper D: Confidentiality-preserving Obfuscation for Cloud-based Power System Contingency Analysis

Ognjen Vuković, György Dán, and Rakesh B. Bobba.
In Proceedings of IEEE SmartGridComm, October 2013.

Summary: In this paper we propose an approach to obfuscate information regarding power flows and the presence of a contingency violation to enable Contingency Analysis in the cloud while allowing the operator to obtain accurate post contingency flows. Our approach doesn't introduce any error for CA using a DC model and our numerical results show that the error introduced when using AC models is tolerable, and that the RMS errors introduced grow linearly with the magnitude of obfuscation.

Contribution: The author of this thesis implemented and carried out the simulations, and analyzed the resulting data. The article was written in collaboration with the co-authors.

Paper E: Mitigating Gray Hole Attacks in Industrial Communications using Anonymity Networks: Relationship Anonymity-Communication Overhead Trade-off

Ognjen Vuković, György Dán, and Gunnar Karlsson.
Submitted to IEEE Transactions on Parallel and Distributed Systems.

Summary: In this paper we consider the problem of mitigating gray hole attacks by providing relationship anonymity among a fixed set of nodes. We describe two anonymity networks, MCrowds and Minstrels. MCrowds is an extension of Crowds, and provides unbounded path length, while Minstrels provides bounded path length. We consider two attack methods the Bayesian inference method and the Maximum posteriori method. We show that MCrowds provides better relationship anonymity than Crowds, but in order to provide anonymity to the receiver the sender is more exposed than in Crowds. Moreover, we show that Minstrels provides better relationship anonymity than MCrowds. We use the two anonymity systems to study the trade-off between relationship anonymity and communication overhead, and show that increased overhead does not always lead to improved relationship anonymity. When comparing the two traffic analysis methods, we show that the Maximum posteriori method performs always better. We study the way relationship anonymity scales with the number of nodes, and show that relationship anonymity improves with the number of nodes but at the price of higher overhead. Our results also indicate that in practice anonymity systems should be optimized for a higher number of attackers than expected.

Contribution: The author of this thesis defined the two anonymity networks in collaboration with the second co-author, derived the analytical expressions for the relationship anonymity for these networks, implemented and carried out the simulations, and analyzed the resulting data. The article was written in collaboration with the second co-author.

Paper F: Peekaboo: A Gray Hole Attack on Encrypted SCADA Communication using Traffic Analysis

Nunzio Marco Torrisi, Ognjen Vuković, György Dán, and Stefan Hagdahl.
In Proceedings of IEEE SmartGridComm, November 2014.

Summary: In this paper we address the vulnerability of SCADA communication to a gray hole attack, in which an attacker drops unsolicited reports sent by an outstation to a SCADA master, while letting through solicited reports in order to avoid detection. We show that such a gray hole attack is possible even if messages are sent through an encrypted tunnel, because due to the strict timing rules used in SCADA protocols traffic analysis can effectively be used to classify protocol messages. We propose a support vector machine based traffic analysis algorithm, used trace-based simulations to evaluate the attack, and show that an attacker would not need exact knowledge of system parameters for a successful attack. We quantified the impact of the attack in terms on monitoring accuracy, and showed that the operator's observation can be up to 10% off on average, and up to 20% off in 5% of the time. Finally, we discuss potential mitigation schemes, and show that the attack can be mitigated by introducing a random delay before answering to poll messages.

Contribution: The author of this thesis participated in designing the traffic analysis algorithm, defined the metric to quantify the attack impact, designed the mitigation algorithm in collaboration with the co-authors, implemented and carried out the simulations, and analyzed the resulting data. The article was written in collaboration with the third co-author.

Publications not included in the thesis:

- Ognjen Vuković and György Dán, "Detection and Localization of Targeted Attacks on Fully Distributed Power System State Estimation", *in Proc. of IEEE SmartGridComm, October 2013.*
- György Dán and Ognjen Vuković, "Utility-based PMU Data Rate Allocation under End-to-end Delay Constraints", *IEEE COMSOC MMTC E-Letter, vol.7, no.8, November 2012.*
- Ognjen Vuković, Kin Cheong Sou, György Dán, and Henrik Sandberg, "Network-layer Protection Schemes against Stealth Attacks on State Estimators in Power Systems", *in Proc. of IEEE SmartGridComm, October 2011.*
- Ognjen Vuković, György Dán, and Gunnar Karlsson, "Traffic Analysis Attacks in Anonymity Networks: Relationship Anonymity-Overhead Trade-off", *n Proc. of 7th Swedish National Computer Networking Workshop (SNCNW), Jun 2011.*
- Ognjen Vuković, György Dán, and Gunnar Karlsson, "On the Trade-off between Relationship Anonymity and Communication Overhead in Anonymity Networks", *in Proc. of IEEE International Conference on Communications (ICC), Jun 2011.*
- Ognjen Vuković, György Dán, and Gunnar Karlsson, "Minstrels: Improving Communications Availability via Increased Relationship Anonymity", *Euro-NF Workshop on Traffic Engineering and Dependability in the Network of the Future, April 2010.*

Chapter 5

Conclusions and Future work

This thesis addresses data integrity, confidentiality, and availability issues in power system information technologies. In the following, we summarize the main contributions of this thesis, and we outline some possible directions for future work.

We developed a framework and proposed security metrics that can be used to evaluate the security of a power system against stealthy attacks on measurements. We provided algorithms to calculate the metrics, and proposed approximations of the metrics, that only consider the communication topology, and therefore, are easier to calculate. We provided an algorithm that could be used to improve the security of the system by applying simpler mitigation strategies, e.g., rerouting, or more involved mitigation strategies, such as multi-path routing and cryptographic protection. Our results emphasized the importance of considering both the communication infrastructure and the power system applications, particularly power system state estimation, when analyzing and improving the security of the system.

We addressed the vulnerability of fully distributed state estimation to data integrity attacks. We considered an attacker that compromises the communication infrastructure of a single control center and can manipulate the state variables exchanged between the control center and its neighbors. We showed that a denial of service attack can be launched against a state of the art state estimator this way. We proposed an attack detection algorithm based on the convergence properties of the distributed state estimation algorithm and based on the evolution of the exchanged state variables. Furthermore, we proposed an attack mitigation algorithm based on the consensus of the beliefs of the individual regions about the attack location, formulated as the stationary distribution of a random walk on a graph. We established existence, uniqueness, and convergence of the stationary distribution. We showed the efficiency of the attack detection and mitigation algorithms via simulations on an IEEE benchmark power system, and we used the simulations to illustrate the trade-off between localization speed and localization accuracy. Our numerical results also show that strong attacks can often be localized and mitigated faster than weak attacks.

We proposed an approach to obfuscate information regarding power flows to enable contingency analysis in the cloud while allowing the operator to obtain accurate post contingency flows. Our approach does not introduce any error for contingency analysis using a DC model and our numerical results show that the error introduced when using AC models is tolerable.

We studied how data availability in power system communication infrastructures could be improved by anonymity networks. Since anonymity networks increase message delay, which could be an issue for power system applications that require timely message delivery, we studied the trade-off between the provided anonymity and the message delay. We found that, contrary to intuition, the anonymity is not always improved with more delay. Moreover, we show that it is better to overestimate than to underestimate the attacker's capabilities when configuring an anonymity network.

Finally, we addressed the vulnerability of SCADA communication to gray hole attacks, in which an attacker drops unsolicited reports sent by an outstation to a SCADA master, while letting through solicited reports in order to avoid detection. We showed that such a gray hole attack is possible even if messages are sent through an encrypted tunnel and the attacker does not know exact system parameters, because due to the strict timing rules used in SCADA protocols traffic analysis can effectively be used to classify protocol messages. We discussed potential mitigation schemes, and showed that the attack can be mitigated by introducing a random delay before answering to poll messages.

Future Work

There are a number of different possibilities for future work. Some of them are complementary studies to the studies included in this thesis, while other studies could address some aspects of data integrity, confidentiality, and availability in power system information technologies not covered in this thesis. We outline some of the possibilities as follows.

Data integrity

We developed a framework and security metrics that evaluate the security of the power system state estimation against attacks on the data integrity of RTU to SCADA server communication. A complementary study could analyze the robustness of the metrics to changes in the power system transmission network topology, as well as to random errors. Moreover, attacks on the data integrity of RTU to SCADA server communication could be also targeted against control messages used to remotely operate control relays. Similar security metrics, and a framework that includes the same model of communication infrastructure complemented with a model of the physical system could be developed to consider such attacks.

We investigated how attacks on data integrity of ICCP data could affect the fully distributed multi-area state estimation. We proposed a detection scheme that could detect such attacks, and outlined a simple mitigation scheme. However, attacks on data integrity of ICCP data could be targeted against data used by other power system applications. It is an open question if such attacks could also disturb those applications.

Data confidentiality

We proposed a scheme to obfuscate information regarding power flows to enable contingency analysis in the cloud, and showed that our scheme introduces tolerable error for AC models of contingency analysis. A complementary study could analytically bound the introduced error. Moreover, similar schemes could be developed to obfuscate sensitive information for other power system applications that utilize cloud computing.

Data availability

We studied how anonymity networks could be used to improve the data availability against targeted DoS attacks, while keeping message delay low. Studies on how targeted DoS attacks could affect power system applications that require timely data delivery, such as fully distributed multi-area state estimation, could help in finding a good balance between the improved data availability and the increased delay.

Furthermore, a subject of future work could be to address the data availability in communication networks used for the acquisition of PMU measurements. The frequency at which a PMU takes and delivers measurements is adjustable, and it may go up to 120Hz. A communication network that acquires measurements from many PMUs at such frequency could experience congestion and losses. Therefore, it is important to understand how congestion could affect the PMU data delivery, and furthermore, to find schemes that would optimally control message generation rate for every PMU in the network so that the losses are minimized [18].

Bibliography

[1] A. Abur and A.G. Exposito. *Power System State Estimation: Theory and Implementation.* Marcel Dekker, Inc., 2004.

[2] Ali Abur. Distributed state estimation for mega grids. In *Proc. of the 15th PSCC Liege*, pages 22–26, Aug. 2006.

[3] B. Akyol. Cyber security challenges in using cloud computing in the electric utility industry. Research report, Pacific Nortwest National Laboratory, September 2012.

[4] S.M. Amin and B.F. Wollenberg. Toward a smart grid: power delivery for the 21st century. *IEEE Power and Energy Magazine*, 3(5):34–41, September 2005.

[5] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. Above the clouds: A berkeley view of cloud computing. Research report, University of California at Berkeley, Februar 2009.

[6] D. Bailey and E. Wright. *Practical SCADA for Industry.* Newnes, 2003.

[7] N. Balu, T. Bertram, A. Bose, V. Brandwajn, G. Cauley, D. Curtice, A. Fouad L. Fink, M. G. Lauby, B. I. Wollenberg, and J. N. Wrjbel. Online power system security analysis. *Proceedings of the IEEE*, 80(2), February 1992.

[8] R.B. Bobba, K.M. Rogers, Q. Wang, H. Khurana, K. Nahrstedt, and T.J. Overbye. Detecting false data injection attacks on DC state estimation. In *Preprints of the First Workshop on Secure Control Systems, CPSWEEK*, Stockholm, Sweden, April 2010.

[9] A.R. Borde, D.K. Molzahn, P. Ramanathan, and B.C. Lesieutre. Confidentiality-preserving optimal power flow for cloud computing. In *Proceedings of Allerton Conference on Communication, Control, and Computing*, pages 1300–1307, October 2012.

[10] M. Brenner, H. Perl, and M. Smith. How practical is homomorphically encrypted program execution? an implementation and performance evaluation. In *Proceddings of IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 375–382, June 2012.

[11] M. Brenner, J. Wiebelitz, G. von Voigt, and M. Smith. Secret program execution in the cloud applying homomorphic encryption. In *Proceedings of IEEE International Conference on Digital Ecosystems and Technologies Conference (DEST)*, pages 114–119, May 2011.

[12] C.W. Brice and R.K. Cavin. Multiprocessor static state estimation. *IEEE Transactions on Power Apparatus Systems*, pages 302–308, February 1982.

[13] G. Clarke and D. Reynders. *Practical Modern SCADA Protocols: DNP3, 60870.5 and Related Systems*. Newnes, 2004.

[14] K.A. Clements, O.J. Denison, and R.J. Ringlee. A multi-area approach to state estimation in power system networks. In *IEEE PES Summer Meeting*, July 1972.

[15] G. Dán, R. B. Bobba, G. Gross, and R. H. Campbell. Cloud computing for the power grid: From service composition to assured clouds. In *Proc. of USENIX HotCloud'13*, Jun 2013.

[16] G. Dán and H. Sandberg. Stealth attacks and protection schemes for state estimators in power systems. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[17] G. Dán, H. Sandberg, M. Ekstedt, and G. Björkman. Challenges in power system information security. *IEEE Security & Privacy*, 10(4):62–70, July 2012.

[18] G. Dán and O. Vuković. Utility-based pmu data rate allocation under end-to-end delay constraints. *IEEE COMSOC MMTC E-Letter*, 7(8), November 2012.

[19] T. Dierks and C. Allen. The tls protocol version 1.0. RFC 2246, IETF, January 1999. URL `http://www.ietf.org/rfc/rfc2246.txt`.

[20] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol, Version 1.2. RFC 5246, IETF, August 2008. URL `http://www.ietf.org/rfc/rfc5246.txt`.

[21] DNP3 IEEE WG. IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3). *IEEE Std 1815-2012 (Revision of IEEE Std 1815-2010)*, pages 1–821, 2012.

[22] D. Dzung, M. Naedele, T.P. Von Hoff, and M. Crevatin. Security for industrial communication systems. *Proceedings of the IEEE*, 93(6):1152–1177, 2005.

[23] R. Ebrahimian and R. Baldick. State estimation distributed processing. *IEEE Trans. on Power Systems*, 4:1240–1246, Nov. 2000.

[24] A.A. El-Keib, J. Nieplocha, H. Singh, and D.J. Maratukulam. A decomposed state estimation technique suitable for parallel processor implementation. *IEEE Trans. on Power Systems*, 3:1088–1097, Aug. 1992.

[25] H. Farhangi. The path of the smart grid. *IEEE Power and Energy Magazine*, 8(1):18–28, January 2010.

[26] Y. Feng, C. Foglietta, A. Baiocco, S. Panzieri, and S.D. Wolthusen. Malicious false data injection in hierarchical electric power grid state estimation systems. In *Proceedings of the Fourth International Conference on Future Energy Systems*, e-Energy '13, pages 183–192, New York, NY, USA, 2013. ACM. URL `http://doi.acm.org/10.1145/2487166.2487187`.

[27] S. Fries, H.J Hof, and M. Seewald. Enhancing IEC 62351 to improve security for energy automation in smart grid environments. In *Proc. of the fifth International Conference on Internet and Web Applications and Services (ICIW)*, pages 135–142, 2010.

[28] A. Giani, S.S. Sastry, K.H. Johansson, and H. Sandberg. The VIKING project: An initiative on resilient control of power networks. In *Proc. of the 2nd International Symposium on Resilient Control Systems*, 2009.

[29] G. Gilchrist. Secure authentication for dnp3. In *IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, pages 1–3, 2008.

[30] A. Gómez-Expósito, A. de la Villa Jaén, C. Gómez-Quiles, P. Rousseaux, and T. Van Cutsem. A taxonomy of multi-area state estimation methods. *Electric Power Systems Research*, 81:1060–1069, 2011.

[31] V.C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G.P. Hancke. Smart grid technologies: Communication technologies and standards. *IEEE Transactions on Industrial Informatics*, 7(4):529–539, November 2011.

[32] B. Hayes. Cloud computing. *Communications of the ACM*, 51(7):9–11, July 2008.

[33] Q. Huang, M. Zhou, Y. Zhang, and Z. Wu. Exploiting cloud computing for power system analysis. In *Proc. of International Conference on Power System Technology (POWERCON)*, pages 1–6, October 2010.

[34] University of Southern California Information Sciences Institute. Internet Protocol. RFC 791, IETF, September 1981. URL `http://www.ietf.org/rfc/rfc791.txt`.

[35] University of Southern California Information Sciences Institute. Transmission Control Protocol. RFC 793, IETF, September 1981. URL `http://www.ietf.org/rfc/rfc793.txt`.

[36] International Electro-technical Commission (IEC) Technical Committee 57. IEC62351 Power systems management and associated information exchange - Data and communications security - Part 3: Communication network and system security - Profiles including TCP/IP. Technical report, IEC Technical Committee 57, Jun 2007.

[37] International Electro-technical Commission (IEC) Technical Committee 57. IEC62351 Power systems management and associated information exchange - Data and communications security - Part 4: Profiles including MMS. Technical report, IEC Technical Committee 57, Jun 2007.

[38] International Electro-technical Commission (IEC) Technical Committee 57. IEC62351 Power systems management and associated information exchange - Data and communications security - Part 5: Security for IEC 60870-5 and derivatives. Technical report, IEC Technical Committee 57, August 2009.

[39] A. Ipakchi and F. Albuyeh. Grid of the future. *IEEE Power and Energy Magazine*, 7(2):52–62, March 2009.

[40] V. Kekatos and G.B. Giannakis. Distributed robust power system state estimation. *IEEE Transactions on Power Systems*, 28(2):1617–1626, 2013.

[41] S. Kent and R. Atkinson. Security architecture for the internet protocol. RFC 2401, IETF, November 1998. URL `http://www.ietf.org/rfc/rfc2401.txt`.

[42] M. Kezunovic, G. Gurrala, A. Bose, P. Yemula, P. Kansal, and Y. Wang. The next generation energy management system design: Final project report. PSERC Publication 13-40, PSERC, September 2013.

[43] T.T. Kim and H.V. Poor. Strategic protection against data injection attacks on power grids. *IEEE Trans. on Smart Grid*, 2:326–333, Jun. 2011.

[44] H. Kobayashi, S. Narita, and M.S.A.A. Hamman. Model coordination method applied to power system control and estimation problems. In *Proc. of the IFAC/IFIP 4th Int. Conf. on Digital Computer Appl. to Process Control*, 1974.

[45] O. Kosut, L. Jia, R. Thomas, and L. Tong. Malicious data attacks on smart grid state estimation: Attack strategies and countermeasures. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[46] F. Li, W. Qiao, H. Sun, H. Wan, J. Wang, Y. Xia, Z. Xu, and P. Zhang. Smart transmission grid: Vision and framework. *IEEE Transactions on Smart Grid*, 1(2):168–177, September 2010.

[47] T.E. Dy Liacco. Real-time computer control of power systems. *In Proc. of IEEE*, 62(7):884–891, July 1974.

[48] S.Y. Lin and C.H. Lin. An implementable distributed state estimator and distributed bad data processing schemes for electric power systems. *IEEE Transactions on Power Systems*, pages 1277–1284, August 1994.

[49] S. Liu, B. Chen, T. Zourntos, D. Kundur, and K. Butler-Purry. A coordinated multi-switch attack for cascading failures in smart grid. *IEEE Transactions on Smart Grid*, 5(3):1183–1195, May 2014.

[50] Y. Liu, P. Ning, and M. Reiter. False data injection attacks against state estimation in electric power grids. In *Proc. of the 16th ACM conference on Computer and Communications Security (CCS)*, pages 21–32, 2009.

[51] J.T. Michalski, A. Lanzone, J. Trent, and S. Smith. Secure ICCP Integration Considerations and Recommendations. Technical report, Sandia National Laboratories, Jun 2007.

[52] A. Monticelli. Electric power system state estimation. *Proc. of the IEEE*, 88 (2):262–282, 2000.

[53] D. Reynders, S. Mackay, and E. Wright. *Practical Industrial Data Communications*. Newnes, 2005.

[54] B. Schneier. *Secret and Lies: Digital Security in a Networked World*. Wiley Publishing, Inc., January 2004.

[55] F.C. Schweppe, J. Wildes, and D.B. Rom. Power system static-state estimation, Part I, II, III. *IEEE Transactions on Power Apparatus and Systems*, 89: 120–135, January 1970.

[56] M. Shahidehpour and Y. Wang. *Communication and Control in Electric Power Systems*. John Wiley and Sons, Inc., 2003.

[57] Symantec Security Response. W32.duq: The precursor to the next stuxnet, November 2011.

[58] A. Teixeira, S. Amin, H. Sandberg, K.H. Johansson, and S.S. Sastry. Cyber-security analysis of state estimators in electric power systems. In *Proc. of IEEE Conf. on Decision and Control (CDC)*, Dec. 2010.

[59] The DNP User Group. DNP Secure Authentication v5. Technical report, The DNP User Group, November 2011.

[60] P.P. Tsang and S.W. Smith. YASIR: A low-latency, high-integrity security retrofit for legacy scada systems. In *Proc. of IFIP/TC11 International Information Security Conference*, 2008.

[61] C. Vecchiola, S. Pandey, and R. Buyya. High-performance cloud computing: A view of scientific applications. In *Proc. of International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN)*, pages 4–16, December 2009.

[62] A. von Meier. *Electric Power Systems: A Conceptual Introduction.* John Wiley and Sons, Inc., 2006.

[63] O. Vuković and G. Dán. Detection and localization of targeted attacks on fully distributed power system state estimation. In *Proceedings of IEEE SmartGridComm*, pages 390–395, October 2013.

[64] O. Vuković, G. Dán, and G. Karlsson. On the trade-off between relationship anonymity and communication overhead in anonymity networks. In *Proceedings of IEEE International Conference on Communications (ICC)*, June 2011.

[65] O. Vuković, K. C. Sou, G. Dán, and H. Sandberg. Network-layer protection schemes against stealth attacks on state estimators in power systems. In *Proceedings of IEEE SmartGridComm*, pages 184–189, October 2011.

[66] F.F. Wu, K. Moslehi, and A. Bose. Power System Control Centers: Past, Present, and Future. *Proceedings of the IEEE*, 93(11):1890–1908, 2005.

Paper A

Network-aware Mitigation of Data Integrity Attacks on Power System State Estimation

Ognjen Vuković, Kin Cheong Sou, György Dán, Henrik Sandberg.

in IEEE Journal on Selected Areas in Communications (JSAC), vol. 30, no. 6, July 2012.

Network-aware Mitigation of Data Integrity Attacks on Power System State Estimation

Ognjen Vuković, Kin Cheong Sou, György Dán, Henrik Sandberg
ACCESS Linnaeus Center, School of Electrical Engineering
KTH Royal Institute of Technology, Stockholm, Sweden
Email: {vukovic,sou,gyuri,hsan}@ee.kth.se

Abstract

Critical power system applications like contingency analysis and optimal power flow calculation rely on the power system state estimator. Hence the security of the state estimator is essential for the proper operation of the power system. In the future more applications are expected to rely on it, so that its importance will increase. Based on realistic models of the communication infrastructure used to deliver measurement data from the substations to the state estimator, in this paper we investigate the vulnerability of the power system state estimator to attacks performed against the communication infrastructure. We define security metrics that quantify the importance of individual substations and the cost of attacking individual measurements. We propose approximations of these metrics, that are based on the communication network topology only, and we compare them to the exact metrics. We provide efficient algorithms to calculate the security metrics. We use the metrics to show how various network layer and application layer mitigation strategies, like single and multi-path routing and data authentication, can be used to decrease the vulnerability of the state estimator. We illustrate the efficiency of the algorithms on the IEEE 118 and 300 bus benchmark power systems.

1 Introduction

Supervisory control and data acquisition (SCADA) systems are used to monitor and to control large-scale power grids. They collect measurement data taken at the substations, multiplex them in remote terminal units (RTUs) located at the substations, and deliver the multiplexed data through the SCADA network to the SCADA master located at the control center. At the control center the measurement data are fed into the power system state estimator (SE). The SE is an on-line application that relies on redundant measurements and a physical model of the power system to periodically calculate an accurate estimate of the power system's state [1, 2]. It includes a Bad Data Detection (BDD) system to detect faulty measurement data.

The state estimate provided by the SE is the basis for a set of application specific software, usually called energy management systems (EMS). Modern EMS provide information support in the control center for a variety of applications related to power network monitoring and control. One example is the optimal routing of power flows in the network, called optimal power flow (OPF), which is to ensure cost-efficient operation. Another example is contingency analysis, which is an essential application to maintain the power system in a secure and stable state despite potential failures, e.g., by using the $n-1$ security criterion. EMS are also expected to be integral components of future SmartGrid solutions, hence the secure and proper operation of the SE is of critical importance [3, 4].

SCADA systems and communication protocols have traditionally been designed to be efficient and to be resilient to failures in order to achieve cost-efficient system operation. Security has been provided through isolating the SCADA infrastructure from the public and the corporate infrastructures, and by following the principle of security by obscurity. SCADA infrastructures are, however, increasingly integrated with corporate infrastructures and equipment are often left unattended, which together with a large installed base of legacy equipment and protocols implies that SCADA systems are potentially vulnerable to cyber attacks [4, 5].

An attacker that gains access to the SCADA communication infrastructure could potentially inject crafted packets or could manipulate the measurement data sent from the RTUs to the control center. While the BDD is supposed to detect faulty measurement data, it was shown recently [6] that measurement data can be manipulated such that they do not trigger the BDD system in the SE. We term such corruptions *stealth attacks* on the SE. Recent experiments on a SCADA/EMS testbed [7] indeed verify that large stealth attacks can be performed without triggering alarms. By fooling the SE the attacker could manipulate the power markets [8], or could hide that the power system is in an unsecure state and eventually can cause cascading failures. The existence of such attacks and their potential security implications make it important to understand how such attacks can be mitigated using various mitigation schemes at a relatively low cost, e.g., without introducing authentication in all system components.

In this paper we address this important question by proposing a framework that captures the characteristics of the power system and of the SCADA communication infrastructure. Our contributions are twofold. First, we develop quantitative metrics to assess the importance of substations and communication equipment with respect to the SE. Second, we use these metrics to evaluate the potential of various mitigation measures to decrease the SE's vulnerability to stealth attacks. As mitigation measures we consider both network layer solutions, such as single-path and multi-path routing, and application layer solutions such as data authentication. We use IEEE benchmark systems to provide numerical results based on the framework. The framework can be used by SCADA system designers and operators to assess the vulnerability of their systems and to evaluate the efficiency of different mitigation schemes to protect the SCADA state estimator against attacks.

The structure of the paper is as follows. In Section 2, we discuss the related work. In Section 3 we outline power system SE and stealth attacks, and a model of modern SCADA communication infrastructures. In Section 4, we introduce system security metrics and show how they can be efficiently computed even for large power systems. In Section 5, we propose an algorithm to mitigate attacks efficiently via various mitigation measures. In Section 6, we use the proposed metrics to evaluate the potential of the mitigation measures to improve security. In Section 7 we conclude the paper.

2 Related Work

Since power system state estimation is a core component of SCADA/EMS systems, there is a wealth of literature on state estimation and bad data detection algorithms [1, 2].

It has long been known that certain bad data are not detectable [9, 10]. Still, the first to study state estimation from a security perspective was [6], where it was pointed out that measurements can be corrupted so that they do not trigger the BDD system, even though the measurements are erroneous. The observation is built on a linearized model of state estimation, but experiments on a SCADA/EMS testbed verified the possibility of stealth attacks under nonlinear models [7].

Several works aimed to quantify the difficulty of performing stealth attacks against some measurements [6, 11, 12, 13, 14, 15]. A common assumption among most of these works is that the measurement values are delivered individually from the meters to the control center [6, 11, 12, 13, 15]. This assumption, while it simplifies the problem formulation, ignores the fact that measurement data taken by different meters at a substation are multiplexed before being sent to the control center. Multiplexing was treated in [14, 15], where the authors considered that measurements taken at the same substation are delivered to the control center over the same point-to-point communication link. This communication model still ignores the network topology, and captures only a fraction of the SCADA communication infrastructures in use today. We, instead, consider a realistic communication model where measurement data are multiplexed and may be relayed through other substations.

Related to our work are studies that use the betweenness centrality [16] and the vertex connectivity [17] in the context of network reliability and in the context of security, respectively. In [18] the authors use the betweenness centrality to assess the importance of individual nodes in routing messages. In [19] the authors use the vertex connectivity to assess network resilience against attacks that compromise communication nodes and communication links. We provide a joint treatment of the communication network topology and stealth attacks against the state estimator, and use these graph theoretical metrics as a comparison to our security metrics.

In this paper we propose a model of the communication infrastructure used in modern power transmission systems. The model accounts for the fact that measurement data can be delivered from a substation to the control center through point-to-point links but also via other substations. Hence an attacker that gains access to a substation, can potentially access and modify all data that traverses the substation. The combination of the power flow model with the model of the communication infrastructure allows us to provide a realistic treatment of stealth attacks and mitigation schemes for power system SE. To our knowledge this paper is the first to consider such a cyber-physical model of power system SE security.

3 Background and system model

In this section, we review steady-state power system modeling and state-estimation techniques, and give an overview of the communication infrastructure used in SCADA systems.

3.1 SCADA Communication Infrastructure

Electric power transmission systems extend over large geographical areas, typically entire countries. Wide-area networks (WANs) are used to deliver the multiplexed measurement data, often together with voice, video and other data traffic, from the RTUs located at the substations to the control center of the transmission system operator (TSO).

For reliability the WAN communication infrastructure is usually owned by the TSO, but the public switched telephone network (PSTN), cellular, and satellite networks are also used. Historically, the WAN infrastructure consisted of point-to-point communication links between RTUs and the control center (e.g., over the PSTN). However, modern WAN infrastructures are increasingly based on overhead ground wire (also called optical ground wire, OPGW) installations that run between the tops of the high voltage transmission towers or along underground cables. In the latter case, SONET or SDH is used to establish communication links (called virtual circuits) between the substations and the control center, but wide-area Ethernet is expected to become prevalent in the near future. As an effect the data sent from a remote substation to the control center might traverse several substations, where switches, multiplexers or cross connects multiplex the data from different substations onto a single OPGW link.

To detect bit errors, SCADA communication protocols include an error detection code calculated by the RTU, which is sent along with the data. The error detection code can be based on, e.g., cyclic redundancy check (CRC) or a cryptographic hash function, such as SHA-1. These codes do not provide message authentication. The operator can achieve message authentication by installing a secret key at the substation in one of two ways. First, by installing a bump-in-the-wire (BITW) device adjacent to a legacy RTU [20]. Data between the RTU and the BITW device are sent in plain-text, hence a BITW does not protect the data if an attacker can gain physical access to the substation. Nevertheless, it protects the data between the BITW device and the control center. Second, by installing an RTU that supports message authentication. A tamper-proof RTU that supports authentication, though more expensive, ensures data integrity even if the attacker can gain physical access to the substation.

3.2 Power System State Estimation and Stealth Attacks

Measurements are taken and sent at a low frequency in SCADA systems, and therefore steady-state estimators are used for state estimation. For a complete treatment of this topic, see for example [1, 2].

Consider a power system that has $n+1$ buses. We consider models of the active power flows P_{ij} (between bus i and j), active power injections P_i (at bus i), and bus phase angles

δ_i, where $i, j = 1, \ldots, n+1$. (A negative P_i indicates a power load at bus i.) The state-estimation problem we consider consists of estimating n phase angles δ_i given M active power flow and injection measurement values z_m ($m \in \{1, ..., M\}$). One has to fix one (arbitrary) bus phase angle as reference angle, for example $\delta_0 := 0$, and therefore only n angles have to be estimated, i.e., the vector $\delta = (\delta_1, \delta_2, .., \delta_n)^T$. The active power flow measurements are denoted by $z = (z_1, \ldots, z_M)^T$, and are equal to the actual power flow plus independent random measurement noise e, which we assume has a Gaussian distribution of zero mean, $e = \big(e_1, \ldots, e_M\big)^T \in \mathcal{N}(0, R)$ where $R := \mathbf{E}ee^T$ is the diagonal measurement covariance matrix.

When the phase differences $\delta_i - \delta_j$ between the buses in the power system are all small, then a linear approximation, a so called DC power flow model, is accurate, and we can write

$$z = H\delta + e, \tag{1}$$

where $H \in \mathbb{R}^{M \times n}$ is a constant known Jacobian matrix that depends on the power system topology and the measurements, see [1, 2] for details. The state estimation problem can then be solved as

$$\hat{\delta} := (H^T R^{-1} H)^{-1} H^T R^{-1} z. \tag{2}$$

The phase-angle estimates $\hat{\delta}$ are used to estimate the active power flows by [2]

$$\hat{z} = H\hat{\delta} = H(H^T R^{-1} H)^{-1} H^T R^{-1} z. \tag{3}$$

The BDD system uses such estimates to identify faulty sensors and bad data by comparing the estimate $\hat{z}$ with z: if the elements $\hat{z}_m$ and z_m are very different, an alarm is triggered because the received measurement value z_m is not explained well by the model. For a more complete treatment of BDD we refer to [1, 2].

An attacker that wants to change measurement m (its *value* z_m) might have to change several other measurements m' to avoid a BDD alarm to be triggered. Consider that the attacker wants to change the measurements from z into $z_a := z + a$. The *attack vector* a is the corruption added to the real measurement vector z. As was shown in [6], an attack vector must satisfy

$$a = Hc, \quad \text{for some } c \in \mathbb{R}^n, \tag{4}$$

in order for it not to increase the risk of an alarm. The corresponding a is termed a *stealth attack* henceforth.

In the recent study [7] it was verified that, despite the simplifying assumptions, stealth attacks can be made large in real (nonlinear) SE software: in the example considered in [7], a power flow measurement was corrupted by 150 MW (57% of the nominal power flow) without triggering alarms.

3.3 Power System Communication Model

The $n+1$ buses of the power system are spread over a set of substations $\mathcal{S}$, $|\mathcal{S}| = S$. We denote the substation at which measurement m is taken by $S(m) \in \mathcal{S}$, and we denote the substation at which the control center is located by $s_{cc} \in \mathcal{S}$. We model the communication network by an undirected graph $\mathcal{G} = (\mathcal{S}, E)$; an edge between two substations corresponds to a communication link between the two substations (e.g., a point-to-point link from a substation to the control-center, or an OPGW link between two substations connected by a transmission line). The graph $\mathcal{G}$ is connected but is typically sparse. Every substation $s \in \mathcal{S}$ can have multiple established routes to the control center s_{cc} through $\mathcal{G}$. We represent route i of substation s by the set of substations $r_s^i \subseteq \mathcal{S}$ that it traverses, including substation s and the control center s_{cc}. The order in which the substations appear in the route is not relevant to the considered problem. For substation s, we denote the set of established routes by $\mathcal{R}_s = \{r_s^1, \ldots, r_s^{R(s)}\}$. If $R(s) = 1$ then all measurement data from substation s are sent over a single route to the control center. If $R(s) > 1$ then unless the data sent over all routes get corrupted in an appropriate way, the control center can detect the data corruption. This can be achieved in a number of ways, e.g., by repeating the measurement data on all the routes or by appending a checksum calculated using an error detection code or a cryptographic hash function, and splitting the data among all the routes. We denote the collection of all $\mathcal{R}_s$ by $\mathcal{R}$.

We consider two forms of end-to-end authentication: non tamper-proof and tamper-proof. We denote the set of substations with *non tamper-proof* authentication (e.g., substations with a BITW device to *authenticate* the data sent to the control center, or an RTU with a non tamper-proof data authentication module) by $\mathcal{E}^N \subseteq \mathcal{S}$. For a route r_s^i we denote by $\sigma_{\mathcal{E}^N}(r_s^i)$ the set of substations in which the data are *susceptible* to attack despite non tamper-proof authentication. Data authenticated in a non tamper-proof way is only *susceptible* to attack at the substation where it originates from, if physical access is possible. Therefore, for every route $r_s^i \in \mathcal{R}_s$ it holds that $\sigma_{\mathcal{E}^N}(r_s^i) = \{s\}$ if $s \in \mathcal{E}^N$ and $\sigma_{\mathcal{E}^N}(r_s^i) = r_s^i$ otherwise.

Similarly, we denote the set of substations with *tamper-proof* authentication (e.g., substations with a tamper-proof RTU that *authenticates* the data sent to the control center) by $\mathcal{E}^P \subseteq \mathcal{S}$. Data authenticated in a tamper-proof way is not susceptible to attack at any substation on the route, hence $\sigma_{\mathcal{E}^P}(r_s^i) = \emptyset$ for every route r_s^i.

Finally, a substation can be *protected* against attacks, e.g., by guards, video surveillance or using tamper-proof system components. We denote the set of protected substations by $\mathcal{P} \subseteq \mathcal{S}$. Protected substations are not susceptible to attacks, therefore $\sigma_{\mathcal{P}}(r_s^i) = r_s^i \setminus \mathcal{P}$. We assume that the substation where the control center is located is protected, that is, $s_{cc} \in \mathcal{P}$.

Fig. 1 illustrates a simple power system and its communication infrastructure. Some substations have applied mitigation schemes, such as non tamper-proof authentication, tamper-proof authentication, and protection.

4 Attack model and security metrics

We consider an attacker whose goal is to perform a *stealth attack* on some power flow or power injection measurement m. To perform the stealth attack, the attacker has to manipulate measurement data from several measurements to avoid a BDD alarm. To manipulate

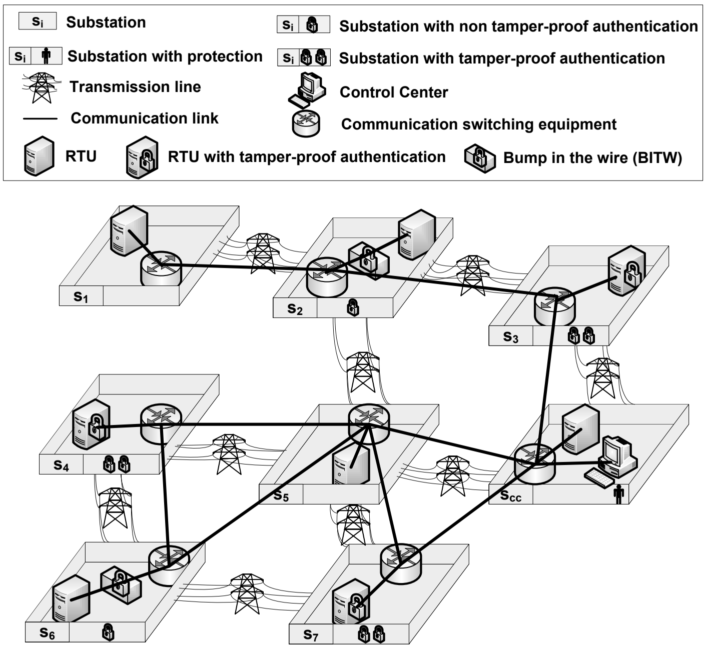

Si Substation
Si Substation with non tamper-proof authentication
Si Substation with protection
Si Substation with tamper-proof authentication
Transmission line
Control Center
Communication link
Communication switching equipment
RTU
RTU with tamper-proof authentication
Bump in the wire (BITW)
S1
S2
S3
S4
S5
Scc
S6
S7

Figure 1: A simple example of a power system and its communication infrastructure. We have $\mathcal{E}^N = \{s_2, s_6\}$, $\mathcal{E}^P = \{s_3, s_4, s_7\}$, and $\mathcal{P} = \{s_{cc}\}$. A measurement taken at substation $s_1 \notin \mathcal{E}^P \cup \mathcal{E}^N$ is susceptible to attacks at substations s_1, s_2, and s_3. A measurement taken at substation $s_6 \in \mathcal{E}^N$ is only susceptible to attacks at substation s_6 ($\sigma_{\mathcal{E}^N}(r^1_{s_6}) = \{s_6\}$). A measurement taken at substation $s_4 \in \mathcal{E}^P$ is not susceptible to attacks ($\sigma_{\mathcal{E}^P}(r^1_{s_4}) = \emptyset$).

measurement data the attacker gets access to the communication equipment located at a subset of the substations. For example, the attacker could get physical access to the equipment in an unmanned substation or could remotely exploit the improper access configuration of the communication equipment. By gaining access to a substation $s \in \mathcal{S}$ (i.e., the switching equipment and the RTU) the attacker can potentially manipulate the measurement data that are *measured in* substation s and the data that are *routed through* substation s, unless multi-path routing, data authentication or protection make that impossible. To perform a *stealth attack* on a particular measurement m (its value z_m) the attacker might need to attack several substations simultaneously, which increases the cost of performing the attack.

In the following we propose two security metrics to characterize the vulnerability of the system with respect to the importance of individual substations and with respect to the vulnerability of individual measurements. Both metrics depend on the mitigation measures implemented by the operator. We also propose an approximation for each metric based on the communication graph topology.

4.1 Substation Attack Impact (I_s)

We quantify the importance of substation s by its *attack impact* I_s, which is the number of measurements on which an attacker can perform a *stealth* attack by getting access to a *single* substation s.

By definition $I_s = 0$ if the substation is protected ($s \in \mathcal{P}$). Otherwise, we define I_s as follows. A measurement m can be attacked if and only if the susceptible parts of all routes from $S(m)$ to the control center pass through substation s. Let us denote by $\mathcal{M}_s \subset \{1,\dots,M\}$ the index set of all such attackable measurements. Then measurement $m \in \mathcal{M}_s$ can be *stealthily* attacked if and only if the following system of equations has a solution with respect to unknowns $a \in \mathbb{R}^M$ and $c \in \mathbb{R}^n$

$$a = Hc, \quad a(m') = 0, \forall\, m' \notin \mathcal{M}_s, \quad \text{and } a(m) = 1. \tag{5}$$

We note that due to the bilinearity of matrix multiplication, the constraint on $a(m)$ in (5) is equivalent to $a(m) \neq 0$. We use $a(m) = 1$ for simplicity. The attack impact I_s is then the cardinality of the set of measurements for which (5) has a solution. That is,

$$I_s = \big|\, \big\{ m \,\big|\, \exists\, a \ \text{ satisfying (5)} \big\} \,\big|. \tag{6}$$

The attack impact of a substation depends on the routing $\mathcal{R}$, the set $\mathcal{E}^N$ of substations with non tamper-proof authentication, the set $\mathcal{E}^P$ of substations with tamper-proof authentication, and the set $\mathcal{P}$ of protected substations.

4.1.1 Calculating I_s

By a linear algebra fact [21], $a = Hc$ for some c if and only if there exists a matrix N_s such that $N_s a = 0$, where $N_s{}^T$ is a basis matrix for the null space of H^T. Let us denote by $N_s(:,\mathcal{M}_s)$ the matrix formed by keeping only the columns of N_s in $\mathcal{M}_s$, $a(\mathcal{M}_s)$ as a vector formed by keeping only the entries of a corresponding to $\mathcal{M}_s$. Then (5) is solvable if and only if

$$N_s(:,\mathcal{M}_s)a(\mathcal{M}_s) = 0, \quad \text{and} \quad v_i{}^T a(\mathcal{M}_s) = 1 \tag{7}$$

can be solved, where v_i denotes the i^{th} column of an identity matrix of dimension $|\mathcal{M}_s|$, and the i^{th} entry of $z(\mathcal{M}_s)$ is $z(m)$. Next, let $\tilde{N}_s$ be a basis matrix for the null space of $N_s(:,\mathcal{M}_s)$. Then (7) is solvable if and only if there exists a vector $\tilde{c}$ s.t.

$$\big(v_i{}^T \tilde{N}_s\big)\tilde{c} = 1. \tag{8}$$

This is possible if and only if the i^{th} row of $\tilde{N}_s$ is not identically zero. The above checking procedure applies to indices other than i. Hence, the calculation of I_s can be summarized as

Proposition 1.

$$I_s = \Big|\, \big\{ i \,\big|\, \tilde{N}_s(i,:) \neq 0 \big\} \,\Big|.$$

The complexity of the calculation is dominated by the singular value decomposition needed to find the basis matrix $N_s{}^T$, and is $O(M^3)$.

4.1.2 Substation Betweenness $\tilde{I}_s$

An intriguing question is whether one can estimate I_s based on the topology of the communication graph $\mathcal{G}$ only, i.e., without considering the power system. The substation betweenness $\tilde{I}_s$, which we describe in the following is inspired by the betweenness centrality of a vertex in a graph [16]. The betweenness centrality of a vertex corresponds to the importance of the vertex in the graph if all nodes communicate with each other; it is often related to the load the vertex is exposed to and to the dependence of the network on the vertex.

To calculate the substation betweenness $\tilde{I}_s$ we assign to every substation s' as weight the number of measurements taken at substation s' (i.e., $|\{m : S(m) = s'\}|$). For a given set of established routes $\mathcal{R}$ the substation betweenness of substation s is then given by the sum of the weights of the substations s' for which it holds that all their established routes to the control center are susceptible to attack at substation s. This is exactly the cardinality of the index set $\mathcal{M}_s$ used to define I_s

$$\tilde{I}_s \quad = \quad |\mathcal{M}_s| \tag{9}$$

The following proposition establishes the relationship between the attack impact and the betweenness of a substation.

Proposition 2. *The substation betweenness is an upper bound for the attack impact, i.e.,* $\tilde{I}_s \geq I_s$.

Proof. The result is trivial if substation $s \in \mathcal{P}$, as $\tilde{I}_s = I_s = 0$. For $s \notin \mathcal{P}$ observe that if a measurement m can be stealthily attacked then by (5) and (6) it must be that $m \in \mathcal{M}_s$. □

Furthermore, if substation s is susceptible to attacks then $\tilde{I}_s$ is no less than the number of measurements taken at substation s, i.e., $\tilde{I}_s \geq |\{m : S(m)\}|$. The complexity of calculating the substation betweenness is that of calculating $\mathcal{M}_s$, which is $O(M)$, and is significantly lower than that of I_s.

4.2 Measurement Attack Cost (Γ_m)

We quantify the vulnerability of measurement m by the minimum number of substations that have to be attacked in order to perform a stealth attack against the measurement, and denote it by Γ_m. If the substation at which the measurement is located is protected and uses non tamper-proof authentication ($S(m) \in \mathcal{P} \cap \mathcal{E}^N$), or it uses tamper-proof authentication ($S(m) \in \mathcal{E}^P$) then the measurement is not vulnerable and we define $\Gamma_m = \infty$.

Otherwise, for a measurement m we define Γ_m as the cardinality of the smallest set of substations $\omega \subseteq \mathcal{S}$ such that there is a stealth attack against m involving some measurements m' at substations $S(m')$ such that every route of the substations $S(m')$ involved in the stealth attack is susceptible to attack at least in one substation in ω. That is,

$$\begin{aligned} \Gamma_m = \min_{\omega \subseteq \mathcal{S}; \omega \bigcap \mathcal{P} = \emptyset} |\omega| \;\; s.t. \;\; \exists\, a, c \;\; s.t. \;\; a = Hc, \;\; a(m) = 1 \text{ and} \\ a(m') \neq 0 \implies \omega \bigcap \sigma_{\mathcal{E}}(r^i_{S(m')}) \neq \emptyset, \quad \forall\, r^i_{S(m')} \in \mathcal{R}_{S(m')}, \end{aligned} \tag{10}$$

where $\sigma_{\mathcal{E}}(r^i_{S(m')})$ denotes the substations in route $r^i_{S(m')}$ that are susceptible to attack despite the authentication applied at substation $S(m')$, i.e., $\sigma_{\mathcal{E}}(r^i_{S(m')}) = \sigma_{\mathcal{E}^P}(r^i_{S(m')}) \bigcap \sigma_{\mathcal{E}^N}(r^i_{S(m')})$. Similar to (5), the constraint on $a(m)$ in (10) is equivalent to $a(m) \neq 0$.

The attack cost of a measurement depends on the routing $\mathcal{R}$, the set $\mathcal{E}^N$ of substations using non tamper-proof authentication, the set $\mathcal{E}^P$ of substations using tamper-proof authentication, and the set $\mathcal{P}$ of protected substations. The following proposition establishes a relationship between the two security metrics; it states that if all measurements have attack cost greater than 1 then all substations have attack impact equal to 0. That is, there is no single substation that would allow attacking a measurement in a stealthy way.

Proposition 3. $I_s = 0\, \forall s \in \mathcal{S} \iff \min_m \Gamma_m > 1$.

Proof. Follows directly from the definitions (6) and (10). If $\nexists s\; I_s > 0$ then a stealth attack against any measurement requires at least two substations to be attacked, $\Gamma_m \geq 2$. If $\exists s$ $I_s > 0$ then attacking substation s is sufficient to attack some measurement m and hence $\Gamma_m = 1$. □

4.2.1 Calculating Γ_m

We can obtain Γ_m by solving a mixed integer linear programming problem (MILP) as follows. Define decision vectors $a \in \mathbb{R}^M$ and $c \in \mathbb{R}^n$. a is the attack vector to be determined. We need a to be a stealth attack targeting measurement m and for the solution to be unique we require the attack magnitude on m to be unit

$$a(m) = 1 \quad \text{and (4) is satisfied.} \tag{11}$$

To describe the connection between the choice of which substations to attack and the set of measurements that can be attacked as a result of the substation attacks, two 0-1 binary decision vectors are needed. One such binary decision vector is $x \in \{0,1\}^{n+1}$, with $x(s) = 1$ if and only if substation s is attacked. Hence, for protected substations (i.e., $s \in \mathcal{P}$)

$$x(s) = 0 \quad \forall\, s \in \mathcal{P}. \tag{12}$$

The other binary decision vector is denoted as $y \in \{0,1\}^M$, with $y(m) = 1$ meaning measurement m might be attacked because of attacks on relevant substations. Conversely, $y(m) = 0$ means measurement m cannot be attacked. To apply y as an indicator for which measurements can be attacked, we impose

$$a \leq Ky \quad \text{and} \quad -a \leq Ky, \tag{13}$$

where the inequality is entry-wise and K is a scalar which is regarded as "infinity". A nontrivial upper bound for K can be obtained from physical insight. Finally, measurement m' can be attacked if and only if the susceptible part of every route between $S(m')$ and s_{cc} goes through at least one of the attacked substations. This is captured by the following constraints

$$y(m') \leq \sum_{s \in \sigma_{\mathcal{E}}(r^i_{S(m')})} x(s), \quad \forall\, r^i_{S(m')} \in \mathcal{R}_{S(m')}, \forall\, m' = 1, \ldots, M \tag{14}$$

Note that by (14) itself it is possible to have $y(m') = 0$ for some m', while the sum on the right-hand-side can be greater than zero. However, this cannot happen at optimality since the objective is to minimize the sum of all entries of x (i.e., the number of substations to be attacked). The following summarizes the calculation.

Proposition 4. *The MILP for finding the attack scheme on measurement m with the minimum number of substation attacks is as follows:*

$$\begin{aligned} \underset{a,c,x,y}{\text{minimize}} \quad & \sum_{s \in \mathcal{S}} x(s) \\ \textit{subject to} \quad & \textit{constraints (11) through (14)} \\ & x(s) \in \{0,1\} \quad \forall\, s \\ & y(m') \in \{0,1\} \quad \forall\, m'. \end{aligned} \tag{15}$$

If (15) is infeasible, then the measurement attack cost is defined to be $\Gamma_m = \infty$. Otherwise, Γ_m is the optimal objective function value in (15).

MILPs are NP-hard in general, but moderate instances of (15) are feasible to solve offline using off-the-shelf MILP solvers.

4.2.2 Measurement Connectivity $\tilde{\Gamma}_m$

The measurement connectivity $\tilde{\Gamma}_m$ is an approximation of the attack cost based on the communication network topology. It is inspired by the minimum vertex cut between two vertices of a graph, i.e., the smallest set of vertices within a graph whose removal disconnects the two vertices.

We define the measurement connectivity of measurement m as the cardinality of the minimum vertex cut for substation $S(m)$ and the control center s_{cc}. Intuitively, if an attacker attacks the substations in the minimum vertex cut for substations $S(m)$ and s_{cc} then it can manipulate the value of measurement m if the measurement data are susceptible to attack at the substations specified in the minimum vertex cut. This is the case if there is no data authentication at $S(m)$ and the substations are not protected. For measurements for which $S(m) = s_{cc}$ or $S(m)$ is adjacent to s_{cc} we define $\tilde{\Gamma}_m = \infty$.

To calculate the measurement connectivity, we can use Menger's theorem [17], which states that the cardinality of the minimum vertex cut for two vertices equals the maximum number of vertex-disjoint paths between the two vertices. The maximum number of vertex-disjoint paths can be efficiently calculated using Ford-Fulkerson-like algorithms. In particular, because capacities are unit, Dinitz's algorithm finds the maximum number of vertex-disjoint paths with complexity $O(min(|\mathcal{S}|^{2/3}, |E|^{1/2})|E|)$ [22].

The measurement connectivity $\tilde{\Gamma}_m$ is not an upper bound for the attack cost Γ_m; it captures the minimum number of substations that have to be attacked in order to tamper measurement m given that substation $S(m)$ is protected ($S(m) \in \mathcal{P}$) and given that the maximum number of node disjoint routes is used.

5 Mitigation measures against attacks

In the following we consider how an operator could improve the security of the system by (i) changing the routes used by the substations (ii) by using multipath routing (iii) and by using data authentication and/or protection.

First, we formulate a result regarding mitigation schemes that make stealth attacks impossible to perform, i.e., mitigation schemes such that $\Gamma_m = \infty$, $\forall m$. For this to hold, the minimum number of *measurements* z_m needed to be protected is the number of buses n [6, 14]. The straightforward way to protect this many measurements is to deploy tamper-proof authentication at all substations. The following result suggests that one can mitigate stealth attacks by deploying authentication in significantly less substations.

Proposition 5. *Consider the power system graph, i.e., the graph with vertex set $\mathcal{S}$, and edges the transmission lines. If $\Gamma_m = \infty$ $\forall m$ then $\mathcal{E}^P \cup \mathcal{P}$ is a dominating set of the power system graph.*

Proof. The dominating set of a graph is a subset of the graph's vertices such that every vertex is either a member of the subset or is adjacent to a vertex in the subset. To prove the proposition, we show that if $\mathcal{E}^P \cup \mathcal{P}$ is not a dominating set of the power system graph then there is at least one measurement m with $\Gamma_m < \infty$.

Since $\mathcal{E}^P \cup \mathcal{P}$ is not a dominating set, there is at least one substation s that is unprotected and not authenticated, and is not adjacent to any substation $s' \in \mathcal{E}^P \cup \mathcal{P}$. Take a measurement m at a bus at substation s. This measurement can be attacked by using an attack vector $a = Hc$ for a vector c whose only non-zero component is that corresponding to a bus at substation s. a has nonzero components corresponding to measurements at adjacent buses, and these measurements are located at substations that do not use either authentication or protection. Hence $\Gamma_m < \infty$. This concludes the proof. □

The cardinality of the dominating set of connected graphs is typically much smaller than the number of vertices, hence perfect protection might be achievable without installing tamper-proof authentication at every substation. The numerical results in Section 6 validate this observation as do the results in [14, 15]. Thus, Proposition 5 can be used to achieve perfect protection with low computational complexity, as follows. First, we find a dominating set of the power system graph. Second, we deploy tamper-proof authentication at the substations in the dominating set. Third, we use the CSF (Critical Substation First) algorithm, described later in this section, to select additional substations at which to deploy tamper-proof authentication, one by one, until perfect protection is achieved.

Next, we turn to the problem of decreasing the vulnerability of the system. A natural goal for the operator would be to improve the most vulnerable part of the system, that is, to minimize $\max_{s \in \mathcal{S}} I_s$ or to maximize $\min_{m \in \mathcal{M}} \Gamma_m$, potentially subject to some constraints on the feasible set of mitigation measures (e.g., due to financial reasons). Maximizing the cost of the least cost stealth attack can lead to increased average attack cost as well, compared to maximizing the average attack cost [14].

Instead of the above formulations, we formulate the operator's goal as a multi-objective optimization problem. As we show later, the solution to this problem formulation is a solution to the max-min formulation. We define the objective γ to be the minimization of the number of measurements with attack cost γ, $|\{m|\Gamma_m = \gamma\}|$. The objectives are ordered: objective γ has priority over objective $\gamma' > \gamma$. Formally, we define the objective vector $w \in \mathbb{N}^{S-1}$ whose γ^{th} component is $w_\gamma = |\{m|\Gamma_m = \gamma\}|$. The goal of the operator can then be expressed as

$$\underset{\mathcal{R},\mathcal{E}^N,\mathcal{E}^P,\mathcal{P}}{\mathrm{l}\mathit{exmin}} \; w(\mathcal{R},\mathcal{E}^N,\mathcal{E}^P,\mathcal{P}), \tag{16}$$

where l*exmin* stands for lexicographical minimization [23], $w(\mathcal{R},\mathcal{E}^N,\mathcal{E}^P,\mathcal{P})$ is the objective vector calculated using Proposition 4 for the established routes $\mathcal{R}$, the sets $\mathcal{E}^N$ and $\mathcal{E}^P$ of authenticated substations, and the set $\mathcal{P}$ of protected substations, and the optimization is performed over all feasible mitigation schemes. The minimal objective vector w, $w_\gamma = 0$ $(1 \leq \gamma \leq S-1)$ corresponds the case when no measurement can be stealthily attacked, i.e., $\Gamma_m = \infty$ for all $m \in \mathcal{M}$.

Proposition 6. *The solution to (16) is a solution to* $\max_{\mathcal{P},\mathcal{E}^N,\mathcal{E}^P,\mathcal{R}} \min_{m\in\mathcal{M}} \Gamma_m$. *Furthermore, if* $\max_{\mathcal{P},\mathcal{E},\mathcal{R}} \min_{m\in\mathcal{M}} \Gamma_m > 1$ *the solution to (16) is a solution to* $\min_{\mathcal{P},\mathcal{E}^N,\mathcal{E}^P,\mathcal{R}} \max_{s\in\mathcal{S}} I_s$.

Proof. We prove the first part of the proposition by contradiction. Let w be the solution to (16), i.e., the lexicographically minimal objective vector, and denote by γ^* the smallest attack cost for which $w_{\gamma^*} > 0$, i.e., $\gamma^* = \min\{\gamma|w_\gamma > 0\}$. Let $\gamma' = max_{\mathcal{P},\mathcal{E}^N,\mathcal{E}^P,\mathcal{R}} \min_{m\in\mathcal{M}} \Gamma_m$ be the max-min solution and w' a corresponding objective vector. Assume now that $\gamma^* < \gamma'$. For $\gamma < \gamma'$ the objective vector has $w'_\gamma = 0$. Since $\gamma^* < \gamma'$, $w'_{\gamma^*} = 0$, and hence according to the definition of lexicographical ordering $w' < w$, which contradicts to the assumption that w is lexicographically minimal.

The second part of the proposition follows directly from Proposition 3 and from the first part of the proposition. □

We solve the lexicographical minimization in (16) in an iterative way [23]. Consider given $\mathcal{R},\mathcal{E}^N,\mathcal{E}^P,\mathcal{P}$ and let $\gamma^* = \min\{\gamma|w_\gamma > 0\}$. If $\gamma^* = \infty$ the system is not vulnerable. Otherwise, we use the *critical substation first* (CSF) algorithm shown in Table 1 to decrease w_γ for some $\gamma \geq \gamma^*$ as long as that is possible.

The algorithm starts by calculating the set $\hat{\mathcal{S}}$ of *critical* substations. In order to find the *critical* substations, the algorithm identifies measurements with attack cost $\Gamma_m = \gamma^*$. Each such measurement has at least one stealth attack ω with attack cost $||\omega|| = \gamma^*$. The substations that are contained in ω for every such stealth attack are *critical* substations. There is at least one such substation, the substation $S(m)$. The critical substations are the candidates for route reconfiguration, authentication or protection.

For every *critical* substation $\hat{s}$ the algorithm considers an alternate mitigation scheme. The alternate mitigation scheme could contain a new set of routes $\mathcal{R}'_{\hat{s}}$ between substation $\hat{s}$ and the control center, or it could be the set of authenticated or protected substations augmented by $\hat{s}$ $(\mathcal{E}^{N\prime}(\hat{s}) = \mathcal{E}^N \cup \hat{s}$, $\mathcal{E}^{P\prime}(\hat{s}) = \mathcal{E}^P \cup \hat{s}$ or $\mathcal{P}'(\hat{s}) = \mathcal{P} \cup \hat{s})$. For every alternate mitigation scheme the algorithm calculates the objective vector $w^{\hat{s}}$ using Proposition 4, and selects the one with the minimal objective vector, $w^{\hat{s}}$. If the alternate mitigation scheme

Table 1: CSF algorithm for given $\mathcal{R}$, $\mathcal{E}^N$, $\mathcal{E}^P$, $\mathcal{P}$ and γ^*

```
1.  Set $\hat{\mathcal{S}} = \emptyset$
2.  for $\forall m$ where $\Gamma_m = \gamma^*$ do
3.      $X = \{x|$ subject to constraints (11) - (14) assuming $\mathcal{E}^N = \mathcal{S}\}$
4.      $\exists X_{\gamma^*} \subseteq X$ s.t. $\forall x \in X_{\gamma^*}, \gamma^* = ||\omega||$
5.      $\hat{\mathcal{S}} = \hat{\mathcal{S}} \cup \{\hat{s}|x(\hat{s}) = 1, \forall x \in X_{\gamma^*}\}$
6.  end for
7.  for $\forall \hat{s} \in \hat{\mathcal{S}}$
8.      create $\mathcal{R}'_{\hat{s}}$ and set $\mathcal{R}'(\hat{s}) = (\mathcal{R} \setminus \mathcal{R}_{\hat{s}}) \cup \mathcal{R}'_{\hat{s}}$ or
9.      set $\mathcal{E}^{N\prime}(\hat{s}) = \mathcal{E}^N \cup \hat{s}$ or $\mathcal{E}^{P\prime}(\hat{s}) = \mathcal{E}^P \cup \hat{s}$ or $\mathcal{P}'(\hat{s}) = \mathcal{P} \cup \hat{s}$
9.      calculate $w^{\hat{s}}(\mathcal{R}'(\hat{s}), \mathcal{E}^{N\prime}(\hat{s}), \mathcal{E}^{P\prime}(\hat{s}), \mathcal{P}'(\hat{s}))$ using Proposition 4
10. end for
11. $\hat{s}^* = \arg\min_{\hat{s}} w^{\hat{s}}$
12. if $w^{\hat{s}^*} < w$
13.     return $\mathcal{R}'(\hat{s}^*)$, $\mathcal{E}^{N\prime}(\hat{s}^*)$, $\mathcal{E}^{P\prime}(\hat{s}^*)$, $\mathcal{P}'(\hat{s}^*)$
14. else if $\gamma^* < S - 1$
15.     Set $\gamma^* = \gamma^* + 1$ and GOTO (1)
16. else
17.     return $\mathcal{R}$, $\mathcal{E}^N$, $\mathcal{E}^P$ and $\mathcal{P}$
18. end if
```

improves the system's level of protection, i.e., $w^{\hat{s}} < w$ then the algorithm terminates. Otherwise the algorithm considers a higher attack cost $\gamma^* = \gamma^* + 1$, and continues from Step 1.

6 Numerical Results

In the following we show numerical results obtained using the algorithms for two IEEE benchmark power systems: the IEEE 118 and 300 bus power systems. Measurements are assumed to be taken at every power injection and power flow.

We considered two communication network topologies. In the first topology every substation communicates directly to the control center, hence the communication network graph is a star graph of order $|S| + 1$: the control center has degree $|S|$ and all substations have degree 1. We refer to this communication network graph as the *star topology*. In the second topology there is an edge between two substations s and s' in the communication network graph if there is a transmission line between any two buses in substations s and s'. The control center is located adjacent to the substation with highest degree s_{cc}. We refer to this communication network graph as the *mesh topology*.

6.1 Baseline Numerical Results

We start with considering a baseline scenario. Authentication is not used at any substation ($\mathcal{E}^N = \emptyset$, $\mathcal{E}^P = \emptyset$). For the mesh topology we consider that all substations use a *single shortest path* ($|\mathcal{R}_s| = 1$) to the control center s_{cc}, and the substation to which the control center is adjacent is protected ($\mathcal{P} = \{s_{cc}\}$). In the following we show the attack impact and the measurement attack cost for the star and for the mesh communication network topologies.

For the star topology, the substation betweenness of substation s is equal to the number of measurements taken at substation s, i.e., $\tilde{I}_s = |\{m : S(m) = s\}$. Then by Proposition 2, this is an upper bound for the attack impact.

For the mesh topology Fig. 2 shows the attack impact I_s and the substation betweenness $\tilde{I}_s$ for the substations for which $I_s > 0$ and $\tilde{I}_s > 0$ for the two power systems. The results show that there are several substations that would enable an attacker to perform a *stealth* attack on a significant fraction of the measurements in the power system, e.g., on about 1000 measurements for the 300 bus system (approx. 90% of all measurements). Almost 50% of the substations have non-zero attack impact, and the attack impact decreases slower than exponentially with the rank of the substation. The substation betweenness $\tilde{I}_s$, is very close to the attack impact for the substations with the highest attack impacts (low ranks), but it overestimates the attack impact significantly for substations with low attack impact.

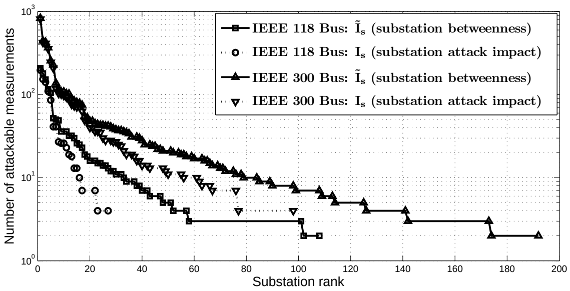


Figure 2: Attack impact I_s of the substations in the IEEE 118 and 300 bus systems in decreasing order of attack impact. The case of shortest path routing.

Table 2 shows the measurement attack costs for the star and the mesh topologies, and the measurement connectivity for the mesh topology. For the star topology and the 118 bus power system there are no measurements with attack cost 1, and most of the measurements (more than 90%) have the attack cost of at least 3. Interestingly, for the 300 bus power system the attack costs are significantly lower. Almost 20% of the measurements have attack cost 1 and only around 45% of the measurements have an attack cost of at least 3. The reason is that in the 300 bus power system topology there are more substations with several buses, and an attacker can tamper with more measurements by accessing such substations.

Table 2: Number of Measurements with Particular Measurement Attack Cost and Measurement connectivity for the IEEE 118 and IEEE 300 systems

System	**Topology**	**1**	**2**	**3**	**4**	**5**	**6**
IEEE118	Star (Γ_m)	0	47	279	71	32	26
	Mesh (Γ_m)	374	78	11	0	0	0
	Mesh ($\tilde{\Gamma}_m$)	53	301	52	18	0	0
IEEE300	Star (Γ_m)	209	251	378	188	41	2
	Mesh (Γ_m)	975	89	3	6	0	0
	Mesh ($\tilde{\Gamma}_m$)	217	403	303	44	0	0

The measurement attack costs for the mesh topology are significantly lower than those for the star topology; e.g., for the 118 bus power system more than 75% of the measurements have attack cost 1 for the mesh topology, while none for the star topology. The significant difference in terms of the attack costs shows the importance of considering the communication network topology when estimating the system security. We also note that the measurement connectivity overestimates the actual attack costs for the mesh topology. This is because the attack costs were calculated for the case of a *single shortest path* for every substation.

Motivated by the large substation attack impacts and low measurement attack costs in the case of shortest path routing, in the following we investigate how the operator can improve the system security by changing single-path routes, using multi-path routing, authentication and protection.

6.2 The Case of Single-path Routing

Modifying single-path routes has the smallest complexity among the mitigation schemes we consider, hence we start with evaluating its potential to decrease the vulnerability of the system. For single-path routing the alternate mitigation schemes differ only in terms of routing. Consequently, $\mathcal{P}'(\hat{s}) = \mathcal{P}$, $\mathcal{E}^{P\prime}(\hat{s}) = \mathcal{E}^P$ and $\mathcal{E}^{N\prime}(\hat{s}) = \mathcal{E}^N$.

In the star topology, substations are directly connected to the control center. Hence, modifying single-path routes is not feasible. For the case of the mesh topology, in order to obtain $\mathcal{R}'(\hat{s})$ from $\mathcal{R}$ for a critical substation $\hat{s}$ we modify the only route $r_1^{\hat{s}}$ in $\mathcal{R}_{\hat{s}}$. For a route $r_1^{\hat{s}}$ we create the shortest alternate route $r_1^{\hat{s}\prime}$ that avoids the substation $s \in r_1^{\hat{s}}$ that appears in most substation attacks ω with cardinality γ^*.

Fig. 3 shows the maximum normalized substation attack impact, i.e., $\max_s I_s/M$, as a function of the number of single-path routes changed in the 118 bus system. The maximum attack impact shows a very fast decay, and decreases by almost a factor of two. At the same time the average path length to the control center increases by only 10%.

Fig. 4 shows the number of measurements that have attack cost 1, 2 and 3 (i.e., w_1, w_2 and w_3) as a function of the number of routes changed in the 118 bus system for the mesh topology. By changing single-path routes the algorithm could increase the attack

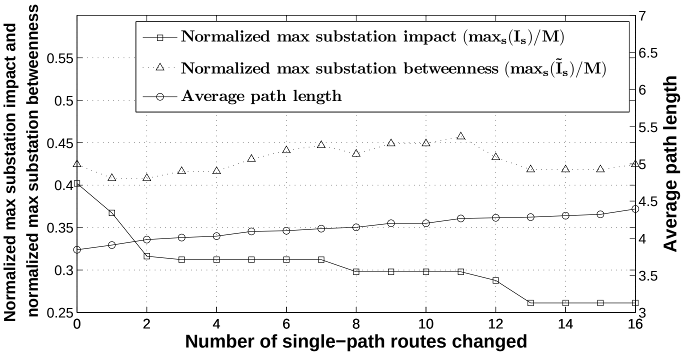


Figure 3: Maximum normalized attack impact, substation betweenness, and average path length vs. the number of single-path routes changed in the IEEE 118 bus system and mesh topology.

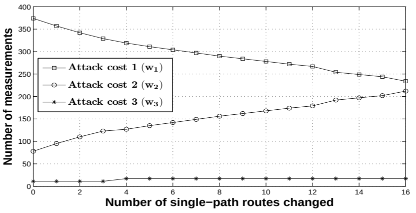


Figure 4: Number of measurements for various attack costs vs. the number of single-path routes changed in the IEEE 118 bus system and mesh topology.

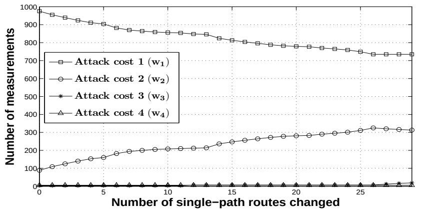


Figure 5: Number of measurements for various attack costs vs. the number of single-path routes changed in the IEEE 300 bus system and mesh topology.

cost for about 200 measurements from $\Gamma_m = 1$ to $\Gamma_m = 2$, and for some measurements to $\Gamma_m = 3$ (e.g., at iteration 5). Fig. 5 shows the corresponding results for the 300 bus system. Note that after 27 iterations w_1 does not decrease, but instead w_2 does. After 16 resp. 29 iterations the algorithm could not find any single-path route that would lead to increased attack cost for any measurement. Hence, we turn to multi-path routing.

6.3 The Case of Multi-path Routing

In the case of multi-path routing the alternate mitigation schemes differ only in terms of routing, as for single-path routing. Consequently, $\mathcal{P}'(\hat{s}) = \mathcal{P}$, $\mathcal{E}^{P\prime}(\hat{s}) = \mathcal{E}^P$ and $\mathcal{E}^{N\prime}(\hat{s}) = \mathcal{E}^N$.

Since in the star topology substations are directly connected to the control center, multi-

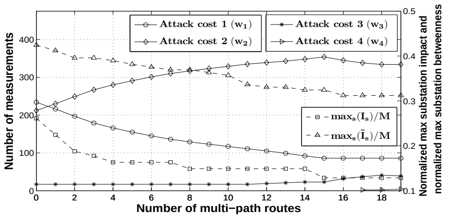


Figure 6: Maximum attack impact, substation betweenness, and number of measurements for various attack costs vs. the number of multi-path routes. IEEE 118 bus system, mesh topology.

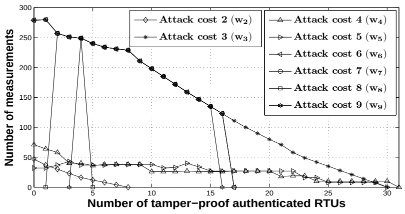


Figure 7: Maximum attack impact and number of measurements for various attack costs vs. the number of tamper-proof authenticated RTUs ($|\mathcal{E}^P|$). IEEE 118 bus system, star topology.

path routing can not decrease the vulnerability of the system. For the mesh topology, to obtain $\mathcal{R}'(\hat{s})$ from $\mathcal{R}$ for a critical substation $\hat{s}$, we consider the single route $r_1^{\hat{s}}$ in $\mathcal{R}_{\hat{s}}$, and construct the shortest route $r_2^{\hat{s}\prime}$ such that $r_2^{\hat{s}\prime}$ and $r_1^{\hat{s}}$ are node-disjoint. The routes in $\mathcal{R}_{\hat{s}}{}'$ are then $r_1^{\hat{s}\prime} = r_1^{\hat{s}}$ and $r_2^{\hat{s}\prime}$.

Multi-path routing introduces complexity in the management of the communication infrastructure. In the case of SDH at the link layer several virtual circuits have to be configured and maintained. In the case of Ethernet some form of traffic engineering is required (e.g., using MPLS). Hence the cost of establishing a multi-path route from a substation to the control center has a higher cost than changing a single-path route, considered in the previous subsection. We therefore take the set of routes $\mathcal{R}$ obtained in the last iteration of the algorithm in the previous subsection as the starting point for deploying multi-path routing.

Fig. 6 shows the maximum normalized substation attack impact and the number of measurements with attack costs 1 to 4 vs. the number of multi-path routes in the 118 bus system and the mesh topology. Multi-path routing could decrease the maximum attack impact by 50% through increasing the number of measurements with attack cost $\Gamma_m = 2$ and $\Gamma_m = 3$. Still, 86 measurements have attack cost 1 when the algorithm terminates. The achieved attack costs are much closer to the measurement connectivity $\tilde{\Gamma}_m$ than in the case of single-path routing. However, the measurement connectivity still overestimates the attack costs. This is because we only consider two node-disjoint paths to the control center. By considering all node-disjoint paths the attack costs would approach and potentially exceed the measurement connectivity.

6.4 The Case of Authentication

In the case of (non) tamper-proof authentication the alternate mitigation schemes differ in terms of the set of (non) tamper-proof authenticated substations $\mathcal{E}^P$ ($\mathcal{E}^N$). Consequently, $\mathcal{P}'(\hat{s}) = \mathcal{P}$ and $\mathcal{R}'(\hat{s}) = \mathcal{R}$.

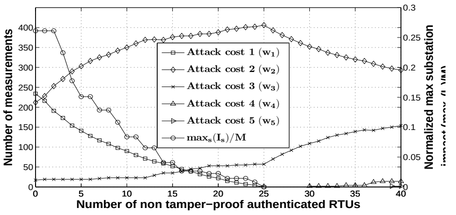


Figure 8: Maximum attack impact and number of measurements for various attack costs vs. the number of non tamper-proof authenticated RTUs ($|\mathcal{E}^N|$). IEEE 118 bus system, mesh topology.

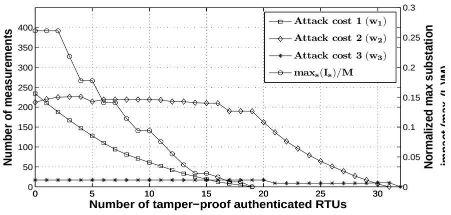


Figure 9: Maximum attack impact and number of measurements for various attack costs vs. the number of non tamper-proof authenticated RTUs ($|\mathcal{E}^N|$). IEEE 118 bus system, mesh topology.

To obtain $\mathcal{E}^{N\prime}(\hat{s})$ from $\mathcal{E}^N$ for a critical substation $\hat{s}$ we add substation $\hat{s}$ to the set of substations using non tamper-proof authentication, i.e., $\mathcal{E}^{N\prime}(\hat{s}) = \mathcal{E}^N \cup \hat{s}$. We follow a similar procedure to augment the set $\mathcal{E}^P$ of substations with tamper-proof authentication.

Apart from the deployment costs (e.g., new equipment), authentication requires that secret keys be protected and managed, which results in costs for the operator. The cost of introducing authentication is certainly higher than that of reconfiguring single-path routing, but it is difficult to compare its cost to that of introducing multi-path routing. We therefore take the set of routes $\mathcal{R}$ obtained in the last iteration of the algorithm for single-path routing as the starting point for deploying authentication.

Fig. 7 shows the number of measurements with attack cost 1 to 9 as a function of the number of tamper-proof authenticated RTUs in the 118 bus system for the star topology. Note that there are no measurements with attack cost 1. With 31 substations using tamper-proof authentication stealth attacks are impossible to perform. The 31 substations form a dominating set of the power system graph, in accordance with Proposition 5. Note that this number is less than one third of the number of substations in the system, which is $S = 109$.

Fig. 8 shows the maximum normalized substation attack impact and the number of measurements with attack cost 1 to 5 as a function of the number of non tamper-proof authenticated RTUs in the 118 bus system for the mesh topology. Authentication eliminates measurements with attack cost $\Gamma_m = 1$ after 25 substations are authenticated. Furthermore, upon termination more measurements have attacks cost $\Gamma_m \geq 3$, than using multi-path routing.

Fig. 9 shows the maximum normalized substation attack impact and the number of measurements with attack cost 1 to 3 as a function of the number of tamper-proof authenticated RTUs in the 118 bus system for the mesh topology. Authentication eliminates measurements with attack cost $\Gamma_m = 1$ ($\Gamma_m = 2$, $\Gamma_m = 3$) after 19 (31,32) substations are authenticated. With 32 using tamper-proof authentication stealth attacks are impossible to perform. These 32 substations also form a dominating set of the power system graph, in accordance with Proposition 5. We note that authenticating the 31 substations found to make stealth attacks impossible for the star topology would also make stealth attacks impossible for the mesh topology.

7 Conclusion

We considered the problem of mitigating data integrity attacks against the power system state estimator. By combining a power flow model with a model of the SCADA communication infrastructure, we developed a framework and proposed security metrics to quantify the importance of substations and the cost of stealthy attacks against measurements. We provided efficient algorithms to calculate the security metrics. We proposed easy to calculate approximations of the security metrics based on the communication network topology only. We proposed an algorithm to improve the system security by using various mitigation measures, such as modified routing and data authentication. We illustrated the potential of the solutions through numerical examples on large IEEE benchmark power systems. Our results show the importance of considering the physical system and the network topology jointly when analyzing the security of the state estimator against attacks. It is subject of our future work to analyze the robustness of our metrics to changes in the power system topology and to random failures.

References

[1] A. Monticelli. Electric power system state estimation. *Proc. of the IEEE*, 88(2):262–282, 2000.

[2] Ali Abur and Antonio Gomez Exposito. *Power System State Estimation: Theory and Implementation*. Marcel Dekker, Inc., 2004.

[3] National Energy Technology Laboratory. Smart grid principal characteristics: Operates resiliently against attack and natural disasters. Technical report, U.S. Department of Energy, September 2009.

[4] A. Giani, Shankar S. Sastry, Karl H. Johansson, and H. Sandberg. The VIKING project: An initiative on resilient control of power networks. In *Proc. of the 2nd International Symposium on Resilient Control Systems*, 2009.

[5] A.A. Cárdenas, S. Amin, and S.S. Sastry. Research challenges for the security of control systems. In *Proc. of 3rd USENIX Workshop on Hot topics in security*, July 2008.

[6] Yao Liu, Peng Ning, and Michael Reiter. False data injection attacks against state estimation in electric power grids. In *Proc. of the 16th ACM conference on Computer and Communications Security (CCS)*, pages 21–32, 2009.

[7] A. Teixeira, György Dán, H. Sandberg, and Karl H. Johansson. A cyber security study of a SCADA energy management system: Stealthy deception attacks on the state estimator. In *Proc. IFAC World Congress*, Aug. 2011.

[8] Le Xie, Yilin Mo, and Bruno Sinopoli. False data injection attacks in electricity markets. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[9] L. Mili, T. Cutsem, and M. Ribbens-Pavella. Bad data identification methods in power system state estimation: A comparative study. *IEEE Trans. Power App. Syst.*, 104(11):3037–3049, Nov. 1985.

[10] F. F. Wu and W. H. E. Liu. Detection of topology errors by state estimation. *IEEE Trans. Power Syst.*, 4(1):176–183, Feb. 1989.

[11] Rakesh B. Bobba, Katherine M. Rogers, Qiyan Wang, Himanshu Khurana, Klara Nahrstedt, and Thomas J. Overbye. Detecting false data injection attacks on dc state estimation. In *Preprints of the First Workshop on Secure Control Systems, CPSWEEK*, Stockholm, Sweden, April 2010.

[12] Andr Teixeira, Saurabh Amin, Henrik Sandberg, Karl H. Johansson, and Shankar S. Sastry. Cyber-security analysis of state estimators in electric power systems. In *Proc. of IEEE Conf. on Decision and Control (CDC)*, Dec. 2010.

[13] Oliver Kosut, Liyan Jia, Robert J. Thomas, and Lang Tong. Malicious data attacks on the smart grid. *IEEE Trans. on Smart Grid*, 2:645–658, Oct 2011.

[14] György Dán and Henrik Sandberg. Stealth attacks and protection schemes for state estimators in power systems. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[15] T. T. Kim and H. V. Poor. Strategic protection against data injection attacks on power grids. *IEEE Trans. on Smart Grid*, 2:326–333, Jun. 2011.

[16] L.C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40:35–41, 1977.

[17] Reinhard Diestel. *Graph Theory*. Springer-Verlag, 2006.

[18] M.W. Bigrigg, K.M. Carley, K. Manousakis, and A. McAuley. Routing through an integrated communication and social network. In *Proc. of IEEE Military Communications Conference (MILCOM)*, 2009.

[19] Y. W. Law, L. Yen, R. Di Pietro, and M. Palaniswami. Secure k-connectivity properties of wireless sensor networks. In *Proc. of IEEE Conference on Mobile Adhoc and Sensor Systems (MASS)*, Oct. 2007.

[20] P.P. Tsang and S.W. Smith. YASIR: A low-latency, high-integrity security retrofit for legacy scada systems. In *Proc. of IFIP/TC11 International Information Security Conference*, 2008.

[21] Gilbert Strang. *Introduction to Applied Mathematics*. Wellesley-Cambridge Press, 1986.

[22] Oded Goldreich, Arnold L. Rosenberg, and Alan L. Selman, editors. *Dinitz' Algorithm: The Original Version and Even's Version*, volume 3895 of *LNCS Festschrift*, pages 218–240. Springer-Verlag, 2006.

[23] J.P. Ignizio and T.M. Cavalier. *Linear Programming*. Prentice Hall, Englewood Cliffs, NJ, 1994.

Paper B

On the Security of Distributed Power System State Estimation under Targeted Attacks

Ognjen Vuković and György Dán.

in Proc. of ACM Symposium on Applied Computing (SAC), March 2013.

On the Security of Distributed Power System State Estimation under Targeted Attacks

Ognjen Vuković and György Dán
Laboratory for Communication Networks
School of Electrical Engineering
KTH Royal Institute of Technology, Stockholm, Sweden
Email: {vukovic,gyuri}@ee.kth.se

Abstract

State estimation plays an essential role in the monitoring and control of power transmission systems. In modern, highly inter-connected power systems the state estimation should be performed in a distributed fashion and requires information exchange between the control centers of directly connected systems. Motivated by recent reports on trojans targeting industrial control systems, in this paper we investigate how a single compromised control center can affect the outcome of distributed state estimation. We describe five attack strategies, and evaluate their impact on the IEEE 118 benchmark power system. We show that that even if the state estimation converges despite the attack, the estimate can have up to 30% of error, and bad data detection cannot locate the attack. We also show that if powerful enough, the attack can impede the convergence of the state estimation, and thus it can blind the system operators. Our results show that it is important to provide confidentiality for the measurement data in order to prevent the most powerful attacks. Finally, we discuss a possible way to detect and to mitigate these attacks.

1 Introduction

Power system operators rely on Supervisory Control and Data Acquisition (SCADA) systems integrated with Energy Management Systems (EMS) to efficiently and safely operate the power grid. The SCADA system collects measurement data from the substations that belong to the operator into a control center. The measurement data are processed at the control center by the EMS. A core component of the EMS is the state estimator (SE), which allows the operator to get an accurate estimate of the state of the power system despite noisy or faulty measurement data by using a steady-state model of the power flows in the physical system [1, 2]. The state estimate is used by various EMS applications, such as contingency analysis and security constrained economic dispatch, and thus an accurate state estimate is crucial both for the safety and for the efficiency of the power system's operation.

In order to improve operational efficiency, modern power systems have become increasingly inter-connected and are managed by several independent operators. Each operator has its own

SCADA/EMS system and control center, which it uses to manage a region of the entire system. Examples of inter-connected systems are the Western Interconnect (WECC) in the U.S., the ENTSO-E in Europe, and some major European national transmission systems managed by various operators. In the future smart grid, inter-connected systems are expected to become even more prevalent, and it is expected that their control and supervision becomes fully distributed, without any central coordinator. The safety of an inter-connected power system depends on the safety of its constituent regions, as demonstrated by recent cascading failures, e.g., the 2003 North-East blackout in the U.S. It is therefore important that the regional operators exchange timely and accurate information about each other's networks state. Due to the sensitivity of the data, the information exchange is in practice very limited. Nevertheless, the exchanged information is used in the regional control centers as an input to the SE. The resulting fully distributed SE [3, 4, 5] are effectively extensions of the basic SE algorithm and aim to achieve a consistent state estimate for the entire power system.

Motivated by recent reports on trojans targeting industrial control systems, such as Stuxnet and Duqu [6], in this work we address the security of distributed state estimation in the presence of a misbehaving control center. We consider an attacker that compromises a single control center so that it can manipulate the data that the control center exchanges with its neighbors. We define various attack strategies that differ in the attacker's knowledge about the system. We show via simulations on an IEEE benchmark power system that attacks can disturb the distributed state estimation in two ways. First, the distributed state estimation could yield a highly erroneous state estimate (up to 30% relative estimation error), and second, the distributed state estimation could fail to provide any state estimate. Moreover, our results show that it is important to protect the confidentiality of measurement data, since the attacker needs those data to perform the strongest kinds of attacks. Finally, we show a possible way to detect convergence problems as a consequence of an attack by relying on a contraction mapping interpretation of distributed state estimation. This detection is a complement to traditional bad data detection (BDD) algorithms, which require the SE to converge.

Several recent works focused on the security of standalone SEs for the case of so called stealth attacks [7, 8, 9, 10, 11, 12, 13, 14]. Stealth attacks are false data injection attacks against SCADA measurement data that bypass the model-based bad data detector used in the SE. The possibility of such attacks was pointed out in [7], and different mitigation schemes were proposed in [9] based on protecting individual data, by changing the bad data detection algorithm [10], and by protecting components of the SCADA network infrastructure [11, 12]. The problem of maintaining operator privacy for distributed state estimation was addressed recently in an information theoretic framework in [15]. To the best of our knowledge we are the first to consider the vulnerability of distributed state estimators to data integrity attacks.

The rest of the paper is organized as follows. In Section 2 we describe the system model and give an outline of distributed state estimation algorithms. In Section 3 we describe the attacker model and define various strategies. Section 4 provides an impact analysis of the attack strategies. In Section 5 we consider a possible detection and mitigation strategy, and Section 6 concludes the paper.

2 System Model

We consider an inter-connected power system that spans several administrative areas, called regions. We denote the set of buses by $\mathcal{B}$, $|\mathcal{B}| = B$, and the set of regions by $\mathcal{R}$. Each bus belongs to a region, and we denote the set of buses that belong to region $r \in \mathcal{R}$ by $\mathcal{B}_r$.

We say that a transmission line $t_{b,b'}$ that connects $b \in \mathcal{B}_r$ and $b' \in \mathcal{B}_{r'}$ is a *tie line* between two regions if $r \neq r'$. We say that $b \in \mathcal{B}_r$ is a border bus to region r' if there is a tie line $t_{b,b'}$ for some $b' \in \mathcal{B}_{r'}$. We denote the set of all tie lines connecting region r to region r' by $\mathcal{T}_{r,r'} = \{t_{b,b'} \mid b \in \mathcal{B}_r,\ b' \in \mathcal{B}_{r'}\ \}$. The set of all border buses of region r to region r' is denoted by $\mathcal{B}_{r,r'} = \{b \mid \exists t_{b,b'} \in \mathcal{T}_{r,r'}\}$ ($B_{r,r'} = |\mathcal{B}_{r,r'}|$). Similarly, the set of border buses from all regions to region r' is denoted by $\mathcal{B}_{b,r'} = \cup_r \mathcal{B}_{r,r'}$ ($B_{b,r'} = |\mathcal{B}_{b,r'}|$). Finally, we say that two regions are *neighbors* if they share at least one tie line. We denote the set of neighbors of region r by $\mathcal{N}(r)$ ($N(r) = |\mathcal{N}(r)|$).

2.1 State Estimation

We consider models of the active and reactive power injections at every bus, and models of the active and reactive power flows between buses (over transmission lines) [1, 2]. The power flow and injection measurement values are denoted by the vector $z \in \mathbb{R}^M$, where M is the number of measurements. The value of a measurement m equals to $z_m = P_m + e_m$, where P_m is the actual power flow or injection (active or reactive) and e_m is independent random measurement noise. The noise is usually assumed to have a Gaussian distribution of zero mean, $e = (e_1, e_2, ..., e_M)^T \in N(0;R)$ where $W = Eee^T$ is the diagonal measurement covariance matrix.

The state-estimation problem consists of estimating B voltage phasor vectors, $\mathcal{V}_b = V_b e^{j\theta_b}\ \forall b \in \mathcal{B}$, given the power flow and injection measurement vector z. One (arbitrary) voltage phasor can be selected as the reference phasor, for example $\mathcal{V}_B = 1e^{j0}$, and then only $n = B - 1$ phasors have to be estimated. We denote by x, the *state vector*, which consists of the voltage phasor angles and magnitudes, i.e., $x = [\theta_1, V_1, \theta_2, V_2, ..., \theta_n, V_n]^T$, where θ_i and V_i are phase angle and voltage magnitude on bus b_i, respectively. We refer to a component of the vector x as a *state variable*.

The most widely used approach to solve the estimation problem is to minimize the squares of the weighted deviations of the estimated variables from the actual measurements [2], which can be formulated as

$$\min_x J(x) = \min_x [z - f(x)]^T [W^{-1}] [z - f(x)], \qquad (1)$$

where $f(x)$ is the vector of functions describing the measurements as a function of the state vector x. Since f is non-linear, the estimation is typically done using an iterative solution scheme known as the Gauss-Newton algorithm [2]. The recurrence relation of this iterative solution scheme is

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)}, \tag{2}$$

and the increment $\Delta x^{(k)}$ can be calculated as

$$\Delta x^{(k)} = [H^{(k)T} W^{-1} H^{(k)}]^{-1} H^{(k)T} W^{-1} \Delta z^{(k)}, \tag{3}$$

where $H^{(k)}$ is the Jacobian of vector $f(x^{(k)})$, $\Delta z^{(k)}$ is the measurement residual vector defined as $\Delta z^{(k)} = z - f(x^{(k)})$, and $x^{(k)}$ is the value of vector x at the k^{th} iteration. The algorithm is said to converge when for some k^* the maximum update of the state variables is smaller than the *convergence threshold* $\varepsilon > 0$, i.e., $||\Delta x^{(k^*)}||_\infty < \varepsilon$, where $||\cdot||_\infty$ denotes the maximum norm of a vector. We refer to the number of iterations k^* required for convergence as the *convergence time*.

Once the state estimator converges, a Bad Data Detection (BDD) algorithm is used to detect and identify faulty measurement data. The BDD algorithm analyses the measurement residual vector ($\Delta z^{(k^*)}$). The most widely used BDD algorithm is the *Largest Normalized Residual Test (LNRT)*. The LNRT suspects the measurement with highest normalized residual, i.e., the largest value of the measurement residual vector divided by its Euclidean norm ($\Delta z^{(k^*)}/||\Delta z^{(k^*)}||_2$), as bad data, if the ratio is above a certain threshold. For a more complete treatment of BDD we refer to [1, 2].

2.2 Distributed State Estimation (DSE)

In an inter-connected power system each regional control center performs the state estimation based the topology and the parameters of the region, and based on the measurements taken in the region. Therefore, the state estimation problem in region r becomes a problem of estimating the voltage phasor vectors for the buses $b \in \mathcal{B}_r$, i.e., the state vector x_r. However, the power flow measurements on the tie lines $\mathcal{T}_{r,r'}$ ($r' \in \mathcal{N}(r)$), which we refer to as the *boundary* measurements, are a function of the state variables of the neighboring regions r' as well. Hence, the control center of region r needs to exchange a few state variables with the control centers of its neighboring regions. These state variables correspond to the buses at the two ends of the tie lines; the control center of region r sends the state variables for the buses in $\mathcal{B}_{r,r'}$ to the control center of region r'. In most of the recently proposed DSE algorithms, e.g., [3, 4, 5], state variables are exchanged at the beginning of every iteration. For the purpose of our study, we consider the algorithm described in [3].

We denote the vector of state variables communicated by region r to region r' (r' to r) at iteration k by $x^{(k)}_{r,r'}$ ($x^{(k)}_{r',r}$), and define it as

$$x^{(k)}_{r,r'} = [\theta^{(k)}_{i_1} \; V^{(k)}_{i_1} \; \theta^{(k)}_{i_2} \; V^{(k)}_{i_2} \; \ldots]^T, \;\; \forall b_{i_j} \in \mathcal{B}_{r,r'}. \tag{4}$$

We denote the vector of state variables that region r receives from its neighbors at iteration k by

$$x^{(k)}_{b,r} = [x^{(k)T}_{r'_{i_1},r} \; x^{(k)T}_{r'_{i_2},r} \; \ldots]^T, \;\; \forall r'_{i_j} \in \mathcal{N}(r).$$

The state estimator of region r uses $x_{b,r}^{(k)}$ to iteratively estimate x_r similar to (2) and (3), but the Jacobian and the measurement residual vector are calculated as

$$H^{(k)} = \left[\frac{\partial f(y_r^{(k)})}{\partial x_r^{(k)}} \right] \quad \Delta z^{(k)} = z - f(y_r^{(k)}), \tag{5}$$

where $y_r^{(k)} = [x_r^{(k)T} \; x_{b,r}^{(k)T}]^T$ is the state vector extended with the boundary state variables received at the beginning of the current iteration, i.e., iteration k. The DSE is said to converge when all regional state estimators converge. If we denote by k_r^* the convergence time of region r, then the *total convergence time* is $c = \max_r(k_r^*)$.

3 Attack Scenario

DSE requires that neighboring control centers periodically exchange data with each other. The most widely used protocol for this purpose is the standardized Inter-Control Center Communications Protocol (ICCP or IEC 60870-6/TASE.2). ICCP defines data structures and encodings, and allows control centers to establish so called associations on a pair-wise basis. An association allows bidirectional data exchange between two control centers. Using ICCP it is possible to implement access control, but ICCP provides no means for key-based authentication of the data sent.

The standard way of providing authentication for ICCP associations is to rely on the authentication provided by standard transport layer protocols, such as TLS and SSL [16], as mandated by IEC 62351. As an effect, ICCP messages might be passed in clear text to the TCP/IP protocol stack or to standard libraries providing authentication. An attacker that compromises the operating system and the TCP/IP protocol stack in a control center, e.g., by installing a trojan, can thus easily manipulate all incoming and outgoing ICCP messages at the compromised control center. The vulnerability of control systems to such an attack is aggravated by the fact that ICCP associations are often established between hosts in demilitarized zones.

3.1 Attack Model

We consider an attacker whose goal is to introduce disturbances in DSE. In order to achieve its goal, the attacker corrupts the control center of a single region $r^a \in \mathcal{R}$ so that it has access to the state variables exchanged between region r^a and its neighbors $\mathcal{N}(r^a)$ at the beginning of every DSE iteration. At iteration k, the state variables are elements of the vectors $x_{r,r^a}^{(k)}$, $\forall r \in \mathcal{N}(r^a)$, and the vectors $x_{r^a,r}^{(k)}$, $\forall r \in \mathcal{N}(r^a)$. In principle, the attacker can tamper with the entire vectors, but the relative differences in voltage magnitudes between neighboring buses are rather small and their manipulation may be easy to detect. Therefore, we focus on an attacker that tampers with the exchanged state variables that correspond to the phase angles. We describe the attack against the state variables sent from regions

$r \in \mathcal{N}(r^a)$ to region r^a (from r^a to r) at the beginning of iteration k by the *attack vector* $a^{(k)}_{r,r^a}$ ($a^{(k)}_{r^a,r}$). We define the attack vector $a^{(k)}_{r,r^a}$ as the vector of phase angles

$$a^{(k)}_{r,r^a} = [\hat{\theta}^{(k)}_{i_1}\ \hat{\theta}^{(k)}_{i_2}\ \ldots]^T \ \ \forall b_{i_j} \in \mathcal{B}_{r,r^a}, \tag{6}$$

where element $\hat{\theta}^{(k)}_{i_j}$ corresponds to the value that the attacker adds to the phase angle $\theta^{(k)}_{i_j}$ that it wants to modify. The attack vector $a^{(k)}_{r,r^a}$ can be defined in a similar way.

In the rest of this Section, we describe the attack against the state variables sent to region r^a from its neighbors $r \in \mathcal{N}(r^a)$. The attack against the state variables sent from region r^a to its neighbors can be described in a similar way, but we omit it for brevity.

Since the attack is additive and it concerns the phase angles of the exchanged vector of state variables $x^{(k)}_{r,r^a}$, it results in a corrupted vector of state variables

$$\tilde{x}^{(k)}_{r,r^a} = x^{(k)}_{r,r^a} + Q_{r,r^a} \cdot a^{(k)}_{r,r^a}, \tag{7}$$

where $Q_{r,r^a} = [q_{i,j}]_{2 \cdot B_{r,r^a} \times B_{r,r^a}}$ is a matrix used to insert the components that correspond to voltage magnitudes with values equal to 0. The elements of matrix Q_{r,r^a} are defined as: $q_{i,j} = 1$ if $\lceil j = i/2 \rceil$ and $i \bmod 2 = 1$, and $q_{i,j} = 0$ otherwise. The resulting vector $\tilde{x}^{(k)}_{r,r^a}$ is used as an input to the iteration k of DSE in region r^a, instead of the originally exchanged vector $x^{(k)}_{r,r^a}$.

For convenience, we introduce the attack vector $a^{(k)}_{b,r^a}$ for the state variables sent to region r^a from all its neighboring regions

$$a^{(k)}_{b,r^a} = [a^{(k)T}_{r_{i_1},r^a}\ a^{(k)T}_{r_{i_2},r^a}\ \ldots]^T \ \ \forall r_{i_j} \in \mathcal{N}(r^a), \tag{8}$$

and the corresponding corrupted vector of state variables

$$\tilde{x}^{(k)}_{b,r^a} = x^{(k)}_{b,r^a} + Q_{r^a} \cdot a^{(k)}_{b,r^a}, \tag{9}$$

where $Q_{r^a} = [q_{i,j}]_{2 \cdot B_{b,r^a} \times B_{b,r^a}}$ is a matrix with the same structure as Q_{r,r^a}. Fig. 1 illustrates an attack on a power system with three regions. Observe that $\tilde{x}^{(k)}_{b,r^a}$ is the input to iteration k of DSE, and thus, the attack $a^{(k)}_{b,r^a}$ leads to a *corrupted* state vector update $\Delta\tilde{x}^{(k)}_{r^a}$.

We define *the size of the attack* as the Euclidean norm of the attack vector, i.e., $||a^{(k)}_{b,r^a}||_2$. We consider that the goal of the attacker is to find an attack vector with a small size but with a big impact on the convergence time c of the distributed state estimator, or formally

$$\max_{a^{(k)}_{b,r^a}, k=1,\ldots} \ c \qquad s.t.\ ||a^{(k)}_{b,r^a}||_2 \leq \beta \ \ \forall k, \tag{10}$$

where $\beta > 0$ is the desired bound on the attack size. By definition, $c = \infty$ if the DSE does not converge.

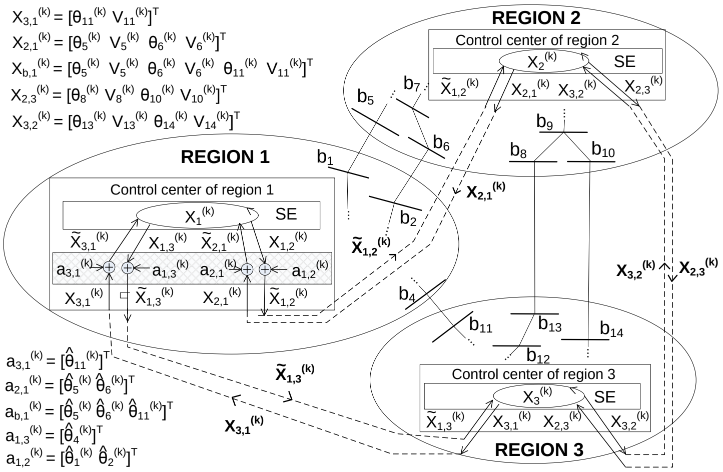

$X_{3,1}^{(k)} = [\theta_{11}^{(k)}\ V_{11}^{(k)}]^T$
$X_{2,1}^{(k)} = [\theta_5^{(k)}\ V_5^{(k)}\ \theta_6^{(k)}\ V_6^{(k)}]^T$
$X_{b,1}^{(k)} = [\theta_5^{(k)}\ V_5^{(k)}\ \theta_6^{(k)}\ V_6^{(k)}\ \theta_{11}^{(k)}\ V_{11}^{(k)}]^T$
$X_{2,3}^{(k)} = [\theta_8^{(k)}\ V_8^{(k)}\ \theta_{10}^{(k)}\ V_{10}^{(k)}]^T$
$X_{3,2}^{(k)} = [\theta_{13}^{(k)}\ V_{13}^{(k)}\ \theta_{14}^{(k)}\ V_{14}^{(k)}]^T$
REGION 2
Control center of region 2
$X_2^{(k)}$
SE
$\tilde{X}_{1,2}^{(k)}$ $X_{2,1}^{(k)}$ $X_{3,2}^{(k)}$ $X_{2,3}^{(k)}$
b_5 b_7 b_6 b_9 b_8 b_{10}
REGION 1
b_1
Control center of region 1
$X_1^{(k)}$
SE
$\tilde{X}_{3,1}^{(k)}$ $X_{1,3}^{(k)}$ $\tilde{X}_{2,1}^{(k)}$ $X_{1,2}^{(k)}$
$a_{3,1}^{(k)}$ $a_{1,3}^{(k)}$ $a_{2,1}^{(k)}$ $a_{1,2}^{(k)}$
$X_{3,1}^{(k)}$ $\tilde{X}_{1,3}^{(k)}$ $X_{2,1}^{(k)}$ $\tilde{X}_{1,2}^{(k)}$
b_2
$\mathbf{X_{2,1}^{(k)}}$
$\mathbf{\tilde{X}_{1,2}^{(k)}}$
$\mathbf{X_{3,2}^{(k)}}$ $\mathbf{X_{2,3}^{(k)}}$
b_4
b_{11} b_{13} b_{14} b_{12}
$a_{3,1}^{(k)} = [\hat{\theta}_{11}^{(k)}]^T$
$a_{2,1}^{(k)} = [\hat{\theta}_5^{(k)}\ \hat{\theta}_6^{(k)}]^T$
$a_{b,1}^{(k)} = [\hat{\theta}_5^{(k)}\ \hat{\theta}_6^{(k)}\ \hat{\theta}_{11}^{(k)}]^T$
$a_{1,3}^{(k)} = [\hat{\theta}_4^{(k)}]^T$
$a_{1,2}^{(k)} = [\hat{\theta}_1^{(k)}\ \hat{\theta}_2^{(k)}]^T$
$\mathbf{\tilde{X}_{1,3}^{(k)}}$
$\mathbf{X_{3,1}^{(k)}}$
Control center of region 3
$X_3^{(k)}$
SE
$\tilde{X}_{1,3}^{(k)}$ $X_{3,1}^{(k)}$ $X_{2,3}^{(k)}$ $X_{3,2}^{(k)}$
REGION 3

Figure 1: Interconnected power system with three regions. The attacker corrupts the control center of Region 1, and tampers with the state variables $x_{1,2}^{(k)}$ and $x_{1,3}^{(k)}$ sent from Region 1, and the state variables $x_{2,1}^{(k)}$ and $x_{3,1}^{(k)}$ received by Region 1. The symbol (+) indicates that the components of the attack vector are added to the corresponding components (phase angles) of the vector of exchanged state variables. The attacker cannot tamper with the state variables exchanged between Regions 2 and 3.

3.2 Attack Strategies

Since the distributed state estimation problem is non-linear, solving (10) is non-trivial. In the following we describe five strategies to construct the attack vector $a_{b,r^a}^{(k)}$.

3.2.1 Maximal Update Vector Attack (MUV)

The MUV attack is an approximation of (10) done by maximizing the Euclidean norm of the corrupted state vector update in every iteration,

$$\max_{a_{b,r^a}^{(k)}} ||\Delta\tilde{x}_r^{(k)}||_2 \;\; s.t. \;\; ||a_{b,r^a}^{(k)}||_2 = \beta. \tag{11}$$

Recall that $\Delta\tilde{x}_r^{(k)}$ depends on $a_{b,r^a}^{(k)}$ through (3) and (5). The objective function and the constraints in (11) are quadratic functions, and therefore the vector $a_{b,r^a}^{(k)}$ can be obtained by solving a quadratically constrained quadratic program [17]. Observe that the attacker cannot solve (11) without knowing the entire state vector $x_r^{(k)}$ and the measurement vector z, but the vectors $x_r^{(k)}$ and z are not exchanged between the regions. We therefore use the MUV attack as a baseline for comparison.

3.2.2 First Singular Vector Attack (FSV)

The FSV attack also aims to solve (11) but in the cases when the vectors $x_r^{(k)}$ and z may be unknown to the attacker. We denote by $x_r^{a(k)}$ the attacker's knowledge of the vector $x_r^{(k)}$ at iteration k. Correspondingly, we denote by $x_{b,r}^{a(k)}$ and by $y_r^{a(k)}$ the attacker's knowledge of the vectors $x_{b,r}^{(k)}$ and $y_r^{(k)}$, respectively. In order to approximate (11), we linearize the function $f(y_r^{(k)})$ at $y_r^{a(k)}$ so that for the measurement residual vector $\Delta\tilde{z}^{(k)}$ we obtain

$$\begin{aligned}\Delta\tilde{z}^{(k)} &\approx z - (f(\begin{bmatrix} x_r^{a(k)} \\ x_{b,r}^{a(k)}\end{bmatrix}) + [H^{a(k)} H_b^{a(k)}] \begin{bmatrix}\mathbf{0}\\ Q_{r^a}\cdot a_{b,r^a}^{(k)}\end{bmatrix}) \\ &\approx \Delta z^{(k)} - [H^{a(k)}H_b^{a(k)}]\begin{bmatrix}\mathbf{0}\\ Q_{r^a}\cdot a_{b,r^a}^{(k)}\end{bmatrix} \approx \Delta z^{(k)} - H_b^{a(k)}\cdot Q_{r^a}\cdot a_{b,r^a}^{(k)},\end{aligned} \tag{12}$$

where $H^{a(k)}$ and $H_b^{a(k)}$ are the Jacobian matrices of $f(y_r^{(k)})$ evaluated at $x_r^{a(k)}$ and $x_{b,r}^{a(k)}$, respectively. After substituting (12) into (3), the corrupted vector $\Delta\tilde{x}_r^{(k)}$ can be approximated as

$$\Delta\tilde{x}_r^{(k)} = \Delta x_r^{(k)} - [H^{a(k)\,T} W^{-1} H^{a(k)}]^{-1} H^{a(k)\,T} W^{-1} H_b^{a(k)} \cdot Q_{r^a}\cdot a_{b,r^a}^{(k)}. \tag{13}$$

Observe that the subtrahend in (13) is a vector with the same number of elements as the vector $\Delta x_r^{(k)}$, and we refer to it as the *subtrahend vector*. The Euclidean norm of the subtrahend vector is maximized if the attack vector $a_{b,r^a}^{(k)}$ is aligned with the first right singular vector of the matrix $[H^{a(k)\,T} W^{-1} H^{a(k)}]^{-1} H^{a(k)\,T} W^{-1} H_b^{a(k)} \cdot Q_{r^a}$, that is, with the singular vector with highest singular value. The complexity of singular vector decomposition is $O(mn^2)$ [18], low enough for the computation to be done on-line.

Observe in (13) that size of the corrupted vector $\Delta\tilde{x}_r^{(k)}$ depends on the direction of the subtrahend vector, and consequently, on the direction of the first singular vector. Whether the attacker will choose the correct direction of the first singular vector depends on its knowledge of the state vector $x_r^{(k)}$, and on the measurement vector z. We consider two variants of the FSV attack.

FSV with State Information (FSV+ST): The FSV+ST attack assumes that the attacker knows the state vector $x_r^{(k)}$, but it does not know the measurement vector z and the correct direction. Consequently, $x_r^{a(k)} = x_r^{(k)}$and $y_r^{a(k)} = y_r^{(k)}$ in (12) and (13). Since the attacker does not know the vector z, and thereby the update vector $\Delta x_r^{(k)}$ without attack, finding the correct direction is not trivial. In order to estimate the direction, we assume that the estimates of the active and reactive power flows on a tie line are closer to their actual values when using the most recent exchanged state variables. The attacker may tamper with the exchanged state variables such that the introduced estimation errors take the estimates closer to the estimates from the previous round. The direction which satisfies this for more tie line power flows is chosen by the attacker.

FSV with Measurement Information (FSV+MEAS): The FSV+MEAS attack assumes that the attacker does not know the state vector $x_r^{(k)}$, but it knows the measurement vector z. Consequently, $x_r^{a(k)} = x_r^{(1)}$ and $y_r^{a(k)} = y_r^{(1)}$ in (12) and (13). The update vector $\Delta x_r^{(k)}$, and thereby the correct direction, is not known by the attacker. In order to estimate the direction, we use a similar approach as for the FSV+ST attack, but the attacker uses the actual measurements, rather than two estimates, when choosing the direction.

3.2.3 Uniform Rotation Attack (UR)

The third strategy we consider is rather naive. The attack vector rotates all voltage phasors by a constant ϕ, thus

$$a_{b,r^a}^{(k)} = \phi \cdot \mathbf{1}, \tag{14}$$

where $\mathbf{1}$ is the column vector of all ones of dimension B_{b,r^a}. The size of the attack is $||a_{b,r^a}^{(k)}||_2 = \phi \cdot \sqrt{B_{b,r^a}}$.

3.2.4 Sign Inversion Attack (SI)

The fourth strategy we consider is adaptive, similar to the FSV attack. The attack only requires knowledge of the exchanged state variables, and at every round it inverts the sign of exchanged phase angles,

$$a_{b,r^a}^{(k)} = [-2\theta_{i_1}^{(k)} \;\; -2\theta_{i_2}^{(k)} \;\; \dots] \;\; \forall b_{i_j} \in \mathcal{B}_{b,r^a}. \tag{15}$$

The size of the attack depends on the system state.

3.2.5 Sign of Difference Inversion Attack (SDI)

The last strategy is based on the insight that the steady state active power flow on a tie line is an odd function of the phase angle difference between the border buses [2],

$$a_{b,r^a}^{(k)} = [-2(\theta_{i_1}^{(k)} - \theta_{i_1'}^{(k)}) \;\; \dots] \forall b_{i_j} \in \mathcal{B}_{b,r^a} \text{ and } t_{b_i,b_i'} \in \mathcal{T}_{b,r^a}. \tag{16}$$

The attack effectively inverts the sign of the phase angle differences for every tie line, which corresponds to reverting the power flow on every tie line of region r^a. Again, the size of the attack depends on the system state.

4 Attack Impact

In the following we evaluate the impact of the attack strategies on the IEEE 118 bus power system. The power system is divided into six regions as shown in Fig. 2. We consider that the attacker corrupts the control center of region r_1, and performs the attacks against the state variables sent from and to region r_1. Measurements are taken at every power injection and power flow (both active and reactive), and the convergence threshold is $\varepsilon = 10^{-3}$.

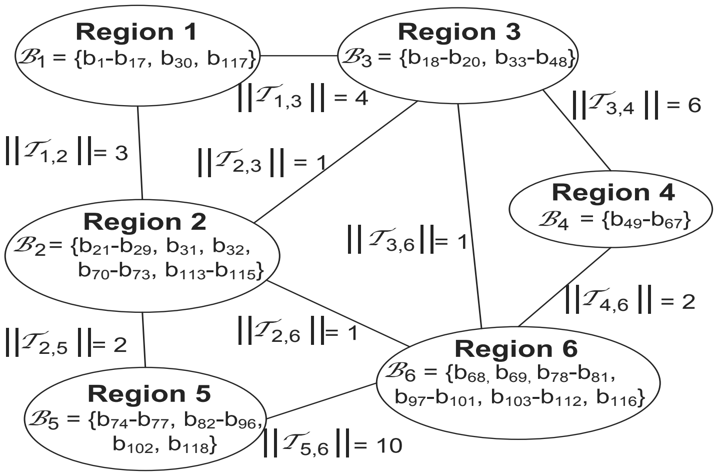

Region 1
$\mathcal{B}_1$ = {b1-b17, b30, b117}
Region 3
$\mathcal{B}_3$ = {b18-b20, b33-b48}
$||\mathcal{T}_{1,3}||$ = 4
$||\mathcal{T}_{3,4}||$ = 6
$||\mathcal{T}_{1,2}||$= 3
$||\mathcal{T}_{2,3}||$ = 1
Region 4
$\mathcal{B}_4$ = {b49-b67}
Region 2
$\mathcal{B}_2$= {b21-b29, b31, b32, b70-b73, b113-b115}
$||\mathcal{T}_{3,6}||$= 1
$||\mathcal{T}_{4,6}||$ = 2
$||\mathcal{T}_{2,5}||$= 2
$||\mathcal{T}_{2,6}||$= 1
Region 6
$\mathcal{B}_6$ = {b68, b69, b78-b81, b97-b101, b103-b112, b116}
Region 5
$\mathcal{B}_5$ = {b74-b77, b82-b96, b102, b118}
$||\mathcal{T}_{5,6}||$= 10

Figure 2: IEEE 118 bus system divided into six regions. Neighboring regions are connected by a line, $||\mathcal{T}_{r,r'}||$ is the number of tie-lines. The buses $\mathcal{B}_r$ are shown for each region.

Fig. 3 shows the total convergence time c as a function of the attack size for the DSE under the MUV, the FSV (both variants), and the UR attacks. The total convergence time increases monotonically with the attack size for all considered attacks. The MUV attack is the most powerful among the considered attacks: the increase of the convergence time is significantly higher for the same attack size, and the DSE stops converging for a much lower attack size. The results show that the FSV+MEAS attack is significantly more powerful than the FSV+ST attack. Therefore, it is important to prevent the attacker from obtaining the measurement data, e.g., by not exchanging the data between neighboring regions and by encrypting the data when transmitting them from the substations to the control center.

Although for small attacks the DSE converges, the estimated state and thus the estimated power flows could be erroneous. Fig. 4 and Fig. 5 show the 50th percentile and the maximum of the relative estimation error for the highest 50% and for the highest 10% of the power flows, respectively as a function of total convergence time (and thus the attack size). The relative estimation error increases monotonically with the total convergence time, and thereby the attack size, and can exceed 25% for some large power flows, which is a significant estimation error that can affect the outcome of EMS applications like contingency analysis.

In principle, the BDD algorithm should identify the measurements whose estimates significantly differ from the measured values (e.g., due to the attack) as bad data, and should thus detect the attack. In the following we use the centralized Largest Normalized Residual Test algorithm [3] for BDD to evaluate the efficiency of bad data detection under the considered attacks. We use a centralized BDD, because a centralized BDD is typically

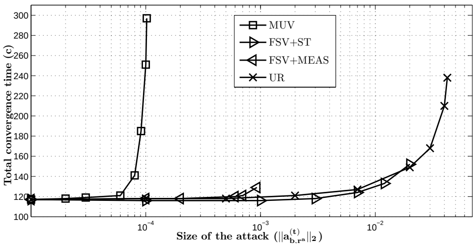


Figure 3: Total convergence time for cases when the DSE converges as a function of the attack size.

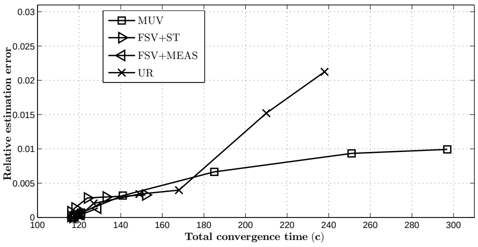


Figure 4: Relative estimation error (50th percentile) for the upper 50% utilized power flows and injections vs. total convergence time.

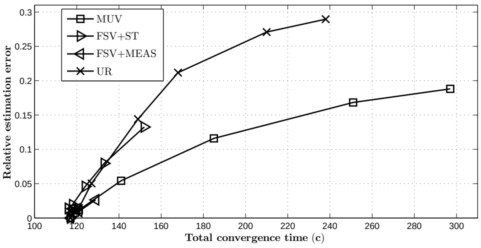


Figure 5: Relative estimation error (maximum) for the upper 10% utilized power flows and injections vs. total convergence time.

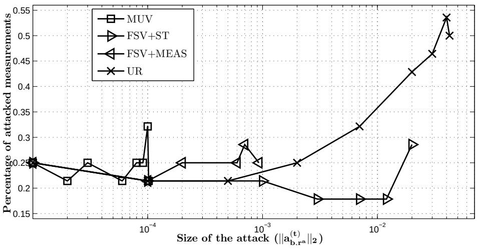


Figure 6: Percentage of the border measurements around the attacked region that are among top candidates for bad data vs. size of the attack.

more efficient in identifying bad data than the fully distributed algorithms, e.g., [19]. We thus consider the strongest BDD possible. We focus on attacks that allow the DSE to converge, as the BDD cannot be performed if the DSE does not converge.

Since the attack concerns the power flow estimates at the tie lines connecting the attacked region with its neighbors, one would expect that the border measurements around the attacked region get identified by the BDD algorithm as bad data. If this was the case then by discarding those measurements, the BDD would isolate the rest of the system from the attacked region. However, this is not the case. Fig. 6 shows the percentage of the border measurements around the attacked region that are identified by the BDD algorithm as the top candidates for bad data as a function of the attack size for the MUV, the FSV (both variants), and the UR attacks. The percentage does not increase monotonically, and it is fairly constant even for strong attacks that cause significant estimation errors. Moreover, the percentage is relatively low for all attacks. This implies that the BDD algorithm may be misled: it can discard measurements in/between non-attacked regions, and does not locate the source of the attack.

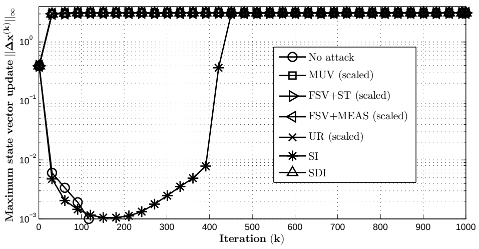


Figure 7: Evolution of the max. value of the state vector update in the entire system ($||\Delta x^{(k)}||_\infty$) with and without data integrity attacks in region r_1.

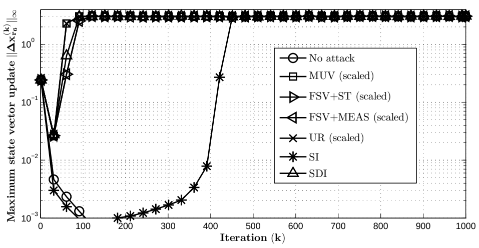


Figure 8: Evolution of the max. value of the state vector update in region r_6 ($||\Delta x_{r_6}^{(k)}||_\infty$) with and without data integrity attacks in region r_1.

Fig. 7 shows the maximum state vector update $||\Delta x||_\infty$ for the FSV+ST, the UR, the SI, and the SDI attacks in each iteration k. In order to make the results comparable, we scaled the FSV+ST and the UR attacks such that their attack size equals to the attack size of the SI attack in every iteration. Under the SI attack the DSE almost converges, but all attack strategies prevent the DSE to converge eventually. One may assume that the DSE does not converge mainly due to the state vector updates in the corrupted region (r_1) and its neighbors, but Fig. 8 shows that this is not the case. Fig. 8 shows the maximum state vector update $||\Delta x_{r_6}||_\infty$ in the non-neighboring region r_6. While $||\Delta x_{r_6}||_\infty$ decreases initially for all attacks, it eventually starts increasing and diverges from the convergence threshold due to the resulting disturbances in r_1, r_2, and r_3 that propagate to the rest of the system through the state variables that are communicated. It is interesting that in the case of the SI attack the state estimator in region r_6 first converges, but not the DSE since at least one of the other regions has not converged yet, and as an effect $||\Delta x_{r_6}||_\infty$ starts increasing.

5 Detection and Mitigation

In the following we discuss a possible way for detecting an attack against the DSE. For the detection, let us first consider the evolution of the state vector without the data integrity attack. Observe that the evolution of the state vector in the DSE can be written as a recurrence relation $x^{(k+1)} = g(x^{(k)})$ for some non-linear mapping $g : \mathbb{R}^n \to \mathbb{R}^n$. Furthermore, when the DSE converges after k^* iterations, it holds that $x^{(k^*)} = g(x^{(k^*-1)}) \approx x^{(k^*-1)}$. In order for the DSE to converge, the mapping g has to satisfy certain conditions. One example is the sufficient condition formulated in [3, Proposition 5.2., Theorem 7.5.], which provides some insight into the behavior of the recurrence relation defined by g. The following proposition summarizes the condition.

Proposition 1. *If the iterative non-linear mapping function $g : \mathbb{R}^n \to \mathbb{R}^n$ is non-expansive*

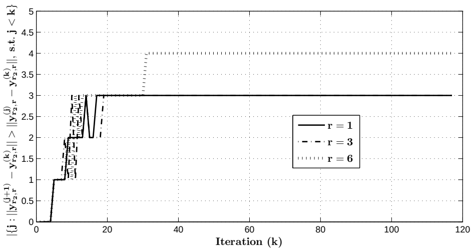


Figure 9: Evolution of the number of outlier state estimates based on $y^{(k)}_{r',r}$ in region $r' = 2$ vs. the number of rounds. No attack.

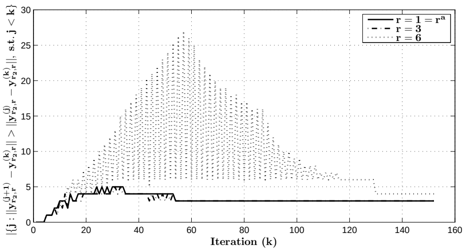


Figure 10: Evolution of the number of outlier state estimates based on $y^{(k)}_{r',r}$ in region $r' = 2$ vs. the number of rounds. FSV+MEAS attack at region $r^a = 1$ that admits convergence.

in the Euclidean norm, then the set X^* of its fixed points is non-empty. If it satisfies

$$||g(x) - x^*||_\infty \leq ||x - x^*||_\infty, \forall x \in \mathbb{R}^n, \forall x^* \in X^*, \tag{17}$$

then the solution sequence $x^{(k)}$ converges to a fixed point x^*.

The above result does not imply that the subsequent state vector updates $\Delta x^{(k)}$ would form a non-increasing sequence in the max norm, i.e., $||g(x^{(k+1)}) - g(x^{(k)})||_\infty \not\leq ||g(x^{(k)}) - g(x^{(k-1)})||_\infty$. Furthermore, the set of fixed points X^* is not known. Nevertheless, for large values of k we can use the approximation that the estimate $x^{(k)}$ is close to a fixed point of g, and thus for large k and $k' < k$ we have

$$||x^{(k'+1)} - x^{(k)}||_\infty \leq ||x^{(k')} - x^{(k)}||_\infty \tag{18}$$

assuming that the state estimator converges. In other words, when the state estimator is close to convergence to a fixed point, the distance of the points on the trajectory of convergence from the current estimate is a non-increasing function of the iteration k'. In the case of DSE the regional control centers only have access to their own state vector $x^{(k)}_r$ and to the last received state variables $x^{(k)}_{b,r}$ from their neighbors, i.e., to the vector $y^{(k)}_r$, and thus (18) has to be verified on these data. In the following we investigate how well (18) indicates convergence problems based on this data.

Fig. 9 shows for every iteration k the number of previous iterations j for which (18) does not hold for the vector $y^{(k)}_{r',r} = [x^{(k)T}_{r'} \; x^{(k)}_{r,r'}]^T$ for region $r' = 2$ for three of its neighbors. The results are for the case without any attack. The results confirm that the number of outliers is small when the DSE converges, and also shows that the few outliers occur during the early iterations of the DSE. Fig. 10 shows results for a small FSV+MEAS attack that allows the DSE to converge, though with an estimation error (c.f. Figs 4 and 5). Initially, the number of outliers increases with the number of iterations, but it decreases as the DSE gets closer to convergence. Surprisingly, most outliers are detected based on $y^{(k)}_{2,6}$, although region 6 is not

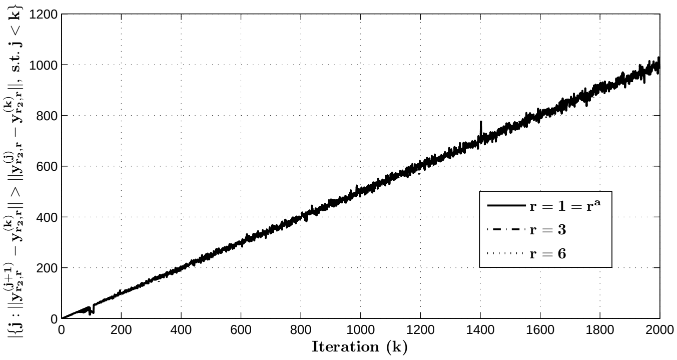


Figure 11: Evolution of the number of outlier state estimates based on $y_{r',r}^{(k)}$ in region $r' = 2$ vs. the number of rounds. FSV+MEAS attack at region $r^a = 1$ that does not admit convergence.

a neighbor of the attacked region ($r^a = 1$). Finally, Fig. 11 shows results for a FSV+MEAS attack that does not allow the DSE to converge. The number of outliers increases linearly with the number of iterations, and indicates the convergence problem immediately.

Fig. 10 and Fig. 11 show that outliers can be used to detect convergence problems due to, e.g., an attack. However, Fig. 11 also shows that localizing the point of the attack is not possible. One possible mitigation scheme could then be to disable the DSE, and let every region perform a local state estimation. Although power injections at border buses and the power flows on the tie lines cannot be estimated in this case, the resulting estimate is not affected by the attack.

6 Conclusion

We considered the vulnerability of distributed state estimation to targeted attacks against the exchanged data between operators. We described five attack strategies, and showed via simulations on an IEEE benchmark power system the effects of the attacks. The presented results led us to the following interesting conclusions. First, already a single compromised control center can cause convergence problems to the distributed state estimator. For small attacks the estimator converges but with significant errors, and the BDD algorithm cannot detect the attack location. For large attacks the estimator fails to converge and to provide a consistent state estimate. Second, it is important to protect the confidentiality of measurement data, since the attacker can perform strong attacks only if it knows the measurement data. Finally, the attacks could be detected by observing the number of outlier state estimates. Based on this detection scheme, we outlined a simple mitigation scheme. It is subject of our future work to extend the detection scheme such that it can localize the point of the attack, which could lead to an improved mitigation scheme.

References

[1] A. Monticelli. Electric power system state estimation. *Proc. of the IEEE*, 88(2):262–282, 2000.

[2] Ali Abur and Antonio Gomez Exposito. *Power System State Estimation: Theory and Implementation*. Marcel Dekker, Inc., 2004.

[3] M. Shahidehpour and Y. Wang. *Communication and Control in Electric Power Systems*. John Wiley and Sons, 2003.

[4] Sebastian de la Torre Antonio J. Conejo and Miguel Canas. An optimization approach to multiarea state estimation. *IEEE Transactions on Power Systems*, 22(1), February 2007.

[5] Soummya Kar Le Xie, Dae-Hyun Choi and H. Vincent Poor. Fully distributed state estimation for wide-area monitoring systems. *IEEE Transactions on Smart Grid*, 3(3), 2012.

[6] Symantec Security Response. W32.duq: The precursor to the next stuxnet, November 2011.

[7] Yao Liu, Peng Ning, and Michael Reiter. False data injection attacks against state estimation in electric power grids. In *Proc. of the 16th ACM conference on Computer and Communications Security (CCS)*, pages 21–32, 2009.

[8] A. Teixeira, G. Dán, H. Sandberg, and Karl H. Johansson. A cyber security study of a SCADA energy management system: Stealthy deception attacks on the state estimator. In *Proc. IFAC World Congress*, 2011.

[9] Rakesh B. Bobba, Katherine M. Rogers, Qiyan Wang, Himanshu Khurana, Klara Nahrstedt, and Thomas J. Overbye. Detecting false data injection attacks on dc state estimation. In *Preprints of the First Workshop on Secure Control Systems, CPSWEEK*, 2010.

[10] Oliver Kosut, Liyan Jia, Robert Thomas, and Lang Tong. Malicious data attacks on smart grid state estimation: Attack strategies and countermeasures. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[11] Gy. Dán and Henrik Sandberg. Stealth attacks and protection schemes for state estimators in power systems. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[12] Ognjen Vuković, Kin Cheong Sou, György Dán, and Henrik Sandberg. Network-aware mitigation of data integrity attacks on power system state estimation. *IEEE JSAC: Smart Grid Communications Series*, 30(6), 2012.

[13] T. T. Kim and H. V. Poor. Strategic protection against data injection attacks on power grids. *IEEE Trans. on Smart Grid*, 2:326–333, Jun. 2011.

[14] Annarita Giani, Eilyan Bitar, Manuel Garcia, Miles McQueen, Pramod Khargonekar, and Kameshwar Poolla. Smart grid data integrity attacks: Characterizations and countermeasures. In *Proc. of IEEE SmartGridComm*, Oct. 2011.

[15] Lalitha Sankar, Soummya Kar, Ravi Tandon, and H. Vincent Poor. Competitive privacy in the smart grid: An information-theoretic approach. In *Proc. of IEEE SmartGridComm*, Oct. 2011.

[16] T. Dierks and E. Rescorla. RFC5246: The transport layer security (TLS) protocol version 1.2. http://tools.ietf.org/html/rfc5246, August 2008.

[17] Stephen Boyd and Vandenberghe Lieven. *Convex Optimization*. Cambridge University Press, 2004.

[18] R.A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1990.

[19] Dae-Hyun Choi and Le Xie. Fully distributed bad data processing for wide area state estimation. In *Proc. of IEEE SmartGridComm*, October 2011.

Paper C

Security of Fully Distributed Power System State Estimation: Detection and Mitigation of Data Integrity Attacks

Ognjen Vuković and György Dán.

in IEEE Journal on Selected Areas in Communications (JSAC), vol. 32, no. 7, July 2014.

Security of Fully Distributed Power System State Estimation: Detection and Mitigation of Data Integrity Attacks

Ognjen Vuković and György Dán
Laboratory for Communication Networks
School of Electrical Engineering
KTH Royal Institute of Technology, Stockholm, Sweden
Email: {vukovic,gyuri}@ee.kth.se

Abstract

State estimation plays an essential role in the monitoring and supervision of power systems. In today's power systems state estimation is typically done in a centralized or in a hierarchical way, but as power systems will be increasingly interconnected in the future smart grid, distributed state estimation will become an important alternative to centralized and hierarchical solutions. As the future smart grid may rely on distributed state estimation, it is essential to understand the potential vulnerabilities that distributed state estimation may have. In this paper, we show that an attacker that compromises the communication infrastructure of a single control center in an interconnected power system can successfully perform a denial of service attack against state-of-the-art distributed state estimation, and consequently it can blind the system operators of every region. As a solution to mitigate such a denial of service attack, we propose a fully distributed algorithm for attack detection. Furthermore, we propose a fully distributed algorithm that identifies the most likely attack location based on the individual regions' beliefs about the attack location, isolates the identified region, and then reruns the distributed state estimation. We validate the proposed algorithms on the IEEE 118 bus benchmark power system.

1 Introduction

Power system state estimation (SE) is an essential functionality of modern Energy Managements Systems (EMS), which allows the power system operators to get an accurate estimate of the system's state despite noisy or faulty measurement data collected by the Supervisory Control and Data Acquisition (SCADA) system at substations [1, 2]. The output of the SE, the estimated state and the resulting power flows, is the basis for various important EMS applications, such as contingency analysis used to assess how an outage would affect system stability, and optimal power flow used to compute the optimal generation profile based on some predefined criteria. Hence, an accurate state estimate is crucial both for system safety and for operating efficiency.

The importance of SE has made its security a major concern, and therefore the vulnerability of standalone SEs to so called stealth attacks has been widely studied [3, 4, 5, 6, 7, 8, 9, 10]. Stealth attacks are false data injection attacks against the measurement data collected by the SCADA system that successfully bypass the model-based bad data detection (BDD) used in the SE [3]. To secure standalone SE, a variety of mitigation schemes were proposed recently against stealth attacks [3, 5, 6, 7, 8].

Power systems are increasingly interconnected and the trend of interconnection is expected to continue in the future smart grid. Interconnected power systems are managed by independent operators; each operator uses SE to estimate the state of the region of the interconnected system that it controls. Examples of interconnected power systems are the Western Interconnect (WECC) in the U.S., and the ENTSO-E in Europe. The safety of an interconnected power system depends on the safety of its constituent regions, as demonstrated by recent cascading failures (e.g., the U.S. North-East blackout in 2003). It is therefore very important that the operators exchange accurate information about their most recent system state in a timely manner. However, the information exchange is very limited in practice due to the sensitivity of the data, and it typically includes only the data needed for a consistent and correct estimate of power flows on the lines connecting two regions. While today the SE in interconnected power systems is mostly done hierarchically, there is an increasing interest for fully distributed SE (DSE) for future smart grids [11, 12, 13, 14, 15], as it eliminates the need for a central authority. DSE is effectively an extension of the basic SE [1, 2], and it can obtain a consistent state estimate for the entire interconnected power system.

Despite its importance, the security of DSE has not received significant attention. In an interconnected power system every region could in principle use an appropriate mitigation scheme to secure its own local SE. Nevertheless, in the case of DSE in an interconnected system, the security of one's local SE may depend on the security of other SEs, and the security of the DSE as a whole may also depend on the security of the data exchange between the regions [16]. In order to design secure and resilient DSEs for future smart grids, it is thus important to understand the potential vulnerabilities of DSE, i.e., whether or not a compromised control center or compromised data exchange between SEs could affect the DSE. If DSE is vulnerable to attacks, it is important to develop mitigation schemes for the vulnerabilities.

In this work we consider false data injection attacks on fully distributed SE. We consider an attacker that compromises a single control center so that it can manipulate the data exchanged between the control center and its neighbors. We consider one of the most recent DSE algorithms [15] and show that an attacker can effectively disable the DSE by manipulating the data exchanged by the attacked control center. We propose an algorithm to detect the attack by identifying discrepancies in the temporal evolution of the exchanged data between regions. Furthermore, we propose a distributed algorithm to mitigate the attack. The algorithm identifies the region with the compromised control center by consolidating the beliefs of the individual regions about the origin of the attack, isolates the identified region, and then restarts the DSE.

The structure of the paper is as follows. In Section 2 we discuss related work. In Section 3 we outline the DSE algorithm used for our study. In Section 4 we describe the attack model and show that the false data injection attacks can disable the DSE. In Section 5 we propose an algorithm for attack detection, and in Section 6 we propose the algorithm for mitigation. Section 7 concludes the paper.

2 Related work

The vulnerability of standalone SE to false data injection attacks was first studied in [3]. There it was shown that the measurement data collected by SCADA can be corrupted so that they do not trigger the BDD system. Such attacks are often called stealth attacks. The observation was made using a linearized model of the SE, but it was shown later on a SCADA/EMS testbed that stealth attacks are also possible under a non-linear model [4]. Since then the security of standalone SE has received much attention [3, 4, 5, 6, 7, 8, 9, 10]. Various schemes were proposed to mitigate stealth attacks, through individual data protection [5], through changes to the BDD algorithm [6], and through the protection of the SCADA infrastructure [7, 8].

The vulnerability of hierarchical multi-area state estimation has been studied in [17], where the authors extended the false data injection attack presented in [3] to the case of a bi-level hierarchical state estimator, and gave some results on how the attack could impede network observability. The security of DSE against false data injection attacks on the exchanged data between neighboring operators was studied in [16] for a simple DSE [11]. It was shown that an attack can disable the DSE, i.e., can prevent it from finding a correct estimate. Furthermore, a detection scheme was proposed to detect an attack along with a simple mitigation scheme. The mitigation scheme suggested that upon detecting an attack, the regions ignore all exchanged data and perform a local SE. However, by using such a mitigation scheme, the power flows on transmission lines connecting any two regions cannot be correctly estimated. Compared to [16], in this paper we consider a state-of-the-art DSE [15], and we propose a mitigation scheme that makes it possible for the DSE to be performed between non attacked regions. Consequently, the power flows on the lines connecting the non attacked regions can be correctly estimated.

Distributed state estimation can be considered a form of consensus. A widely studied model of consensus under attack is the byzantine consensus problem [18, 19, 20], in which a number of processors have to agree on a value even if some processors may report a false value to influence the consensus. In our work the processors are the regions, but the attack is fundamentally different; its goal is to impede the convergence of the distributed state estimation, and the mitigation scheme we propose not only provides convergence but it also allows to localize the attack.

3 System Model and State Estimation

We consider an inter-connected power system that consists of several control areas, which we call regions. We denote the set of regions by $\mathcal{R}$, and use $|\mathcal{R}| = R$. A region $r \in \mathcal{R}$ includes a subset of all buses, and a subset of the transmission lines. Regions have no common buses, but there are shared transmission lines, which connect two regions. We refer to the shared transmission lines as *tie lines*, and to the buses connected by these lines as *border buses*.

We consider models of the active power injections at every bus, and active power flows on transmission lines [1, 2]. The active power injection and flow measurements taken in region r are denoted by the vector $z_r \in \mathbb{R}^{M_r}$, where M_r is the number of measurements in region r. The measurements equal to the actual power injections/flows plus independent random measurement noise, $z_r = f_r(x_r) + e$, where x_r is the vector of phase angles used to compute the power flows in region r. The noise e is usually assumed to have a Gaussian distribution of zero mean. We denote by W_r the diagonal measurement covariance matrix.

We refer to the vector of phase angles x_r as the state vector in region r, and we refer to a component of the vector x_r as a *state variable*. The state variables of the vector x_r correspond to the phase angles on buses that belong to region r, and to the phase angles on border buses in other regions that are needed to describe the measurements on the tie lines and to describe power injection measurements at border buses in region r. Consequently, the state variables included in vectors x_r, $\forall r \in \mathcal{R}$ are overlapping. We denote by $x_{r,r'}$ the vector of state variables of region r that correspond to state variables shared between regions r and r'. Observe that all components in the vector $x_{r,r'}$ are also contained in the vector x_r. We say that region r and region r' are neighbors if the vector $x_{r,r'}$ has at least one component, and we denote the set of all neighbors of region r by $\mathcal{N}(r)$ $(|\mathcal{N}(r)| = N(r))$. For convenience, we introduce the vector $x_{r,b}$ for all state variables that region r shares with its neighboring regions $\mathcal{N}(r)$, i.e., the components in the vectors $x_{r,r'}, \forall r' \in \mathcal{N}(r)$ form the vector $x_{r,b}$. The vectors $x_{r',r}$ and $x_{b,r}$ can be defined in a similar way.

3.1 Distributed State Estimation (DSE)

The state-estimation problem consists of estimating the voltage phase angles x at all buses given the power flow and injection measurement vector [2]. In the case of DSE each control center needs to estimate those phase angles that are related to its measurements, but it has to cooperate with neighboring control centers, typically by exchanging the state variables of the border buses, to ensure that the power flows on the tie lines are correctly estimated. In most of the recently proposed DSE algorithms, e.g., [11, 12, 13, 15], state variables are exchanged at the beginning or at the end of every iteration, and are used as an input when calculating the next state vector update. For the purpose of our study, we consider a state-of-the-art algorithm proposed in [15], which is highly robust and obtains accurate estimates of the power flows on the tie lines. The algorithm works as follows.

The goal of the DSE is to estimate x_r in every region under the condition that the estimates of shared state variables match between neighboring regions. One (arbitrary) phase angle in the entire interconnected system is selected as the reference angle, and its value is fixed to zero. Each region estimates x_r by minimizing the squares of the weighted deviations of the estimated active power flows and injections (which are functions of x_r) from the measured values (comprehended in z_r). Therefore, the distributed state estimation problem can be formulated as

$$\begin{aligned} &\min_{x_r,\, r\in\mathcal{R}} \sum_{r\in\mathcal{R}} [z_r - f_r(x_r)]^T [W_r^{-1}][z_r - f_r(x_r)] \\ &s.t. \quad x_{r,r'} = x_{r',r} \quad \forall r \in \mathcal{R}\, and\, \forall r' \in \mathcal{N}(r), \end{aligned} \tag{1}$$

where $f_r(x)$ is the vector of non-linear functions describing the active power flows and power injections in region r as a function of the state vector x_r.

The constraints in (1) couple the estimation across regions. In order to have a fully distributed algorithm, auxiliary variables can be introduced so that the problem can be solved using the alternating direction method of multipliers (ADMM) [15]. The resulting iterative solution scheme is

$$\begin{aligned} x_r^{(k+1)} &= (H_r^{(k)T} W^{-1} H_r^{(k)} + cD_r)^{-1} (H_r^{(k)T} z_r + cD_r p_r^{(k)}) \\ s_r^{(k+1)} &= U_{x_r} \cdot \sum_{\forall r' \in \mathcal{N}(r)} Y_{r,r'} \cdot x_{r',r}^{(k+1)} \\ p_r^{(k+1)} &= p_r^{(k)} + s_r^{(k+1)} - \frac{1}{2}(Y_{r,b} \cdot Y_{r,b}^T \cdot x_r^{(k)} - s_r^{(k)}), \end{aligned}$$

where $c > 0$ is a predefined constant, the matrix $H_r^{(k)}$ is the Jacobian of vector $f_r(x^{(k)})$, and matrices D_r, U_{x_r}, $Y_{r,r'}$ are defined as follows. D_r is a diagonal matrix whose element $d_{i,i}$ equals the number of regions sharing the ith component (state variable) of the vector x_r. U_{x_r} is a diagonal matrix whose elements are defined as: $u_{i,i}$ equals to the inverse of the number of regions (if greater than 0) sharing the ith component (state variable) of the vector x_r, and zero otherwise. Finally, $Y_{r,r'}$ is a matrix that determines the connection between vector x_r and vector $x_{r,r'}$, and its elements are: $y_{i,j} = 1$ if the ith element (state variable) in x_r corresponds to the jth element (state variable) in $x_{r,r'}$, and $y_{i,j} = 0$ otherwise. Consequently, we have

$$x_{r,r'} = Y_{r,r'}^T \cdot x_r \ . \tag{2}$$

Similar to (2), we introduce the matrix $Y_{r,b}$, which has a similar structure as $Y_{r,r'}$ so that we have

$$x_{r,b} = Y_{r,b}^T \cdot x_r \tag{3}$$

The matrix $Y_{b,r}$ can be defined in a similar way.

The DSE is said to converge when for some k^* the maximum change of the state variables in every region is smaller than the *convergence threshold* $\varepsilon > 0$, i.e., $\forall r \in \mathcal{R},\ ||x_r^{(k^*+1)} - x_r^{k^*}||_\infty < \varepsilon$, where $||\cdot||_\infty$ denotes the maximum norm of a vector. We refer to the number of iterations k^* required for convergence as the *convergence time*.

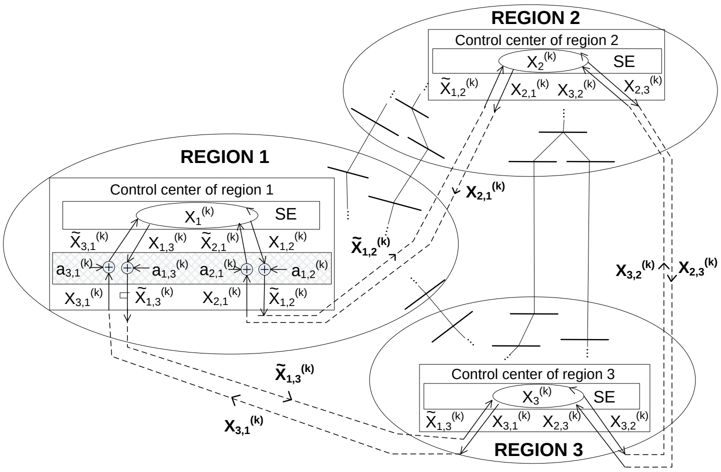


Figure 1: Interconnected power system with three regions. The attacker corrupts the control center of Region 1, and tampers with the state variables $x_{1,2}^{(k)}$ and $x_{1,3}^{(k)}$ sent from Region 1, and the state variables $x_{2,1}^{(k)}$ and $x_{3,1}^{(k)}$ received by Region 1. The symbol (+) indicates that the components of the attack vector are added to the corresponding components (phase angles) of the vector of exchanged state variables. The attacker cannot tamper with the state variables exchanged between Regions 2 and 3.

4 A DoS Attack on DSE

We consider an attacker whose goal is to perform a Denial-of-Service (DoS) attack against the DSE, i.e., to disable the DSE by preventing it from converging. The attacker compromises the communication infrastructure of a region $r^a \in \mathcal{R}$ used for data exchange between r^a and its neighbors $\mathcal{N}(r^a)$, so it can manipulate the exchanged data used as an input to the DSE. The exchanged data are the state variables defined by the vectors $x_{r,r^a}^{(k)}$, $\forall r \in \mathcal{N}(r^a)$, and the vectors $x_{r^a,r}^{(k)}$, $\forall r \in \mathcal{N}(r^a)$. We describe the attack against the state variables sent from regions $r \in \mathcal{N}(r^a)$ to region r^a (from r^a to r) at the end of iteration k by the *attack vector* $a_{r,r^a}^{(k)}$ ($a_{r^a,r}^{(k)}$). We define the attack vector $a_{r,r^a}^{(k)}$ as the vector of phase angles whose elements correspond to the value that the attacker adds to that phase angle, that is,

$$\tilde{x}_{r,r^a}^{(k)} = x_{r,r^a}^{(k)} + a_{r,r^a}^{(k)}, \tag{4}$$

where $\tilde{x}_{r,r^a}^{(k)}$ is the resulting corrupted vector of state variables. The vector $\tilde{x}_{r,r^a}^{(k)}$ is used as input to the next iteration of DSE in region r^a, instead of the originally exchanged vector $x_{r,r^a}^{(k)}$. The attack vector $a_{r^a,r}^{(k)}$ can be defined in a similar way.

In the rest of this Section, we describe the attack against the state variables sent to region r^a from its neighbors $r \in \mathcal{N}(r^a)$. The attack against the state variables sent from region r^a to its neighbors can be described in a similar way, but we omit it for brevity. For convenience, we introduce the attack vector $a^{(k)}_{b,r^a}$ for the state variables that region r^a receives from all its neighboring regions

$$a^{(k)}_{b,r^a} = [a^{(k)T}_{r_{i_1},r^a}\ a^{(k)T}_{r_{i_2},r^a}\ \dots]^T\ \ \forall r_{i_j} \in \mathcal{N}(r^a), \tag{5}$$

and the corresponding corrupted vector of state variables

$$\tilde{x}^{(k)}_{b,r^a} = x^{(k)}_{b,r^a} + a^{(k)}_{b,r^a}, \tag{6}$$

Fig. 1 illustrates an attack on a power system with three regions. Observe that $\tilde{x}^{(k)}_{b,r^a}$ is the input to iteration $k+1$ of DSE, and thus, the attack $a^{(k)}_{b,r^a}$ leads to a *corrupted* state vector $\tilde{x}^{(k+1)}_{r^a}$.

We define *the size of the attack* as the Euclidean norm of the attack vector, i.e., $||a^{(k)}_{b,r^a}||_2$. Intuitively, a smaller attack size implies smaller corruption added to the exchanged values, which could make the detection and the localization of the attack harder; as our results will show later, this is indeed the case. Thus, it would be natural for the attacker to look for the smallest attack vector that prevents the DSE from converging ($k^* = \infty$), or formally

$$\min_{a^{(k)}_{b,r^a},k=1,\dots} \beta \qquad \text{s.t. } k^* = \infty \ \text{ and } \ \beta = ||a^{(k)}_{b,r^a}||_2; \forall k. \tag{7}$$

Since the distributed state estimation problem is non-linear, solving (7) is non-trivial. In the following we propose an approximation of the above objective.

4.1 First Singular Vector Attack (FSV)

The First Singular Vector (FSV) attack is an approximation of (7) done by maximizing the introduced disturbances for a given attack size. Note that the attack vector $a^{(k)}_{b,r^a}$ results in corrupted vectors

$$\begin{aligned} \tilde{s}^{(k+1)}_{r^a} &= s^{(k+1)}_{r^a} + U_{x_r} \cdot Y_{b,r^a} \cdot a^{(k)}_{b,r^a} \\ \tilde{p}^{(k+1)}_{r^a} &= p^{(k+1)}_{r^a} + U_{x_r} \cdot Y_{b,r^a} \cdot a^{(k)}_{b,r^a}, \end{aligned} \tag{8}$$

which yield a corrupted state vector

$$\tilde{x}^{(k+1)}_{r^a} = x^{(k+1)}_{r^a} + K \cdot a^{(k)}_{b,r^a}, \tag{9}$$

where $K = (H^{(k)T}_r W^{-1} H^{(k)}_r + cD_r)^{-1} \cdot cD_r U_{x_r} Y_{b,r^a}$. Note that the addend in (9) is a vector with the same number of elements as the vector $x^{(k+1)}_{r^a}$, and we refer to it as the *addend*

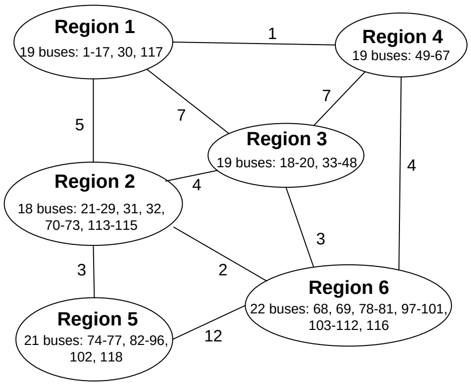


Figure 2: IEEE 118 bus system divided into six regions. Neighboring regions are connected by a line and the number next to the line represents the number of shared state variables. Note that the reference bus (69) is not a state variable.

vector. The Euclidean norm of the addend vector is maximized if the attack vector $a_{b,r^a}^{(k)}$ is aligned with the first right singular vector of the matrix K, that is, with the singular vector with highest singular value. The complexity of singular vector decomposition is $O(mn^2)$ [21], low enough for the computation to be done on-line. Nevertheless, the computation of the Jacobian $H_r^{(k)}$ requires knowledge of the current system state $x_{r^a}^{(k)}$ for the attacked region r^a. Since the entire current system state is not exchanged between the regions, and consequently the attacker does not have access to all entries in $x_{r^a}^{(k)}$, we approximate $H_r^{(k)}$ with the Jacobian calculated at the initial state $H_r^{(0)}$. Such an approximation can be easily used by a sophisticated attacker that knows the system model, which is also sufficient to obtain the matrices U_{x_r} and Y_{b,r^a}.

Observe that in (9) the size of the corrupted vector $\tilde{x}_{r^a}^{(k+1)}$ depends on the direction of the addend vector, and consequently, on the direction of the first singular vector. Since the attacker does not know the state vector $x_{r^a}^{(k)}$, finding the correct direction is not trivial. In order to estimate the direction, the attacker can assume that the estimates of the power flows on a tie line are closer to their actual values when using the most recent exchanged state variables. Then, the attacker applies the attack so that the introduced estimation errors take the estimates in the direction towards the previous iteration estimates.

4.2 Impact of FSV Attack on DSE

We show the impact of the FSV attack on the IEEE 118 bus power system, divided into six regions as shown in Fig. 2. We consider that the attacker corrupts the control center of one of the regions, and performs the attacks against the state variables sent from and to that

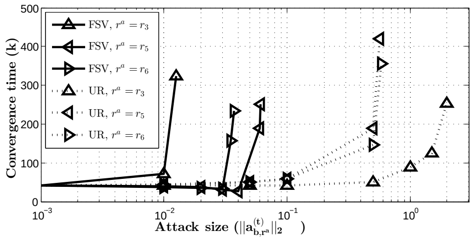


Figure 3: Convergence time for cases when the DSE converges as a function of the attack size.

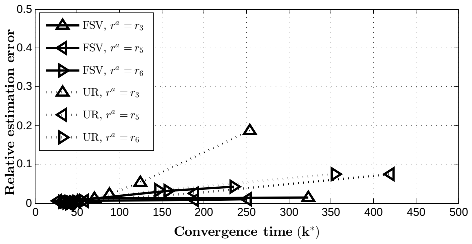


Figure 4: Relative estimation error (maximum) for the upper 10% utilized power flows and injections vs. convergence time.

region. Bus 69, located in region r_6, is used as the reference bus, as specified in the IEEE 118 bus power system. Measurements are taken at every power injection and power flow, and the convergence threshold is $\varepsilon = 10^{-3}$. The phase angles, thereby the state variables and the attack vector, are in radians.

As a baseline for comparison we use a simple attack, the Uniform Rotation (UR) attack, which adds a constant ϕ to every compromised state variable. The attack vector of the UR attack is thus $a^{(k)}_{b,r^a} = \phi \cdot \mathbf{1}$, where $\mathbf{1}$ is the column vector of all ones with the same dimension as the vector $a^{(k)}_{b,r^a}$. The size of the attack is $||a^{(k)}_{b,r^a}||_2 = \phi \cdot \sqrt{|a^{(k)}_{b,r^a}|}$, where $|a^{(k)}_{b,r^a}|$ denotes the number of elements in the vector $a^{(k)}_{b,r^a}$.

Fig. 3 shows the convergence time k^* (when the DSE converges) as a function of the attack size for the FSV attack and for the UR attack considering regions r_3, r_5, and r_6 individually as the attacked region. For all considered cases, both the FSV attack and the UR attack can prevent the DSE from converging, i.e., lead to denial of service. The FSV attack is more powerful than the UR attack: FSV requires a much smaller attack size for a successful denial of service attack than UR. One might expect that the DSE is more sensitive when the region containing the reference bus is attacked, since it may be harder for other regions to synchronize with the reference bus. However, the results show that this is not the case: there is no significant difference when the region containing the reference bus is attacked (region r_6), and when some other region is attacked using either the FSV attack or the UR attack.

Observe that in Fig. 3 it does not take a big FSV attack to prevent the DSE from converging. For example, the FSV attack with size $||a^{(k)}_{b,r^a}||_2 = 0.07$ prevents the DSE from converging regardless of which region is attacked. This size corresponds to an average value of the attack vector elements of 0.0265 radians (1.51 degrees) if region r_1 is attacked, or 0.019 (1.07 degrees) if region r_6 is attacked.

Although for small attacks the DSE converges, the estimated state and thus the estimated power flows could be erroneous. Fig. 4 shows the maximum of the relative estimation error for the highest 10% of the power flows and injections as a function of convergence time (and thus the attack size). The relative estimation error increases monotonically with the convergence time, and thereby the attack size, and can exceed 15% for some power flows.

Given the potential of the FSV attack and the UR attack to prevent the DSE from converging, a natural question is whether the attacks can be detected and mitigated. In the following, we show that this is possible.

5 Detection of Attacks

Let us start by elaborating on the convergence of the DSE. Recall that in order to solve (1) in a fully distributed fashion, the right-hand side of the condition $x_{r,r'} = x_{r',r}$ is replaced with an auxiliary variable for each $r \in \mathcal{R}$ and $\forall r' \in \mathcal{N}(r)$. In iteration k and for regions r and r', the auxiliary variable equals to the average of the shared state variables between the regions, i.e., $(x^{(k)}_{r,r'} + x^{(k)}_{r',r})/2$ [15]. Consequently, the condition in (1) can be expressed as $x^{(k)}_{r,r'} = (x^{(k)}_{r,r'} + x^{(k)}_{r',r})/2$, or $(x^{(k)}_{r,r'} - x^{(k)}_{r',r})/2 = 0$. The resulting decomposed problem is solved with the ADMM, which guarantees convergence if the following criteria are satisfied (based on [22]).

Proposition 1. *If for* $\forall r \in \mathcal{R}$ *the function* $J_r(x_r) = [z_r - f_r(x_r)]^T [W_r^{-1}][z_r - f_r(x_r)]$ *that region r minimizes (the summand in (1)), is closed, proper, and convex, and the augmented Lagrangian*

$$\mathcal{L} = \sum_{\forall r \in \mathcal{R}} J_r(x_r) + y^T \frac{x^{(k)}_{r,r'} - x^{(k)}_{r',r}}{2} + c||\frac{x^{(k)}_{r,r'} - x^{(k)}_{r',r}}{2}||_2^2 \tag{10}$$

(y is Lagrange multiplier) has a saddle point, then the ADMM converges and $||(x^{(k)}_{r,r'} - x^{(k)}_{r',r})/2||_2^2 \to 0$ *as* $k \to \infty$ *[22, Appendix A,p. 106–110].*

Observe that if the conditions in Proposition 1 are satisfied, and therefore the DSE converges without an attack, the disagreement $||(x^{(k)}_{r,r'} - x^{(k)}_{r',r})/2||_2^2$ may not decrease monotonically. However, for large k and when the DSE approaches a solution, one may expect that

$$||(x^{(k+1)}_{r,r'} - x^{(k+1)}_{r',r})/2||_2^2 < ||(x^{(k)}_{r,r'} - x^{(k)}_{r',r})/2||_2^2 \tag{11}$$

holds for all state variables exchanged between regions. In what follows we investigate if a normalized version of (11) can be used to detect convergence problems due to an attack.

Definition. *The mean squared disagreement (MSD) between regions r and r′ at iteration k is*

$$d^{(k)}_{r,r'} = \frac{||(x^{(k)}_{r,r'} - x^{(k)}_{r',r})/2||_2^2}{|x^{(k)}_{r,r'}|}, \tag{12}$$

where $|x^{(k)}_{r,r'}|$ *denotes the number of elements in vector* $x^{(k)}_{r,r'}$*. Observe that by definition* $d^{(k)}_{r,r'} = d^{(k)}_{r',r}$*.*

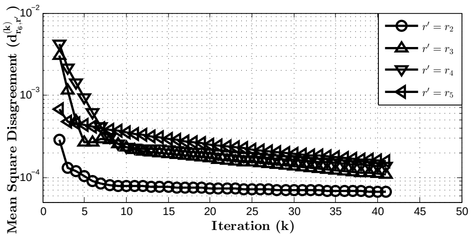


Figure 5: Evolution of the MSDs $d^{(k)}_{r_6,r'}$ observed in region r_6 for $r' \in \mathcal{N}(r_6)$. No attack.

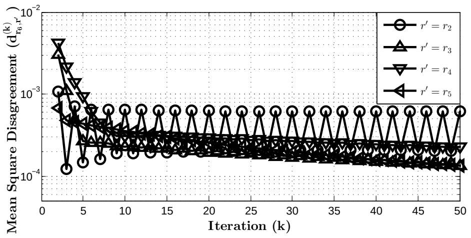


Figure 6: Evolution of the MSDs $d^{(k)}_{r_6,r'}$ observed in region r_6 for $r' \in \mathcal{N}(r_6)$ in presence of FSV attack in region $r^a = r_2 \in \mathcal{N}(r_6)$ for attack size 0.1.

Fig. 5 shows the evolution of the MSD $d^{(k)}_{r_6,r'}$ between region r_6 and its neighbors $r' \in \mathcal{N}(r_6)$ without an attack: it decreases for all $r' \in \mathcal{N}(r_6)$. Fig. 6 and Fig. 7 show the evolution of the MSDs of regions r_6 and r_5, which are neighbors of the attacked regions $r^a = r_2 \in \mathcal{N}(r_6)$ and $r^a = r_6 \in \mathcal{N}(r_5)$, for the FSV attack and for the UR attack, respectively. Observe that not all MSDs decrease with the iterations, which is in contrast to Proposition 1. This is the phenomenon we use to detect convergence problems as described in the following.

Proposition 2. *Let* $\sup\{\cdot\}$ *be the supremum of a set. If the conditions in Proposition 1 are satisfied, but for large k there are some r and* $r' \in \mathcal{N}(r)$ *such that* $\sup\{d^{(k')}_{r,r'} : k' > k\} > 0$, $||x^{(k+1)}_r - x^{(k)}_r||_\infty > \varepsilon$, *and* $\nexists t \in \mathbb{N}$ *so that*

$$\sup\{d^{(k')}_{r,r'} : k' > k\} > \sup\{d^{(k')}_{r,r'} : k' > k+t\} \tag{13}$$

then there is a convergence problem (an attack).

Proof. The proof follows from Proposition 1. If the conditions of Proposition 1 hold, then $||(x^{(k)}_{r,r'} - x^{(k)}_{r',r})/2||^2_2 \to 0$ and $d^{(k)}_{r,r'} \to 0$ as $k \to \infty$. Consequently, $\sup\{d^{(k')}_{r,r'} : k' > k\} \to 0$. □

The regions can thus use Proposition 2 to detect an attack.

6 Mitigation of Attacks

Given that we can detect an ongoing attack, the next important question is whether it is possible to mitigate the attack. In the following we propose a mitigation algorithm that first aims at localizing the region where a detected attack originates from, and then isolates the region so that the DSE can converge.

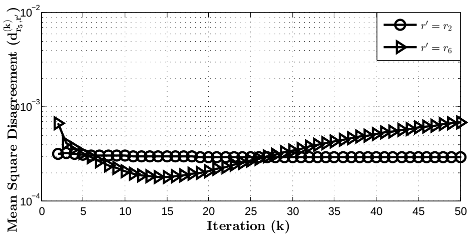


Figure 7: Evolution of the MSDs $d^{(k)}_{r_5,r'}$ observed in region r_5 for $r' \in \mathcal{N}(r_5)$ in presence of UR attack in region $r^a = r_6 \in \mathcal{N}(r_5)$ for attack size 0.7.

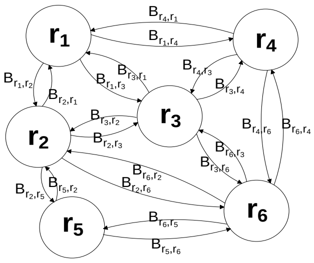


Figure 8: Markov chain based on BALs used for the attack localization.

6.1 Distributed localization and mitigation algorithm

We start with the definition of the beliefs of the individual regions, which is the basis for the localization algorithm.

Definition. *Let $\tilde{d}^{(k)}_{r,r'} = \alpha^{(k)} d^{(k)}_{r,r'} + (1-\alpha^{(k)})\tilde{d}^{(k-1)}_{r,r'}$ be the weighted moving average (WMA) of the MSD $d^{(k)}_{r,r'}$. The smoothing factor $\alpha^{(k)} \in (0,1)$ and satisfies $\sum_{k=0}^{\infty} \alpha^{(k)} = \infty$. The belief of attack direction of region r that its neighbor $r' \in \mathcal{N}(r)$ is the attacked region at iteration k is defined as*

$$B^{(k)}_{r,r'} = \frac{\tilde{d}^{(k)}_{r,r'}}{\sum\limits_{\forall r' \in \mathcal{N}(r)} \tilde{d}^{(k)}_{r,r'}}. \tag{14}$$

Observe that regions have beliefs only about their neighbors. i.e., $B^{(k)}_{r,r'} = 0, \forall r' \notin \mathcal{N}(r)$. Furthermore, the beliefs are not necessarily symmetric, i.e., $B^{(k)}_{r,r'} \neq B^{(k)}_{r',r}$ is possible.

Given the beliefs $B^{(k)}_{r,r'}$ of the regions, our goal is to find the region that is most likely to be compromised consistent with all beliefs. Before we introduce the distributed localization algorithm we describe a hypothetical localization scheme based on a global observer, which motivates the proposed algorithm.

Motivation: Assume there exists a token that the regions use to express their beliefs about the attack location: when region r receives the token, it will pass the token to region r' with probability $B_{r,r'}$. Moreover, assume that there exists a global observer that observes every passing of the token and that keeps count of how many times the token visits each region. The observer uses the counts to calculate for every region the empirical frequency of token visits: the number of token visits to the region divided by the number of token visits to all regions. It then identifies the region $\hat{r}^a$ with the highest empirical frequency as the most likely compromised region.

This hypothetical token passing scheme defines a random walk on a graph: the vertices are the regions and there is an edge between vertices r and r' if $B_{r,r'} > 0$. The random walk can then be modeled by a Markov chain. Fig. 8 shows the Markov chain for the interconnected system in Figure 2. The state transition matrix $B^{(k)}$ of the Markov chain is the right stochastic $R \times R$ matrix in which every row and every column corresponds to a region, and the entries of the matrix are the beliefs of attack direction $B^{(k)}_{r,r'}$ $\forall r' \in \mathcal{R}$. Thus, row r contains the beliefs of region r. Under appropriate conditions, which we will discuss later, the empirical frequency computed by the global observer converges to the stationary distribution $\pi^{(k)}$ of the Markov chain, which satisfies $\pi^{(k)}B^{(k)} = \pi^{(k)}$ [23]. Consequently, $\hat{r}^{a(k)} = argmax_r \pi^{(k)}$.

The Belief Consensus Localization (BCL) Algorithm:

The *BCL* algorithm with convergence threshold ε^L consists of five steps executed by the regional control centers.

1. Flood the MSDs $d^{(k)}_{r,r'}$ so that every region obtains all MSDs in the system. A flooding protocol, such as the one used in OSPF [24] can be used for this purpose.
2. Every region verifies that $d^{(k)}_{r,r'} = d^{(k)}_{r',r}$ $\forall r \in \mathcal{R}, r' \in \mathcal{N}(r)$.
3. Compute the beliefs $B^{(k)}_{r,r'}$, $\forall r \in \mathcal{R}, r' \in \mathcal{N}(r)$ according to (14). Construct the state transition matrix $B^{(k)}$.
4. Compute the stationary distribution $\pi^{(k)}$, the solution to $\pi^{(k)}B^{(k)} = \pi^{(k)}$.
5. If $||\pi^{(k)} - \pi^{(k-1)}||_\infty < \varepsilon^L$ then $k^L = k$. BCL reached convergence, $\hat{r}^{a(k^L)} = argmax_r \pi^{(k^L)}$.

Figure 9: Pseudo-code of the BCL Algorithm

Observe that due to Step 2 the attacker cannot tamper with the MSDs sent from region r^a without being noticed, and as a consequence all regions obtain the same matrix $B^{(k)}$ in Step 3. In what follows we show that the proposed *BCL* algorithm is correct, i.e., all regions identify the same region $\hat{r}^{a(k^L)}$ and the algorithm leads to a solution.

Proposition 3. *Consider a system with $R > 2$ regions. If (i) there exists a 3-clique in the graph $G = (\mathcal{R}, E)$ where $E = \{e_{r,r'} | r \in \mathcal{R}, r' \in \mathcal{N}(r)\}$, and (ii) for finite k the DSE does not converge, then the stationary distribution $\pi^{(k)}$ exists, it is unique and it can be computed.*

Proof. For sufficiently small k the disagreements between neighboring regions $d^{(k)}_{r,r'} > 0$, because of the initial disagreements on the shared state variables and because of the lack of synchronization to the reference bus. Consequently, the moving average $\tilde{d}^{(k)}_{r,r'} > 0$ since $\alpha^{(k)} > 0$, and so are the beliefs $B^{(k)}_{r,r'} > 0$, $\forall r, r'$ s.t. $r \in \mathcal{N}(r')$. This implies that the state transition diagram of the Markov chain described by $B^{(k)}$ is a symmetric directed graph, and thus all states of the Markov chain lie in a single communicating class, i.e., the chain is irreducible. Since the Markov chain is irreducible, it has a stationary distribution [23, Proposition 1.14] and this distribution is unique [23, Corollary 1.17]. Although $B^{(k)}_{r,r} = 0$ $\forall r \in \mathcal{R}$, for $R > 2$ condition (i) ensures that the Markov chain is aperiodic. Aperiodicity in turn is a sufficient condition for the (irreducible) Markov chain to converge to its stationary distribution [23, Theorem 4.9], i.e., the chain is ergodic. Since all regions obtain the same matrix $B^{(k)}$, and the stationary distribution $\pi^{(k)}$ is unique, all regions obtain the same distribution $\pi^{(k)}$. □

The above proposition shows that after a particular iteration k the *BCL* algorithm is correct. Nonetheless, the stationary distribution $\pi^{(k)}$ is a function of the matrix $B^{(k)}$, which can change at every iteration k. The following proposition establishes the convergence of $\pi^{(k)}$, which implies that the BCL algorithm eventually terminates.

Proposition 4. *If $\alpha^{(k)} \to 0$ as $k \to \infty$, then $\pi^{(k)} - \pi^{(k-1)} \to \mathbf{0}_{1\times R}$. Furthermore, if the attacked system state follows an asymptotically periodic orbit then the stationary distributions $\pi^{(k)}$ converge in k to a stationary distribution vector π^*, and $\hat{r}^{a(k)} \to \hat{r}^{a*}$.*

Proof. We start the proof of Proposition 4 by formulating the following lemma based on results in [25], which will allow us to prove the first part of the proposition ($\pi^{(k)} - \pi^{(k-1)} \to \mathbf{0}_{1\times R}$).

Lemma 5. *Let C be a right stochastic matrix that describes an irreducible Markov chain with stationary distribution vector $\pi_C = \pi_C C$, and let Π_C be the matrix with the same size as C and all columns equal π_C. Let us denote by $Z = [I - C + \Pi_C]^{-1}$ the fundamental matrix of C. Furthermore, let D be another right stochastic matrix that describes an irreducible Markov chain, and is sufficiently close to C so that all eigenvalues of the differential matrix $U = [D - C]Z$ are strictly less than unity in magnitude. Then*

$$\pi_D = \pi_C + \sum_{n=1}^{\infty} \pi_C U^n, \tag{15}$$

and consequently $\pi_D - \pi_C \to \mathbf{0}_{\mathbf{1}\times|\pi_\mathbf{C}|}$ as $D - C \to \mathbf{0}_{|\pi_\mathbf{C}|\times|\pi_\mathbf{C}|}$, where $|\pi_C|$ denotes the number of elements in the vector π_C.

Observe that by definition (14)

$$B^{(k)}_{r,r'} - B^{(k-1)}_{r,r'} = \alpha^{(k)} \frac{d^{(k)}_{r,r'} \sum\limits_{r''\in\mathcal{N}(r)} \tilde{d}^{(k-1)}_{r,r''} - \tilde{d}^{(k-1)}_{r,r'} \sum\limits_{r'\in\mathcal{N}(r)} d^{(k)}_{r,r''}}{\left(\sum\limits_{r''\in\mathcal{N}(r)} \tilde{d}^{(k)}_{r,r''}\right)\left(\sum\limits_{r''\in\mathcal{N}(r)} \tilde{d}^{(k-1)}_{r,r''}\right)}.$$

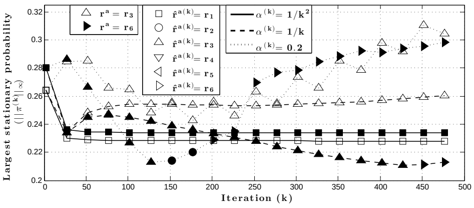


Figure 10: Evolution of the largest probability in the stationary probability vector $\pi^{(k)}$ for six different scenarios. Three smoothing factors $\alpha^{(k)} = 1/k^2$, $\alpha^{(k)} = 1/k$ or $\alpha^{(k)} = 0.2$, and two attack locations $r^a = r_3$ and $r^a = r_6$. The attack size is 0.05.

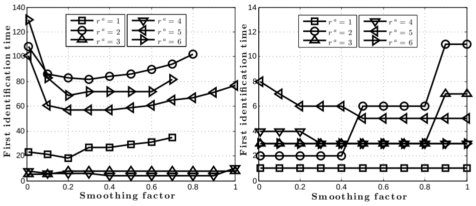


Figure 11: The first identification time (k^F) as a function of the smoothing factor ($\alpha^{(k)}$) considering the attacks in every region $\forall r \in \mathcal{R}$ individually with attack size 0.1 (left) and 0.5 (right).

Consequently $B^{(k)} - B^{(k-1)} \to \mathbf{0}_{|\pi^{(k)}|x|\pi^{(k)}|}$ as $\alpha^{(k)} \to 0$. Let $C = B^{(k-1)}$ and $D = B^{(k)}$. Since the Markov chains described by $B^{(k)}$ and $B^{(k-1)}$ are irreducible (c.f. Proposition 3), the conditions of Lemma 5 are satisfied for k big enough, and thus $\pi^{(k)} - \pi^{(k-1)} \to \mathbf{0}_{|\pi|}$. Consider now the orbit of the system state for large k. If the attacked system state follows an asymptotically periodic orbit then the disagreements follow an asymptotically periodic orbit too. The smoothing factor by definition satisfies $\sum_{k=0}^{\infty} \alpha^{(k)} = \infty$, and thus if it also satisfies $\alpha^{(k)} \to 0$ then the smoothed disagreements $\tilde{d}_{r,r'}^{(k)}$ converge to the mean disagreement of the limiting periodic orbit, and so do the beliefs $B_{r,r'}^{(k)} \to B_{r,r'}^{*}$. Consequently $\pi^{(k)} \to \pi^*$ and $\hat{r}^{a(k)} \to \hat{r}^{a*}$. □

Observe that the proposition does not hold for constant $\alpha^{(k)}$, but it does hold, for example, if $\alpha^{(k)} = 1/k$.

The mitigation algorithm uses BCL to identify the region $\hat{r}^{a*}$ that is most likely to be attacked, isolates the region, and reruns the DSE for the remaining regions until the DSE eventually converges.

6.2 Numerical results

Fig. 10 shows the evolution of the largest element of the stationary probability vector $\pi^{(k)}$ for different values of the smoothing factor $\alpha^{(k)}$ ($1/k^2$, $1/k$, 0.2), each considered in a separate scenario. We considered two attacked regions, $r^a = r_3$ and $r^a = r_6$; the attack size is 0.05 for which the DSE does not converge. In the case of $\alpha^{(k)} = 1/k^2$, the largest element of $\pi^{(k)}$ converges relatively quickly. The fast convergence of $\pi^{(k)}$ compared to $\alpha^{(k)} = 1/k$ and $\alpha^{(k)} = 0.2$ does, however, come at a price: the identified region $\hat{r}^{a(k)} = argmax_r \pi^{(k)}$ is not the attacked region, for both $r^a = r_3$ and $r^a = r_6$ region $\hat{r}^{a(k)} = r_1$ is erroneously identified as attacked. Observe that $\alpha^{(k)} = 1/k^2$ does not satisfy the condition $\sum_{k=0}^{\infty} \alpha^{(k)} = \infty$ required in the definition of $\tilde{d}_{r,r'}^{(k)}$ in Section 6.1, and shows the importance of the condition.

For $\alpha^{(k)} = 1/k$ and $\alpha^{(k)} = 0.2$, which do satisfy the condition, the largest element of $\pi^{(k)}$ converges slower, but the attacked region is correctly identified ($\hat{r}^{a(k)} = r^a$) eventually.

Although convergence cannot be guaranteed for a constant smoothing factor $\alpha^{(k)}$, because the condition $\alpha^{(k)} \not\to 0$ in Proposition 4 is not satisfied, a constant weighting factor is nevertheless useful for exploring the impact of smoothing on the localization time. Since in this case the stationary distribution vector $\pi^{(k)}$ may not converge, there may not exist a k^L for which $||\pi^{(k^L)} - \pi^{(k^L-1)}||_\infty < \varepsilon^L$. Still, after some number of iterations k^F the algorithm can correctly identify $\hat{r}^{a(k^F)}$ as the attacked region, that is, $\hat{r}^{a(k^F-1)} \neq r^a$, but $\hat{r}^{a(k^F)} = r^a \; \forall k \geq k^F$. We refer to k^F as the *first identification time*.

Fig. 11 shows the first identification time k^F as a function of $\alpha^{(k)}$ considering an attack in various regions $r \in \mathcal{R}$ for attack size 0.1 (left) and 0.5 (right). The first identification time depends on the region that is attacked as well as on the attack size: for larger attack size the localization is significantly faster (localization time is lower). For most of the regions, the optimal $\alpha^{(k)}$ is in the range $(0.2, 0.3)$ and a very high $\alpha^{(k)} > 0.7$ may even make localization fail for the smaller considered attack size (0.1). This observation supports that a small smoothing factor is in general preferable, even if it may lead to a larger localization time.

7 Conclusion

We addressed the vulnerability of fully distributed state estimation to data integrity attacks. We considered an attacker that compromises the communication infrastructure of a single control center and can manipulate the state variables exchanged between the control center and its neighbors. We showed that a denial of service attack can be launched against a state of the art state estimator this way. We proposed an attack detection algorithm based on the convergence properties of the distributed state estimation algorithm and based on the evolution of the exchanged state variables. Furthermore, we proposed an attack mitigation algorithm based on the consensus of the beliefs of the individual regions about the attack location, formulated as the stationary distribution of a random walk on a graph. We established existence, uniqueness, and convergence of the stationary distribution. We showed the efficiency of the attack detection and mitigation algorithms via simulations on an IEEE benchmark power system, and we used the simulations to illustrate the trade-off between localization speed and localization accuracy. Our numerical results also show that strong attacks can often be localized and mitigated faster than weak attacks.

Acknowledgments

This work was partially funded by the EIT ICTLabs through activity ASES 13030 and by the Swedish Research Council through project 2010:5812.

References

[1] A. Monticelli. Electric power system state estimation. *Proc. of the IEEE*, 88(2):262–282, 2000.

[2] Ali Abur and Antonio Gomez Exposito. *Power System State Estimation: Theory and Implementation*. Marcel Dekker, Inc., 2004.

[3] Yao Liu, Peng Ning, and Michael Reiter. False data injection attacks against state estimation in electric power grids. In *Proc. of the 16th ACM conference on Computer and Communications Security (CCS)*, pages 21–32, 2009.

[4] A. Teixeira, G. Dán, H. Sandberg, and Karl H. Johansson. A cyber security study of a SCADA energy management system: Stealthy deception attacks on the state estimator. In *Proc. IFAC World Congress*, Aug. 2011.

[5] Rakesh B. Bobba, Katherine M. Rogers, Qiyan Wang, Himanshu Khurana, Klara Nahrstedt, and Thomas J. Overbye. Detecting false data injection attacks on dc state estimation. In *Preprints of the First Workshop on Secure Control Systems, CPSWEEK*, Stockholm, Sweden, April 2010.

[6] Oliver Kosut, Liyan Jia, Robert Thomas, and Lang Tong. Malicious data attacks on smart grid state estimation: Attack strategies and countermeasures. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[7] Gy. Dán and Henrik Sandberg. Stealth attacks and protection schemes for state estimators in power systems. In *Proc. of IEEE SmartGridComm*, Oct. 2010.

[8] Ognjen Vuković, Kin Cheong Sou, György Dán, and Henrik Sandberg. Network-aware mitigation of data integrity attacks on power system state estimation. *IEEE JSAC: Smart Grid Communications Series*, 30(6):176–183, July 2012.

[9] T. T. Kim and H. V. Poor. Strategic protection against data injection attacks on power grids. *IEEE Trans. on Smart Grid*, 2:326–333, Jun. 2011.

[10] Annarita Giani, Eilyan Bitar, Manuel Garcia, Miles McQueen, Pramod Khargonekar, and Kameshwar Poolla. Smart grid data integrity attacks: Characterizations and countermeasures. In *Proc. of IEEE SmartGridComm*, Oct. 2011.

[11] M. Shahidehpour and Y. Wang. *Communication and Control in Electric Power Systems*. John Wiley and Sons, 2003.

[12] Antonio J. Conejo, Sebastian de la Torre, and Miguel Canas. An optimization approach to multiarea state estimation. *IEEE Transactions on Power Systems*, 22(1):213–221, February 2007.

[13] Le Xie, Dae-Hyun Choi, Soummya Kar, and H. Vincent Poor. Fully distributed state estimation for wide-area monitoring systems. *IEEE Transactions on Smart Grid*, 3(3):1154–1169, September 2012.

[14] Xiao Li and Anna Scaglione. Robust decentralized state estimation and tracking for power systems via network gossiping. *IEEE Journal on Selected Areas in Communications*, 31(7):1184–1194, July 2013.

[15] Vassilis Kekatos and Georgios B. Giannakis. Distributed robust power system state estimation. *IEEE Transactions on Power Systems*, 28(2):1617–1626, 2013.

[16] Ognjen Vuković and György Dán. On the security of distributed power system state estimation under targeted attacks. In *Proc. of ACM Symposium on Applied Computing (SAC)*, March 2013.

[17] Yangyue Feng, Chiara Foglietta, Alessio Baiocco, Stefano Panzieri, and Stephen D. Wolthusen. Malicious false data injection in hierarchical electric power grid state estimation systems. In *Proc. of the Fourth International Conference on Future Energy Systems*, e-Energy ’13, pages 183–192, 2013.

[18] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, July 1982.

[19] M. J. Fischer, N. A. Lynch, and M. Merritt. Easy impossibility proofs for distributed consensus problems. In *Proc. of the ACM symposium on Principles of distributed computing*, pages 59–70, 1985.

[20] N. H. Vaidya and Vijay K. Garg. Byzantine vector consensus in complete graphs. In *Proc. of the ACM Symposium on Principles of distributed computing*, pages 65–73, July 2013.

[21] R.A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1990.

[22] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

[23] David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. *Markov Chains and Mixing Times*. American Mathematical Society, 2009.

[24] R. Coltun, D. Ferguson, J. Moy, and A. Lindem. OSPF for IPv6. RFC 5340, RFC Editor, July 2008.

[25] Paul J. Schweitzer. Perturbation theory and finite markov chains. *Journal on Applied Probability*, 5(2):401–413, August 1968.

Paper D

Confidentiality-preserving Obfuscation for Cloud-based Power System Contingency Analysis

Ognjen Vuković, György Dán, and Rakesh B. Bobba.

in Proc. of IEEE SmartGridComm, October 2013.

Confidentiality-preserving Obfuscation for Cloud-based Power System Contingency Analysis

Ognjen Vuković and György Dán
Laboratory for Communication Networks
School of Electrical Engineering
KTH Royal Institute of Technology, Stockholm, Sweden
Email: {vukovic,gyuri}@ee.kth.se

Rakesh B. Bobba
University of Illinois at Urbana-Champaign
Urbana, IL, USA.
Email: rbobba@illinois.edu

Abstract

Power system operators are looking to adopt and migrate to cloud technologies and third-party cloud services for customer facing and enterprise IT applications. Security and reliability are major barriers for adopting cloud technologies and services for power system operational applications. In this work we focus on the use of cloud computing for Contingency Analysis and propose an approach to obfuscate information regarding power flows and the presence of a contingency violation while allowing the operator to analyze contingencies with the needed accuracy in the cloud. Our empirical evaluation shows, i) that the errors introduced into power flows due to the obfuscation approach are small, and ii) that the RMS errors introduced grow linearly with the magnitude of obfuscation.

1 Introduction

Power grids around the world are undergoing a transformation to accommodate more renewable generation, allow consumer interaction with the infrastructure, and improve efficiencies through modernization. At the heart of this transformation are new sensor deployments, such as smart meters and phasor measurement. These new sensors are producing large volumes of data that a power system operator has to process and store, and increasing the number of devices that a power utility has to connect to and manage. To manage the data and connectivity to these devices utilities are looking to cloud based services. Smart meters in particular, given their numbers (in Millions even for medium size Utility) and geographic distribution, pose a challenge. Responding to this demand, many companies (*e.g.,* GE's GRID IQ, Honeywell's Akuacom, AutoGrid *etc.*) are offering cloud-based software-as-a-service models to manage smart meters and associated applications such as automated Demand Response (DR). Apart from customer facing applications such as Demand Response, utilities are also looking into leveraging cloud computing for other services such as managing security of their infrastructure as evidenced by the new CIGRE working group (D2.37) on cloud technologies for managed security [1]. The primary drives towards cloud computing are lower costs, improved efficiencies and elasticity of computing provided.

Power system applications related to operations, such as Contingency Analysis, forecasting, Optimal Power Flow, *etc.*, could also benefit from the advantages cloud technologies provide [2]). Security and reliability concerns are, however, a major barrier for adopting cloud technologies for power system operations [2,3], especially with third-party providers. Recent work has addressed this issue from two sides. First, by improving the reliability and security provided by the cloud infrastructure for power grid applications, as is being done in the GridCloud [4,5] project. Second, by transforming power system applications to preserve security properties such as confidentiality, integrity and availability in third party infrastructures. Borden *et al.,* [6] focus on transforming the optimal power flow problem before instantiating it in the cloud to preserve confidentiality.

In this paper we focus on contingency analysis (CA), which is a core application in power system operation. A contingency corresponds to the failure of one or more system components, such as a transmission line, a transformer, or a generator. The failure of any of these components would lead to a change in the power flows on the transmission lines, and could potentially result in an unstable system (e.g., power flows that exceed the thermal capacity of transmission lines). The aim of contingency analysis is to determine whether the power system would be unstable in case any of a potentially large set of contingencies would happen.

Contingency analysis is performed in modern energy management systems every time a new state estimate becomes available as a result of state estimation - as often as every few minutes. The number of contingencies that needs to be considered depends on the instantaneous load of the power system, the higher the load the more contingencies might have to be considered. The number of contingencies considered in practice is limited by the computational power available in the control center, and is often constrained to considering the loss of single components known as $N-1$ security. Cloud-based contingency analysis could allow an operator to scale the number of considered contingencies freely as a function of the instantaneous system state and enable $N-x$ security that is considered desirable, but it could expose the current system state and possible critical contingencies, thereby facilitating targeted attacks.

In this paper we propose an algorithm to obfuscate information regarding power flows and the presence of a contingency violation while allowing the operator to analyze contingencies with the needed accuracy in the cloud. We show that our approach doesn't introduce any error for CA using DC model. Further our empirical evaluation shows that the error introduced by the approach when using an AC model is quite small and that RMS error grows linearly with the magnitude of obfuscation applied.

The rest of the paper is organized as follows. Section 2 provides necessary background on contingency analysis. Section 3 presents our adversary model and usage scenario and Section 4 describes our obfuscation approach. Section 5 discusses some preliminary evaluation results and Section 6 concludes the paper.

2 Background

We consider a power system that consists of N buses. We denote by P_n, $1 \leq n \leq N$ the power injection (load or generation) at bus n, and P_I is the vector of power injections. We denote the state of the power system by x. For simplicity, we consider active power flows only, in which case the system state is determined by the phase angles at the buses, and thus x is the vector of phase angles.

Given the system state x, the power flow between buses n and m can be computed as

$$P_{nm} = V_n V_m (G_{nm} \cos x_{nm} + B_{nm} \sin x_{nm}) = f_{nm}(x_{nm}), \tag{1}$$

where $x_{nm} = x_n - x_m$ is the phase angle difference between buses n and m, and G_{nm} and B_{nm} are the real and imaginary parts of the bus admittance matrix corresponding to buses n and m. The power injections can be computed using Kirchhoff's nodal law, and we denote the power injections as a function of the system state by $P_I = f_I(x)$. Finally, one can express the vector of power injections and power flows as a function of the system state as $P = f(x)$

2.1 AC Load-flow based Contingency Analysis

Let c be a contingency (e.g., the failure of two transmission lines), and let f^c be the function that describes the power flows under contingency c as a function of the system state, i.e., $P^c = f^c(x)$. Observe that a contingency might change the system topology and thus $f^c(.) \neq f(.)$. Similarly, the vector of power injections P_I^c under contingency c might be different from P_I, e.g., if the contingency involves the loss of one or more generators. To describe the relationship between the power injections before and after the contingency we introduce the fault matrix F^c such that $P_I^c = F_I^c P_I$. If contingency c does not affect the power injections then F_I^c is the identity matrix.

Given the vector of power injections P_I^c under contingency c, contingency analysis requires the solution of the load-flow problem, i.e., finding the state vector x^c that solves $P_I^c = f_I^c(x^c)$. The state vector is obtained through solving the power balance equations,

$$\Delta P_n \overset{d}{=} -P_n + \sum_m P_{nm} = 0. \tag{2}$$

Since the sum of the injections over all buses is zero, there are in total $N-1$ power balance equations and $N-1$ unknowns, as the phase angle of the reference bus is set to zero.

The equations (1) are non-linear, thus the solution to (2) is obtained using an iterative numerical method, typically the Newton-Raphson method [7]. Starting from an initial guess $x^c(0)$, the Newton-Raphson method obtains an updated estimate at iteration k by computing

$$\Delta x^c(k+1) = -J_k^{-1}\Delta P_I(k), \tag{3}$$

where $J_k = \frac{\partial P_I}{\partial x}|_{x=x^c(k)}$ is the Jacobian evaluated at the most recent guess $x^c(k)$, and then letting $x^c(k+1) = x^c(k) + \Delta x^c(k+1)$. Observe that the Jacobian is a non-singular square matrix of size $(N-1) \times (N-1)$. The algorithm terminates when the power mismatch ΔP_I is below a certain threshold. Let x^c be computed system state under contingency c.

Given the system state x^c under the contingency, the power flows can be calculated as $P^c = f^c(x^c)$. If any of the power flows exceeds the capacity limit (e.g., thermal capacity) of the transmission line then the system is said to be in an insecure state, and a corrective action must be taken by the operator to move the system to a state in which no contingency results in a capacity violation.

3 Adversary Model and Scenario

3.1 Adversary Model

We assume that the adversary has knowledge about the topology of the system but that he doesn't have access to the current state of the system. That is he does not know what the instantaneous power injections and power flows are. This adversarial model is inline with the recent body of work on false data injection attacks (*e.g.,* [8–10]) where the adversary is assumed to have full or partial knowledge of the H matrix for a DC model.

The goal of the adversary is to find the current system state (flows and injections) so he can determine if there are any contingencies with critical violations. Correspondingly, the goal of the obfuscation algorithm is to mask the real power flows from the adversary and to hide the existence of a violating contingency.

3.2 Usage Scenario

As shown in Figure 1, when a power system operator wants to undertake CA he will create an obfuscation vector and send the system with obfuscated flows to the cloud for contingency analysis. On obtaining the result of the CA for the various contingencies, the operator performs a deobfuscation step to obtain the power flows and injections that correspond to the non-obfuscated (actual) system. While obfuscation is performed only once, deobfuscation is performed for every contingency. Nevertheless, much of the computation of the deobfuscation can be done a-priori for a particular system topology.

4 Obfuscated Contingency Analysis

In the following we first introduce the proposed obfuscation algorithm. We then show that for DC load flow calculation the proposed obfuscation does not introduce an error.

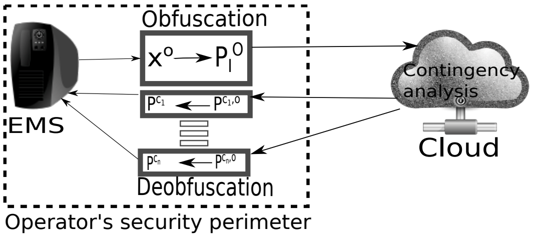

Obfuscation
$x^o \rightarrow P_I^O$
EMS
$P^{c_1} \leftarrow P^{c_1,O}$
$P^{c_n} \leftarrow P^{c_n,O}$
Deobfuscation
Operator's security perimeter
Contingency analysis
Cloud

Figure 1: Considered scenario: Obfuscation is done once before contingency analysis is performed in the cloud, deobfuscation is done for all results.

4.1 Obfuscation Algorithm

Consider the known power injections P_I^c under a contingency c. If an adversary has access to the power injections P_I^c and the computed power flows P^c under the contingency, it can infer which part of the system is most critical for stability and could perform a targeted attack. It is therefore important to obfuscate the information exposed to a potential attacker.

In the following we propose an algorithm that limits the attacker's ability to infer potential system instability. We do so by obfuscating the system state on which contingency analysis is performed, and by compensating the contingent system for the modification after contingency analysis is performed. The important property of the proposed algorithm is that the computational cost of the obfuscation and of the de-obfuscation is much less than that of the contingency analysis.

4.1.1 Obfuscation

Given P, the actual power flows in the system, obfuscation consists of adding a randomly chosen vector of power flows to the actual power flows. We refer to the latter as the *power flow obfuscation* vector,

$$P^o = Hx^o, \tag{4}$$

where $H = \frac{\partial P}{\partial x}$ is the Jacobian evaluated at the most recent system state (prior to the CA), and x^o is a non-negative vector of phase angles, the *state obfuscation* vector. We will discuss in Section 4.3 how to create the state obfuscation vector. We use the state obfuscation vector to create the obfuscated system state, $x^O = x + x^o$. The obfuscated system state can be used to compute the obfuscated power injections as

$$P^O = f(x^O) \tag{5}$$

The obfuscated system state, and the corresponding obfuscated power injections $P_I^O = f_I(x^O)$ are the basis for the contingency analysis performed in the cloud.

For a particular contingency c, the obfuscated power injections $P_I^{c,O}$ are created, and are used as the input to the non-linear load-flow problem. The solution to the load-flow problem, i.e., the result of the analysis for contingency c is the state $x^{c,O}$ of the obfuscated contigent system.

4.1.2 Deobfuscation for a Contingency

Given the result $P^{c,O} = f_c(x^{c,O})$ of the contingency analysis performed on the obfuscated power flows for contingency c, deobfuscation consists of compensating for the power flows introduced through obfuscation.

To describe deobfuscation we define $H_c = \frac{\partial P^c}{\partial x}$, the Jacobian of the system under contingency c evaluated at the most recent system state (as in (4)). The deobfuscated power flows under contingency c are then obtained as

$$\tilde{P}^c = P^{c,O} - H_c J_c{}^{-1} P_I^{c,o}, \tag{6}$$

where $P_I^{c,o}$ is the vector of obfuscation power injections under contingency c. Note that if the contingency involves the loss of a generator then at least one or two entries in $P_I^{c,o}$ are changed and thus $P_I^{c,o} \neq P_I^o$.

Due to the non-linearity of the power balance equations, obfuscation will introduce an error in the result of the contigency analysis. We quantify this error by the difference of the power flows under a contingency with and without obfuscation

$$e_P = P^c - \tilde{P}^c. \tag{7}$$

To express the relative error we furthermore define the maximum componentwise relative error

$$\varepsilon_P = \max_i \frac{e_{P,i}}{P_i^c}, \tag{8}$$

where P_i^c is the i^{th} component of the vector P^c.

4.2 Correctness under DC Load Flow-based CA

In the following we consider DC load flow computation and show that if contingency analysis is performed using DC load flow then the proposed obfuscation algorithm does not affect the result of the contingency analysis, i.e., the error e_P is zero.

The DC load flow model is based on the observation that in a system in normal operation the angular separation along any transmission line is small. This allows one to obtain a linear approximation for (1) of the form

$$P_{nm}^{DC} = V_n V_m (B_{nm} x_{nm}), \tag{9}$$

If one further considers that the per-unit voltages are approximately equal to one, then the power balance equations can be written as

$$\Delta P_n^{DC} \stackrel{d}{=} -P_n + \sum_m B_{nm} x_{nm} = 0. \tag{10}$$

Observe that due to the linearity of the power balance equations in the DC power flow model, the load flow problem for power injection vector P_I can be solved as $x = J^{-1} P_I$.

Proposition 1. *Under DC load flow based contingency analysis the error introduced through obfuscation $e_P = \mathbf{0}$, where $\mathbf{0}$ is the vector of all zeros.*

Proof. Consider the error e_P introduced by obfuscation in the result of the contingency analysis, as defined in (7),

$$\begin{aligned} e_P &= P^c - \tilde{P}^c \\ &= P^c - (P^{c,O} - H_c {J_c}^{-1} P_I^{c,o}) \\ &= H_c {J_c}^{-1} F_I^c Jx - (H_c {J_c}^{-1} P_I^{c,O} - H_c {J_c}^{-1} F_I^c P_I^o) \\ &= H_c {J_c}^{-1} F_I^c Jx - (H_c {J_c}^{-1} F_I^c (P_I + P_I^o) - H_c {J_c}^{-1} F_I^c P_I^o) \\ &= H_c {J_c}^{-1} F_I^c Jx - (H_c {J_c}^{-1} F_I^c (Jx + Jx^o) - H_c {J_c}^{-1} F_I^c Jx^o) \\ &= \mathbf{0}, \end{aligned}$$

where ${J_c}^{-1} P_I^O = x^O$ because of (4). □

Note that the proof relies on the linearity of the power balance equations in the DC model, which implies that the DC load-flow problem can be solved in one iteration. Thus, the proof does not hold for AC load-flow based contingency analysis.

4.3 Choosing the Obfuscation Vector

In order to make obfuscation suitable for AC load-flow based contingency analysis, the choice of the obfuscation vector should be such that obfuscation does not introduce a significant error in the result of the contingency analysis, thus obfuscation should not be too big. At the same time, obfuscation should be big enough to hide the actual power flows from an attacker in the following sense. On the one hand, it should be ambiguous for an attacker whether a contingency exists in the actual system in case a critical contingency exists in the obfuscated system. On the other hand, if there is no critical contingency in the obfuscated system, the attacker can be aware of that there is no critical contingency in the actual system either, as this information cannot be used against the system.

The above two requirements imply that the power flow obfuscation P^o has to be bounded, and the obfuscation should have maximal entropy. We use the following result from [11] to construct the maximum entropy distribution.

Lemma 1. *Fix real numbers $a < b$ and $\mu \in (a,b)$. The continuous probability density function on the interval $[a,b]$ with mean μ that maximizes entropy among all such densities (on $[a,b]$ with mean μ) is a truncated exponential density*

$$q_\alpha(x) = \begin{cases} C_\alpha e^{\alpha x} & if x \in [a,b] \\ 0 & otherwise \end{cases} \tag{11}$$

where C_α is chosen so that $\int_a^b C_\alpha e^{\alpha x} dx = 1$, and α is the unique real number such that $\int_a^b x C_\alpha e^{\alpha x} dx = \mu$.

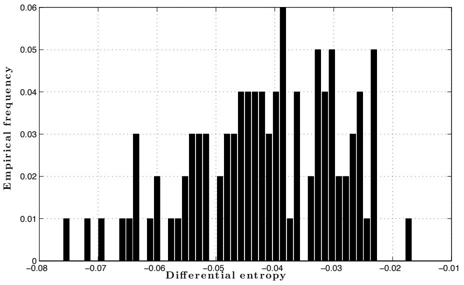


Figure 2: Histogram of the differential entropy of the relative obfuscation power flows for $u_{max} = 0.1$, computed over 100 runs.

For $\alpha = 0$ *the distribution is uniform on* $[a,b]$*, and its differential entropy is*

$$h(X) = \int_a^b q_0(x)\log q_0(x)dx = \log(b-a). \tag{12}$$

Proof. We refer to [11] for the proof. □

As our objective is to obfuscate the power flows, we define the obfuscation vector in terms of the obfuscation power flows P^o, and use the uniform distribution for obfuscation. We thus define the diagonal matrix U with diagonal elements $U_{i,i} \sim U(0,0.1)$, and create the vector

$$\hat{P}^o = UP. \tag{13}$$

This vector cannot be used directly for the obfuscation because it does not necessarily correspond to any system state. We therefore perform a linearized state estimation on this vector to obtain the state obfuscation vector

$$x^o = (H^T H)^{-1} H\hat{P}^o, \tag{14}$$

where H^T is the transpose of H. Note that the components of x^o do not follow a uniform distribution, but the components of the power flow obfuscation vector $P^o = Hx^o$ are likely close to uniform (relative to the actual power flows). Numerical results presented in Section 5 show that this is indeed the case.

5 Performance Evaluation

In the following we illustrate the efficiency of the proposed algorithm via simulations.

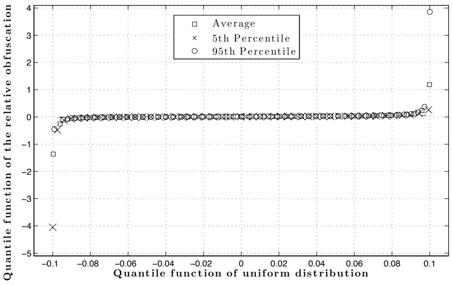


Figure 3: QQ plot of the distribution of the relative obfuscation power flows and injections P^o/P for $u_{max} = 0.1$ vs. a uniform distribution $U(0, 0.1)$, computed over 100 runs.

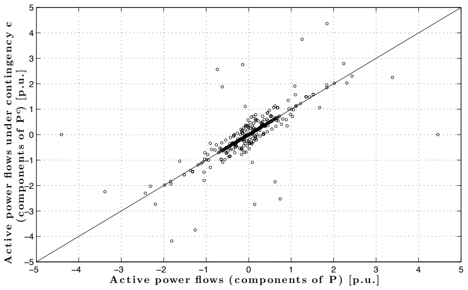


Figure 4: Active power flows after the contingency vs. before the contingency.

5.1 Simulation methodology

We used the IEEE 118 bus test system and and used Matpower for the AC load flow based CA. The power flows and injections are represented using p.u, where 1 p.u. equals to 100MW. For the obfuscation, we considered all active power injections and all active power flows, both 'to' and 'from' buses (hence negative values in the Figures).

5.2 Obfuscation performance

We first consider the performance of the algorithm in terms of the obfuscation it provides. Note that the level of obfuscation does not depend on the particular contingency considered, it depends on the system topology and the actual system state. These results are thus general for the IEEE 118 bus system.

Figure 2 shows a histogram of the differential entropy of the relative obfuscation power flows, i.e., that of P^o/P in a component-wise sense, computed over 100 randomly chosen obfuscation power flows for $u_{max} = 0.1$. We approximated the differential entropy by creating a histogram with 200 bins and using the histogram bins width for numerical integration. The differential entropy of $U(0, 0.1) \approx -3.2$, thus aligning the power flows with the range space of the Jacobian in (14) does alter the distribution of the power flows, but it does not decrease its entropy. In fact, the obfuscation of some power flows by far exceeds $u_{max} = 0.1$, which is the reason for the significantly higher entropy than with the uniform distribution.

Figure 3 shows the QQ plot of the distribution of $P^o = Hx^o$ defined in (14) normalized by P, compared to a uniform distribution on $[0, 0.1]$, computed over 100 randomly chosen obfuscation vectors. Recall that the components of $\hat{P}^o$ follow a uniform distribution, but due to (14) the components of P^o do not necessarily do so. Figure 3 shows that the distribution of P^o indeed differs from uniform, especially at the tails, which is also the reason for the increased differential entropy, as discussed above. At the same time, the body of

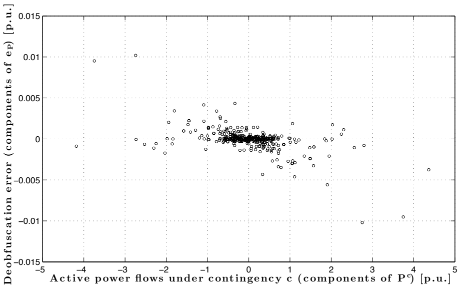


Figure 5: Error introduced by obfuscation for $u_{max} = 0.1$ vs. the power flows under contingency obtained with regular CA.

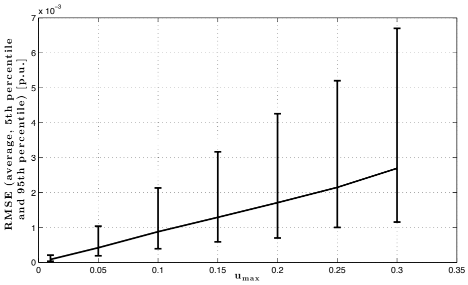


Figure 6: Impact of domain of the uniform distribution $U(0, u_{max})$ on the average root mean square error of obfuscated CA, mean, 5 and 95 percentiles.

the distribution is close to uniform. The figure also shows that there is not much difference between the individual obfuscation vectors, as the percentiles are rather close to the average.

These two figures indicate the choice of the obfuscation vector P^o provides a good level of randomness thus making it hard for an adversary to guess the real power flows. We now turn to the evaluation of the error introduced by obfuscation for AC load flow-based CA.

5.2.1 Obfuscation vs. CA accuracy

In the following we consider a contingency that affects branch 9, which is a transmission line that connects buses 9 and 10. The effect of the contingency on active power flows is shown in Figure 4. The figure shows that the pre-contingency power flow on branch 9 is above 4 p.u., and is the largest power flow in the system, thus the scenario corresponds to a severe contingency.

Figure 5 shows a scatter plot of the error vs. the power flows under the considered contingency after deobfuscation: the power flows P^c obtained without the proposed scheme are shown on the horizontal axis, and the errors e_P remaining in the corresponding power flows $\hat{P}^c$ after deobfuscation are shown on the vertical axis. Thus, every dot shown corresponds to an error in a power flow or a power injection. The results shown were obtained for $u_{max} = 0.1$. The figure shows that the errors introduced by obfuscation are small, all dots are located close to zero, which corresponds to no error, i.e., $e_P = \mathbf{0}$. The figure thus shows that the errors are very small compared to the actual power flows.

Figure 6 shows the average root mean square error (RMSE) introduced in the result of the CA by obfuscation as a function of the upper bound u_{max} of the uniform distribution used for obfuscation in (13). The average RMSE is defined as $\frac{||e_P||_2}{|e_P|}$, where $|.|$ is the number of components in the vector. For every u_{max} value the figure shows the mean over 100 simulations together with the 5 and 95 percentiles. The figure shows that the average

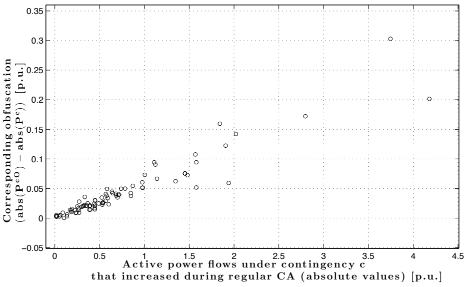


Figure 7: Obfuscated power flows under the contingency vs. power flows that increased due to the contingency with the regular CA. All obfuscated power flows exceed the actual power flows.

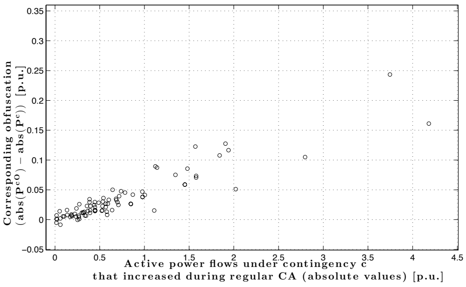


Figure 8: Obfuscated power flows under the contingency vs. power flows that increased due to the contingency with the regular CA. Most of the obfuscated power flows exceed the actual power flows.

RMSE increases approximately linearly over a wide range of u_{max} values, and so do the percentile values. The average RMSE is very small compared to the actual power flows in the system, which confirms that obfuscated CA would be viable.

Figures 7 and 8 show the difference between the obfuscated power flows $P^{c,O}$ and the power flows obtained without obfuscation P^c for two different obfuscation vectors x^o, but only for those power flows that increase due to the contingency. The vertical axis is thus effectively the introduced obfuscation. Both figures show that the amount of obfuscation grows with the power flow, but the actual values differ because they depend on the obfuscation vector x^o. The two obfuscation vectors used for Figure 7 and for Figure 8 were chosen from the considered 100 obfuscation vectors for $u_{max} = 0.1$ so as to represent two different scenarios in terms of the signs of the introduced obfuscations per flow. In the first scenario (Figure 7), all power flows that increased due to the contingency without obfuscation have a positive amount of obfuscation, while in the second scenario (Figure 8), there are a few relatively small power flows for which the obfuscation is negative. Power flows that have a negative obfuscation are determined by the obfuscation vector x^o, which is unknown to the attacker. Consequently, by just observing $P^{c,O}$, an attacker cannot be certain how much obfuscation is introduced and for which flows the obfuscation is negative. Thus, the fact that there is a thermal capacity violation in the obfuscated system does not imply that it is also the case after de-obfuscation, and thus an attacker that observes a violating contingency based on $P^{c,O}$ cannot be certain that there is a violating contingency in the actual system, according to P^c.

6 Conclusion

We proposed an approach to obfuscate information regarding power flows to enable CA in the cloud while allowing the operator to obtain accurate post contingency flows. Our approach doesn't introduce any error for CA using a DC model and our numerical results show that the error introduced when using AC models is tolerable. It is subject of our future work to extend the obfuscation algorithm so that it always introduces positive obfuscation to the power flows that increase due to contingency. Furthermore, our future work will include analytically bounding the error introduced by the proposed obfuscation and an analytical characterization of the randomness of the obfuscation vector.

Acknowledgements

This material is based upon work supported in part by the Department of Energy under Award Number DE-OE0000097[1].

References

[1] [Online]. Available: http://d2.cigre.org/WG-Area/D2.37-Guidelines-for-outsourcing-managed-security-services-using-Cloud-Technologies

[2] G. Dan, R. B. Bobba, G. Gross, and R. H. Campbell, "Cloud Computing for the Power Grid: From Service Composition to Assured Clouds," in *In Proceedings of 5th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud '13)*, June 2013.

[3] K. P. Birman, L. Ganesh, and R. van Renesse, "Running smart grid control software on cloud computing architectures," Workshop on Computational Needs for the Next Generation Electric Grid, April 2010.

[4] K. Maheshwari, M. Lim, L. Wang, K. Birman, and R. van Renesse, "Toward a reliable, secure and fault tolerant smart grid state estimation in the cloud," *IEEE PES Innovative Smart Grid Technologies*, 2013.

[5] K. Maheshwari, K. Birman, J. M. Wozniak, and D. Van Zandt, "Evaluating cloud computing techniques for smart power grid design using parallel scripting," in *IEEE/ACM International Symposium On Cluster, Cloud And Grid Computing (CC-Grid)*, 2013.

[6] A. R. Borden, D. K. Molzahn, P. Ramanathan, and B. C. Lesieutre, "Confidentiality-preserving optimal power flow for cloud computing," in *Allerton Control Conference*, 2012.

[1] This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

[7] B. Stott, “Review of load-flow calculation methods,” *Proc. of the IEEE*, vol. 62, no. 7, pp. 916–929, 1974.

[8] Y. Liu, P. Ning, and M. Reiter, “False data injection attacks against state estimation in electric power grids,” *ACM Transactions in Information and Systems Security (TISSEC), 2011*, vol. 14, no. 1, pp. 13:1–13:33, June 2011.

[9] R. B. Bobba, K. M. Rogers, Q. Wang, H. Khurana, K. Nahrstedt, and T. J. Overbye, “Detecting false data injection attacks on dc state estimation,” in *1st Workshop on Secure Control Systems (SCS ’10)*, 2010.

[10] G. Dán and H. Sandberg, “Stealth attacks and protection schemes for state estimators in power systems,” in *First IEEE International Conference on Smart Grid Communications (SmartGridComm), 2010*, Oct. 2010, pp. 214 –219.

[11] K. Conrad, “Probability distributions and maximum entropy,” 2012. [Online]. Available: http://www.math.uconn.edu/~kconrad/blurbs/analysis/entropypost.pdf

Paper E

Mitigating Gray Hole Attacks in Industrial Communications using Anonymity Networks: Relationship Anonymity-Communication Overhead Trade-off

Ognjen Vuković, György Dán, and Gunnar Karlsson.

Submitted to IEEE Transactions on Parallel and Distributed Systems.

Mitigating Gray Hole Attacks in Industrial Communications using Anonymity Networks: Relationship Anonymity-Communication Overhead Trade-off

Ognjen Vuković, György Dán, and Gunnar Karlsson
ACCESS Linnaeus Center, School of Electrical Engineering,
KTH Royal Institute of Technology, Stockholm, Sweden
Email: {vukovic,gyuri,gk}@ee.kth.se

Abstract

Gray hole attacks are a significant threat to mission critical communication infrastructures, such as industrial control systems. They are relatively easy to perpetrate, as an attacker that has access to communication links or equipment could observe the source and destination addresses for every message, and can identify and discard the messages exchanged between particular communication participants. Anonymity networks could render these attacks more difficult by providing anonymous communication via relaying. Nevertheless, relaying introduces overhead as it increases end-to-end message delivery delay and introduces additional traffic, which both in practice must often be low. Hence, an important question is how to optimize anonymity for limited overhead. In this paper we address this question by studying two anonymity networks: MCrowds, an extension of Crowds, which provides unbounded communication delay and Minstrels, which provides bounded communication delay. We derive exact analytical expressions for the relationship anonymity for these systems. Using MCrowds and Minstrels we show that, contrary to intuition, increased overhead does not always improve anonymity. We investigate the impact of the system's parameters on anonymity and on the optimal anonymity network parameters, and the sensitivity of anonymity to the misestimation of the number of attackers.

1 Introduction

Many modern industrial systems, such as electric power systems and smart grids, require high communication availability between a fixed set of nodes on a pairwise basis [1, 2]. The nodes can be remote sensors, controllers and operation centers in a centralized wide area monitoring and controlling system, e.g., supervisory control and data acquisition (SCADA) system, or local operation and control units in a fully distributed monitoring and controlling system e.g., many emerging paradigms in smart grid such as microgrids [3] and distributed state estimation [4, 5]. Cryptography may provide authentication, confidentiality and data integrity for the communication, but source and destination addresses would still be visible to an outside attacker who is able to observe one or more network links. The outside attacker may identify traffic patterns: who is communicating with whom, when and how often. Using this information the attacker can infer the importance of messages, and may perform a targeted message dropping attack, i.e. gray hole attack, on the communication between any two nodes. It may, for example, drop messages carrying important status or control information and cause incorrect system operation, e.g., it can destabilize a modern industrial control system [6, 7, 8].

Gray hole attacks could be mitigated by using anonymity networks that provide relationship anonymity, i.e., make it untraceable who communicates with whom [9], and therefore make targeted message dropping more difficult. Relationship anonymity against an outside attacker can be provided by a set of mixes [10] that relay messages in such a way that an outside attacker cannot link an outgoing message with an incoming message, and therefore ensures sender-receiver unlinkability against an eavesdropper observing communication links. While relaying renders outside attacks more difficult, it introduces the possibility of inside attacks. Due to the often long life-cycles of industrial systems, software corruption is a threat and due to the complexity of the code-base it is hard to detect. Corrupted nodes that are part of the mix network can perform inside attacks to determine the sender-receiver pair for messages that are relayed through them, and perform the gray hole attack. Therefore, certain anonymity networks, such as [11, 12], also provide some level of relationship anonymity against inside attackers by hiding the sender or the receiver from the relay nodes. Unfortunately, good sender (or receiver) anonymity in itself does not necessarily lead to good relationship anonymity [13], hence we focus on relationship anonymity in this paper.

The relationship anonymity provided by anonymity networks comes at the price of increased overhead: end-to-end delivery delay as well as total network traffic are increased due to relaying. Excessive delays can negatively impact the system performance, while increased traffic leads to high resource requirements, so that in practice both have to be kept low. At the same time, the relationship anonymity may be a function of the number of nodes in the system and the number of nodes controlled by the attacker. Since the number of attacker nodes is unknown, finding the optimal level of overhead can be challenging.

We consider an attacker that wants to perform a gray hole attack on the communication between a particular pair of nodes by dropping the messages that they exchange. As we use anonymity networks to defend against such attacks, the attacker first performs traffic analysis, i.e., estimates the likelihood of that a message is exchanged between the targeted pair of nodes, and then decides to drop the message based on the outcome of the analysis (the likelihood). We compare our attack model to a more traditional approach used in many other studies on relationship anonymity, where the attacker aims to classify every observed message with the sole goal of identifying who is communicating with whom, e.g., [11, 12, 14, 13, 15]. The difference between the two models is important in anonymity networks where the receiver keeps forwarding the message in order to increase overall anonymity. We consider two methods for traffic analysis: the *Maximum posteriori method* and the *Bayesian inference method*. According to the *Maximum posteriori method* the attacker only considers the most likely pairs of nodes as possible sender-receiver pairs for an observed message. According to the *Bayesian inference method* the attacker considers all pairs of nodes as possible sender-receiver pairs for an observed message.

We consider two anonymity networks that provide relationship anonymity. First, MCrowds, a modification of Crowds [12], which provides anonymity by introducing unbounded message delivery delay. MCrowds provides sender anonymity using the same mechanism as Crowds, which was shown to provide optimal sender anonymity for given average path length [15], but, unlike Crowds, it also hides the receiver among a small subset of anonymity network nodes. Second, Minstrels, which provides relationship anonymity by introducing bounded message delivery delay. Bounding the path length is achieved by limiting the number of visited nodes for each message. We use these two anonymity networks to investigate the inherent trade-off between the introduced overhead and the level of provided relationship anonymity. While intuition says that increased overhead should result in better anonymity, our results show that this is not necessarily the case for either the delay overhead or the traffic overhead. The results also show that larger anonymity networks provide better relationship anonymity for the same ratio of attacker nodes. Moreover, we show that it is in general better to overestimate the number of attacker nodes when choosing the level of overhead.

The rest of the paper is organized as follows. In Section 2, we discuss the related work. Section 3 describes our system model, the attack model, the anonymity metric, and the traffic analysis methods. Section 4 describes the MCrowds and Minstrels anonymity networks. In Section 5, we develop analytical models of the relationship anonymity provided by MCrowds and Minstrels, and we show numerical results based on the models in Section 6. Section 7 concludes the paper.

2 Related Work

Early works on traffic analysis attacks against anonymity networks by an external global attacker considered long term intersection attacks [13, 16, 17]. These attacks exploit the distribution of message destinations to decrease the relationship anonymity by relying on cases when the sender's anonymity is not *beyond suspicion*, i.e., the sender is distinguishable from other nodes. Disclosure attacks considered in [18] formulate traffic analysis as an optimization problem, under more general assumptions. More recent works have formulated traffic analysis attacks by an external global adversary in the context of Bayesian inference [13, 19, 20]. These attacks consider that the receiver is outside the anonymity network. In our system the sender and the receiver are part of the anonymity network, and message destinations can have an arbitrary distribution. We use Bayesian inference, but we consider an internal adversary instead of an external global observer.

The relationship between anonymity and traffic overhead was investigated in [21] for a global adversary. The authors considered an anonymity network in which routes have a fixed length, and padding (i.e., dummy traffic) is sent over links to hide traffic patterns. In our work the overhead is measured in terms of message delivery delay, quantified by the route length, and in terms of additional traffic introduced due to relaying. Furthermore, the adversary cannot observe the global traffic, only traffic traversing compromised nodes. Sender anonymity in the presence of compromised nodes was considered for Crowds [15] and for systems inspired by Crowds [21]. In our work, we consider relationship anonymity instead of sender anonymity, and address the trade-off between anonymity and overhead.

The effects of targeted denial-of-service (DoS) attacks on reliability of various anonymity networks and on relationship anonymity they provide was studied in [22]. The authors considered an attacker in control of a number of relays that deny to forward messages for which they cannot identify the sender-receiver pair, and therefore, causing the sender to keep resending the messages over different paths until either the attacker identifies the sender-receiver pair or no attacker relay is visited. The attack leverages the fact that in the considered anonymity networks the sender or the receiver are distinguishable from the relay nodes, e.g., Tor [14] and Mixminion [23] use dedicated nodes as relays, or the attack requires the receiver to be also compromised in order to be able to identify the sender-receiver pair, e.g., in the case of Cashmere [24]. In our work, we consider a closed system with predetermined nodes that act as senders, receivers and relays for each-others messages, and therefore, both sender and receiver are not easily distinguishable from relay nodes. Furthermore, as our goal is to protect against gray hole attacks, we consider that the receiver is not compromised.

Related to our work are also studies on DoS attacks [25], particularly DoS attacks in industrial control systems [26, 27, 7, 28, 29]. In [25], the authors present taxonomies for classifying DoS attacks and defenses in any networked system. DoS attacks against industrial control systems can significantly degrade the performance of such systems [27], and even destabilize them, e.g., power systems in [29]. There have been a number of techniques proposed for detection of DoS attacks caused by malicious communication nodes flooding the network with packets to cause congestion [25, 26, 27]. To protect against such attacks, the system can identify the source of the attack, i.e., the flooding node, and filter the traffic coming from the node at the point where the traffic enters the network [25, 26, 27]. In the case of DoS attacks that result in packet loss on links, e.g., due to link jamming or gray hole attacks, the system can optimize the control loop in order to decrease effects of the attacks [7, 28]. In our work, we protect the system against gray hole attacks by using anonymity networks: anonymity networks make the attacker uncertain about the sender-receiver pair for the messages it observes, and therefore, renders the targeted message dropping much more difficult.

3 System model and metrics

We consider an anonymity network that consists of a set $\mathscr{N}$ of nodes, $N = ||\mathscr{N}||$. The nodes act as *sources*, *destinations* and as *relay* nodes for each others' messages, and do not send messages to themselves over the anonymity network. The underlying communication network is a complete graph: messages can be exchanged between any two nodes without visiting other nodes. We consider that encryption and authentication are done end-to-end between the sender and the receiver, but the relay nodes do not perform cryptographic operations on the messages in order to limit their computational burden.

We use a and b to denote any two nodes in the network ($a \in \mathcal{N}, b \in \mathcal{N} \setminus \{a\}$). We use $(a \to b)$ to denote a sender-receiver pair, where a represents the sender, i.e., the node that originates a particular message, and b represents the receiver, i.e., the node for which the message is intended. We denote by $S = a$ the event that the sender is node a and by $R = b$ the event that the receiver is node b.

3.1 Attack Model

We consider an *inside attacker* that is in control of a set $\mathcal{C} \subset \mathcal{N}$ ($C = ||\mathcal{C}||$) of compromised nodes. The attacker can observe the messages traversing the nodes in $\mathcal{C}$ and the protocol specific information contained in the messages. It can make use of the payload of the messages to recognize if the same message visits several compromised nodes. The attacker has an *a-priori* belief of the system traffic matrix in the form of the distribution $T(S = a, R = b)$ for every pair of nodes $(a \to b)$. Entry $T(S = a, R = b)$ of the traffic matrix is the message sending rate from a to b normalized by the total message rate. For example, the distribution T could be uniform if the attacker has no a-priori knowledge of the actual traffic matrix. Based on its a-priori belief and based on the information contained in an observed message, the attacker calculates its a-posteriori belief $B(S = a, R = b)$ for every message it observes. It uses this belief to perform an attack against the communication between a targeted sender-receiver pair ($s \in \mathcal{N}$), which we refer to as the *targeted s-r pair*. We consider the following two attacks against the targeted s-r pair.

Classification attack: The aim of the attacker is to determine if and how often the targeted s-r pair communicates, i.e., if and how often s sends messages to r. For every message it observes, the attacker classifies that the message belongs to $(s \to r)$ with a probability that is a function of the belief $B(S = s, R = r)$, i.e., with probability $g(B(S = s, R = r))$.

Gray hole attack: The aim of the attacker is to perform a gray hole attack on the communication between the targeted s-r pair. In principle, the attacker could drop every message that gets relayed over the nodes $\mathcal{C}$ it controls to maximize the effect of the attack, but then such an attack could be detected easier as no message would ever been successfully relayed over the nodes in $\mathcal{C}$. Instead, for every message it observes, the attacker decides whether to drop or to continue relaying the message based on its belief $B(S = s, R = r)$ that the message is sent by s to r. The attacker decides to drop the message with probability $g(B(S = s, R = r))$.

For both attacks, the attacker needs to compute the belief $B(S = s, R = r)$, which is used as the input to the function $g()$. However, there is an important difference between these two attacks in anonymity networks where: (i) the receiver continues forwarding messages in order to improve overall anonymity, or (ii) the receiver receives messages at the same time as a number of other nodes, e.g., the receiver is in a multicast group. In such anonymity networks, the analysis of messages that are observed for the first time after, or at the same time when, they are received by the receiver benefits the attacker only in the case of classification attack. In the case of gray hole attack, analyzing and dropping these messages would not affect the communication.

3.2 Overhead and Anonymity Metrics

We define four metrics the communication overhead introduced by the anonymity network and the ability of the anonymity network to protect the system against the classification and the gray hole attacks.
Delay overhead: We quantify the delay overhead by the average number of nodes $E[K^d]$ that an arbitrary message visits until it reaches (including) the receiver.
Traffic overhead: We quantify the traffic overhead by the average number of nodes $E[K^t]$ that a message visits in total. Note that $E[K^d]$ does not equal $E[K^t]$ if the receiver forwards the message further or the anonymity network uses multicast or broadcast.

We consider two metrics that quantify the efficiency of the anonymity network to protect the system against the two attacks: the relationship anonymity, against the attacker whose aim is to classify every message it observes, and the message delivery ratio, against the attacker that aims to perform a gray hole attack.
Relationship Anonymity: We quantify the relationship anonymity by the probability that a message that belongs to the targeted s-r pair will not be correctly classified to belong to $(s \to r)$ by the attacker. For a message not intercepted by the attacker, the probability equals 1.
Message Delivery Ratio: We quantify the message delivery ratio by the probability that a message that belongs to the targeted s-r pair will not be dropped by the attacker before it reaches the receiver, i.e., the probability that it will be successfully delivered. In essence, the message delivery ratio is the average false-negative rate considering that false-negative is 1 for messages not observed by the attacker before the message reaches the receiver. In the rest of this paper, we will refer to the message delivery ratio as the delivery ratio for the sake of brevity.

The relationship anonymity and the delivery ratio depend on three factors. First, on the probability of having an attacker node on the path. In the case of the delivery ratio, the path includes the nodes visited before the message reaches the receiver and it includes the receiver, while in the case of the relationship anonymity, the path includes all nodes that the message visits. Second, on the attacker's a-posteriori belief $B(S = s, R = r)$ that the message belongs to the targeted s-r pair. Third, on the function $g()$. The first two factors are functions of the anonymity protocol, the number of nodes N and the number of inside attacker nodes C. The function $g(B(S = s, R = r))$ depends on the method used by the attacker.

3.3 Attack Methods

We consider two attack methods, which define the function $g()$, used by the attacker for the classification and the gray hole attack.

3.3.1 Maximum posteriori method

Using the Maximum Posteriori (MP) method, when the attacker intercepts a message, it uses its belief to populate the set $\mathcal{Q} = \{(a \rightarrow b) : B(S = a, R = b) \geq B(S = a', R = b'), \forall a, a', b, b' \in \mathcal{N} \setminus \mathcal{C}\}$ of most likely sender-receiver pairs. If $(s \rightarrow r) \in \mathcal{Q}$ then the attacker correctly classifies (the relationship anonymity) or correctly drops (the delivery ratio) the message with probability $1/||\mathcal{Q}||$. The set $\mathcal{Q}$ may be a singleton, $||\mathcal{Q}|| = 1$, in which case the relationship anonymity and the delivery ratio are likely to be low, but it may just as well contain all possible sender-receiver pairs, $||\mathcal{Q}|| = (N - C) \cdot (N - C - 1)$, which would correspond to perfect relationship anonymity and delivery ratio. Note that some $(a \rightarrow b) \in \mathcal{Q}$ does not imply that $(a \rightarrow b)$ is the actual sender-receiver pair, not even when $||\mathcal{Q}|| = 1$. Thus, $g(B(S = s, R = r)) = 1/||\mathcal{Q}||$ if $(s \rightarrow r) \in \mathcal{Q}$, and $g(B(S = s, R = r)) = 0$ otherwise.

We denote by H the event when an attacker node observes the message. Note that H happens if any of the following two mutually exclusive events occurs. (i) the message visits an attacker node for the first time before it visits the receiver, we denote this event by H^r, or (ii) the message does not visit any attacker node before it reaches the receiver, but it visits an attacker node at the same time as the receiver (e.g., multicast or broadcast) or at some point after it visited the receiver (e.g., as a relay after the message leaves the receiver), we denote this event by H^{nr}. Thus, $H = H^r \cup H^{nr}$

We use this notation to express the relationship anonymity under the MP method as

$$A_{MP}(s \rightarrow r) = 1 - \frac{P((s \rightarrow r) \in \mathcal{Q} | H, S = s, R = r)}{||\mathcal{Q}||} \cdot P(H | S = s, R = r). \qquad (1)$$

The delivery ratio under the MP method $D_{MP}(s \rightarrow r)$ can be expressed similarly to $A_{MP}(s \rightarrow r)$ in (1), with the difference that only the cases when the message is observed before the receiver are considered, i.e., the event H^r is considered instead of the event H.

3.3.2 Bayesian inference method

Using the Bayesian Inference (BI) method, $g(B(S = s, R = r)) = B(S = s, R = r)$. Unlike under the MP method, the attacker may drop or classify a message even if $(s \rightarrow r)$ is not the most likely sender-receiver pair.

Using the above notation we can express the relationship anonymity under the BI method as

$$A_{BI}(s \rightarrow r) = 1 - B(S = s, R = r | H, S = s, R = r) \cdot P(H | S = s, R = r). \qquad (2)$$

The delivery ratio under the BI method $D_{BI}(s \rightarrow r)$ can be expressed similarly to $A_{BI}(s \rightarrow r)$ in (2), with the difference that the event H^r is considered instead of the event H.

Observe that the attacker may incorrectly classify (to belong to $(s \rightarrow r)$) or drop messages for which $S \neq s$ or $R \neq r$: with probability $1/||\mathcal{Q}|$ if $(s \rightarrow r) \in \mathcal{Q}$ under the MP method, or with probability $B(S = s, R = r)$ under the BI method. However, these cases do not contribute to neither the relationship anonymity nor the delivery ratio; they are considered as false-positive errors and are evaluated as such in Section 6. It is clear that these errors do not affect the communication between the targeted s-r pair, and consequently, do not degrade the probability that messages that belong to $(s \to r)$ will be successfully delivered, i.e., the delivery ratio. For the classification attack, these errors might mislead the attacker in its goal to determine if the targeted s-r pair is communicating and how often. However, for the sake of comparison to the delivery ratio and for the sake of simplicity of the metric, the relationship anonymity considers only the true positives: the attacker correctly classifies messages that belong to $(s \to r)$.

4 Anonymity system descriptions

In the following we describe the two considered anonymity networks: MCrowds and Minstrels.

4.1 MCrowds system description

MCrowds is an anonymity network inspired by Crowds [12], which was proven to provide optimal sender anonymity [15]. In MCrowds the sender specifies a set $\mathscr{M}$ of nodes as receiver for a message. The number $M = ||\mathscr{M}||$ of receiver nodes is a system parameter. Nodes specified in the set $\mathscr{M}$ are not used for relaying. For a message to reach its intended receiver r it must be that $r \in \mathscr{M}$; the other $M-1$ nodes are chosen uniformly at random. The sender then relays the message to one of the $\mathscr{N} \setminus \mathscr{M}$ nodes (including itself) selected uniformly at random. A relay node relays the message with probability p_f to one of the $\mathscr{N} \setminus \mathscr{M}$ nodes chosen uniformly at random. Note that a node can relay the message to itself, in which case the message does not leave the node. Otherwise, the message is sent as a multicast message to all receiver nodes specified in $\mathscr{M}$ (i.e., with probability $1-p_f$). Upon multicasting, the receiver set is removed from the message. Node r recognizes that it is the receiver while the other $\mathscr{M} \setminus \{r\}$ nodes discard the message. For $M = 1$ MCrowds is equivalent to Crowds, except that the receiver node is part of the anonymity network, $r \in \mathscr{N}$. In principle the nodes could use different values of M and p_f, but to ease the analysis we consider that all nodes use the same parameter values.

4.2 Minstrels system description

Minstrels uses nodes as message relays in the same way as Crowds, but it design ensures that the number of nodes visited by a message is bounded.

When a node s wants to send a message to a node r it picks a node uniformly at random among the other $N-1$ nodes (excluding s) and forwards the message. The next node forwards the message to one of the other $N-2$ nodes (excluding itself and the sender node s) chosen uniformly at random. Every subsequent forwarder picks one of the non-visited nodes to forward the message. When node r receives the message, it will send the message

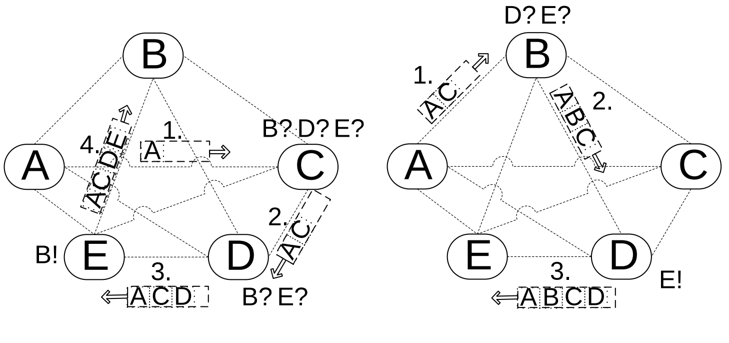


Figure 1: A simple example of Minstrels with five nodes.

further in order to improve the receiver anonymity. The path ends when all N nodes have been visited.

The message, or part of it, is encrypted with the receiver's public key. When a node receives the message, it checks whether it is the receiver by trying to decrypt the encrypted part of the message. If the decrypted part of the message represents valid data, the node is the receiver. Note that a node does not know who the receiver is, but it can check whether it is the receiver itself.

To bound the path length, every message records the set $\mathcal{V}$ of the visited nodes in its header. The set can be implemented, for example, using a Bloom filter, to keep its size small. When a relaying node receives a message, it adds itself to the set $\mathcal{V}$ and relays the message to one of the remaining non-visited nodes. To control the maximum path length (i.e., delay) the sender can initialize the set $\mathcal{V}$ of visited nodes with a number $f \in \{0,...,N-2\}$ of the nodes in the system. These initialized nodes are considered as visited so that the message can not be relayed to them. A message traverses all nodes except for the initialized nodes in the set $\mathcal{V}$ and hence the sender must not include the receiver in the set $\mathcal{V}$. The sender picks the number of initialized nodes at random: it initializes the set with f nodes with probability $P(F=f)$, where $\sum_{f=0}^{N-2} P(F=f) = 1$. For $f=0$ the set is empty, for $f=1$ the set is initialized only with the sender and for $f>1$ the set is initialized with the sender and $f-1$ other nodes. Note that for $f>0$, the sender always includes itself in the set. The distribution of F is a system parameter, and we use it to explore the anonymity-overhead trade-off. In principle the nodes could use different distributions for F, but again, to ease the analysis we consider that all nodes use the same distribution.

Fig. 1 shows two simple examples with five nodes, node A as sender and node D as receiver. Fig. 1 (left) shows a case when the set $\mathcal{V}$ is initialized with the sender node A and the message is forwarded to node C. Node C checks if it is the receiver, puts itself in the set and chooses the next hop uniformly at random among nodes (B,D,E). The next hop, node D, follows the same procedure with only two forwarding options (B,E). Fig. 1 (right) shows another case when the set $\mathcal{V}$ is initialized with the sender and node C, and the message is forwarded to node B. Node B adds itself to the set and decides to which of the remaining nodes (D,E) to forward the message. Node C is considered as already visited.

5 Overhead and Anonymity

In the following we derive expressions for the communication overhead (delay and traffic), for the relationship anonymity and for the delivery ratio provided in face of inside attackers by MCrowds and by Minstrels.

5.1 Communication Overhead

We start with calculating the communication overhead of MCrowds and of Minstrels. For MCrowds, the delay overhead $E[K^d]$ and the traffic overhead $E[K^t]$ are functions of the expected value of a geometric distribution with success probability $1-p_f$, i.e., of the expected number of visited relays excluding the first mandatory relay. The delay overhead equals the expected value plus two more visits, the first mandatory relaying and the last hop to the receiver, and the traffic overhead equals the expected value plus the first mandatory relaying and the multicast messages, i.e.,

$$E[K^d] = \frac{p_f}{1-p_f} + 2, \text{ and } \quad E[K^t] = \frac{p_f}{1-p_f} + 1 + M, \tag{3}$$

respectively.

For Minstrels and for a given number f of initialized nodes in the set $\mathscr{V}$, the delay overhead is the average number of visited nodes before and including the receiver, which is uniformly distributed on $\{1,..,N-f\}$. The traffic overhead for given f always equals $N-f$. Thus, the delay overhead and the traffic overhead can be expresses as

$$\begin{aligned} E[K^d] &= \sum_{f=0}^{N-2} P(F=f) \cdot (N-f+1)/2, \text{ and} \\ E[K^t] &= \sum_{f=0}^{N-2} P(F=f) \cdot (N-f). \end{aligned} \tag{4}$$

Observe that for both MCrowds and Minstrels, $E[K^t]$ can be easily calculated from $E[K^d]$, and vice versa, given M and the distribution of F.

5.2 The Relationship Anonymity for MCrowds

We start the calculation of the relationship anonymity with expressing the probability of having an attacker node on the path as a relay. This probability depends on the number of receiver nodes M, and on the number of attacker nodes in the set $\mathscr{M}$ of receiver nodes.

Number of Initialized Attacker Nodes in the set $\mathscr{M}$

We denote by c_M the number of attacker nodes in the receiver set. c_M is a realization of the random variable $C_M \in \{max(0, M-(N-C-1)),...,min(M-1,C)\}$. For $M=1$ there cannot be attacker nodes in the receiver set, only the receiver r, and therefore $P(C_M=0)=1$. For $M>1$, the sender selects the other $M-1$ nodes uniformly at random from $N-2$

nodes (excluding the sender and the receiver). Thus, once k trusted and j attacker nodes have been selected, the next selected node is a trusted node with probability $\frac{N-C-2-k}{N-2-k-j}$, and is an attacker node with probability $\frac{C-j}{N-2-k-j}$. Observe that it does not matter in what order the c_M attacker nodes were selected, and thus the probability that there are c_M attacker nodes in the set of receiver nodes is

$$P(C_M = c_M) = \left(\begin{array}{c} M-1 \\ c_M \end{array} \right) \frac{\prod_{k=2}^{M-c_M}(N-C-k)\prod_{k=0}^{c_M-1}(C-k)}{\prod_{k=2}^{M}(N-k)}. \tag{5}$$

Attacker Node Occurs as Relay

Let us denote by H_i the event that the position of the first attacker node is i (it is the i^{th} relay). The event H_i happens if the message is first relayed $i-1$ times through trusted nodes, i.e., not through attacker nodes in the set $\mathscr{N} \setminus \mathscr{M}$, but the i^{th} relay is an attacker node. Since a message is relayed to one of the $C - c_M$ attacker nodes with probability $\frac{C-c_M}{N-M}$ and the sender must relay the message initially, conditioned on $C_M = c_M$ we have

$$P(H_i|c_M, S = a, R = b) = \frac{C-c_M}{N-M} p_f^{(i-1)} \left(1 - \frac{C-c_M}{N-M}\right)^{(i-1)}, \tag{6}$$

for $a \in \mathscr{N} \setminus (\mathscr{C} \cup \mathscr{M})$ and $b \in \mathscr{M} \setminus \mathscr{C}$. Note that for brevity we use c_M to denote the condition $C_M = c_M$ in (6) and henceforth. If the message is again relayed over an attacker node on any position after i, the attacker does not gain any additional information about the sender-receiver pair (s,r) of the message: any node from the set $\mathscr{N} \setminus \mathscr{M}$ is equally likely to be used as relay, and the receiver is still one of the nodes in $\mathscr{M}$. Hence, the probability assigned to the sender-receiver pair does not change. Thus, it is enough to focus on the position of the first attacker node on the path. Let us now denote by H_{1+} the event that there is an attacker on the path as a relay. Observe that for MCrowds, the event H_{1+} is the same as the event H^r. The event H_{1+} happens if the event H_i happens for any $i > 0$. Note that the events H_i $\forall i$ are mutually exclusive. Therefore, conditioned on $C_M = c_M$, the event H_{1+} happens with probability

$$P(H_{1+}|c_M, S = a, R = b) = \sum_{i=1}^{\infty} P(H_i|c_M, S = a, R = b) = \frac{C-c_M}{N-M-p_f(N-C-M+c_M)}. \tag{7}$$

This expression is obtained using the same approach as in [12], but considering that the number of attacker nodes is $C - c_M$ and that the total number or relaying nodes is $N - M$. We omit the derivation for brevity.

Predecessor Node

Consider now that there is an attacker on the path as a relay. When the first attacker node on the path gets the message, the attacker knows the nodes in the set $\mathscr{M}$, the number of attacker nodes c_M in the set, and the node that the message is received from, i.e., the predecessor p. Let us denote by I_a the event that the predecessor is node a ($p = a$), and by $\bar{I}_a$ the event that the predecessor is not node a ($p \neq a$).

If H_1 happens and thus the attacker node is on position $i = 1$, then the sender of the message is the predecessor and the event I_a happens if a is the sender. Otherwise, if H_{2+} happens, i.e., the attacker is at position $i > 1$, we have to distinguish two cases. If $S = a$ then any trusted node from the set $\mathcal{N} \setminus \mathcal{M}$ is equally likely to be the predecessor, and we have $P(I_a|H_{2+}, c_M, S = a, R = b) = \frac{1}{N-C-M+c_M}$ for any $b \in \mathcal{N} \setminus \mathcal{C}$ and $b \neq a$. If $S = s$ then I_a for $a \neq s$ can only happen if $a \notin \mathcal{M}$, but any $a \notin \mathcal{M}$ is equally likely to be the predecessor. The event $a \notin \mathcal{M}$ conditioned on $S = s$ $(a \neq s)$ happens with probability $P(a \notin \mathcal{M}|c_M, S = s, R = b) = \frac{N-C-M-c_M-1}{N-C-2}$, for any $b \in \mathcal{N} \setminus \mathcal{C}$ and $b \notin \{s, a\}$. Thus, $P(I_a|H_{2+}, c_M, S = s, R = b) = \frac{P(a \notin \mathcal{M}|c_M, S=s, R=b)}{N-C-M+c_M}$. Putting it all together, the event I_a conditioned on H_{1+} and $S = a$ $(s \neq a)$ happens with probability

$$\begin{aligned} P(I_a|H_{1+}, c_M, S = a, R = b) = P(H_1|c_M, S = a, R = b) + \\ P(I_a|H_{2+}, c_M, S = a, R = b) \cdot P(H_{2+}|c_M, S = a, R = b), \end{aligned} \tag{8}$$

and for $S = s$ with probability

$$P(I_a|H_{1+}, c_M, S = s, R = b) = P(I_a|H_{2+}, c_M, S = s, R = b) \cdot P(H_{2+}|c_M, S = s, R = b). \tag{9}$$

Anonymity with Attacker as Relay

Let us now consider the case when node s sends a message and the attacker appears as a relay, i.e., the events $S = s$ and H_{1+} happen. If node s is the predecessor (I_s) then the attacker's belief that node s is the sender of the message is

$$B(S = s|I_s, H_{1+}, c_M, S = s, R = b) = \frac{\sum_b P(I_s, H_{1+}, c_M|S = s, R = b) \cdot T(S = s, R = b)}{\sum_{(a,b)} P(I_s, H_{1+}, c_M|S = a, R = b) \cdot T(S = a, R = b)}, \tag{10}$$

where $a \in \mathcal{N} \setminus (\mathcal{M} \cup \mathcal{C})$ and $b \in \mathcal{M} \setminus \mathcal{C}$. Recall that $T(S = a, R = b)$ is the attacker's a-priori belief of the traffic matrix, which it uses as the probability that node a sends a message to node b. The attacker's belief $B(S = s|\bar{I}_s, H_{1+}, c_M, S = s, R = b)$ that node s is the sender when node s is not the predecessor ($\bar{I}_s$) can be expressed in a similar way.

Based on the above, a relaying attacker's belief that node s is the sender, given $S = s$, H_{1+} and $C_M = c_M$, is

$$\begin{aligned} B(S = s|H_{1+}, c_M, S = s, R = b) = \\ B(S = s|I_s, H_{1+}, c_M, S = s, R = b) P(I_s|H_{1+}, c_M, S = s, R = b) + \\ B(S = s|\bar{I}_s, H_{1+}, c_M, S = s, R = b) P(\bar{I}_s|H_{1+}, c_M, S = s, R = b). \end{aligned} \tag{11}$$

The attacker's belief that node r is the receiver equals $B(R = r|H_{1+}, c_M, S = s, R = r) = \frac{1}{M-c_M}$. Note that the events are conditionally independent since the receiver is one of the trusted nodes in $\mathcal{M}$, and the sender is one of the trusted nodes in $\mathcal{N} \setminus \mathcal{M}$. Hence, the attacker's belief that the targeted s-r pair is the sender-receiver pair is the product of the two beliefs.

Anonymity with no Attacker as Relay

Let us now consider the case when there is no attacker on the path. We denote by $\overline{H}_{1+}$ the event that a message does not visit any attacker node as a relay, the complement event of H_{1+}. If $\overline{H}_{1+}$ and $C_M = 0$ happen then the attacker does not observe the message. Otherwise, if $\overline{H}_{1+}$ happens but $C_M > 0$ then the attacker nodes in the receiver set $\mathscr{M}$ get the multicast message from the last relay node (the one that decides to send the message to the receivers with probability $1 - p_f$). If the events $\overline{H}_{1+}$ and $C_M > 0$ happen, that corresponds to the event H^{nr}. Observe that any trusted node from the set $\mathscr{N} \setminus \mathscr{M}$ is equally likely to be the last relay (the predecessor), and therefore for H^{nr} we have $\forall(a \to b)$

$$P(I_a|H^{nr}, S = a, R = b) = P(I_a|H_{2+}, c_M, S = a, R = b)$$

Consequently, given H^{nr}, and I_s or $\bar{I}_s$, the probability that the attacker assigns to node s being the sender can be expressed similar to (10). Finally, the attacker's belief $B(S = s|H^{nr}, S = s, R = b)$ that node s is the sender, given $S = s$ and H^{nr}, can be expressed using the law of total probability conditioned on I_s and $\bar{I}_s$, similar to (11).

Since the last relay node removes the receiver set $\mathscr{M}$ from the message, the receiver is hidden among $N - C - 1$ trusted nodes (it cannot be the last relay). However, the attacker's belief that node r is the receiver depends on whom the attacker guesses to be the sender. If the attacker believes that the predecessor is the sender, each of the other $N - C - 1$ trusted nodes is equally likely to be the receiver. Therefore, if I_s happens and the attacker assumes $S = s$ then it's belief that r is the receiver equals $B(R = r|S = s, I_s, H^{nr}, S = s, R = r) = \frac{1}{N-C-1}$. If the attacker believes that the predecessor is not the sender then each of the $N - C - 2$ trusted nodes apart from the predecessor and the sender is equally likely to be the receiver. Thus, if $\bar{I}_s$ happens and the attacker assumes $S = s$ then the attacker's belief that node r is the receiver equals $B(R = r|S = s, \bar{I}_s, H^{nr}, S = s, R = r) = \frac{1}{N-C-2}$. Thus, given H^{nr}, the attacker's belief that the targeted s-r pair is the sender-receiver pair is

$$\begin{aligned} &B(S = s, R = r|H^{nr}, S = s, R = r) = \\ &\qquad \frac{B(S = s|I_s, H^{nr}, S = s, R = r)}{N - C - 1} P(I_s|H^{nr}, S = s, R = r) + \\ &\qquad \frac{B(S = s|\bar{I}_s, H^{nr}, S = s, R = r)}{N - C - 2} P(\bar{I}_s|H^{nr}, S = s, R = r). \end{aligned} \tag{12}$$

Tying it all together

We are now ready to express the relationship anonymity under the BI method $A_{BI}(s \to r)$ using the law of total probability accounting for all possible values of C_M, and for all cases when the attacker receives the message, i.e., either H_{1+} (H^r) or $\overline{H}_{1+} \cup C_M = c_M > 0$ (H^{nr}),

$$\begin{aligned} &A_{BI}(s \to r) = 1 - (\\ &\quad \sum_{c_M} B(S = s, R = r|H_{1+}, c_M, S = s, R = r) \cdot P(H_{1+}|c_M, S = s, R = r) \cdot P(C_M = c_M) \\ &\quad + \sum_{c_M \neq 0} B(S = s, R = r|H^{nr}, S = s, R = r) \cdot P(\overline{H}_{1+}|c_M, S = s, R = r) \cdot P(C_M = c_M)). \end{aligned} \tag{13}$$

In order to calculate the relationship anonymity $A_{MP}(s \to r)$ under the MP method, we need to determine the probability that the sender-receiver pair $(s \to r)$ is one of the most likely sender-receiver pairs, i.e., $(s \to r) \in \mathscr{Q}$. This can be easily done for an arbitrary traffic matrix T given particular events, e.g., I_s and H_{1+}. In the special case when the attacker's a-priori belief is that the traffic matrix is homogeneous, all pairs $(a \to b)$ of trusted nodes are equally likely to be the sender-receiver pair. Hence, if either H_{1+} or H^{nr} happens, then the predecessor is the sole most likely sender. Therefore, $(s \to r) \in \mathscr{Q}$ only if I_s happens, and in this case every trusted node in the receiver set $\mathscr{M}$ is equally likely to be the receiver, thus $||\mathscr{Q}|| = M - c_M$.

5.3 Delivery Ratio for MCrowds

The analysis of the delivery ratio differs from the analysis of the relationship anonymity only in that it considers only the event H^r, i.e., H_{1+}, instead of both H^r and H^{nr}. Thus, the delivery ratio $D_{BI}(s \to r)$ under the BI method can be expressed as

$$\begin{aligned} D_{BI}(s \to r) = 1 - \sum_{c_M} & B(S = s, R = r | H_{1+}, c_M, S = s, R = r) \\ & \cdot P(H_{1+} | c_M, S = s, R = r) \cdot P(C_M = c_M) \end{aligned} \tag{14}$$

In order to calculate the delivery ration $D_{MP}(s \to r)$ under the MP method, we need to determine the probability $(s \to r) \in \mathscr{Q}$, and that can be done as in the case of $A_{MP}(s \to r)$ but not considering the event H^{nr}.

5.4 Relationship Anonymity and Delivery Ratio for Minstrels

When the first attacker node on the path gets the message, the attacker knows the number c_F of attacker nodes that the set of visited nodes was initialized with by the sender. c_F is a realization of the random variable C_F, whose distribution depends on the number f of initialized nodes in the set of visited nodes, $\mathscr{V}$.

In Minstrels the attacker's belief that the targeted s-r pair is the sender-receiver pair does not only depend on the node that the message is received from, i.e., the predecessor p, but also on the contents of the set $\mathscr{V}$ of visited nodes that the message carries. Consequently, the attacker distinguishes between three disjoint sets of nodes: the predecessor node $(\{p\})$, nodes in the set of visited nodes except the predecessor $(\mathscr{V} \setminus \{p\})$, and nodes not in the set of visited nodes $(\overline{\mathscr{V} \cup \{p\}})$. These sets form a partition of the set of all nodes in the system, and trusted nodes belonging to the same set are equally likely to be the sender (and the receiver). As a shorthand for the universe of distinguishable events we use the notation $\Omega_s = \{s = p, s \in \mathscr{V} \setminus \{p\}, s \in \overline{\mathscr{V} \cup \{p\}}\}$, where, for example, $s = p$ is the event that the predecessor is the sender. Similarly, we define $\Omega_r = \{r = p, r \in \mathscr{V} \setminus \{p\}, r \in \overline{\mathscr{V} \cup \{p\}}\}$ for the distinguishable events regarding the receiver. For the delivery ratio, only the cases when the message has not yet reached the receiver need to be considered, i.e., only the event $r \in \overline{\mathscr{V} \cup \{p\}}$ from Ω_r as that corresponds to the event H^r. When it comes to the relationship anonymity, all cases in Ω_r need to be considered.

If the message visits multiple attacker nodes on its path then the attacker can identify the nodes that were visited between the different attacker nodes. However, since any node that has not been visited yet is equally likely to be visited by the message, the attacker does not gain additional information that it could use to assign higher probability to the sender-receiver pair $(s \to r)$. Hence, it is enough to consider the first attacker node on the path that gets the message.

Given the information on $\mathscr{V}$, c_F, and p available to the attacker, we can use the law of total probability to expand (1) and (2) conditional on the size $||\mathscr{V}|| = v$ of the set of visited nodes, $\omega_s \in \Omega_s$,$\omega_r \in \Omega_r$, and $C_F = c_F$ to express the relationship anonymity as

$$A_{BI}(s \to r) = 1 - \sum_{c_F}\sum_{v}\sum_{\omega_s}\sum_{\omega_r}$$

$$B(S = s, R = r|\omega_r, \omega_s, c_F, H_{1+}, v, S = s, R = r) \tag{15}$$

$$\cdot P(\omega_r, \omega_s, c_F, H_{1+}, v|S = s, R = r), \tag{16}$$

$$A_{MP}(s \to r) = 1 - \sum_{c_F}\sum_{v}\sum_{\omega_s}\sum_{\omega_r}$$

$$\frac{P((s \to r) \in \mathscr{Q}|\omega_r, \omega_s, c_F, H_{1+}, v, S = s, R = r)}{||\mathscr{Q}||} \tag{17}$$

$$\cdot P(\omega_r, \omega_s, c_F, H_{1+}, v|S = s, R = r). \tag{18}$$

Similarly, we can expand (1) and (2) conditional on $||\mathscr{V}|| = v$, $\omega_s \in \Omega_s$, and $C_F = c_F$ to express the delivery ratio as

$$D_{BI}(s \to r) = 1 - \sum_{c_F}\sum_{v}\sum_{\omega_s}$$

$$B(S = s, R = r|r \in \overline{\mathscr{V} \cup \{p\}}, \omega_s, c_F, H_{1+}, v, S = s, R = r) \tag{19}$$

$$\cdot P(r \in \overline{\mathscr{V} \cup \{p\}}, \omega_s, c_F, H_{1+}, v|S = s, R = r), \tag{20}$$

$$D_{MP}(s \to r) = 1 - \sum_{c_F}\sum_{v}\sum_{\omega_s}$$

$$\frac{P((s \to r) \in \mathscr{Q}|r \in \overline{\mathscr{V} \cup \{p\}}, \omega_s, c_F, H_{1+}, v, S = s, R = r)}{||\mathscr{Q}||} \tag{21}$$

$$\cdot P(r \in \overline{\mathscr{V} \cup \{p\}}, \omega_s, c_F, H_{1+}, v|S = s, R = r). \tag{22}$$

Observe that the probabilities (19-22) are just special cases of (15-18), respectively, and we treat them like that in the rest of this paper. Note that the eq. (16) and (18) are the probability that a message that belongs to $((s \to r))$ is received by an attacker node and carries particular information. The numerator in (17) corresponds to the probability that the sender-receiver pair $((s \to r)) \in \mathscr{Q}$.

The key to calculate both the relationship anonymity and the delivery ratio is to calculate the attacker's belief that the targeted s-r pair is the sender-receiver pair in (15), for which we have to rely on the information available to the attacker upon receiving a message. A message contains the information ($||\mathscr{V}|| = v$, $\omega_s \in \Omega_s$, $\omega_r \in \Omega_r$, and $C_F = c_F$), and based on these the attacker would compute the probability that (s,r) is the sender-receiver pair as

$$B(S=s,R=r|\omega_r,\omega_s,c_F,H_{1+},v) = \frac{P(\omega_r,\omega_s,v,c_F,H_{1+}|S=s,R=r)\cdot T(S=s,R=r)}{\sum_{(a,b)} P(\omega_r,\omega_s,v,c_F,H_{1+}|S(a),R(b))\cdot T(S(a),R(b))} \tag{23}$$

where the summation in the denominator is over all possible non-attacker sender-receiver pairs ($a \rightarrow b$). $T(S(a),R(b))$ is the a-priori probability that node a sends a message to node b, i.e., the attacker's a-priori belief of the traffic matrix. In the special case when the attacker's a-priori belief is that the traffic matrix is homogeneous, $T(S(a),R(b)) = \frac{1}{(N-C)(N-C-1)}$ for all $(a \rightarrow b)$, and these probabilities cancel out each other in (23). In what follows we compute the probabilities in (23).

Number of Initialized Attacker Nodes

Before we turn to the calculation of the probability $P(\omega_r,\omega_s,v,c_F,H_{1+}|S=s,R=r)$ we introduce the notation $H(v,c_F|F=f)$ for the joint event $||\mathscr{V}|| = v$, H_{1+}, and $C_F = c_F$ for a given number of initialized nodes f. Clearly, $v \geq f$. The probability of this event can be expressed as

$$P(H(v,c_F|F=f)) = \begin{array}{ll} \frac{C}{N-1} & v=0, f=0 \\ P(C_F=0|F=f)\frac{N-C-1}{N-1}\frac{C}{N-v}\prod_{z=1}^{v-1}\frac{N-C-z}{N-z} & v \geq 1, f=0 \\ P(C_F=c_F|F=f)\frac{C-c_F}{N-v}\prod_{z=f}^{v-1}\frac{N-C+c_F-z}{N-z} & v \geq 1, f>0, \end{array} \tag{24}$$

where $P(C_F|F=f)$ is the probability that the set of visited nodes is initialized with c_F attacker nodes, given that it is initialized with f nodes by the sender. Due to the rules of initialization in Minstels, $c_F \in \{max(0,f-1-(N-2-C)),min(f-1,C)\}$. For $F=0$ and $F=1$ there cannot be any initialized attackers, hence $P(C_F=0|F \in \{0,1\}) = 1$ and $P(C_F>0|F \in \{0,1\}) = 0$. For $f>1$ we have

$$P(C_F|F=f) = \left(\begin{array}{c} f-1 \\ c_F \end{array}\right)\frac{\prod_{k=2}^{f-c_F}(N-C-k)\prod_{k=0}^{c_F-1}(C-k)}{\prod_{k=2}^{f}(N-k)} \tag{25}$$

Visited nodes and the Predecessor

We now turn to the calculation of the probability $P(\omega_r,\omega_s,v,c_F,H_{1+}|S=s,R=r)$, i.e., the probability that the attacker would receive a particular message sent by s to r. If the sender is the predecessor ($s=p$) the receiver cannot be the predecessor, hence $P(r=p,s=p,v,c_F,H_{1+}|S=s,R=r) = 0$. For the rest of the cases we show the probabilities in a tabular form to improve readability.

Table 1: $P(\Omega_r, \Omega_s, ||\mathcal{V}|| \in \{0,1\}, C_F = 0, H_{1+}|S = s, R = r)$

Ω_s, Ω_r	$\lvert\lvert\mathcal{V}\rvert\rvert$	
$s = p, r \in \overline{\mathcal{V} \cup \{p\}}$	0	$P(F = 0)P(H(0,0\lvert F = 0))$
$s = p, r \in \overline{\mathcal{V} \cup \{p\}}$	1	$P(F = 1)P(H(1,0\lvert F = 1))$
$s \in \overline{\mathcal{V} \cup \{p\}}, r = p$	1	$P(F = 0)P(H(1,0\lvert F = 0))\frac{1}{N-C-1}$
$s \in \overline{\mathcal{V} \cup \{p\}}, r \in \overline{\mathcal{V} \cup \{p\}}$	1	$P(F = 0)P(H(1,0\lvert F = 0))\frac{N-C-2}{N-C-1}$

For $||\mathcal{V}|| = 0$ and $||\mathcal{V}|| = 1$ there can be no attackers in the set of visited nodes (when received by the first attacker), because if the sender initializes the set of visited nodes with $f > 0$ nodes, it has to include itself in the set. Hence, for $||\mathcal{V}|| = 0$ and $||\mathcal{V}|| = 1$ we have $C_F > 0$ with probability 0. Furthermore, for $||\mathcal{V}|| = 0$ the sender must be the predecessor ($s = p$) and the receiver cannot be in the set of visited nodes ($r \in \overline{\mathcal{V} \cup \{p\}}$). Every other tuple in $\{(\omega_s, \omega_r) : \omega_s \in \Omega_s, \omega_r \in \Omega_r\}$ has probability 0. The first row of Table 1 shows the corresponding probability, i.e., the probability that the sender initializes the message with an empty set, and chooses the attacker as next hop. For $||\mathcal{V}|| = 1$ the sender and the receiver cannot both be in the set of visited nodes. Furthermore, if the sender or the receiver is in the set of visited nodes, it must be the predecessor, hence $s \in \mathcal{V} \setminus \{p\}$ and $r \in \mathcal{V} \setminus \{p\}$ have probability 0. The probabilities for the remaining cases for $||\mathcal{V}|| = 1$ are shown in Table 1. As an example, the third row in the table is the probability that the sender initializes the set empty, forwards the message to the receiver, which then forwards the message to the attacker.

For $||\mathcal{V}|| > 1$ there may or may not be attackers in the set of initialized nodes. When there are attackers in the set of initialized nodes ($C_F > 0$), the sender has to be in the set of visited nodes. Furthermore, if the sender is the predecessor ($s = p$) then the receiver cannot be in the set of visited nodes ($r \in \mathcal{V} \setminus \{p\}$), because this could only happen if the sender had initialized the set of visited nodes with the receiver, but then the receiver would never receive the message. The corresponding probabilities for $||\mathcal{V}|| > 1$ are shown in Table 2 and Table 3 in the Appendix.

We already calculated the numerator of (23), so in order to finish our calculations we only have to express $P(\omega_r, \omega_s, v, c_F, H_{1+}|S(a), R(b))$ and only for the cases when the numerator of (23) is non-zero, and when $a \neq s$ or $b \neq r$.

The attacker can receive a message with an empty set of visited nodes ($||\mathcal{V}|| = 0, C_F = 0$) only if the sender is the predecessor, hence, $P(\omega_r, \omega_s, ||\mathcal{V}|| = 0, C_F = 0, H_{1+}|S(a), R(b)) > 0$ only for $a = s$. Nevertheless, the receiver of the message can be any trusted node $b \neq s$ (we use $\forall b$ as a shorthand notation). The corresponding probability $P(\Omega_r, \Omega_s, ||\mathcal{V}|| = 0, C_F = 0, H_{1+}|S = a, R = b)$ is given in Table 4 in the Appendix.

The attacker can receive a message with only one node in the set of visited nodes ($||\mathcal{V}|| = 1$), in which case the node in the set is the predecessor. The set could have been sent by the predecessor ($a = p$) or by a node not in the set ($a \in \overline{\mathcal{V} \cup \{p\}}$), but in either case there cannot be any attacker node initialized in the set ($C_F = 0$). The receiver could be

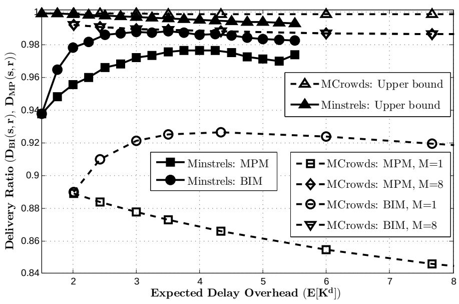


Figure 2: Relationship anonymity vs. delay overhead for $N = 10$ and $C = 1$

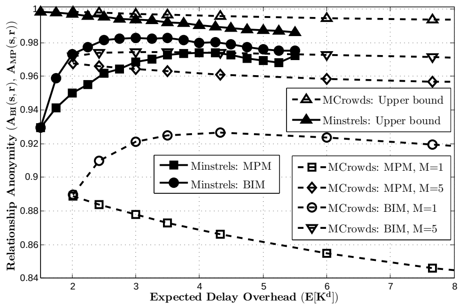


Figure 3: Delivery ratio vs. delay overhead for $N = 10$ and $C = 1$

any other node ($\forall b$).The probability of receiving such a message $P(\Omega_r, \Omega_s, ||\mathscr{V}|| = 1, C_F = 0, H_{1+}|S = a, R = b)$ is given in Table 5 in the Appendix.

The probabilities for $||\mathscr{V}|| > 1$ can be obtained following a similar reasoning. In order to maintain the readability of the paper we describe the probabilities in the Appendix.

5.5 Upper Bounds

In order to have a better understanding of the relationship anonymity and the delivery ratio provided by the described anonymity networks, we define the upper bound for the relationship anonymity and the upper bound for the delivery ratio. To obtain the upper bounds, we consider that whenever the attacker intercepts a message, it assumes that any trusted pair of nodes is equally likely to be the sender-receiver pair with belief $B(S = s, R = r|H_{1+}, S = s, R = r) = \frac{1}{(N-C)(N-C-1)}$.

6 Numerical Results

In the following, we first use the analytical results to investigate the relationship anonymity-overhead trade-off provided by MCrowds and by Minstrels. We then show simulation results that confirm the analytical results.

6.1 Relationship anonymity-overhead trade off

We use the analytical results developed in Section 5 to explore the trade-off between relationship anonymity and overhead, and between delivery ratio and overhead for MCrowds and for Minstrels. For MCrowds we use a relaying probability $p_f \in (0,1)$ and $M \in \{1, \ldots, N-2\}$, and for Minstrels we use various uniform, binomial, and triangular distributions to choose the number F of initialized nodes. The attacker's a-priory belief is that the traffic matrix is homogeneous.

Fig. 2 and Fig. 3 show the delivery ratio and the relationship anonymity, respectively, as a function of the expected delay overhead ($E[K^d]$) for $C = 1$ attacker node in a system of

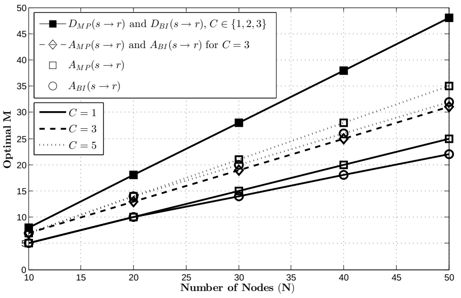


Figure 4: Optimal receiver set size *M* vs. number of nodes for MCrowds

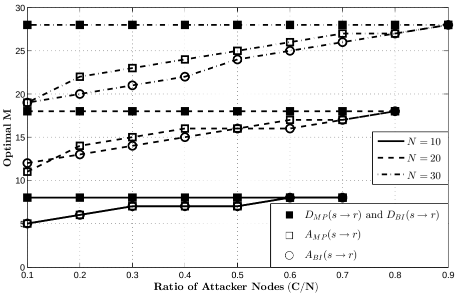


Figure 5: Optimal receiver set size *M* vs. ratio of attacker nodes for MCrowds

$N = 10$ nodes. An expected delay overhead of $E[K^d] = 2$ corresponds to one relay visited before the receiver on average, while $E[K^d] = (N+1)/2$ is the maximum expected delay overhead for Minstrels which happens when the list is always initialized empty $F = 0$. The upper bounds are obtained by finding the distribution of F for Minstrels, and the receiver set size M for MCrowds, that results in the lowest $P(H_{1+}|S = s, R = r)$ for a given overhead.

One would expect that higher delay overhead always provides better relationship anonymity and delivery ratio, but surprisingly this is not the case. A further increase of the delay overhead (more relaying) can have a negative effect on both the delivery ratio and the relationship anonymity under the considered traffic analysis methods for both anonymity networks. The reason is that as the expected number of relays increases, the probability $P(H_{1+}|S = s, R = r)$ of having an attacker node on the path increases faster than the certainty of the attacker about the identity of the sender-receiver pair decreases. Interestingly, for MCrowds and the MP method increased overhead almost always results in worse relationship anonymity. We also observe that the delivery ratio is larger or equal than the relationship anonymity.

Observe that in Fig. 2 for $M = 8$, the MP method and the BI method result in the same delivery ratio. That happens due to the fact that for $M = N-2$, the delivery ratio is affected only in the cases when the only relay is in fact the attacker node, which consequently is completely certain in the sender of the message while the receiver stays perfectly hidden among the rest of the nodes. Hence, $B(S = s, R = r)$ under the BI method equal 0 for all $(s \to r) \notin \mathscr{Q}$, and therefore $D_{MP}(s \to r) = D_{BI}(s \to r)$ in the cases when $M = N-2$.

The results suggest that MCrowds performs better for larger values of the receiver set size M. However, this may not be necessarily the case as a larger M hides the receiver better but, at the same time, exposes more the sender because there are fewer potential relays. Hence there should be an optimal receiver set size M. Fig. 4 shows the optimal value of M as a function of the number N of nodes in the system. The optimal receiver set size M increases with the number of nodes in the system. In the case of the delivery ratio ($D_{MP}(s \to r)$ and $D_{BI}(s \to r)$), the optimal M equals the largest possible value $(N-2)$ as that value minimizes the probability of having the attacker node as a relay and the multicast messages are not accounted in the analysis. However, in the case of the relationship anonymity,

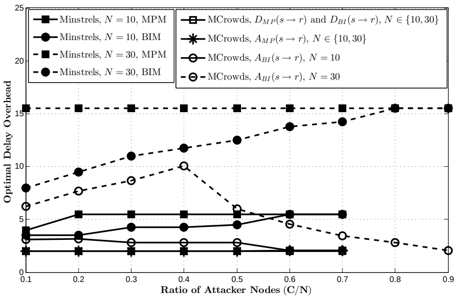


Figure 6: Optimal delay overhead vs. ratio of attacker nodes

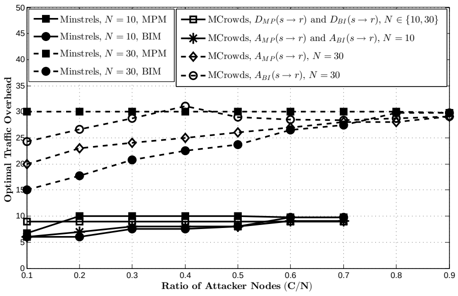


Figure 7: Optimal traffic overhead vs. ratio of attacker nodes

the multicast messages are accounted and the optimal M is found as a result of the trade-off between the probability of having the attacker as a relay and the anonymity against a non-relaying attacker that receives the multicast messages. Thus, the optimal M for the relationship anonymity is always lower than for the delivery ratio. The values of $M > 1$ used in Fig. 2 and Fig. 3 are in fact optimal for $N = 10$ and $C = 1$.

Fig. 5 shows the optimal receiver set size M as a function of the ratio $\frac{C}{N}$ of attacker nodes in the system. The optimal value of M for the delivery ratio is always the largest possible ($M = N - 2$) regardless of the number of attacker nodes C in the system, and the method used. In the case of the relationship anonymity, the optimal value of M is a non-decreasing function of the ratio of attacker nodes. For a given ratio of attacker nodes the optimal receiver set size M for the MP method is always greater than or equal to the optimal M for the BI method (they completely overlap for $N = 10$), but they always have the same maximum value, which is equal to the optimal M for the delivery ratio. As the system gets larger, the highest optimal value of M for the MP method and for the BI method is reached at higher values of the ratio of attacker nodes. Hence, with more attacker nodes in the system it is better to increase the receiver set size M if it is lower than the highest optimal value.

Fig. 6 and Fig. 7 show the optimal delay overhead overhead and the optimal traffic overhead (where the delivery ratio or the relationship anonymity is the highest) as a function of the ratio of attacker nodes ($\frac{C}{N}$), respectively. For Minstrels, the optimal delay overhead and the optimal traffic overhead are the same for the delivery and for the relationship anonymity. The optimal delay overhead and the optimal traffic overhead for the BI method increase with the system size N and the ratio ($\frac{C}{N}$) and they are lower than the optimal delay overhead and the optimal traffic overhead for the MP method, respectively. Under the MP method, the optimal values of delay and traffic overheads equal their maximum values ($E[K^d] = (N+1)/2$ and $E[K^t] = N$) except for $N = 10$ and $\frac{C}{N} = 0.1$.

For MCrowds, the optimal overhead values are always the smallest ($E[K^d] = 2$ and $E[K^t] = N - 1$) for the delivery ratio as the minimum relaying and maximum $M = N - 2$ achieve the best ($D_{MP}(s \to r)$ and $D_{BI}(s \to r)$). In the case of the relationship anonymity, the optimal overhead values for both the BI method and the MP method increase with the

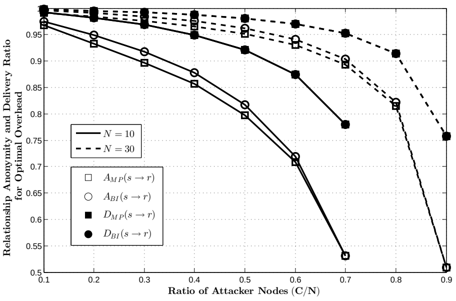


Figure 8: Relationship anonymity and Delivery ratio for optimal overhead vs. ratio of attacker nodes for MCrowds

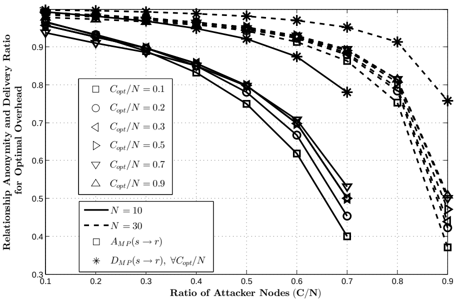


Figure 9: Relationship anonymity and Delivery ratio for optimal overhead vs. ratio of attacker nodes for MCrowds

system size N. For a given ratio of attacker nodes $\frac{C}{N}$ the optimal overhead values for the BI method are greater than or equal to the optimal overhead values for the MP method. It is interesting to note that for the considered system sizes N the optimal delay overhead is upper bounded by the maximum Minstrels delay overhead, and the optimal traffic overhead rarely exceeds the maximum Minstrels traffic overhead.

Fig. 8 shows the relationship anonymity and the delivery ratio at the optimal overhead as a function of the ratio of attacker nodes $(\frac{C}{N})$. As the ratio of attacker nodes increases, the relationship anonymity and the delivery ratio monotonically decrease. However, for larger systems the relationship anonymity and the delivery ratio are higher for the same ratio of attacker nodes. Consequently, with an increase in the system size the attacker needs to corrupt more than proportional number of nodes in order to achieve the same values of the relationship anonymity and the delivery ratio. Hence, both for Minstrels and for MCrowds, it is always beneficial to have more nodes in the network for the same ratio of attacker nodes $\frac{C}{N}$. Furthermore, the results show that both relationship anonymity and the delivery ratio are lower under the MP method than under the BI method, i.e., for the attacker it is always better to use the MP traffic analysis method than the BI traffic analysis method.

In practice the ratio of the attacker nodes is not known by the system designer, hence the anonymity network must be inevitably optimized for an unknown parameter. In Fig. 9 we investigate the sensitivity of the relationship anonymity and of the delivery ratio under the MP method to misestimating the ratio of attacker nodes for MCrowds. The expected overhead is selected to be optimal for various ratios of attacker nodes, from $\frac{C}{N} = 0.1$ to $\frac{C}{N} = 0.9$. Interestingly, both the relationship anonymity and the delivery ratio are less sensitive to the actual ratio of attacker nodes when the anonymity network is optimized for a higher ratio of attacker nodes. The anonymity network optimized for a lower ratio of attacker nodes performs worse for higher $\frac{C}{N}$ ratios than the anonymity network optimized for a higher ratio of attacker nodes for lower $\frac{C}{N}$ ratios. Therefore, it is better to optimize the anonymity network for a higher ratio of attacker nodes than the actual ratio. We observed similar behavior for bigger system sizes N and the BI method.

The presented results lead us to the following interesting conclusions. First, best re-

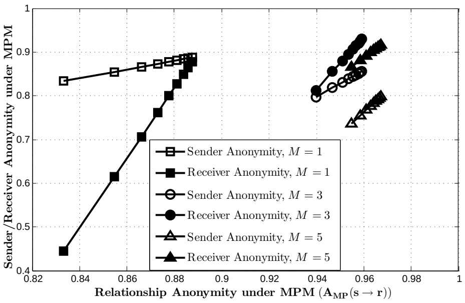


Figure 10: Sender and receiver anonymity vs. relationship anonymity under the MP method for MCrowds.

lationship anonymity and best delivery ratio might not be achieved at the highest possible overhead. The optimal overhead depends on the anonymity network, traffic analysis method, system size, and the number of attacker nodes. Second, for an attacker it is always better to use the Maximum posteriori method than the Bayesian inference method for traffic analysis in case of the MCrowds and the Minstrels anonymity networks. Third, MCrowds and Minstrels can achieve better relationship anonymity and delivery ratio in bigger systems, but at the price of higher overhead. Fourth, when the number of attacker nodes is unknown MCrowds and Minstrels are less sensitive if they are optimized for a high ratio of attacker nodes. Finally, for MCrowds it always beneficial to have more than one node specified as the receiver of the message ($M > 1$).

6.2 Trade off between Relationship Anonymity and Sender-Receiver Anonymity

In the following, we explore the trade off between the relationship anonymity and the sender or receiver anonymity in order to justify our approach to consider the relationship anonymity instead of the sender and the receiver anonymity separately. We quantify the sender (receiver) anonymity similarly to the relationship anonymity: the probability that a message sent from s (sent to r) is correctly classified when s (r) is the targeted sender (receiver).

Fig. 10 shows the trade-off between the sender or receiver anonymity and the relationship anonymity for a system with $N = 10$ nodes that uses MCrowds with $M \in \{1,3,5\}$ and $p_f \in (0.1, 0.9)$. The attacker is in control of one node ($C = 1$), and it uses the MP method assuming that $T(S = a, R = b)$ is uniform. To calculate the sender (receiver) anonymity, we used the analytical results developed in Section 5.2 while assuming that the probability assigned to the receiver (sender) equals to 1. Both sender and receiver anonymity increase with the relationship anonymity as a function of p_f. However, the best relationship anonymity is not achieved together with the best sender or receiver anonymity. The best relationship anonymity is achieved for $M = 5$, while the best sender anonymity and the best receiver anonymity are achieved for $M = 1$ and $M = 3$, respectively. Thus, the results show that it is better to consider the relationship anonymity instead of the sender and the receiver anonymity separately when optimizing an anonymity network to protect pair-wise communication.

7 Conclusions

In this paper we considered the problem of mitigating gray hole attacks by providing relationship anonymity among a fixed set of nodes. We described two anonymity networks, MCrowds and Minstrels. MCrowds is an extension of Crowds, and provides unbounded path length, while Minstrels provides bounded path length. We considered two attack methods: the Bayesian inference method and the Maximum posteriori method. We found that MCrowds provides better relationship anonymity than Crowds, but in order to provide anonymity to the receiver the sender is more exposed than in Crowds. Moreover, we found that Minstrels provides better relationship anonymity than MCrowds. We used the two anonymity systems to study the trade-off between relationship anonymity and communication overhead, and found that increased overhead does not always lead to improved relationship anonymity. When comparing the two traffic analysis methods, we found that the Maximum posteriori method performs always better. We studied the way relationship anonymity scales with the number of nodes, and observed that relationship anonymity improves with the number of nodes but at the price of higher overhead. Our results also show that in practice anonymity systems should be optimized for a higher number of attackers than expected.

References

[1] D. Dzung, M. Naedele, T. V. Hoff, and M. Crevatin. Security for industrial communication systems. In *Proc. of IEEE*, volume 93, pages 1152–1177, 2005.

[2] C. W. Ten, C. C. Liu, and M. Govindarasu. Vulnerability assessment of cybersecurity for scada systems. *IEEE Trans. Power Syst.*, 23(4), 2008.

[3] N. Hatziargyriou, H. Asano, R. Iravani, and C. Marnay. Microgrids. *IEEE Power and Energy Magazine*, 5(4):78–94, July 2007.

[4] M. Shahidehpour and Y. Wang. *Communication and Control in Electric Power Systems*. John Wiley and Sons, 2003.

[5] A. Gómez-Expósito, A. Abur, A. De La Villa Jaén, and C. Gómez-Quiles. A multilevel state estimation paradigm for smart grids. *Proceedings of the IEEE*, 99(6):952–976, June 2011.

[6] R. J. Turk. Cyber incidents involving control systems. Technical report, Idaho National Laboratory, 2005.

[7] Saurabh Amin, Alvaro A. Cárdenas, and S. Shankar Sastry. Safe and secure networked control systems under denial-of-service attacks. In *Proc. of the 12th International Conference on Hybrid Systems: Computation and Control*, pages 31–45. Springer-Verlag, 2009.

[8] Keith Stouffer, Joe Falco, and Karen Scarfone. Guide to industrial control systems (ICS) security. Technical report, NIST SP 800-82, 2011.

[9] Andreas Pfitzmann and Marit Köhntopp. Anonymity, unobservability, and pseudonymity - a proposal for terminology. In *Designing Privacy Enhancing Technologies*, volume 2009, pages 1–9. Springer Berlin Heidelberg, 2001.

[10] D. Chaum. Untraceable electronic mail, return addresses and digital pseudonyms. *Commun. of the ACM*, 24(2):84–88, 1981.

[11] P. Syverson, D. Goldschlag, and M. Reed. Anonymous connections and onion routing. In *Proc. IEEE Symp. on Security and Privacy*, pages 44–54, May 1997.

[12] M. Reiter and A. Rubin. Crowds: Anonymity for web transactions. *ACM Trans. Inform. Syst. Secur.*, 1(1):66–92, 1998.

[13] V. Shmatikov and M. H. Wang. Measuring relationship anonymity in mix networks. In *Proc. of Workshop on Privacy in the Electronic Society (WPES)*, 2006.

[14] Roger Dingledine, Nick Mathewson, and Paul F. Syverson. Tor: The second-generation onion router. In *Proceedings of the 13th Conference on USENIX Security Symposium*, volume 13 of *SSYM'04*, pages 303–320. USENIX Association, 2004.

[15] G. Danezis, C. Díaz, E. Käsper, and C. Troncoso. The wisdom of crowds: attacks and optimal constructions. In *Proc. of ESORICS*, 2009.

[16] J. Feigenbaum, A. Johnson, and P. Syverson. Probabilistic analysis of onion routing in a black-box model. In *Proc. of Workshop on Privacy in the Electronic Society (WPES)*, 2007.

[17] M. Wright, M. Adler, B. N. Levine, and C. Shields. The predecessor attack: An analysis of a threat to anonymous communications systems. *ACM Trans. Inform. Syst. Secur.*, 7(4):489–522, November 2004.

[18] C. Troncoso, B. Gierlichs, B. Preneel, and I. Verbauwhede. Perfect matching disclosure attacks. In *Proc. of Privacy Enhancing Technologies Symposium (PETS)*, 2008.

[19] G. Danezis and C. Troncoso. Vida: How to use bayesian inference to de-anonymize persistent communications. In *Proc. of Privacy Enhancing Technologies Symposium (PETS)*, 2009.

[20] C. Troncoso and G. Danezis. The bayesian traffic analysis of mix networks. In *Proc. of Conference on Computer and Communications Security (CCS)*, 2009.

[21] C. Diaz, S. J. Murdoch, and C. Troncoso. Impact of network topology on anonymity and overhead in low-latency anonymity networks. In *Proc. of Privacy Enhancing Technologies Symposium (PETS)*, 2010.

[22] Nikita Borisov, George Danezis, Prateek Mittal, and Parisa Tabriz. Denial of service or denial of security? In *Proc. of ACM CCS*, October 2007.

[23] George Danezis, R. Dingledine, and N. Mathewson. Mixminion: Design of a type iii anonymous remailer protocol. In *2003 Symposium on Security and Privacy*, pages 2–15, May 2003.

[24] Li Zhuang, Feng Zhou, Ben Y. Zhao, and Antony Rowstron. Cashmere: Resilient anonymous routing. In *Proceedings of the 2nd Conference on Symposium on Networked Systems Design & Implementation*, volume 2 of *NSDI'05*, pages 301–314. USENIX Association, 2005.

[25] Jelena Mirkovic and Peter Reiher. A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communications Review*, 34(2):39–53, April 2004.

[26] M. Long, C.-H.J. Wu, J.Y. Hung, and J.D. Irwin. Mitigating performance degradation of network-based control systems under denial of service attacks. In *In Proc. of IEEE Industrial Electronics Society Conference IECON*, volume 3, pages 2339–2342, November 2004.

[27] Men Long, Chwan-Hwa Wu, and John Y. Hung. Denial of Service Attacks on Network-Based Control Systems: Impact and Mitigation. *IEEE Transactions on Industrial Informatics*, 1(2):85–96, May 2005.

[28] H.S. Foroush and S. Martinez. On event-triggered control of linear systems under periodic denial-of-service jamming attacks. In *In Proc. of IEEE 51st Annual Conference on Decision and Control (CDC)*, pages 2551–2556, December 2012.

[29] Liu Shichao, X.P. Liu, and A. El Saddik. Denial-of-service (DoS) attacks on load frequency control in smart grids. In *In Proc. of IEEE PES Innovative Smart Grid Technologies (ISGT)*, pages 1–6, February 2013.

Appendix

In the following we show calculation of the probabilities introduced in Section 5.4 in Table 2, 3, 4, and 5. Moreover, we describe the probabilities $P(\Omega_s, \Omega_r, ||\mathscr{V}||, C_F, H_{1+}|S = a, R = b)$ for $||\mathscr{V}|| > 1$.

Table 2: $P(\Omega_r, \Omega_s, ||\mathscr{V}|| > 1, C_F = 0, H_{1+}|S = s, R = r)$

Ω_s, Ω_r	
$s = p$, $r \in \mathscr{V} \setminus \{p\}$	$P(F=0)P(H(v,0 \mid F=0))\frac{v-1}{(N-C-1)^2}$
$s = p$, $r \in \overline{\mathscr{V} \cup \{p\}}$	$P(F=0)P(H(v,0 \mid F=0))\frac{(N-C-v)}{(N-C-1)^2}$ $+P(F=v)P(H(v,0 \mid F=v))$
$s \in \mathscr{V} \setminus \{p\}$, $r = p$	$P(F=0)P(H(v,0 \mid F=0))\frac{v-2}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\frac{1}{N-C-k}$
$s \in \mathscr{V} \setminus \{p\}$, $r \in \mathscr{V} \setminus \{p\}$	$P(F=0)P(H(v,0 \mid F=0))\frac{(v-2)^2}{(N-C-1)^2}$ $+\sum_{k=1}^{v-2} P(F=k)P(H(v,0 \mid F=k))\frac{v-k-1}{N-C-k}$
$s \in \mathscr{V} \setminus \{p\}$, $r \in \overline{\mathscr{V} \cup \{p\}}$	$P(F=0)P(H(v,0 \mid F=0))\frac{(N-C-v)(v-2)}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\frac{N-C-v}{N-C-k}$
$s \in \overline{\mathscr{V} \cup \{p\}}$, $r = p$	$P(F=0)P(H(v,0 \mid F=0))\frac{(N-C-v)}{(N-C-1)^2}$
$s \in \overline{\mathscr{V} \cup \{p\}}$, $r \in \mathscr{V} \setminus \{p\}$	$P(F=0)P(H(v,0 \mid F=0))\frac{(v-1)(N-C-v)}{(N-C-1)^2}$
$s \in \overline{\mathscr{V} \cup \{p\}}$, $r \in \overline{\mathscr{V} \cup \{p\}}$	$P(F=0)P(H(v,0 \mid F=0))\frac{(N-C-v)(N-C-v-1)}{(N-C-1)^2}$

Table 3: $P(\Omega_r, \Omega_s, ||\mathscr{V}|| > 1, C_F > 0, H_{1+}|S = s, R = r)$

Ω_s, Ω_r	
$s = p$, $r \in \overline{\mathscr{V} \cup \{p\}}$	$P(F=v)P(H(v,c_F \mid F=v))$
$s \in \mathscr{V} \setminus \{p\}$, $r = p$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F \mid F=k))\frac{1}{N-C+c_F-k}$
$s \in \mathscr{V} \setminus \{p\}$, $r \in \mathscr{V} \setminus \{p\}$	$\sum_{k=c_F+1}^{v-2} P(F=k)P(H(v,c_F \mid F=k))\frac{v-k-1}{N-C+c_F-k}$
$s \in \mathscr{V} \setminus \{p\}$, $r \in \overline{\mathscr{V} \cup \{p\}}$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F \mid F=k))\frac{N-C+c_F-v}{N-C+c_F-k}$

Table 4: $P(\Omega_r, \Omega_s, ||\mathscr{V}|| = 0, C_F = 0, H_{1+}|S = a, R = b)$

Ω_s, Ω_r, a, b	
$s = p$, $r \in \overline{\mathscr{V} \cup \{p\}}$, $a = s$, $\forall b$	$P(F=0)P(H(0,0 \mid F=0))$

When there are no initialized attackers ($C_F = 0$) the set could have been initialized with $F \in [0..||\mathscr{V}||]$ nodes. Let us first consider the case when node s is the predecessor ($s = p$) and node r is in the set ($r \in \mathscr{V} \setminus \{p\}$). For any sender-receiver pair (a, b), the prerequisite for this to happen is that node s has to be visited just before the attacker,

Table 5: $P(\Omega_r, \Omega_s, ||\mathscr{V}|| = 1, C_F = 0, H_{1+} | S = a, R = b)$

Ω_s, Ω_r, a, b	
$s = p,\ r \in \overline{\mathscr{V} \cup \{p\}},\ a = s,\ \forall b$	$P(F=1)P(H(1,0 \vert F=1))$
$s = p,\ r \in \overline{\mathscr{V} \cup \{p\}},\ a \neq s,\ \forall b$	$P(F=0)P(H(1,0 \vert F=0))\frac{1}{N-C-1}$
$s \in \overline{\mathscr{V} \cup \{p\}},\ r = p,\ a = r,\ \forall b$	$P(F=1)P(H(1,0 \vert F=1))$
$s \in \overline{\mathscr{V} \cup \{p\}},\ r = p,\ a \neq r,\ \forall b$	$P(F=0)P(H(1,0 \vert F=0))\frac{1}{N-C-1}$
$s \in \overline{\mathscr{V} \cup \{p\}},\ r \in \overline{\mathscr{V} \cup \{p\}},\ a \in \{s,r\},\ \forall b$	$P(F=0)P(H(1,0 \vert F=0))\frac{N-C-2}{N-C-1}$
$s \in \overline{\mathscr{V} \cup \{p\}},\ r \in \overline{\mathscr{V} \cup \{p\}},\ a \notin \{s,r\},\ \forall b$	$P(F=0)P(H(1,0 \vert F=0))\frac{N-C-3}{N-C-1}$ $+P(F=1)P(H(1,0 \vert F=1))$

while node r has to be either initialized or be visited. The corresponding probabilities $P(s = p, r \in \mathscr{V} \setminus \{p\}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$ are given in Table 6.

Table 6: $P(s = p, r \in \mathscr{V} \setminus \{p\}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$

a, b	
$a = s,\ b \neq r$	$P(F=0)P(H(v,0 \vert F=0))\frac{v-1}{(N-C-1)^2} + P(F=v)P(H(v,0 \vert F=v))\frac{v-1}{N-C-2}$
$a = r,\ \forall b$	$P(F=0)P(H(v,0 \vert F=0))\frac{v-2}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \vert F=k))\frac{1}{N-C-1}$
$a \notin \{s,r\}$, $b = s$	$P(F=0)P(H(v,0 \vert F=0))\left(\frac{1}{(N-C-1)^2} + \frac{(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \vert F=k))\frac{v-2}{(N-C-2)(N-C-k)}$
$a \notin \{s,r\}$, $b = r$	$P(F=0)P(H(v,0 \vert F=0))\left(\frac{1}{(N-C-1)^2} + \frac{(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+P(F=1)P(H(v,0 \vert F=1))\frac{v-2}{(N-C-1)(N-C-2)}$ $+\sum_{k=2}^{v-1} P(F=k)P(H(v,0 \vert F=k))\frac{v-k-1}{(N-C-2)^2}$
$a \notin \{s,r\}$, $b \notin \{s,r\}$	$P(F=0)P(H(v,0 \vert F=0))\left(\frac{1}{(N-C-1)^2} + \frac{(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \vert F=k))\cdot$ $\left(\frac{(k-1)(N-C-k-1)}{(N-C-2)(N-C-3)(N-C-k)} + \frac{(v-k-1)(N-C-k-2)}{(N-C-2)(N-C-3)(N-C-k)}\right)$

The case when node s is the predecessor ($s = p$) but node r is not in the set ($r \in \overline{\mathscr{V} \cup \{p\}}$) is similar to the previous case. The only difference is that node r has to be neither initialized nor be visited. The probabilities $P(s = p, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$ are given in Table 7.

When we have $s \in \mathscr{V} \setminus \{p\}$ and $r = p$, node s has to be either initialized or be visited, while node r has to be visited just before the attacker. The probabilities $P(s \in \mathscr{V} \setminus \{p\}, r = p, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$ are given in Table 8.

Table 7: $P(s=p, r\in\overline{\mathscr{V}\cup\{p\}}, ||\mathscr{V}||=v>1, C_F=0, H_{1+}|S=a, R=b)$

a,b	
$a=s,$ $b\neq r$	$P(F=0)P(H(v,0\vert F=0))\frac{N-C-v}{(N-C-1)^2}$ $+P(F=v)P(H(v,0\vert F=v))\frac{N-C-v-1}{N-C-2}$
$a=r,\forall b$	$P(F=0)P(H(v,0\vert F=0))\frac{N-C-v}{(N-C-1)^2}$
$a\notin\{s,r\},$ $b\in\{s,r\}$	$P(F=0)P(H(v,0\vert F=0))\frac{(N-C-3)(N-C-v)}{(N-C-1)^2(N-C-2)}$ $+\sum_{k=1}^{v-1}P(F=k)P(H(v,0\vert F=k))\frac{N-C-v}{(N-C-2)(N-C-k)}$
$a\notin\{s,r\},$ $b\notin\{s,r\}$	$P(F=0)P(H(v,0\vert F=0))\frac{(N-C-3)(N-C-v)}{(N-C-1)^2(N-C-2)}$ $+\sum_{k=1}^{v-1}P(F=k)P(H(v,0\vert F=k))\frac{(N-C-k-2)(N-C-v)}{(N-C-2)(N-C-3)(N-C-k)}$

Table 8: $P(s\in\mathscr{V}\setminus\{p\}, r=p, ||\mathscr{V}||=v>1, C_F=0, H_{1+}|S=a, R=b)$

a,b	
$a=s,\forall b$	$P(F=0)P(H(v,0\vert F=0))\frac{v-2}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1}P(F=k)P(H(v,0\vert F=k))\frac{N-C-k-1}{N-C-2}\frac{1}{N-C-k}$
$a=r, b=s$	$P(F=0)P(H(v,0\vert F=0))\frac{v-1}{(N-C-1)^2}$
$a=r,\ b\neq s$	$P(F=0)P(H(v,0\vert F=0))\frac{v-1}{(N-C-1)^2}+P(F=v)P(H(v,0\vert F=v))\frac{v-1}{N-C-2}$
$a\notin\{s,r\},$ $b=r$	$P(F=0)P(H(v,0\vert F=0))\left(\frac{1}{(N-C-1)^2}+\frac{(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1}P(F=k)P(H(v,0\vert F=k))\frac{v-2}{(N-C-2)(N-C-k)}$
$a\notin\{s,r\},$ $b=s$	$P(F=0)P(H(v,0\vert F=0))\left(\frac{1}{(N-C-1)^2}+\frac{(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+P(F=1)P(H(v,0\vert F=1))\frac{v-2}{(N-C-1)(N-C-2)}$ $+\sum_{k=2}^{v-1}P(F=k)P(H(v,0\vert F=k))\frac{v-k-1}{(N-C-2)^2}$
$a\notin\{s,r\},$ $b\notin\{s,r\}$	$P(F=0)P(H(v,0\vert F=0))\left(\frac{1}{(N-C-1)^2}+\frac{(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1}P(F=k)P(H(v,0\vert F=k))\cdot$ $\left(\frac{(k-1)(N-C-k-1)}{(N-C-2)(N-C-3)(N-C-k)}+\frac{(v-k-1)(N-C-k-2)}{(N-C-2)(N-C-3)(N-C-k)}\right)$

For $s\in\mathscr{V}\setminus\{p\}$ and $r\in\mathscr{V}\setminus\{p\}$, both nodes ($s$, r) have to be either initialized or be visited before the message reaches the attacker. The probabilities $P(s\in\mathscr{V}\setminus\{p\}, r\in\mathscr{V}\setminus\{p\}, ||\mathscr{V}||=v>1, C_F=0, H_{1+}|S=a, R=b)$ are given in Table 9.

For the case when we have $s\in\mathscr{V}\setminus\{p\}$ and $r\in\overline{\mathscr{V}\cup\{p\}}$, the only difference from

Table 9: $P(s \in \mathscr{V} \setminus \{p\}, r \in \mathscr{V} \setminus \{p\}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$

a,b	
$a = s,\, b = r$ $a = r,\, b = s$	$P(F=0)P(H(v,0\|F=0))\frac{(v-2)^2}{(N-C-1)^2}$ $+\sum_{k=1}^{v-2} P(F=k)P(H(v,0\|F=k))\frac{v-k-1}{N-C-k}$
$a = s,\, b \neq r$ $a = r,\, b \neq s$	$P(F=0)P(H(v,0\|F=0))\frac{(v-2)^2}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0\|F=k))\left(\frac{k-1}{N-C-2} + \frac{(v-k-1)(N-C-k-1)}{(N-C-2)(N-C-k)}\right)$
$a \notin \{s,r\}$, $b \in \{s,r\}$, $v > 2$	$P(F=0)P(H(v,0\|F=0))\left(\frac{2(v-2)}{(N-C-1)^2} + \frac{(v-2)(v-3)(N-C-3)}{(N-C-1)^2(N-C-2)}\right)$ $+P(F=1)P(H(v,0\|F=1))\frac{(v-2)(v-3)}{(N-C-1)(N-C-2)}$ $+\sum_{k=2}^{v-3} P(F=k)P(H(v,0\|F=k))\frac{(v-k-1)^2}{(N-C-2)(N-C-k)}$ $+P(F=v-2)P(H(v,0\|F=v-2))\frac{v-3}{(N-C-2)(N-C-v+2)}$
$a \notin \{s,r\}$, $b \notin \{s,r\}$ $v > 2$	$P(F=0)P(H(v,0\|F=0))\left(\frac{2(v-2)}{(N-C-1)^2} + \frac{(v-2)(v-3)(N-C-3)}{(N-C-1)^2(N-C-2)}\right)$ $\sum_{k=1}^{v-1} P(F=k)P(H(v,0\|F=k))\Bigg(\frac{(k-1)(k-2)}{(N-C-2)(N-C-3)}$ $\frac{(v-k-1)(v-k-2)(N-C-k-2)}{(N-C-k)(N-C-k-1)(N-C-3)} + \frac{2(N-C-k-1)(k-1)(v-k-1)}{(N-C-2)(N-C-3)(N-C-k)}\Bigg)$ $+P(F=v)P(H(v,0\|F=v))\frac{(v-1)(v-2)}{(N-C-2)(N-C-3)}$

the case above is that node r must not have been initialized or visited. The probabilities $P(s \in \mathscr{V} \setminus \{p\}, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$ are given in Table 10.

When we have the opposite case of the above, $s \in \overline{\mathscr{V} \cup \{p\}}$ and $r \in \mathscr{V} \setminus \{p\}$, the same reasoning applies but in this case node s must not have been initialized or visited, and node r has to be either initialized or visited before the message reaches the attacker. The probabilities $P(s \in \overline{\mathscr{V} \cup \{p\}}, r \in \mathscr{V} \setminus \{p\}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b$ are given in Table 11.

For $s \in \overline{\mathscr{V} \cup \{p\}}$ and $r = p$, node s must not have been initialized or visited, while node r has to be visited just before the attacker. The corresponding probabilities $P(s \in \overline{\mathscr{V} \cup \{p\}}, r = p, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$ are given in Table 12.

Finally, for the case when neither s nor r are in the set ($s \in \overline{\mathscr{V} \cup \{p\}}$, $r \in \overline{\mathscr{V} \cup \{p\}}$), they must not have been initialized or visited. The probabilities $P(s \in \overline{\mathscr{V} \cup \{p\}}, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$ are given in Table 13.

Until now we considered the cases when there are no initialized attackers in the set of visited nodes ($C_F = 0$). However, the attacker can receive a message with $||\mathscr{V}|| = v > 1$ visited nodes and with $C_F = c_F > 0$ initialized attackers. In this case the sender node must have initialized the set with c_F attackers. Hence $F \in [c_F + 1..v]$. Let us now consider

Table 10: $P(s \in \mathscr{V} \setminus \{p\}, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$

a, b	
$a = s$, $b \neq r$	$P(F=0)P(H(v,0 \vert F=0))\frac{(v-2)(N-C-v)}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \vert F=k))\frac{(N-C-k-1)(N-C-v)}{(N-C-2)(N-C-k)}$
$a = r$, $\forall b$	$P(F=0)P(H(v,0 \vert F=0))\frac{(v-1)(N-C-v)}{(N-C-1)^2}$
$a \notin \{s,r\}$, $b = s$	$P(F=0)P(H(v,0 \vert F=0))\left(\frac{N-C-v}{(N-C-1)^2} + \frac{(N-C-v)(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-2} P(F=k)P(H(v,0 \vert F=k))\frac{(v-k-1)(N-C-v)}{(N-C-2)(N-C-k)}$
$a \notin \{s,r\}$, $b = r$	$P(F=0)P(H(v,0 \vert F=0))\left(\frac{N-C-v}{(N-C-1)^2} + \frac{(N-C-v)(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \vert F=k))\frac{(v-k-1)(N-C-v)}{(N-C-2)(N-C-k)}$ $+P(F=v)P(H(v,0 \vert F=v))\frac{v-1}{N-C-2}$
$a \notin \{s,r\}$, $b \notin \{s,r\}$	$P(F=0)P(H(v,0 \vert F=0))\left(\frac{N-C-v}{(N-C-1)^2} + \frac{(N-C-v)(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \vert F=k)) \cdot$ $\left(\frac{(k-1)(N-C-k-1)(N-C-v)}{(N-C-2)(N-C-3)(N-C-k)} + \frac{(v-k-1)(N-C-k-2)(N-C-v)}{(N-C-2)(N-C-3)(N-C-k)}\right)$ $+P(F=v)P(H(v,0 \vert F=v))\frac{(v-1)(N-C-v-1)}{(N-C-2)(N-C-3)}$

different values of Ω_s and Ω_r. For $s = p$ and $r \in \overline{\mathscr{V} \cup \{p\}}$, node s has to be visited just before the attacker. At the same time, node r must not have been initialized or visited. The corresponding probabilities $P(s = p, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+} | S = a, R = b)$ are given in Table 14.

A similar reasoning applies when we have $s \in \mathscr{V} \setminus \{p\}$ and $r = p$. Node s has to be either initialized or visited, while node r has to appear as the predecessor. The probabilities $P(s \in \mathscr{V} \setminus \{p\}, r = p, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+} | S = a, R = b)$ are given in Table 15.

When nodes s and r are both in the set ($s \in \mathscr{V} \setminus \{p\}, r \in \mathscr{V} \setminus \{p\}$), the sender a must have initialized them or the message must have visited them. The corresponding probabilities $P(s \in \mathscr{V} \setminus \{p\}, r \in \mathscr{V} \setminus \{p\}, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+} | S = a, R = b)$ are given in Table 16.

For $s \in \mathscr{V} \setminus \{p\}$ and $r \in \overline{\mathscr{V} \cup \{p\}}$, the sender a must have initialized node s or the message must have visited it before the attacker received the message. At the same time, node r must not have been initialized or visited. The corresponding probabilities $P(s \in \mathscr{V} \setminus \{p\}, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+} | S = a, R = b)$ are given in Table 17.

Table 11: $P(s \in \overline{\mathscr{V} \cup \{p\}}, r \in \mathscr{V} \setminus \{p\}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+}|S = a, R = b)$

a,b	
$a = s, \forall b$	$P(F=0)P(H(v,0 \mid F=0))\frac{(v-1)(N-C-v)}{(N-C-1)^2}$
$a = r,$ $b = s$	$P(F=0)P(H(v,0 \mid F=0))\frac{(v-2)(N-C-v)}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\frac{N-C-v}{N-C-k}$
$a = r,$ $b \neq s$	$P(F=0)P(H(v,0 \mid F=0))\frac{(v-2)(N-C-v)}{(N-C-1)^2}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\frac{(N-C-k-1)(N-C-v)}{(N-C-2)(N-C-k)}$
$a \notin \{s,r\},$ $b = s$	$P(F=0)P(H(v,0 \mid F=0))\left(\frac{N-C-v}{(N-C-1)^2} + \frac{(N-C-v)(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\frac{(v-k-1)(N-C-v)}{(N-C-2)(N-C-k)}$ $+P(F=v)P(H(v,0 \mid F=v))\frac{v-1}{N-C-2}$
$a \notin \{s,r\},$ $b = r$	$P(F=0)P(H(v,0 \mid F=0))\left(\frac{N-C-v}{(N-C-1)^2} + \frac{(N-C-v)(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-2} P(F=k)P(H(v,0 \mid F=k))\frac{(v-k-1)(N-C-v)}{(N-C-2)(N-C-k)}$
$a \notin \{s,r\},$ $b \notin \{s,r\}$	$P(F=0)P(H(v,0 \mid F=0))\left(\frac{N-C-v}{(N-C-1)^2} + \frac{(N-C-v)(N-C-3)(v-2)}{(N-C-1)^2(N-C-2)}\right)$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\cdot$ $\left(\frac{(k-1)(N-C-k-1)(N-C-v)}{(N-C-2)(N-C-3)(N-C-k)} + \frac{(v-k-1)(N-C-k-2)(N-C-v)}{(N-C-2)(N-C-3)(N-C-k)}\right)$ $+P(F=v)P(H(v,0 \mid F=v))\frac{(v-1)(N-C-v-1)}{(N-C-2)(N-C-3)}$

Table 12: $P(s \in \overline{\mathscr{V} \cup \{p\}}, r = p, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+}|S = a, R = b)$

a,b	
$a = s, \forall b$	$P(F=0)P(H(v,0 \mid F=0))\frac{N-C-v}{(N-C-1)^2}$
$a = r, b = s$	$P(F=0)P(H(v,0 \mid F=0))\frac{N-C-v}{(N-C-1)^2} + P(F=v)P(H(v,0 \mid F=v))$
$a = r,$ $b \neq s$	$P(F=0)P(H(v,0 \mid F=0))\frac{N-C-v}{(N-C-1)^2}$ $+P(F=v)P(H(v,0 \mid F=v))\frac{N-C-v-1}{N-C-2}$
$a \notin \{s,r\},$ $b \in \{s,r\}$	$P(F=0)P(H(v,0 \mid F=0))\frac{(N-C-3)(N-C-v)}{(N-C-1)^2(N-C-2)}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\frac{N-C-v}{(N-C-2)(N-C-k)}$
$a \notin \{s,r\},$ $b \notin \{s,r\}$	$P(F=0)P(H(v,0 \mid F=0))\frac{(N-C-3)(N-C-v)}{(N-C-1)^2(N-C-2)}$ $+\sum_{k=1}^{v-1} P(F=k)P(H(v,0 \mid F=k))\frac{(N-C-k-2)(N-C-v)}{(N-C-2)(N-C-3)(N-C-k)}$

Table 13: $P(s \in \overline{\mathscr{V} \cup \{p\}}, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = 0, H_{1+} | S = a, R = b)$

a,b	
$a \in \{s,r\}$, $\forall b$	$P(F=0)P(H(v,0\|F=0))\frac{(N-C-v)(N-C-v-1)}{(N-C-1)^2}$
$a \notin \{s,r\}$, $b \in \{s,r\}$	$P(F=0)P(H(v,0\|F=0))\frac{(N-C-3)(N-C-v)(N-C-v-1)}{(N-C-1)^2(N-C-2)}$ $+\sum_{k=1}^{v} P(F=k)P(H(v,0\|F=k))\frac{(N-C-v)(N-C-v-1)}{(N-C-2)(N-C-k)}$
$a \notin \{s,r\}$, $b \notin \{s,r\}$	$P(F=0)P(H(v,0\|F=0))\frac{(N-C-3)(N-C-v)(N-C-v-1)}{(N-C-1)^2(N-C-2)}$ $+\sum_{k=1}^{v} P(F=k)P(H(v,0\|F=k))\frac{(N-C-v)(N-C-v-1)(N-C-k-2)}{(N-C-2)(N-C-3)(N-C-k)}$

Table 14: $P(s = p, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+} | S = a, R = b)$

a,b	
$a = s$, $b \neq r$	$P(F=v)P(H(v,c_F\|F=v))\frac{N-C-v-1+c_F}{N-C-2}$
$a \notin \{s,r\}$, $b \in \{s,r\}$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{N-C-v+c_F}{(N-C-k+c_F)(N-C-2)}$
$a \notin \{s,r\}$, $b \notin \{s,r\}$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{(N-C-v+c_F)(N-C-k-2+c_F)}{(N-C-k+c_F)(N-C-2)(N-C-3)}$

Table 15: $P(s \in \mathscr{V} \setminus \{p\}, r = p, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+} | S = a, R = b)$

a,b	
$a = s$, $b \neq r$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{N-C-k+c_F-1}{(N-C-k+c_F)(N-C-2)}$
$a = r$, $b \neq s$	$P(F=v)P(H(v,c_F\|F=v))\frac{v-1-c_F}{N-C-2}$
$a \notin \{s,r\}$, $b = s$	$\sum_{k=c_F+1}^{v-2} P(F=k)P(H(v,c_F\|F=k))\frac{v-1-k}{(N-C-k+c_F)(N-C-2)}$
$a \notin \{s,r\}$, $b = r$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{v-c_F-2}{(N-C-k+c_F)(N-C-2)}$
$a \notin \{s,r\}$, $b \notin \{s,r\}$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\cdot$ $\left(\frac{(N-C-k+c_F-1)(k-c_F-1)+(N-C-k+c_F-2)(v-k-1)}{(N-C-k+c_F)(N-C-2)(N-C-3)}\right)$

Table 16: $P(s \in \mathscr{V} \setminus \{p\}, r \in \mathscr{V} \setminus \{p\}, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+} | S = a, R = b)$

a,b	
$a = s$, $b \neq r$ $a = r$, $b \neq s$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\cdot$ $\left(\frac{(N-C-k+c_F-1)(v-k-1)}{(N-C-k+c_F)(N-C-2)} + \frac{k-c_F-1}{N-C-2}\right)$
$a = r$, $b = s$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{v-k-1}{N-C-k+c_F}$
$a \notin \{s,r\}$, $b \in \{s,r\}$	$\sum_{k=c_F+1}^{v-2} P(F=k)P(H(v,c_F\|F=k))\frac{v-k-1}{N-C-k+c_F}\cdot$ $\left(\frac{(N-C-k+c_F-1)(v-k-2)}{(N-C-k+c_F-1)(N-C-2)} + \frac{k-c_F-1}{N-C-2}\right)$
$a \notin \{s,r\}$, $b \notin \{s,r\}$	$\sum_{k=c_F+1}^{v} P(F=k)P(H(v,c_F\|F=k))\left(\frac{(k-c_F-1)(k-c_F-2)}{(N-C-2)(N-C-3)}\right.$ $\left.+\frac{(N-C-k+c_F-2)(v-k-1)}{(N-C-k+c_F)(N-C-2)(N-C-3)}\frac{(N-C-k+c_F-1)(v-k-1)(k-c_F-1)}{(N-C-k+c_F)(N-C-2)(N-C-3)}\right)$

Table 17: $P(s \in \mathscr{V} \setminus \{p\}, r \in \overline{\mathscr{V} \cup \{p\}}, ||\mathscr{V}|| = v > 1, C_F = c_F > 0, H_{1+}|S = a, R = b)$

a,b	
$a = s,\, b \neq r$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{(N-C-k+c_F-1)(N-C+c_F-v)}{(N-C-k+c_F)(N-C-2)}$
$a \notin \{s,r\}, b = s$	$\sum_{k=c_F+1}^{v-2} P(F=k)P(H(v,c_F\|F=k))\frac{(N-C+c_F-v)(v-k-1)}{(N-C-k+c_F)(N-C-2)}$
$a \notin \{s,r\}, b = r$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{(N-C+c_F-v)}{(N-C-k+c_F)} \cdot$ $\left(\frac{k-c_F-1}{N-C-2} + \frac{(N-C-k+c_F-1)(v-k-1)}{(N-C-2)(N-C-k+c_F)}\right)$ $+P(F=v)P(H(v,c_F\|F=v))\frac{v-c_F-1}{N-C-2}$
$a \notin \{s,r\}$, $b \notin \{s,r\}$	$\sum_{k=c_F+1}^{v-1} P(F=k)P(H(v,c_F\|F=k))\frac{(N-C+c_F-v)(N-C+c_F-k-1)}{(N-C-2)(N-C-k+c_F)} \cdot$ $\left(\frac{k-c_F-1}{N-C-3} + \frac{(N-C-k+c_F-2)(v-k-1)}{(N-C-3)(N-C-k+c_F)}\right)$ $+P(F=v)P(H(v,c_F\|F=v))\frac{(N-C-v+c_F-1)(v-c_F-1)}{(N-C-2)(N-C-3)}$

Paper F

Peekaboo: A Gray Hole Attack on Encrypted SCADA Communication using Traffic Analysis

Nunzio Marco Torrisi, Ognjen Vuković, György Dán, and Stefan Hagdahl.

In Proc. of IEEE SmartGridComm, November 2014.

Peekaboo: A Gray Hole Attack on Encrypted SCADA Communication using Traffic Analysis

Nunzio Marco Torrisi
Centro de Matemática, Computação e Cognição
Universidade Federal do ABC, Santo André, Brazil 09.210-170
Email: nunzio.torrisi@ufabc.edu.br

Ognjen Vuković, György Dán
School of Electrical Engineering,
KTH Royal Institute of Technology, Stockholm, Sweden
Email: {vukovic,gyuri}@ee.kth.se

Stefan Hagdahl
Security & Defense Solutions
Saab AB, Stockholm, Sweden
Email: stefan.hagdahl@saabgroup.com

Abstract

We consider a potential gray hole attack against SCADA substation to control center communications using DNP3. We propose a support vector machine-based traffic analysis algorithm that relies on message direction and timing information only, and we use trace-based simulations to show that even if SCADA traffic is sent through an encrypted tunnel, as often done in practice, the gray hole attack can be effectively performed based on the timing and direction of three consecutive messages. Our results show that the attacker does not need accurate system information to be successful, and could affect monitoring accuracy by up to 20%. We discuss possible mitigation schemes at different layers of the communication protocol stack, and show that a minor modification of message timing could help mitigate the attack.

1 Introduction

Electric power systems have to be continuously monitored and controlled via Supervisory Control and Data Acquisition (SCADA) systems in order to be kept in a secure operating state. Meters at remote substations measure power flows and voltages, and the measurements are communicated to one or more SCADA control centers over a communication infrastructure using some SCADA communication protocol, such as Distributed Network Protocol 3 (DNP3) [1]. The dynamic visibility provided by SCADA systems has long been important in transmission systems and is becoming more important in power distribution systems, because the proliferation of intermittent distributed generation sources (e.g., solar) results in faster changes in power flows, which in turn requires that protection devices and integrated voltage and VAR control (iVVC) be adjusted in real time.

Motivated by the reliance of power systems on monitoring, estimation and control, a large body of recent work considered the impact of data integrity attacks on power system state estimation, from single systems [2, 3, 4, 5, 6, 7] to interconnected systems [8]. These works assume that the attacker is able to manipulate measurement data in lack of proper authentication.

Authentication is often indeed not possible in legacy remote terminal units (RTUs), and therefore in most SCADA systems the measured data are sent through an encrypted and authenticated tunnel between a substation gateway and the control center. Tunneling protects integrity and may provide confidentiality against an attacker that has access to one or more communication links or routers, and should be used to conform with NERC CIP. Since encryption hides the message contents from an attacker along the tunnel, one would expect that it would also make it impossible for an attacker to identify and to drop mission critical measurement and/or control messages without dropping all messages in a tunnel, and thus remain undetected or difficult to be detected.

In this paper we show through the example of DNP3, one of the two standardized SCADA substation to control center communication protocols, that targeted gray hole attacks may be feasible despite sending messages through an encrypted tunnel. We propose a support vector machine based traffic analysis attack that can distinguish between reports sent spontaneously by an RTU to the control center and messages sent by the RTU in response to messages by the control center. The attack is computationally simple, and is based on the inter-arrival times and directions of consecutive encrypted messages. We use measurement data sets from medium voltage substations to evaluate the effectiveness of the attack and its sensitivity, and to quantify the impact that the attack may have on monitoring accuracy. We finally discuss mitigation schemes to alleviate the attack. Our results give evidence to that the strict timing rules used in SCADA communication protocols facilitate traffic analysis attacks and appropriate countermeasures should be applied.

The rest of the paper is organized as follows. In Section 2 we review related work, and in Section 3 we give an overview of DNP3. We describe the system and the attack model in Section 4, followed by the attack in Section 5. We evaluate the attack and propose a mitigation scheme in Section 6. Section 7 concludes the paper.

2 Related Work

The vulnerability of SCADA systems to cyber attacks has received significant attention recently. In [9], the authors discuss challenges and difficulties of achieving all-encompassing component-level cyber security in power systems due to its cost and potential performance implications. False data injection attacks against common control system communication protocols were considered in [10, 11]; the authors proposed intrusion detection systems to detect the attacks based on neural networks [10] and based on the concepts of critical state analysis and state proximity [11]. Certain false data injection attacks can bypass the bad data detection algorithm used in SCADA state estimators [2], and can thus be used to deceive the system operators regarding the actual state of the system [2, 5, 3, 4, 6, 7]. Mechanism were proposed to protect against these attacks by securing a subset of measurements [5, 3, 4], and by securing a part of the SCADA infrastructure [5, 6, 7]. In [8], the authors showed that false data injection attacks against distributed state estimation in an interconnected power system can disable state estimation in the entire interconnected system, and proposed a detection and a mitigation scheme against such attacks. Our work differs from these recent works as we consider an attack that is limited to dropping messages, and we investigate the effectiveness of such an attack.

Related to ours are works that aim to identify application layer protocols sent through a tunnel using pattern recognition methods [12]. A support vector machine was used in [13] to identify protocols other than HTTP and SSH tunneled over HTTP or SSH by looking at the message size, the block cipher size (involved in the message encryption), and the MTU size. Application-layer protocols sent through an encrypted tunnel that carries traffic from many TCP connections simultaneously were classified in [14, 15] using a *k*-Nearest-Neighbor classifier based on Hidden Markov Models with the message size, the message direction, and message inter-arrival times as features. In [16], the authors compared Bayesian Networks, Decision Trees and Multilayer Perceptrons for the flow-based classification of six different types of Internet traffic, including peer-to-peer and content delivery traffic, and showed the importance of correctly classifying training instances. In [17], the authors proposed an unsupervised machine learning method for network traffic classification based on information entropy techniques. Furthermore, they combined the unsupervised method with a supervised learning method and showed that the combination can improve classification. Unlike these works that aim to identify different protocols in a tunnel, the attack we consider aims at classifying messages that belong to the same application layer protocol, DNP3, and we investigate the ability of such an attack to interfere with SCADA monitoring. To the best of our knowledge ours is the first work to consider a targeted gray hole attack against tunneled DNP3 traffic.

3 DNP3 Background

DNP3 is one of the two standardized communication protocols for SCADA substation to control center communication [1]. Its design follows the master/slave communication model; the *master* is the SCADA master station at the control center and the slaves, called *outstations*, are Remote Terminal Units (RTUs), Intelligent Electronic Devices (IEDs) and Programmable Logic Controllers (PLCs) at the substations.

3.1 Polling vs. Report by exception

DNP3 allows two types of data acquisition, *polling* and *report-by-exception*. In the case of polling, the master solicits data from an outstation and the outstation replies immediately with all data. In the case of report-by-exception the outstation reports only the values that have changed since the last report by more than a predefined threshold, instead of reporting all data. The advantage of this choice is significant saving in bandwidth.

These two types of data acquisition can be combined, and result in four modes of operation for DNP3: (i) polled static, (ii) polled report-by-exception, (iii) quiescent, and (iv) unsolicited report-by-exception. In the case of (i) the master polls and the outstation reports all data. In case of (ii) the master polls but the outstation only reports changed values. In case of (iii) the master does not poll, an unsolicited response is generated by the outstation whenever a value changes by a predefined threshold. In case of (iv) the master polls periodically (typically at a low frequency) and an unsolicited response is generated by the outstation whenever a value changes by a predefined threshold. This last mode is the most commonly used in practice, as it allows for the detection of communication failures and keeps the bandwidth usage low.

3.2 DNP3 over IP

DNP3 includes a link layer specification (addressing, framing, etc), but it can also operate on top of a transport layer protocol, such as TCP and UDP, when used in IP networks [1]. In practice DNP3 is often used over UDP, because using UDP keeps the outstation implementation simple, using UDP does not require many connections to be kept alive in the master station, and if the operator has to pay for the amount of SCADA traffic then using UDP would also be less costly. Furthermore, DNP3 itself implements reliable transmission, hence reliability at the transport layer is not needed.

3.3 Sequence numbers and the Vulnerability

In order to achieve reliable transmission, every message is identified with a sequence number, and message reception is acknowledged so that lost messages can be retransmitted, if needed. For unsolicited responses DNP3 allows two retransmission strategies. One strategy allows the outstation to send a new unsolicited response without receiving the acknowledgement for the previous one, while the other strategy requires the outstation to wait for the acknowledgement before sending a new unsolicited response. An important feature of DNP3 is that the sequence numbers sent by an outstation in unsolicited responses are independent from the sequence numbers used in solicited responses (i.e., in response to polls).

This design choice makes a **gray hole attack** possible: in lack of signaling from the outstation to the master, as long as solicited responses are delivered, the master station can not tell if an attacker drops all unsolicited responses. This is the attack we consider, and we investigate whether the attack can be performed even if the DNP3 messages are sent through an encrypted tunnel, as is usually done in SCADA systems.

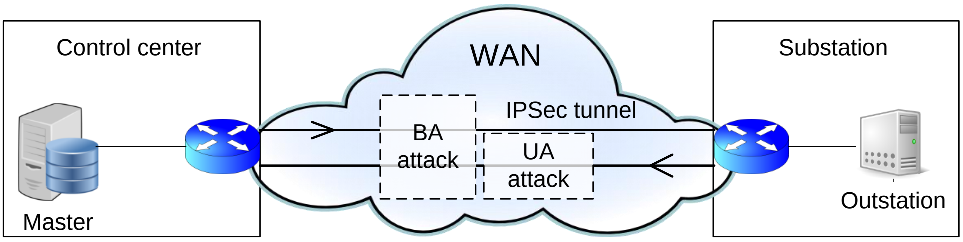


Figure 1: Considered system: Master and outstation communicate using DNP3 through an IPSec tunnel over a WAN.

4 System and Attack Model

We consider a master and an outstation that use DNP3 for communication over a wide area network; the outstation reports measurement data, such as power flow and voltage measurements. DNP3 is used in unsolicited report-by-exception mode, as commonly done in real systems: the outstation reports measurement data by replying to poll messages sent by the master or by sending an unsolicited response when the relative change of a measured value exceeds a configured reporting threshold Δ. We consider a modern WAN deployment, based on the TCP/IP protocol stack, and consider that UDP is used at the transport layer. For reliable delivery the master is configured to send a confirmation message for each unsolicited response it receives. If the outstation does not receive a confirmation for an unsolicited response, the outstation retries sending until the confirmation is received or until the number of retries exceeds a predefined threshold. Since the communication infrastructure is typically not trusted, end-to-end data integrity and confidentiality are achieved through establishing an IPSec tunnel for the DNP3 traffic between the substation gateway and the master station, in ESP mode [18]. In order to avoid non-mission critical data (such as video, voice or engineering data traffic) to interfere with DNP3, the tunnel typically carries DNP3 traffic only. There is thus one IPSec tunnel per DNP3 connection, as shown in Fig 1.

4.1 System Model

We denote the set of polling messages sent by the master by $\mathcal{M}^p = \{m_1^p, m_2^p, \ldots\}$, the set of solicited responses sent by the outstation by $\mathcal{M}^s = \{m_1^s, m_2^s, \ldots\}$, and the set of unsolicited responses (including retranmissions) sent by the outstation by $\mathcal{M}^u = \{m_1^u, m_2^u, \ldots\}$. We denote the set of all DNP3 messages sent by the outstation to the master by $\mathcal{M}^o = \mathcal{M}^s \cup \mathcal{M}^u$ and the set of messages exchanged by the master and the outstation by $\mathcal{M} = \mathcal{M}^p \cup \mathcal{M}^o$.

We denote by t_n^p the time instant when the master sends polling message $m_n^p \in \mathcal{M}^p$ ($n \in \mathbb{N}$), and by $t_n^s > t_n^p$ the time when the outstation replies with solicited response $m_n^s \in \mathcal{M}^p$. The time $t_n^s - t_n^p$ is determined by the one-way delay and the message processing time at the outstation, and is typically rather small compared to the polling period. Similarly, we denote by t_k^u the time when the outstation sends unsolicited response $m_k^u \in \mathcal{M}^u$ ($k \in \mathbb{N}$). If the response is not confirmed, the outstation sends a retransmission $m_{k+1}^u \in \mathcal{M}^u$ at time t_{k+1}^u. Note that the index of a message in a set is determined by the time the message is sent, e.g., $t_{n-1}^p < t_n^p < t_{n+1}^p$.

4.2 Attack Model

The goal of the attacker is to perform a gray hole attack on the data reported by the outstation to the master, while remaining undetected. The attacker has access to a component of the communication network between the substation and the control center, such as a switch, a router or a communication link. The attacker can observe the IPSec tunnels traversing the network component and can identify an IPSec tunnel that carries DNP3 traffic; it can observe the encapsulated DNP3 messages and it can *drop* individual messages. The attacker cannot observe the payload of the messages due to the use of IPsec in ESP mode, but for each message it intercepts it can observe the size of the message's payload, which it can use to differentiate between DNP3 messages and IPsec session management messages, similarly to [13, 14, 19].

Depending on the physical layer technology, the network topology and the routing, the messages sent by the master to the outstation ($\mathcal{M}^p$) and the messages sent by the outstation to the master ($\mathcal{M}^o$) may travel over separate physical links and paths. We therefore consider two models for the attack, the Unidirectional Access (UA) attack and the Bidirectional Access (BA) attack, shown in Fig 1. In the case of the *UA* attack, the attacker can only observe the messages sent from the outstation to the master, i.e., the messages in $\mathcal{M}^o$. In the case of the *BA* attack the attacker can observe the messages sent in both directions, i.e., the messages in $\mathcal{M}$.

Upon intercepting a message the attacker can record the actual time. We denote by t_n^a the time instant when the attacker observes message m_n; in case of the *BA* attack $m_n \in \mathcal{M}$, while in case of the *UA* attack $m_n \in \mathcal{M}^o$.

To perform the attack, the attacker should discard the unsolicited response messages; as long as no unsolicited responses are delivered to the master, the master cannot detect missing sequence numbers, since in DNP3 the sequence numbers are not related in the two directions. To remain undetected, the attacker should discard very few solicited responses as the master can notice the loss of solicited responses (in response to polls). Thus, in order to succeed the attacker has to identify whether an intercepted message is a DNP3 unsolicited response or a DNP3 solicited response. For a sequence of messages, we denote by $\mathcal{M}^{au}$ the set of messages the attacker classifies as unsolicited response.

5 Peekaboo: Binary Classifier Attacks

Clearly, there is a trade-off between correctly classifying the two kinds of messages. We formulate the goal of the attacker as maximizing the probability of correctly classifying an unsolicited response, while keeping the probability of incorrectly classifying a solicited response under a certain threshold c, or formally

$$\begin{aligned} \max \quad & P(m \in \mathcal{M}^{au} | m \in \mathcal{M}^{u}), \\ & \text{s.t. } P(m \in \mathcal{M}^{au} | m \in \mathcal{M}^{s}) < c. \end{aligned} \tag{1}$$

We describe two classes of attack algorithms to solve the problem based on past message inter-arrival times, and if available, based on past message directions.

The considered attacks identify the unsolicited responses by using a support vector machine (SVM) with an appropriately chosen feature space $X \subseteq \mathbb{R}^p$ [20]. Given l training feature vectors $x_n \in X$, $n = 1, \ldots, l$ and for each vector the corresponding class $y_n \in \{-1, 1\}$, an SVM is a supervised learning model that finds a hyperplane w that solves

$$\min_{w, \xi_n, b} \left(\frac{1}{2} |w|^2 \quad + \quad C \sum_{n=1}^{l} \xi_n \right) \tag{2}$$

subject to

$$y_n(w * x_n - b) \quad \geq \quad 1 - \xi_n, \quad n = 1, \ldots, l, \tag{3}$$

where $\xi_n \geq 0$ are slack variables, $C > 0$ is a constant that allows to trade-off between false negatives and false positives, $*$ is an operator that defines the type of the classifier, and b is a scalar. If the operator $*$ is the scalar product, then the classifier is linear and w defines a hyperplane in the feature space. If the operator $*$ is a non-linear kernel function, then the classifier is non-linear, and w defines a hyperplane in the transformed feature space. A widely used non-linear kernel function is the Gaussian radial basis function, for which the transformed feature space is a Hilbert space of infinite dimensions.

Given the trained SVM, i.e., w and b computed, the attacker constructs feature vector x_n for message m_n it intercepts, and decides whether to drop the message based on the sign of $w * x_n - b$. The *UA* and the *BA* attack models differ in terms of the feature space, and are both parameterized by an integer $k > 0$.

UA(k) attack: Under the *UA(k)* attack, the attacker uses the k inter-arrival times between the last $k+1$ messages it observes. The feature vector that corresponds to message $m_n \in \mathcal{M}^o$ is $x_n = (t_n^a - t_{n-1}^a, \ldots, t_{n-k+1}^a - t_{n-k}^a)^T$. The feature space of the SVM in the case of the *UA*(k) attack is $\mathbb{R}^k$.

BA(k) attack: Under the *BA(k)* attack, the attacker uses the k inter-arrival times between the last $k+1$ messages together with the direction of the messages. The feature vector that corresponds to message $m_n \in \mathcal{M}$ is $x_n = (t_n^a - t_{n-1}^a, \ldots, t_{n-k+1}^a - t_{n-k}^a, d_n, \ldots, d_{n-k})^T$, where $d_n \in \{-d, d\}$ for some constant $d > 0$, depending on whether the message is sent by the outstation or by the master, respectively. Since the feature vector includes information about the message direction, the feature space for the *BA*(k) attack is $\mathbb{R}^{2k+1}$.

6 Attack Impact and Mitigation

In the following we evaluate the efficiency of the attacks, we illustrate their potential impact and we consider potential mitigation schemes using traced-based simulations.

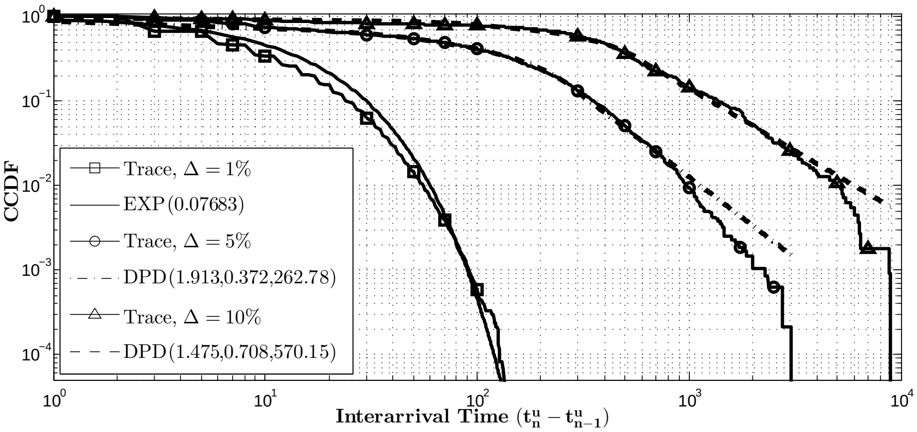


Figure 2: CCDF of unsolicited response inter-arrival times with best fit Double Pareto distributions, DPD(α,β,ω), and Exponential distribution, Exp(λ).

6.1 Measured traces

Our evaluation is based on three measurement data sets collected at medium voltage substations of a European power distribution system operator. The measurements were taken every 3 seconds over 7 consecutive days, and include the voltage and current phasors for the three phases. As RTUs typically report RMS voltage magnitude and active and reactive power flows, we computed these quantities from the traces.

Fig. 2 shows the complementary cumulative distribution function (CCDF) of unsolicited report inter-arrival times, i.e., the CCDF of $t^u_{k+1} - t^u_k$ ($k \in \{1,2,...\}$), assuming three different reporting threshold values $\Delta \in \{1\%,5\%,10\%\}$ based on one of the traces. We observe that the CCDF decays slower for higher values of Δ as unsolicited reports are sent less often due to the higher relative change required to trigger an unsolicited report. It is important to note that the range of inter-arrival times is very wide, between 2 and 4 orders of magnitude and correspondingly the standard deviations are high, 11.2s, 234s, and 932s for $\Delta = 1\%$, $\Delta = 5\%$ and $\Delta = 10\%$, respectively.

The figure shows for each empirical CCDF the CCDF of the best fit double Pareto distribution [21] and the best fit exponential distribution, together with the parameters α, β, and ω, and λ, respectively. The figure shows that for higher threshold values Δ, the double Pareto distribution is a rather good fit and captures large part of the tail. The two regions with different power-law exponents are due to the different power demand dynamics during daytime (fast changing) and nightime (slow changing). For $\Delta = 1\%$ large inter-arrival times are rare because even small power flow and voltage fluctuations trigger unsolicited responses, and thus the exponential distribution seems to provide a very good fit.

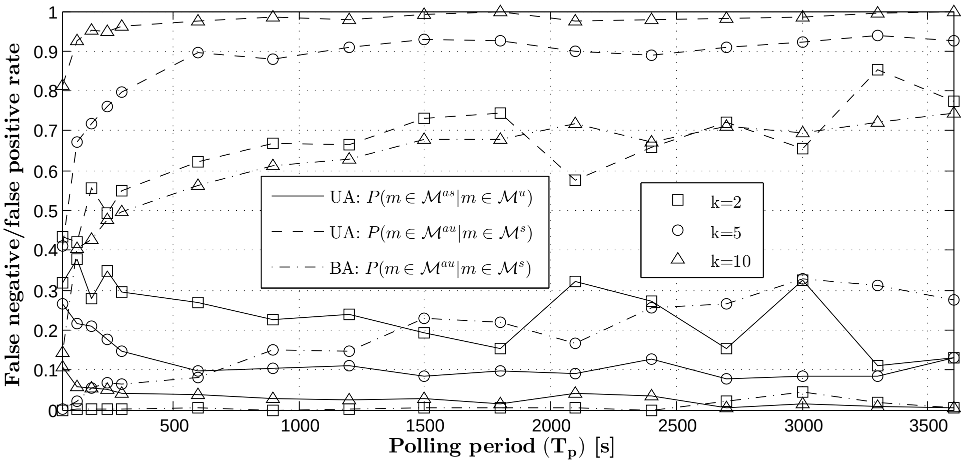


Figure 3: False negative and positive rate vs. polling period.

6.2 Attack Success Rate

We evaluate the efficiency of the attacks for the scenario shown in Fig. 1, i.e., DNP3 traffic exchanged over UDP/IP between an outstation and a master station transmitted through an IPSec tunnel. The unsolicited reports are generated by the outstation based on the measurement data sets in response to voltage magnitude, and active and reactive power flow changes with a threshold of $\Delta = 1\%$. The master is configured to send polling messages every T_p seconds and the outstation sends a solicited report with the most recently measured values immediately after receiving a polling message. The round-trip time (RTT) between the master and the outstation, including the delay due to encryption, authentication and processing at the outstation, equals $1s$ in the baseline scenario.

Fig. 3 shows the false negative and false positive rates $P(m \notin \mathcal{M}^{au}|m \in \mathcal{M}^u)$ and $P(m \in \mathcal{M}^{au}|m \notin \mathcal{M}^u)$, as a function of the polling period T_p for the two classes of attacks for various k values. The kernel function used is the Gaussian radial basis function. The false negative rates and the false positive rates of the *UA(k)* attacks are rather high, which would make the *UA(k)* attacks easy to detect. Interestingly, relying on more messages makes the attack even weaker. The *BA(k)* attacks are, however, very effective. First, the false negative rate of the *BA(k)* attacks is zero (hence it is not shown). Second, the false positive rate is consistently lower for low k. The strongest attack is *BA(2)*, and hence we use it in the sequel. The *BA* attack's efficiency is due to the ability of the attacker to observe the messages in both directions; intuitively any report coming from the outstation shortly after a polling message is classified as a solicited request, and all others as unsolicited requests. Thus, for an attack to be successful, the attacker needs to be able to observe messages sent in both directions.

The results in Fig. 3 were obtained assuming that the attacker knows the (RTT) between

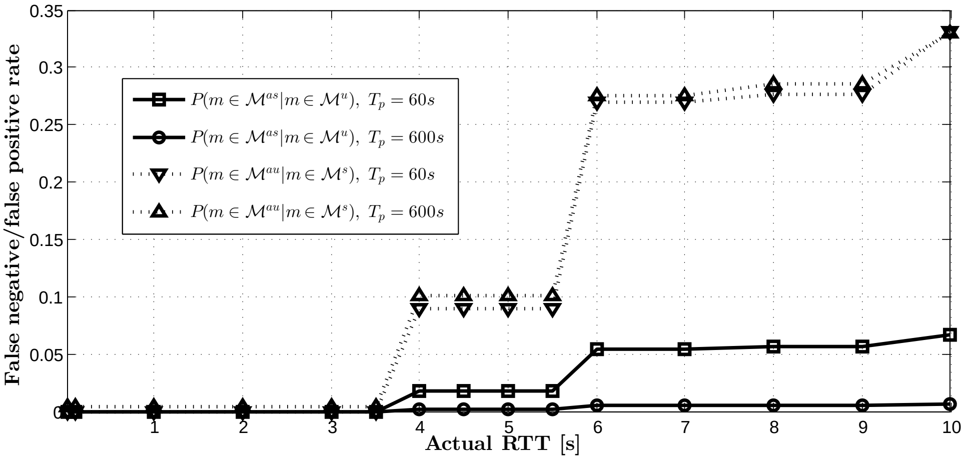


Figure 4: False negative and positive rate vs. actual RTT. SVM is trained for RTT=1s.

the master and the outstation. Figure 4 shows the sensitivity of the false negative and of the false positive rate for the *BA(*2*)* attack using an SVM that was trained with RTT=1s as a function of the actual RTT. The figure shows that the *BA(*2*)* attack is effective as long as the actual RTT is below 4s, i.e., as long as the attacker's estimate of the RTT is within a factor of four, which is a rather wide margin of error. Above a factor of four the false negative and the false positive rates start to increase and the attack could be detected. The stepwise increase in the misclassification rates at RTT 4s and 6s is due to that measurements in the data sets were taken every 3s.

6.3 Attack Impact

The results so far show that the *BA* attack could effectively be used for selectively dropping unsolicited reports and this way blind an operator. We quantify the potential effect of the attack on the situational awareness of an operator through the error that the attacker would introduce in the power flow measurements available to an operator under the attack. We define the error at time t as the difference between the measured value $P(t_n^s)$ received in the most recent solicited response and the measured value $P(t_k^u)$ the operator should have received in the most recent unsolicited response (had it not been dropped by the attacker), i.e., for $t_n^s < t < t_{n+1}^s$

$$E_P^a(t) = \begin{cases} P(t_k^u) - P(t_n^s) & if\ \exists t_k^u \text{ s.t. } t_n^s < t_k^u \leq t < t_{k+1}^u, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

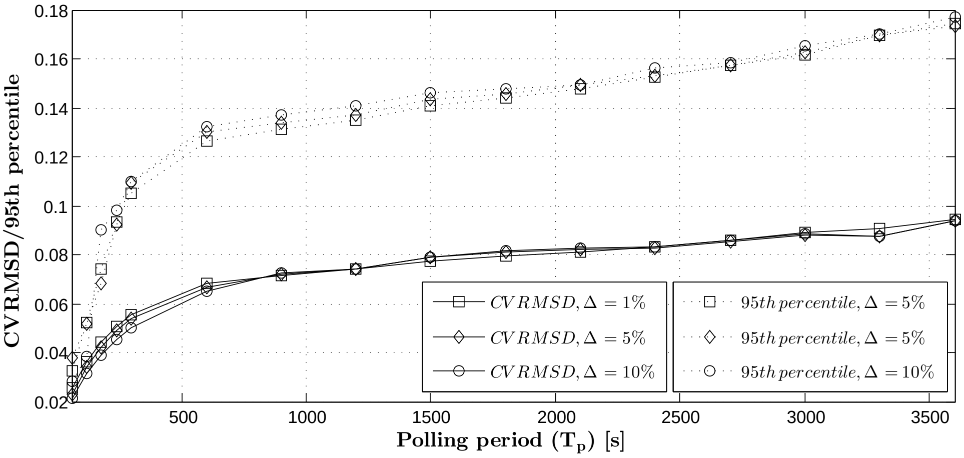


Figure 5: $CV_{RMSE}(E_P^a(t))$ and $NRSE(E_P^a(t),t)$ for an active power flow and the *BA* attack vs. T_p. $\Delta \in \{1\%, 5\%, 10\%\}$.

We define the mean squared error over the time interval $[t_1,t_2]$ as $\overline{E_P^a(t)^2} = \frac{1}{t_2-t_1}\int_{t_1}^{t_2} E_P^a(t)^2$, and the coefficient of variation of the root mean squared error as

$$CV_{RMSE}(t_1,t_2) = \sqrt{\frac{\overline{E_P^a(t_1,t_2)^2}}{\overline{P(t_1,t_2)}^2}}, \tag{5}$$

where $\bar{\cdot}$ stands for the mean. In practice, reacting to sudden short changes of $P(t)$ is important for proper operation of the power system, we therefore also compute the normalized root squared error for every time instant as

$$NRSE(t) = \sqrt{\frac{E_P^a(t)^2}{\overline{P(t))}^2}}. \tag{6}$$

Fig. 5. shows CV_{RMSE} and the 95th percentile of $NRSE(t)$ over the 7 days measurement period as a function of the polling interval T_p for one of the active power flow measurements for the *BA* attack (the attacker successfully drops all unsolicited responses). Both the mean and the 95 percentile increase monotonically with T_p, with a decreasing marginal gain. These results indicate that under an attack the operator's observation of the active power flow would be almost 20% off in 5% of the time and it would be on average up to 10% off. Interestingly, the results are not sensitive to the reporting threshold Δ.

6.4 Attack Mitigation

Motivated by the effectiveness of the *BA* attack and its potential impact, we finally discuss a number of mitigation schemes. At the transport layer one could mitigate the attack by

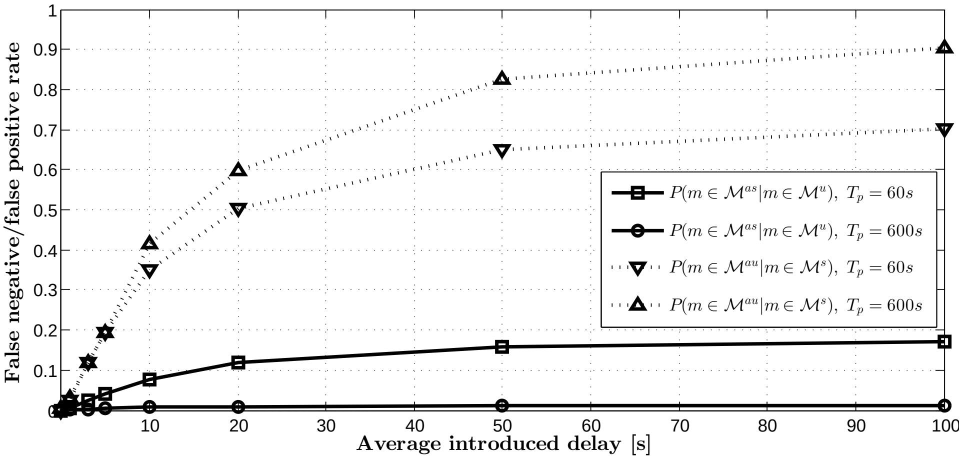


Figure 6: False negative and positive rate vs. average introduced delay.

using TCP, as the attack would cause head of line blocking and would lead to a reset of the TCP connection. This mitigation may, however, not be feasible if the legacy equipment does not support TCP or server resources are insufficient.

At the application layer, the DNP3 solicited response could be extended by a field that contains the sequence number of the most recently sent unsolicited response. As an alternative, the outstation could introduce a random delay before sending a solicited response (in response to a poll). The random delay would make an attack using statistical pattern recognition more difficult. To assess this latter mitigation scheme, Figure 6 shows the false negative rate and the false positive rate as a function of the average delay introduced in the outstation for the case of an exponential distribution and the *BA(*2*)* attack. The choice of the exponential distribution is motivated by the observation that the inter-arrival times of unsolicited responses are well modeled by an exponential distribution for a small reporting threshold. The false negative and the false positive rates increase with a decreasing marginal gain with the introduced delay, and for a relatively small average delay of a few seconds they would be high enough for the *BA(k)* attack to be detected. An interesting open question is whether such delays would be compatible with legacy SCADA masters.

7 Conclusion

We addressed the vulnerability of SCADA communication to a gray hole attack, in which an attacker drops unsolicited reports sent by an outstation to a SCADA master, while letting through solicited reports in order to avoid detection. We showed that such a gray hole attack is possible even if messages are sent through an encrypted tunnel, because due to the strict timing rules used in SCADA protocols traffic analysis can effectively be used to classify protocol messages. We proposed a support vector machine based traffic analysis algorithm, used trace-based simulations to evaluate the attack, and showed that an attacker would not need exact knowledge of system parameters for a successful attack. We quantified the impact of the attack in terms on monitoring accuracy, and showed that the operator's observation can be up to 10% off on average, and up to 20% off in 5% of the time. Finally, we discussed potential mitigation schemes, and showed that the attack can be mitigated by introducing a random delay before answering to poll messages.

Acknowledgment

N. Torrisi was supported by the Swedish-Brazilian Research and Innovation Centre (CISB) and by the SAAB group (Grant CNPq-SAAB No. 236756/2012-5). Dán and Vuković were supported by EIT ICTLabs activity SES 14306.

References

[1] DNP3 IEEE WG, "IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3)," *IEEE Std 1815-2012 (Revision of IEEE Std 1815-2010)*, pp. 1–821, 2012.

[2] Y. Liu, P. Ning, and M. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proc. of the 16th ACM conference on Computer and Communications Security (CCS)*, 2009, pp. 21–32.

[3] R. B. Bobba, K. M. Rogers, Q. Wang, H. Khurana, K. Nahrstedt, and T. J. Overbye, "Detecting false data injection attacks on dc state estimation," in *Preprints of the First Workshop on Secure Control Systems, CPSWEEK, Stockholm, Sweden*, April 2010.

[4] T. T. Kim and H. V. Poor, "Strategic protection against data injection attacks on power grids," *IEEE Trans. on Smart Grid*, vol. 2, no. 2, pp. 326–333, Jun 2011.

[5] G. Dán and H. Sandberg, "Stealth attacks and protection schemes for state estimators in power systems," in *Proc. of IEEE SmartGridComm*, October 2010.

[6] A. Giani, E. Bitar, M. Garcia, M. McQueen, P. Khargonekar, and K. Poolla, "Smart grid data integrity attacks: Characterizations and countermeasures," in *Proc. of IEEE SmartGridComm*, October 2011.

[7] O. Vuković, K. C. Sou, G. Dán, and H. Sandberg, "Network-aware mitigation of data integrity attacks on power system state estimation," *IEEE JSAC: Smart Grid Communications Series*, vol. 30, no. 6, pp. 176–183, July 2012.

[8] O. Vuković and G. Dán, "Detection and localization of targeted attacks on fully distributed power system state estimation," in *Proc. of IEEE SmartGridComm*, October 2013, pp. 390–395.

[9] G. Dán, H. Sandberg, G. Björkman, and M. Ekstedt, "Challenges in power system information security," *IEEE Security and Privacy*, 2011.

[10] W. Gao, T. Morris, B. Reaves, and D. Richey, "On SCADA control system command and response injection and intrusion detection," in *eCrime Researchers Summit (eCrime), 2010*. IEEE, 2010, pp. 1–9.

[11] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. N. Fovino, and A. Trombetta, "A multidimensional critical state analysis for detecting intrusions in SCADA systems," *IEEE Trans. on Industrial Informatics*, vol. 7, no. 2, pp. 179–186, 2011.

[12] K. Fukunaga, *Introduction to statistical pattern recognition*. Academic Press, 1990.

[13] M. Dusi, M. Crotti, F. Gringoli, and L. Salgarelli, "Tunnel hunter: Detecting application-layer tunnels with statistical fingerprinting," *Computer Networks*, vol. 53, no. 1, pp. 81–97, 2009.

[14] C. V. Wright, F. Monrose, and G. M. Masson, "On inferring application protocol behaviors in encrypted network traffic," *J. Mach. Learn. Res.*, vol. 7, pp. 2745–2769, Dec. 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1248547.1248647

[15] G. Maiolini, A. Baiocchi, A. Iacovazzi, and A. Rizzi, "Real time identification of ssh encrypted application flows by using cluster analysis techniques," in *Proc. of IFIP/TC6 Networking*, 2009, pp. 182–194.

[16] M. Soysal and E. G. Schmidt, "Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison," *Performance Evaluation*, vol. 67, no. 6, pp. 451–467, 2010.

[17] J. Yuan, Z. Li, and R. Yuan, "Information entropy based clustering method for unsupervised internet traffic classification," in *Proc. of IEEE ICC*, 2008, pp. 1588–1592.

[18] S. Kent, "IP Encapsulating Security Payload (ESP)," RFC 4303 (Standard Track), Internet Engineering Task Force, 2005. [Online]. Available: http://www.ietf.org/rfc/rfc4303.txt

[19] X. Tan, X. Su, and Q. Qian, "The classification of ssh tunneled traffic using maximum likelihood classifier," in *Proc. of IEEE Conf. on Electronics, Comm. and Control (ICECC)*, 2011, pp. 2347–2350.

[20] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[21] W. J. Reed and M. Jorgensen, "The double Pareto-lognormal distribution - a new parametric model for size distribution," *Communications in Statistics - Theory and Methods*, vol. 33, no. 8, pp. 1733–1753, 2004.