Trust Management and Security in the Future Communication-Based "Smart" Electric Power Grid

Jose Fadul[1], Kenneth Hopkinson[1], Christopher Sheffield[1], James Moore[2] and Todd Andel[1]
[1]Department of Electrical & Computer Engineering, Air Force Institute of Technology, Dayton, OH, 45433, USA
[2]Department of Operational Sciences, Air Force Institute of Technology, Dayton, OH, 45433, USA
{jose.fadul, kenneth.hopkinson, christopher.sheffield.ctr, james.moore, todd.andel}@afit.edu

Abstract

New standards and initiatives in the U.S. electric power grid are moving in the direction of a smarter grid. Media attention has focused prominently on smart meters in distribution systems, but big changes are also occurring in the domains of protection, control, and Supervisory Control and Data Acquisition (SCADA) systems. These changes promise to enhance the reliability of the electric power grid and to allow it to safely operate closer to its limits, but there is also a real danger concerning the introduction of network communication vulnerabilities to so-called cyber attacks. This article advocates the use of a reputation-based trust management system as one method to mitigate such attacks. A simulated demonstration of the potential for such systems is illustrated in the domain of backup protection systems. The simulation results show the promise of this proposed technique.

1. Introduction

There have been a number of significant efforts in recent years to replace legacy protection, control, and Supervisory Control and Data Acquisition (SCADA) systems in the electric power grid with modern communication network alternatives. Legacy technology uses relatively limited communication and proprietary protocols. New efforts based on a modern communication network approach, and often employing Internet protocols and equipment, include Utility Communications Architecture version 2.0 (UCA 2.0) [1] in the 1990s followed by the International Electrotechnical Commission's (IEC) 61850 standard [2] and more recently, the Wide Area Measurement System [3], North American SynchroPhasor Initiative (NASPI) [4], and the general push towards "smart grid" technology [5]. While the media's portrayal of smart grid is often synonymous with smart metering, a more expansively definition encompass modern technologies for the power production and transmission system (aka power grid), such as those mentioned above. The holistic result of smart grid technology is likely to be protection and control equipment that is more reliable and aware than its predecessors. SCADA systems leveraging these smart grid technologies will be capable of faster sweeps through the grid with higher data transfer rates and may include information not previously provided.

The term "smart" in smart grid is perhaps misleading. It evokes images of a grid that adapts its behavior through learning using artificial intelligence to improve its performance over time with little user monitoring or control. This type of smart grid is unlikely since it would be too unpredictable to trust. A more realistic view of the smart grid is that protection and control schemes that previously operated on a stand-alone basis with limited or no communication will be able to use network communication capabilities to gain context-awareness about the regional status of the grid. This context-awareness has the potential to greatly improve the effectiveness of the protection and control schemes in question. A disadvantage is that this new reliance on communication and digital software may unleash large vulnerabilities to malicious network attacks, i.e. viruses, and other such problems seen in the Internet.

This article advocates the use of a reputation-based trust management system to mitigate network vulnerabilities in future smart electric power grids. A basic approach towards this end is described using a communication-based backup protection system as an example of how a trust management system can be introduced to smart grid devices. A simulation is also conducted to illustrate the benefits of the reputation-based trust management system.

The views expressed in this document are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

1530-1605 2011

2. Background

Smart Grid enabled technologies are needed to help mitigate the growing energy demands on the United States (U.S.) electric power grid. The increasing energy demands are attributed to an increasing population size [6] [7] and technological improvements [8]. The U.S. population is expected to increase by 0.9 percent annually [7]. This increase in population also increased the demands place on the nation's power grid. Furthermore, increases in the nation's standard of live and improved medical technologies [8] have also increased the demands placed on the nation's power grid. The increased energy demands have stressed the U.S. power grid and amplified the effects of power outages [9]. Smart Grid enabled technologies, such as demand response [10] and microgrid technologies [11], are expected to mitigate some of the stress placed on the U.S. power grid [12] and reduced the impacts of future power outages. Demand response [10] advanced metering infrastructure (AMI) attempts to optimize the power grid's efficiency by enabling the average consumer to automatically control how much energy they draw, from the power grid, based on energy costs. Microgrid technologies [11] refers to small power grids with local producers of energy, such as wind mills or solar panels, which draw little to no energy from the nation's power grid in order to supply their customers.

Smart Grid technology is currently in its infancy with its first report delivered to Congress this year, 2010 [5]. The holistic improvements of Smart Grid technology is achieved in part by leveraging internet like packet-switch network technologies to increase the nation's electrical power grid communications capabilities. This improvement logically follows previous SCADA communication efforts, such as UCA 2.0 [1], IEC 61850 standard [2] and NASPI [4]. These previous SCADA efforts focused on improving certain SCADA system limitations, such as the lack of standardization between SCADA communication standards. This should improve SCADA component interoperability, but also increase the SCADA system's susceptibility to cyber attacks.

In this article, cyber security risks is defined as cyber attacks that misuse information generated within the smart electric power grid. This broad definition encompasses customer profiling, Internet Protocol (IP) spoofing, Man-in-the-middle (MITM), denial of service (DOS) and system hijacking attacks. Smart grid power usage information can be used by thieves to determine when residential customers are and are not home. IP spoofing can be used to redirect smart grid information to a cyber attacker's computer system. IP spoofing can caused a smart grid communication node to appear unresponsive and faulty—lowering its assigned trust value in the proposed technique. MITM attacks covertly monitor or manipulate smart grid information between two smart grid communication nodes. DOS attacks prevent smart grid services from responding to legitimate requests. System hijacking is gaining unauthorized remote access to a smart grid communication node. This article advocates the use of a reputation-based trust management system to mitigate some cyber attacks; namely, system hijacking.

3. Reputation-based trust

The idea behind reputation-based trust is that shared sensor readings can be evaluated by peers to determine the trustworthiness of an individual peer. This is different from an internal monitor, which is sometimes used in trusted computing, to evaluate whether components are operating correctly. In this article, the reputation-based trust algorithm used is a majority rule trust algorithm, where trust values are assigned to individual nodes based on received information from multiple sources. The sensor node/relays are partitioned into overlapping network neighborhoods. The nodes within these neighborhoods monitor common power grid entities, i.e. a common power line. These nodes share information concerning the state of the power line from their point of view, i.e. their sensor readings. The nodes that agree with the trusted majority are assigned a high trusted value and those that disagree are assigned a low trust value. The trusted nodes can trust the sensor readings and make decision accordingly, i.e. a trusted node can safely disregard a breaker trip request when its sensor readings are within tolerance and indicate that no fault has occurred. This is how this reputation-based trust management system mitigates some network vulnerabilities, such as a false trip breaker signal.

The information shared by these nodes in the experiment run in this article are voltage and current tolerance values. Power line losses cause the voltage and current value to differ from power grid location to power grid location, i.e. sensor readings are not identical throughout the power grid. Using tolerance results provides a means for accounting for these power line losses. Power line losses are caused by power line impedance, which in this article includes power line component losses, such as power transformers and capacitors. In this article, the line impedances between nodes are considered constant.

These constant impedances are used to establish voltage and current tolerances at each node. The information shared by the nodes is an ordered set of binary values, where a 1 indicates the corresponding voltage or current value is within tolerance and a 0 indicates it is not within tolerance. A previously developed simple trust algorithm [13] uses this shared information to determine the trust values of the individual nodes. These trust values are shared with all nodes defined as peers to determine which nodes to call upon for backup protection support.

4. Trust management

The central premise in this article is that reputation-based trust levels, once determined, can be used as inputs in electric power protection and control schemes to make decisions that can allow "smart", or context-aware, elements to make better decisions. These smart devises (aka Intelligent Electronic Devices (IEDs)) can help circumvent parts of the system that have been disabled or otherwise compromised through accidental or malicious circumstances. Fundamental algorithms, such as Dijkstra's shortest-paths [14] and network flows [15], are used in this trust management system to optimize the decision making process and minimize the damage associated with a detected fault. The main idea behind such a system is that there is usually a trade-off to be made between level of trust and the optimality of the system, where the definition of optimal varies based on the type of protection or control scheme in question. Optimization algorithms can be simple enough to run on an IED and yet allow good decisions regarding these fundamental trade-offs.

The proposed reputation-based trust management system increases the current level of system complexity. This trust management system (TMS) requires additional memory, processing power and network bandwidth. IEDs are expected to provide the additional memory and processing resources, while the increased communication capabilities of Smart Grid technologies shall satisfy the bandwidth requirements. A small example is used to explore these complexities, but a larger real world scenario should be simulated before practical implementation is attempted.

5. A communication-based backup protection system (BPS)

Zone 3 backup relay-breaker assemblies are required to clear a fault when the zone 1 primary relay-breaker assemblies fail or become inoperable [16], where the term relay-breaker assembly includes the relay and breaker assemblies. Hence, a relay-breaker assembly failure is indicative of a failed relay, breaker or both. Throughout this article the term relay-breaker assembly is used to signify a relay and breaker assembly. Traditional backup protection systems (BPS) have many challenges to overcome. First, the region they isolate is often larger than it need be. Second, they traditionally act without explicit intra-communication. The need for small isolated regions imposes long lag times for zone 3 relay-breaker assemblies, which is a major cause of power system instability.

A new agent-based design for zone 3 backup protection relay-breaker assemblies was first introduced by Wang et al. in [17]. The agent-based relay-breaker assemblies are able to communicate their relay status information, breaker trip signal events, and local measurements when a zone 1 primary protection event occurs. The agent-based protection system has many benefits over traditional systems. Zone 3 relay-breaker assemblies can be monitored to allow corrections to prevent false breaker trips. These corrections have the potential to greatly reduce the number of incorrect trips in cases where there is a heavy load. In addition, in the case of both primary and local backup relay failure, the agent can locate the faulted line using notification messages and can send a trip signal to clear the faulted line. Traditional backup systems clear such faults using remote backup relay-breaker assemblies with a bigger isolated region and with a greater time delay. If an incorrect primary relay trip is found by an agent, then a block signal can be sent to stop an unwarranted breaker trip. In most bus arrangements, breaker failure protection schemes trip all breakers connected in the same bus and induces a big disturbance leading to poor overall reliability. The reliability gains from an agent-based protection system can be significant in those cases.

The proposed TMS is representative of the types of benefits that arise when communication is added to protection and control systems. By adding context-awareness over much wider areas than typical devices, new smart schemes are able to performing much better than their traditional counterparts. The drawback is that network-based communication introduces the same types of vulnerabilities present in networks like the Internet. These threats must be managed for communication-based smart schemes to be practical.

6. Simulation environment

To illustrate the potential of such a trust management system in electric power protection, a reputation-based trust management system is applied to a communication-based backup protection system

inside a simulated environment. The ***e***lectric ***po***wer and ***c***ommunication sync***h***ronizing ***s***imulator (EPOCHS) [18] is used in these simulated experiments. EPOCHS federates, or combines, General Electric's (GE's) Positive Sequence Load Flow (PSLF) [19], Siemen's Power System Simulation for Engineering (PSS/E) [20] electromechanical transient simulators, HVDC Manitoba's Power Systems Computer Aided Design / ElectroMagnetic Transients including Direct Current (PSCAD/EMTDC) simulator [21], and the University of California at Berkeley's network simulator 2 (ns2) [22] together to allow users to study electric power protection and control systems that depend on network communication [18]. In this article, EPOCHS is used with PSCAD/EMTDC to simulate electromagnetic transient situations and ns2 is used to simulate smart grid technologies' network communication capabilities. These individual simulators are seamlessly integrated from a modeler's perspective.

Software agents, called sensor node/relays, are created to mimic the behavior of real systems. These agents can access and modify the power line, relay and breaker data maintained in the electric power simulator and communicate with other agents via the ns2 network simulator. These sensor node/relays are able to interact within an integrated world containing both electric power and networking state, i.e., these agents can take local measurements, set electrical state, and can send or receive messages across a communication network. The run-time infrastructure (RTI) module ensures the simulators are properly synchronized so that if an event happens at simulation time *t* in the power simulator, then it occurs at this same time in the network simulator and vice versa. Figure 1 gives an overview of the EPOCHS simulation system.

EPOCHS has been used in previous experiments to

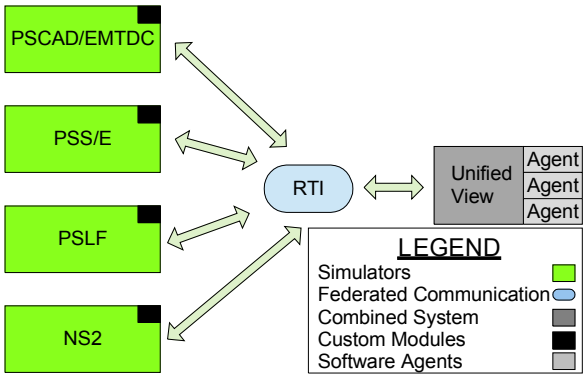

Figure 1. The EPOCHS simulation system [18]

show how network communications, provided by a Utility Intranet such as the smart grid wide area utility network, could be used to enhance the capabilities of protection and control systems. These previous experiments include past work that looked at zone 3 backup protection relay-breaker assemblies augmented with such communication capabilities [18]. The current experiments in this article use EPOCHS to show the promise and potential pitfalls of the proposed reputation-based TMS implementation within the next-generation smart electric power grid.

7. Simulation scenarios

Three simulation scenarios are presented in this article. The first scenario illustrates that the proposed TMS does not interfere with the primary relay-breaker assemblies' protection functions. The second scenario simulates a shorted power line in a SCADA power grid containing trusted and untrusted sensor node/relays. The third scenario simulates a cyber attacker's attempt to cause a power outage by gaining unauthorized remote access of a single sensor node/relay. These three scenarios are mitigated by the proposed reputation-based TMS.

All three scenarios are based on the communication-based BPS, which was described in section 5. A BPS engages circuit breakers to isolate power line faults when the primary protection systems fails or becomes inoperable. This implies that the proposed reputation-based TMS must not interfere with the primary protection system. The first scenario (see Figure 2 and Figure 3) illustrates that the proposed TMS does not interfere with the primary protection system.

Figure 2 represents an electric power grid with 2 generators and 8 trusted sensor node/relays. The "X" between sensor node/relays *S5* and *S6* represents a fault on the power line between these two nodes. The high trust values at all the nodes indicated that the primary protection system mitigate the faulty line by opening the line breakers located at nodes *S5* and *S6*. The proposed TMS does not interfere with the primary protection system and reaches the same conclusion.

The reputation-based TMS uses the power grid topology in Figure 2 to generate the graph in Figure 3. Four fictitious nodes are created and added to the graph; namely a super source, super sink, left junction and right junction nodes. The super source and super sink nodes are the starting and ending nodes for Dijkstra's Shortest Path algorithm [14], respectively. The left and right junction nodes represent the starting nodes for the nodes on the left and right side of the detected fault, respectively. The detected fault is located between nodes *S5* and *S6*. Hence, the nodes on

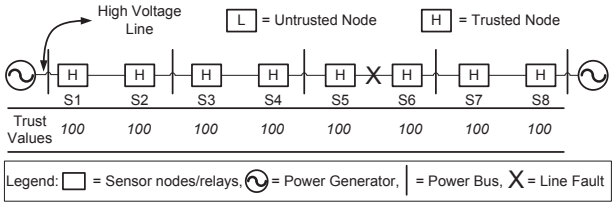

Figure 2. Primary protection system example

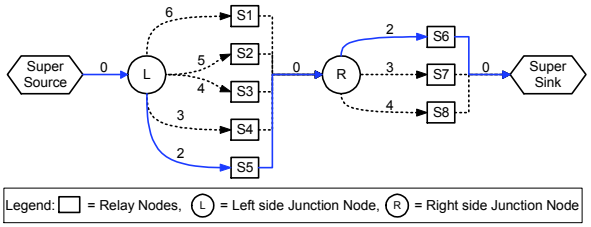

Figure 3. Graph for Figure 2

the left side of the fault are *S1* through *S5* in series in Figure 2 and appear in parallel in between the left and right junction nodes in Figure 3. The nodes on the right side of the fault are *S6* through *S8* in series in Figure 2 and appear in parallel in between the right junction and the super sink nodes in Figure 3. The edge values for the generated graph in Figure 3 are determined as follows:

1. the edges entering a fictitious node are assigned a value of zero
2. the edges entering relay nodes are assigned values based on their distance from the fault and their assigned trust values.

Trust values of 100 add a 1 to the edge values entering the relay nodes. The distance from the fault is determined by a breadth first search algorithm with the results added to the edge values entering the relay nodes. The closest nodes to the fault are *S5* and *S6*, i.e. these two nodes are 1 hop away from the fault. Hence, the edge values for *S5* and *S6* is 2, i.e. 1 for the hop distance plus 1 for the trust value of 100. Similarly, nodes *S4* and *S7* are 2 hops away from the fault with trust value of 100 and are assigned edge values of 3. The remaining relay node entering edge values are shown in Figure 3. Now Dijkstra's Shortest Path algorithm [14] is used to determine the shortest path from the super source node to the super sink node. The resulting path traverses relay nodes *S5* and *S6*. This indicates that the primary relay nodes *S5* and *S6* should open their associated line breakers to isolate the fault. This is in line with allowing the primary protection relay-breaker assembly a chance to mitigate the fault.

In scenario 2 the same line fault and power grid topology is considered with lower trust values assigned to sensor node/relays *S4*, *S5* and *S6*. This is illustrated in Figure 4 and Figure 5.

Figure 4 represents the same power grid topology with the same detected power line fault between nodes *S5* and *S6*—as in Figure 2. The difference between these two figures is the assigned sensor node/relay trust values. In Figure 4, nodes *S4*, *S5* and *S6* are untrusted with trust values of 10%, 10% and 40%, respectively. These lower trust values correspond with the higher edge costs in Figure 5. This causes Dijkstra's algorithm to select a different path from the super source node to the super sink node. This new path is highlighted in blue in Figure 5 and indicates that the line breakers associated with *S3* and *S7* should be opened to clear the fault. These two nodes, *S3* and *S7*, are the closest trusted nodes to the detected line fault. The selection of these two nodes by the backup protection system minimizes the affected service area and the associated damages. The details on how the graph is constructed from a given power grid topology with a detected line fault is explained in section 8.

The third scenario considers the cyber threat associated with hijacking a sensor node/relay. In this scenario an attacker gains remote control of a sensor node/relay. The attacker attempts to trip the hijacked node's associated breaker by initiating a relay trip signal. If the hijacked node is considered trusted, then it would attempt to confirm the trip signal internally via its sensor readings and error detectors. Note: an unconfirmed trip signal is blocked (i.e. ignored) by sensor node/relays. If the hijacked node is not trusted, then it would wait for confirmation from two external trusted nodes via broadcast messages. The internal sensor readings and error detectors cannot be modified remotely. Unwarranted broadcast messages indicate the possible presents of a cyber attacker and alert the power grid control center of a probable attack in

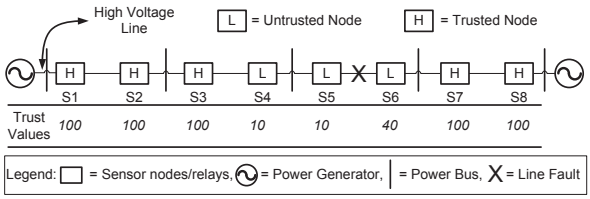

Figure 4. Power grid network connectivity

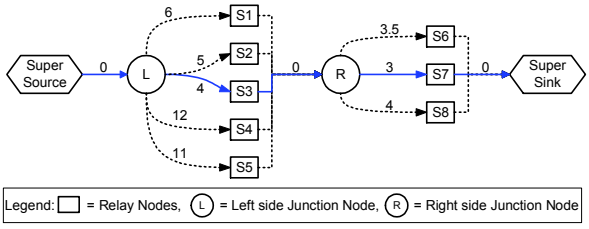

Figure 5. Graph for Figure 4.

progress. This process, see Figure 6, provides some protection against a hijacking type cyber attack.

Traditional legacy SCADA systems cannot disobey a given trip signal. Hence, an attacker gaining remote control of a SCADA node can cause its associated breaker to open and deny power to a target area.

8. Creating the graph

Creating the graph solved by the shortest path algorithm in the proposed TMS requires 1) knowledge of the power grid topology, 2) knowledge of all sensor node/relays' assigned trust values, and 3) knowledge of the location of the detected line fault. The first requirement, "knowledge of the power grid topology," is provided by SCADA system operators or a network discovery program. This information is fairly static and need only be updated when changes occur.

The second requirement, "knowledge of all sensor node/relays' assigned trust values," is satisfied by the Simple Trust algorithm [13] (a reputation-based trust algorithm). The power grid is partitioned into overlapping network neighborhoods containing sensor node/relays, gateway nodes and a process control system (PSC) node, see Figure 7. Figure 7 is a notional illustration of our concept. Care must be given to scalability and reliability in designing solutions for more realistic systems. In Simple Trust [13], the sensor node/relays within a neighborhood monitor common entities, such as the same power line. Some sensor node/relays, called gateway nodes, are elected to receive sensor data from all the sensor node/relays within its neighborhood and assigns corresponding trust values. If two or more gateway nodes disagree on a given sensor node/relay's trust value, then a higher level node, called the PCS node, is provided the gateways' data for arbitration. The gateway and PCS nodes use the simple trust algorithm to determine each node's trust value. Each sensor node/relay determines if its sensor data is within its tolerance values. These tolerance results are shared with its assigned gateway nodes. The gateway nodes use the tolerance results from trusted nodes to establish a consensus regarding the state of the common monitored entities—e.g. is the power line functioning within tolerance. The sensor node/relays results that agree with the consensus are considered trusted (assigned a high trust value) and those that disagree with the consensus are not trusted (assigned a low trust value). Network flow techniques are used to prevent false positives and false negatives trust assignments, where a false positive is defined to be a non-malicious trust agent assigned a low trust value and a false negative is defined as a malicious trust agent assigned a high trust value. Power grid operators and technicians are informed of sensor node/relays with low trust values for appropriate maintenance actions.

The calculated trust values are indirectly used in the objective function shown in Equation 1 below. These trust values are updated on a recurring basis, normally within 12 milliseconds (ms)—the simulations in this article used a simulation time step of 2 ms. Traditional SCADA systems exchange routine message within 540 ms [24] [25] [26]. It is possible for a trusted node to be compromised between updates, resulting in a false negative. Additional research is needed to resolve this small limitation. Also, this proposed technique is meant to run in parallel with the traditional relay protection system. This redundancy will help address this "false negative" occurrence. The malicious node's

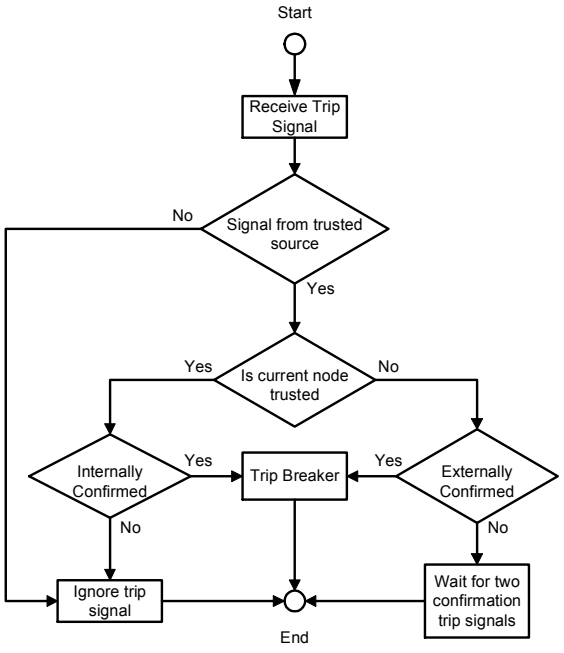

Figure 6. Receive trip signal flowchart

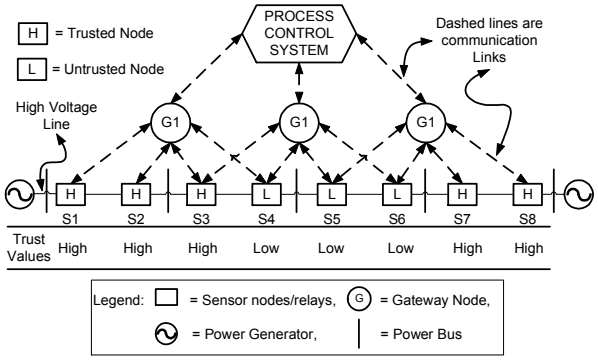

Figure 7. Abstract simple trust algorithm example [23]

failure to respond to a valid trip request would be detected by both the traditional system and the proposed system. This results in both systems initiating corrective action to mitigate the detected fault.

The third requirement, "knowledge of the location of the detected line fault," is provided by the sensor node/relays detecting the fault. This information is used to construct the graph traversed by Dijkstra's shortest path algorithm [14].

Once the constructed graph is created with the trust values are established for the relay-breaker assemblies in the system, then a means of balancing trade-off considerations, such as distance versus trust, is required. Trade-off preferences need to be set by the operators of the system since some may assess the trade-off differently than others. This is accomplished by two weighting variables α and β. The α variable is the weighting factor for distance from the detected fault location and β is the weighting factor for component trust values. The weighted assigned trust values combined with the weighted breakers' distance from the faults is used to quantify a fault's associated damage area. Minimizing a fault's damage area should minimize the total effect of the fault. The goal here is to quickly minimize the trip area by engaging trusted circuit breakers near the detected fault. Minimizing the following objective equation achieves this goal:

$$\text{Minimize:} \quad \sum_{i=1}^{|\hat{N}|}\sum_{j=1}^{|\hat{N}|} u(n_i,n_j)h(n_i,n_j) \tag{1}$$

Subject to:

$$\sum_{j=1}^{|\hat{N}|} h(n_i,n_j) - \sum_{k=1}^{|\hat{N}|} h(n_k,n_i) = \begin{cases} 1 & if\ n_i = Source \\ 0 & if\ n_i \neq Source\ \wedge n_i \neq Sink \\ -1 & if\ n_i = Sink \end{cases} \tag{2}$$

(aka Conservation Constraints)

$$h(n_i,n_j) \geq 0 \qquad i,j \in \{1,2 \ldots |\hat{N}|\} \tag{3}$$

$$u(n_i,n_j) = 0 \qquad if\ \langle n_i,n_j\rangle \notin \hat{E} \text{ or } n_j \notin N \tag{4}$$

$$u(n_i,n_j) = \alpha \cdot h_j + \beta \cdot \left(\frac{1}{\tau_j}\right) \qquad if\ n_j \in N \tag{5}$$

Where

$\hat{E}$	is the set of edges in the constructed graph
$\lvert\hat{N}\rvert$	is the number of nodes in the constructed graph
$\lvert N\rvert$	is the number of sensor node/relays
i,j	are index variables used to enumerate the sensor nodes/relays
h_j	is the number of hops node *j* is away from the fault with a predetermine upper bound of h_{max}
τ_j	is the trust value assigned to sensor node/relay *j*
α	is a weighting factor, used to control the importance of distance
β	is a weighting factor, used to control the importance of trust
$u(n_i,n_j)$	is a function mapping edges $\langle n_i,n_j\rangle$ to their edge cost
$h(n_i,n_j)$	is a function mapping edges $\langle n_i,n_j\rangle$ to 1 if the edge is used within the shortest path, otherwise 0.

The values of $|N|$, α, β and h_{max} are given by the SCADA system operator. The values of τ_j are determined by the trust management system; namely by the simple trust algorithm [13]. The values of h_j are determined by a breadth first search (BFS) like algorithm with a given maximum depth or distance constraint, h_{max}. The binary function $h(n_i,n_j)$ is used to select the trusted sensor node/relay n_j circuit breakers to engage, i.e. a value of 1 indicates that the corresponding sensor node/relay's circuit breaker is selected and a value of 0 indicates that the corresponding sensor node/relay is not selected. A graph generator (Algorithms 1 and 2 below) converts the sensor node/relay power grid connectivity topology in to a graph, which is solved using Dijkstra's shortest path algorithm [14]. The binary function $h(n_i,n_j)$ returns a 1 when sensor node/relay n_j is traversed along the shortest path, otherwise zero. Equation 1 is minimized by the identified shortest path from the super source to super sink nodes within the graph.

The algorithm used to select trusted sensor node/relay circuit breakers to open, in order to isolate/clear a detected SCADA fault, must be time efficient. This efficiency is quantified in terms of an algorithm's order of growth asymptotic notation, e.g., $O(N^2)$, O((N+E) * Log N), … etc. Dijkstra's shortest path algorithm [14], with a O((N+E) * Log N) order of growth [27] may be the best option—given the SCADA power grid topology transformation technique. In this analysis, $|N|$ represents the number of

nodes in a given graph, G, and $|E|$ represents the number of edges in G. The transformation technique takes the input SCADA power grid topology in Figure 4 and generates the graph in Figure 5. The edge cost for the edges going in to the sensor node/relays are calculated using $\alpha \cdot h_j + \beta \cdot \left(\frac{1}{\tau_j}\right)$, with α=1, β=100, h_{max}=10, τ_j trust values from Figure 4 and h_j values also determine by the fault location in Figure 4. The edges going into the Left Junction, Right Junction and Super Sink nodes are assigned a cost of 0. Dijkstra's shortest path algorithm [14] determines the shortest path from the super source node to the super sink node, which in this case traverses through *S3* and *S7*. This indicates that tripping the breakers at *S3* and *S7* would isolate the fault and minimize the affected damage area. The transformation algorithms are as follows:

Algorithm 1 ***SCADA_TMS_Transformation Pseudocode*** (N, E, F)

1. Procedure ***SCADA_TMS_Transformation***
// Inputs: N is the set of SCADA nodes within the SCADA Power Grid, see Figure 4
// E is the set of SCADA edges within the SCADA Power Grid, see Figure 4
// F is the SCADA edge experiencing a Fault, i.e. $F \in E$
// Note: Each edge is represented by a pair of nodes, i.e., $e \in E$, $e = \{left, right\}$
// where $left, right \in N$ and accessed by $e.left \rightarrow left\ node$ and $e.right \rightarrow right\ node$
2. **Begin**
// Initialization
3. $\hat{N} \leftarrow \varnothing$
4. $\hat{E} \leftarrow \varnothing$
5. $h \leftarrow 0$
6. h_{max}=10
7. $Left_{root} \leftarrow F.left$
8. $Right_{root} \leftarrow F.right$
// Update Given sets of Nodes and Edges by removing
// *F* from set *E* and the end nodes of *F* from set *N*.
9. $E \leftarrow E-F$
10. $N \leftarrow N-Left_{root}$
11. $N \leftarrow N-Right_{root}$
// Add Super Source, Left Junction, Right Junction and Super Sink Nodes to return variables, $\hat{N}$ and $\hat{E}$ — see Figure 5
12. $\hat{N} \leftarrow \hat{N} \cup \{S\} \cup \{L\} \cup \{R\} \cup \{T\}$
13. $\hat{E} \leftarrow \hat{E} \cup \{\{S, L, \cos t \leftarrow 0\}\}$
14. $\hat{E} \leftarrow \hat{E} \cup \{\{L, F.left, \text{cost} \leftarrow \alpha * h + \beta * \left(\frac{1}{F.left.trust}\right)\}\}$
15. $\hat{E} \leftarrow \hat{E} \cup \{\{F.left, R, \cos t \leftarrow 0\}\}$
// Call recursive helper function for left side nodes
16. Build$\left(N, E, \hat{N}, \hat{E}, Left_{root}, L, R, h, h_{\max}\right)$
17. $\hat{E} \leftarrow \hat{E} \cup \{\{R, F.right, \text{cost} \leftarrow \alpha * h + \beta * \left(\frac{1}{F.right.trust}\right)\}\}$
18. $\hat{E} \leftarrow \hat{E} \cup \{\{F.right, \Omega, \cos t \leftarrow 0\}\}$
// Call recursive helper function for right side nodes
19. Build$\left(N, E, \hat{N}, \hat{E}, Right_{root}, R, T, h, h_{\max}\right)$
20. **Return** $\hat{N}$, $\hat{E}$
21. **End** procedure S***CADA_TMS_Transformation***

Note: This algorithm uses a SCADA power grid connectivity graph (N, E) and a fault location edge (F) to construct a graph, solvable by Dijkstra's shortest path algorithm [14].

Algorithm 2 ***Build Pseudocode*** $\left(N, E, \hat{N}, \hat{E}, root, j, \Omega, h, h_{\max}\right)$

1. Procedure ***Build_Graph***
// Inputs:*N* is the set of SCADA nodes within the SCADA Power Grid, see Figure 4
// *E* is the set of SCADA edges within the SCADA Power Grid, see Figure 4
// $\hat{N}$ is the set of graph nodes
// $\hat{E}$ is the set of graph edges
// *root* is the starting node in *N*
// *j* is the source node in $\hat{N}$
// Ω is the end node in $\hat{N}$
// h_{max} is the maximum allowed hop distance from the fault
// Note: The above variables are passed by reference
// *h* is the hop distance from the fault
2. **Begin**
// Check Exit conditions
3. **IF** (*E*==Ø) or (*N*==Ø) or ($h \geq h_{max}$)
4. **Return**

5. **End If**
// Initialization
6. temp_edges ← Ø
7. temp_nodes ← Ø
8. $\alpha \leftarrow 1$
9. $\beta \leftarrow 100$
10. $h \leftarrow h + 1$
// find all nodes adjacent to *root* in *N*
11. **For each** *e* in *E*
12. **If** (*e.left* == *root*) and (*e.right* ∈ *N*)
13. *temp_edges* ← *temp_edges* ∪ *e*
14. *temp_nodes* ← *temp_nodes* ∪ *e.right*
15. **End If**
16. **If** (*e.right* == *root*) and (*e.left* ∈ *N*)
17. *temp_edges* ← *temp_edges* ∪ *e*
18. *temp_nodes* ← *temp_nodes* ∪ *e.left*
19. **End If**
20. **End For each**
// Update the given nodes and edges
21. $E \leftarrow E - temp\_edges$
22. $N \leftarrow N - temp\_nodes$
23. **If** (*temp_edges* == Ø) or (temp_nodes == Ø)
24. **Return**
25. **End If**
26. **If** (|*temp_nodes*| == 1)
27. $\hat{N} \leftarrow \hat{N} \cup temp\_nodes$
28. $\hat{E} \leftarrow \hat{E} \cup \{\{j, temp\_nodes.element, \text{cost} = \alpha * h + \beta * \left(\frac{1}{temp\_nodes.element.trust}\right)\}\}$
29. $\hat{E} \leftarrow \hat{E} \cup \{\{temp\_node.element, \Omega, \text{cost} \leftarrow 0\}\}$
30. $\text{Build}\left(N, E, \hat{N}, \hat{E}, temp\_nodes.element, j, \Omega, h, h_{\max}\right)$
31. **Return**
32. **End If**
// spit point found requiring addition virtual junction nodes
33. $\text{junction}_{\text{in}}$ ← new virtual junction node
34. $\hat{N} \leftarrow \hat{N} \cup \{junction_{in}\}$
35. $\hat{E} \leftarrow \hat{E} \cup \{\{j, junction_{in}, \text{cost} = 0\}\}$
36. **For each** *node* in *temp_nodes*
37. $\text{junction}_{\text{in}}$ ← new virtual junction node
38. $\hat{N} \leftarrow \hat{N} \cup \{junction_{out}\}$
39. $\text{Build}\left(N, E, \hat{N}, \hat{E}, node, junction_{in}, junction_{out}, h, h_{\max}\right)$
40. $\text{junction}_{\text{in}} \leftarrow \text{junction}_{\text{out}}$
41. **End For each**
42. $\hat{E} \leftarrow \hat{E} \cup \{\{junction_{out}, \Omega, \text{cost} = 0\}\}$
43. **Return**
44. **End** procedure ***Build_Graph***

9. Results

Scenario 2 was run inside the EPOCHS environment with the results presented here (result for scenarios 1 and 3 are self contained within the text). Based on the results from scenario 2, the proposed reputation-based trust management system does not interfere with the primary protection system, provides some protection against cyber attacks, such as hijacking attacks, and improves backup protection scheme response time by ~96.7%, i.e. the proposed TMS response time of 50 ms is better than traditional SCADA backup protection schemes response times of 1.5 seconds, see Figure 8 and Figure 9. This last benefit is mostly due to the additional shared knowledge available in a smart grid enabled power grid. The traditional power grid backup protection schemes utilized pilot wires, timeout timers and point to point communication lines. This illustrates some of the strong benefits provided by context-aware smart protection schemes. These experiments provide an indication that the employment of such a TMS can help mitigate some security concerns in using smart protection and control systems.

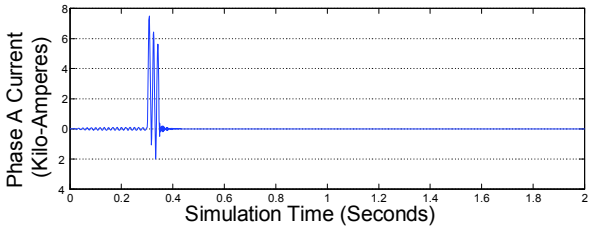

Figure 8. Faulty line isolated within 50 ms

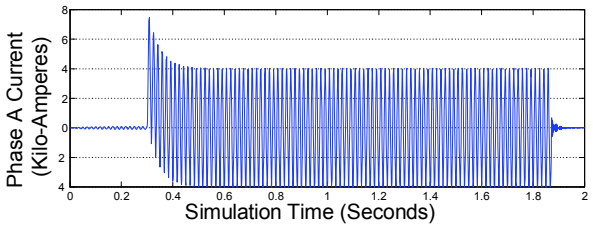

Figure 9. Faulty line isolated within ~1.5 sec

10. Summary

The increased communication capabilities of the future smart electric power grid promises to decrease the stresses placed on the current electric power grid by enabling new technologies, i.e. demand response smart meters and micro grids. The increased communication capabilities increase the power grids susceptibility to cyber attacks. The potential of the proposed reputation-based trust management system has been demonstrated by mitigating some cyber type attacks and the improved backup protection system response time, but further research is required before implementation is attempted. In particular, the small example used to illustrate the proposed technique should be expanded to larger and more realistic scenarios in future work.

11. References

[1] S. Sutanto and R. Page Heller, "Utility communications architecture review," *ISA Transactions,* vol. 32, pp. 297-300, 1993.

[2] T. Kostic, O. Preiss, and C. Frei, "Understanding and using the IEC 61850: a case for meta-modelling," *Computer Standards & Interfaces,* vol. 27, pp. 679-695, 2005.

[3] B. Naduvathuparambil, M. C. Valenti, and A. Feliachi, "Communication Delays in Wide Area Measurement Systems," 2002, pp. 118 - 122.

[4] J. E. Dagle, "North American SynchroPhasor Initiative," 2008, pp. 165-165.

[5] Office of the National Coordinator for Smart Grid Interoperability, "NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 1.0," U.S. Department of Commerce and National Institute of Standards and Technology, NIST Special Publication 1108, 2010.

[6] J. M. Ortman and C. E. Guarneri, "United States Population Projections: 2000 to 2050," United States Census Bureau, 2009.

[7] A. Schmidt, "A Population Perspective of the United States," Population Resource Center, 2004.

[8] L. Neergaard, "Emergency officials struggle to find those on life-support during power outages," 2009.

[9] J. Ballman, "The Great Blackout of 2003 Aug. 14 Power Outage Largest in U.S. History," *Disaster Recovery Journal,* vol. 16, 2003.

[10] Y. Sverdlik, "Is the smart grid an intelligent move," *Datacenter Dynamics FOCUS,* vol. 3, pp. 21--22, February/March 2010.

[11] R. Lamb, "How a Microgrid Works," HowStuffWorks.com, 2009.

[12] U.S. Engery Information Administration (EIA), "Annual Energy Outlook 2010 with Projections to 2035," U.S. Department of Energy (DOE) DOE/EIA-0383(2010), 2010.

[13] J. E. Fadul, K. M. Hopkinson, T. R. Andel, J. T. Moore, and S. H. Kurkowski, "Simple Trust Protocol for Wired and Wireless SCADA networks," in *5th International Conference on Information Warfare and Security*, Dayton, OH, 2010, pp. 89-97.

[14] E. W. Dijkstra, "A Note on Two Problems in Connection with Graphs," *Numerische Mathematik,* vol. 1, pp. 269-271, 1959.

[15] L. R. Ford Jr. and D. R. Fulkerson, *Flows in Networks*. Princeton, N.J., U.S.A.: Princeton University Press, 1962.

[16] IEEE, *IEEE 100: The authoritative dictionary of IEEE standards terms*, 7th ed.: IEEE Press, 2000.

[17] X. R. Wang, K. M. Hopkinson, J. S. Thorp, R. Giovanini, K. Birman, and D. Coury, "Developing an Angent-based Backup Protection System for Transmission Networks," in *First International Conference on Power Systems and Communication Systems Infrastructures for the Future*, Beijing, China, 2002.

[18] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman, and D. Coury, "EPOCHS: A Platform for Agent-based Electric Power and Communication Simulation Built from Commercial Off-The-Shelf Components," *IEEE Transactions on Power Systems,* vol. 21, pp. 548-558, May 2006.

[19] General Electric. *PSLF Manual*. 2003. Available: http://www.gepower.com/dhtml/corporate/en_us/assets/software_solns/prod/pslf.jsp

[20] Shaw Power Technologies Inc., *PSS/E 30 User's Manual*. Schenectady, NY, USA, 2004.

[21] Manitoba HVDC Research Centre, *PSCAD/EMTDC Manual Getting Started*. Winnipeg, Manitoba, Canada, 1998.

[22] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidermann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in Network Simulation," *IEEE Computer,* vol. 33, pp. 59-67, May 2000.

[23] J. E. Fadul, K. M. Hopkinson, T. R. Andel, J. T. Moore, and S. H. Kurkowski, "SCADA Trust Management System," in *2010 International Conference on Security and Management (SAM'10) (under The 2010 World Congress in Computer Science, Computer Engineering, and Applied Computing (WORLDCOMP'2010))*, Las Vegas, Nevada USA, 2010, pp. 548 - 554.

[24] G. M. Coates, K. M. Hopkinson, S. R. Graham, and S. H. Kurkowski, "Collaborative, Trust-Based Security Mechanisms for a Regional Utility Intranet," *Power Systems, IEEE Transactions on,* vol. 23, pp. 831-844, 2008.

[25] C. L. Bowen, III, T. K. Buennemeyer, and R. W. Thomas, "Next generation SCADA security: best practices and client puzzles," in *Information Assurance Workshop, 2005. IAW '05. Proceedings from the Sixth Annual IEEE SMC*, 2005, pp. 426-427.

[26] M. G. Adamiak, A. P. Apostolov, M. M. Begovic, C. F. Henville, K. E. Martin, G. L. Michel, A. G. Phadke, and J. S. Thorp, "Wide Area Protection-Technology and Infrastructures," *Power Delivery, IEEE Transactions on,* vol. 21, pp. 601-609, 2006.

[27] G. Heineman, G. Pollice, and S. Selkow, *Algorithms in a Nutshell*: O'Reilly Media, Inc., 2008.