Mitigating Event Confidentiality Violations in Smart Grids: An Information Flow Security-Based Approach

Thoshitha T. Gamage, *Member, IEEE*, Thomas P. Roth, *Student Member, IEEE*, Bruce M. McMillin, *Senior Member, IEEE*, and Mariesa L. Crow, *Fellow, IEEE*

***Abstract*—Modern smart grids, by and large, merge physical interconnections and cyber controllers. Invariably, this tight coupling results in cyber commands manifesting in the physical layer as observable changes, leading to possible disclosure of sensitive system settings. Thus, cyber event confidentiality of the smart grid is violated. Attacks on confidentiality can ultimately lead to integrity and availability attacks; with adequate knowledge of the system topology, internal settings, and how the physical layer responds to cyber commands, a malicious adversary gains knowledge to attack the system. This work shows how to develop self-obfuscating systems based on information flow security properties that can mitigate event confidentiality violations in smart grids.**

***Index Terms*—Cyber-physical systems, distributed control, distributed detection, logic, power system protection, power system security, security.**

I. Introduction

POWER SYSTEM components in emerging smart grids are controlled by distributed cyber commands raising a concern over protecting the confidentiality of commands. This paper formalizes this loss of confidentiality and proposes an information flow security based mitigation scheme for preserving confidentiality of cyber commands. The inherent external observability and the tight coupling between the cyber and physical components in smart grid environments make sensitive system settings and events prone to unauthorized disclosure. When a cyber command is executed, its physical consequences manifest as externally observable changes. Through systematic recording of system-wide observations and using the knowledge of the physical topology, it is possible to reverse engineer the originating cyber command. In other words, cyber commands are exposed purely through external physical observations. This, in terms of security, is a violation of the system's cyber confidentiality.

Manuscript received March 03, 2012; revised July 20, 2012; accepted December 28, 2012. Paper no. TSG-00093-2012.

T. T. Gamage was with the Missouri University of Science and Technology, Rolla, MO 65409-0350 USA. He is now with the School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164-2752 (e-mail: thoshitha.gamage@wsu.edu).

T. P. Roth and B. M. McMillin are with the Department of Computer Science, Missouri University of Science and Technology, Rolla, MO 65409-0350 USA (e-mail: tprfh7@mst.edu; ff@mst.edu).

M. L. Crow is with the Department of Electrical Engineering, Missouri University of Science and Technology, Rolla, MO 65409-0350 USA (e-mail: crow@mst.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2013.2243924

The problem is motivated by the fact that the form of confidentiality violation described here raises concerns over privacy [1] and confidentiality of command settings [2]. The attack model is similar in concept to the way in which the physical location of a person is tied to his/her credit card and mobile phone usage, or the energy usage signature revealing activities within a house [1]. Consider the following system model: power distribution networks may develop cascading failures if a line outage due to a contingency forces other lines to exceed their capacity in response to the fault. As a remedy, power electronic controllers are used to control the power flow on overloaded lines. However, since the physical components are bound by constraints (e.g., power line capacities) under which cyber algorithms operate (e.g., avoid line overloads), these command responses produce unique observation patterns on the physical system [3]. Thus, a group of coordinated physical system observers[1] can exploit these pattern-like behaviors, not only to deduce the origin of the cyber action, but also the actual setting itself.

The knowledge a potential adversary gains in this manner is useful in exploiting smart grid interfaces with a high risk for confidentiality violations such as load management/demand-response management systems [4], and can easily lead to attacks on resource availability and reliable operation; the aforementioned electronic controllers are identified as critical to the derivation of interconnection reliability operating limits and their associated contingencies in NERC CIP-004 standard.

The main contribution of this work is the introduction of an information flow security-based confidentiality security model that can prevent cyber command disclosure through inherent physical observations. Also presented are the corresponding event confidentiality violation mitigation scheme, a cyber algorithm based on this confidentiality model, its implementation details, and applicability based on system topology. Section II lists background and related work. Section III presents the proposed confidentiality model followed by proposed mitigation scheme in Section IV. Section V presents the cyber algorithm with Sections VI and VII showing the applicability to structured and unstructured topologies. Finally, Section VIII lists the conclusions.

II. Security Models for Smart Grids

Strict confidentiality and its maintenance at the cyber level are well studied topics in the field of security. Access control

[1]referred to as external observers hereafter

based multi-level security models [5] are one good example. While capable of imposing spatial restrictions on information and resources, access control fails in preventing unintended information flows in tightly connected cyber-physical systems, such as smart grids. Furthermore, they violate write down properties [5] through covert *information* channels. Thus, a general paradigm shift in looking at cyber-physical system security as an integrative functional composition [6] is on the rise.

The prominent approach to enforcing security through strict safety property [7], which immediately terminates the execution of the target application upon detecting a violation, also falls short in meeting smart grid security needs. Termination will prevent a cyber process from making further progress, but can't prevent the physical consequences of the action from becoming evident. A command executing at the cyber controller can be made secure and confidential among other cyber peers. However, its physical consequence is visible to physical system observers and has the propensity to violate system level confidentiality. Moreover, safety properties preclude monitoring for and enforcing information flow violations.

A. Related Work

Security models that capture information flow violations are collectively termed information flow security properties (IFPs) where information flowing from the high-level domain $(\mathcal{D}_H)$ to the low-level domain $(\mathcal{D}_L)$ is a violation. IFPs provide a semantically integral and expressive model to analyze integrative system security requirements. Consider commands executed at the cyber level as $\mathcal{D}_H$ and the external physical observations as $\mathcal{D}_L$. Here, the ability to deduce information about the $\mathcal{D}_H$ commands (origin, setting, value, etc.) based on $\mathcal{D}_L$ observations violates the confidentiality of the $\mathcal{D}_H$ commands. Thus, an IFP based security model can capture information flows that are inexorably intertwined between and among cyber and physical components in a smart grid.

Relaxing the safety property requirement empowers enforcement mechanisms to take appropriate remedial actions at the point of violation [8]: i) Inject corrective actions or ii) backup the application to a previously verified safe state. Since the physical consequences of a cyber action are irreversible, the only feasible option is to inject corrective actions. However, care must be taken to ensure that the functional integrity of the target system is maintained, even after modifying the execution. One way of doing this is to employ an emulator to emulate the behavior of the system by running a subsequence [9]. The outcome of the emulation can determine whether a particular modification itself is safe. In essence, this approach extends the safety property based security automata in [7] by providing an enforceable mechanism for IFPs.

III. Mitigating Event Confidentiality Violations

For the rest of this discussion, consider the act of enforcing a $\mathcal{D}_H$ cyber setting as a $\mathcal{D}_H$ event and physically observable flow changes as $\mathcal{D}_L$ events (or projection). The key to mitigating external observation based confidentiality violations is to prevent or limit information flows from the $\mathcal{D}_H$ (cyber layer) to the $\mathcal{D}_L$ (physical layer). From a pure IFP perspective, the violation occurs as a result of violating the *Nondeducibility* property [10], [11]; a particular cyber command becomes deducible if it results in a unique observation. Thus, by definition, Nondeducibility security can be preserved by eliminating uniqueness in $\mathcal{D}_L$ observations. In other words, Nondeducibility security is preserved if every $\mathcal{D}_L$ projection has more than one $\mathcal{D}_H$ trace associated with it [11].

In a brute force attack on confidentiality, a potential system attacker can develop an attack vector over time utilizing the topological knowledge of the system and modern computational capabilities. However, ascertaining whether a particular $\mathcal{D}_H$ event produces a unique $\mathcal{D}_L$ projection or determining which $\mathcal{D}_H$ events produce unique projections is not a simple and straightforward process. A system of n controllable elements each with K discrete $\mathcal{D}_H$ events will produce $K^n + n * K$ number of distinct $\mathcal{D}_L$ projections. On this projection pool, a total of $((K^n + n * K) - 1)!$ comparisons are necessary to identify unique ones. Thus, from a system security point of view, the process of identifying unique $\mathcal{D}_L$ projections, especially at runtime, is computationally expensive.

Instead, ensuring that every $\mathcal{D}_H$ action has multiple equivalent alternatives that result in the same $\mathcal{D}_L$ projection will, by default, establish $\mathcal{D}_H$ event confidentiality as any $\mathcal{D}_L$ projection is guaranteed to have multiple $\mathcal{D}_H$ possibilities. Once this is met, it doesn't matter whether the original action or one of the alternative actions is executed as long as the functional requirements of the system are met. This way, the changes are made only of there are multiple equivalent actions. For example, a control response to a power line overload selected out of a pool of solutions will not only ensure confidentiality, but also the intended functional integrity of the action.

IV. Replacement Solutions

In order to systematically calculate $\mathcal{D}_L$ projection equivalent alternative actions, this work categorizes systems into two broad topologies—*structured* and *unstructured*. Structured topologies consist of series and parallel connections while unstructured topologies have additional loop connections. The rationale behind such an approach is to discover the relationship between alternative solutions and the topology of a system. This allows system operators to switch between solutions, if necessary, without having to manually verify their $\mathcal{D}_L$ projection equivalence.

This work achieves the above objective through a method termed "*Replacement Solutions*" that provide Nondeducibility security to $\mathcal{D}_H$ events by design, and provides the system operator with a pool of $\mathcal{D}_H$ event solutions to select at random on successive implementations. Replacement solutions are based on two primary schemes termed *Upstream Replacement* $(\mathcal{R}_{\blacktriangle})$ and *Downstream Replacement* $\mathcal{R}_{\blacktriangledown}$ respectively based on the direction of the move relative to the direction of flow. Fig. 1 shows these two basic schemes. In $\mathcal{R}_{\blacktriangle}$, the originally intended $\mathcal{D}_H$ event is moved to the immediate parent and sibling couple, as shown in Fig. 1(a). Similarly, the original intended $\mathcal{D}_H$ event is moved to the immediate children In $\mathcal{R}_{\blacktriangledown}$, as shown in Fig. 1(b).

Definition. [Upstream Replacement $(\mathcal{R}_{\blacktriangle})$*]:* The act of replacing a $\mathcal{D}_H$ change with its immediate parent and, if they exist, all its siblings is termed Upstream Replacement $(\mathcal{R}_{\blacktriangle})$.

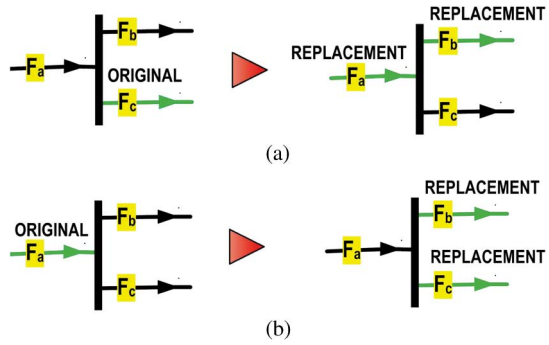

Fig. 1. Single high-level domain ($\mathcal{D}_H$) action replacement schemes.

Definition. [Downstream Replacement ($\mathcal{R}_{\blacktriangledown}$)]: The act of replacing a $\mathcal{D}_H$ change with its immediate children is termed Downstream Replacement ($\mathcal{R}_{\blacktriangledown}$).

The rationale behind replacement solutions is explained as follows. For the purpose of analysis consider the model used here as a flow network abstraction of a smart grid—power lines as edges (E) and busbars as vertices (V). Each power line is equipped with an intelligent cyber controller capable of enforcing either a positive (increase flow) or a negative (decrease flow) flow constraint, considered in this model as $\mathcal{D}_H$ events.

The same flow constraint, however, can be achieved by enforcing flow values on other edges in such a way that the intended amount of flow goes through the initially selected set of edges. Consider the intelligent cyber controllers F_a, F_b and F_c in Fig. 1 have line flows $\vec{f_a}$, $\vec{f_b}$, and $\vec{f_c}$ respectively. At equilibrium, the relationship between the line flows is $\vec{f_a} = \vec{f_b} + \vec{f_c}$. For $\mathcal{R}_{\blacktriangle}$, [Fig. 1(a)] F_c enforcing $\vec{f_c}$ is equivalent to F_a and F_b enforcing flow values $\vec{f_a}$ and $\vec{f_b}$. Similarly argument applies to the $\mathcal{R}_{\blacktriangledown}$ scheme in Fig. 1(b). Thus, any flow value of the original line can be easily replicated using its replacement counterparts. In theory, this would yield the same $\mathcal{D}_L$ observation as the original change.

Theorem 1. [Nondeducibility of Replacement Solutions]: Any replacement solution of a $\mathcal{D}_H$ change is Nondeducibility secure.

Proof: The proof of this theorem follows a flow invariant argument. Consider two sets of real value variables $|P| = n$, $|Q| = m$: $n \geq 1, m > 1$ and their relationship as shown in (1). For the purpose of this analysis, consider the sets P and Q as incoming and outgoing flows and their relationship preserves the *flow conservation property*, i.e., $flow_{in} = flow_{out}$ at a particular vertex with no storage.

$$p_1 + p_2 + \ldots + p_n = q_1 + q_2 + \ldots + q_m \tag{1}$$

Assume a Δq_i: $q_i \in Q$ change in one of the right hand side variables. Thus, $\hat{q_i} - q_i = \Delta q_i$ (action). Due to flow conservation, this change will excite all other variables in the equation to new values $\hat{P}$ and $(\hat{Q} - q_i)$ (consequence). Now consider the converse of this assignment where $\hat{P}$ and $(\hat{Q} - q_i)$ force Δq_i in q_i. If $\hat{q_i} \rightarrow \hat{P} \cup (\hat{Q} - q_i)$ is considered the original $\mathcal{D}_H$ change, the converse $\hat{P} \cup (\hat{Q} - q_i) \rightarrow \hat{q_i}$ is the $\mathcal{R}_{\blacktriangle}$. The flow values are equivalent in both cases and thus, will produce the same $\mathcal{D}_L$ projection. Any subsequent recursive application of $\mathcal{R}_{\blacktriangle}$ will also produce the same $\mathcal{D}_L$ projection. Thus, $\mathcal{R}_{\blacktriangle}$ solutions are Nondeducibility secure.

Assume $\forall p_i \in P$: $p_i \rightarrow \hat{p_i}$ for all incoming flows (R.H.S. variables) where $\hat{p_i} - p_i = \Delta p_i$. Such a change will consequently result in $\forall q_i \in Q$: $q_i \rightarrow \hat{q_i}$. If this transition $\hat{P} \rightarrow \hat{Q}$ is considered the original $\mathcal{D}_H$ change, its converse $\hat{Q} \rightarrow \hat{P}$ becomes the $\mathcal{R}_{\blacktriangledown}$. Both cases maintain the same flow values and the same $\mathcal{D}_L$ projections. Following a similar recursive analysis of $\mathcal{R}_{\blacktriangle}$ above, $\mathcal{R}_{\blacktriangledown}$ will also produce the same $\mathcal{D}_L$ projection, resulting in becoming nondeducible.

Additionally, a $\mathcal{R}_{\blacktriangledown}$ of a $\mathcal{R}_{\blacktriangle}$ and its converse are also nondeducible since any combination of replacements will preserve the flow conversation property, flow values, and $\mathcal{D}_L$ projections. Thus, any replacement solution of a $\mathcal{D}_H$ action is nondeducible. ■

The inset 4(b) is a $\mathcal{R}_{\blacktriangledown}$ of the $\mathcal{D}_H$ cyber setting on l_{1-3} in inset 4(a). Similarly, inset 4(c) is a $\mathcal{R}_{\blacktriangle}$ of the same cyber setting. Note that in all three cases, the set of $\mathcal{D}_L$ observations of the network stay the same. Just by looking at the set of $\mathcal{D}_L$ observations, an adversary is not able to deduce which $\mathcal{D}_H$ event took place. Thus, from a Nondeducibility perspective, the two replacement solutions preserve the event confidentiality of the original $\mathcal{D}_H$ event. The two replacement schemes can be applied recursively to produce additional replacement solutions.

A. Calculating Replacement Solutions

The total number of replacement solutions for a given structured network depends on the initial $\mathcal{D}_H$ event placement and how it partitions the network. The subnetwork rooted under the initial $\mathcal{D}_H$ placement is termed the $\mathcal{R}_{\blacktriangledown}$–tree. The rest of the network which is eligible for $\mathcal{R}_{\blacktriangle}$ is termed the $\mathcal{R}_{\blacktriangle}$—tree. A level is defined as a measure of distance from a certain root. Replacement solutions from these two subnetworks are mutually exclusive, meaning, no replacement solution has alternative $\mathcal{D}_H$ event placements in both subnetworks. For a given initial $\mathcal{D}_H$ event placement, the basic formula to calculate the total number of all possible replacement solutions takes the following form:

$$total_{sol} = downstream_{sol} + upstream_{sol} + original_{sol} \tag{2}$$

Calculating $downstream_{sol}$ is relatively straightforward. For $\mathcal{R}_{\blacktriangledown}$s, the total number of replacement solutions is the sum of the original placement solution and the total number of permutations in each subnetwork rooted under the original placement multiplied by each other. For example, consider the basic single-level $\mathcal{R}_{\blacktriangledown}$–tree in Fig. 2. Here, the network rooted under bus_5 has a single subnetwork (bus_{11}) with a single $\mathcal{R}_{\blacktriangledown}$ permutation (l_{11-23}, l_{11-24}). The total number of replacement solutions equal $1 * 1 = 1$. Added to this is the original placement on line l_{5-11}, which brings up the total number of nondeducible solutions to 2.

This process can be recursively applied to subnetworks with more levels. For example, the two subnetworks under bus_5 in Fig. 3, form two single-level $\mathcal{R}_{\blacktriangledown}$-trees similar to Fig. 2. The immediate result of this is that the total number of replacement solutions in the subnetwork rooted under bus_2 is $2 * 2 = 4$. Thus, the total number of replacement solutions for a two-level $\mathcal{R}_{\blacktriangledown}$—tree is $replacements + original = 4 + 1 = 5$.

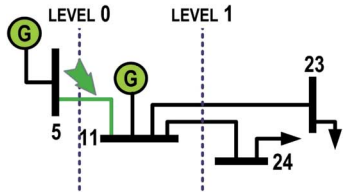

Fig. 2. Basic single-level $\mathcal{R}_{\blacktriangledown}$—tree.

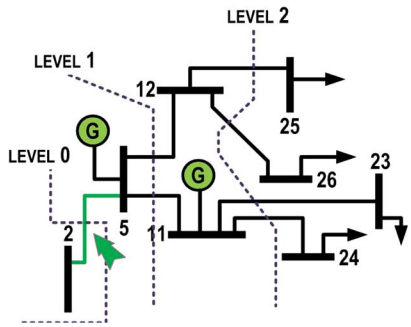

Fig. 3. Basic two-level $\mathcal{R}_{\blacktriangledown}$—tree.

The above method for $\mathcal{R}_{\blacktriangledown}$ solution calculation can be used to calculate the $\mathcal{R}_{\blacktriangle}$ solution count as follows. Consider each subtree rooted under each direct ancestor of the original placement, i.e., parent, grandparent, great-grandparent, etc., as a $\mathcal{R}_{\blacktriangledown}$—subtree under the $\mathcal{R}_{\blacktriangle}$—tree. The subtree rooted under the parent node is considered the *first* $\mathcal{R}_{\blacktriangledown}$*—subtree of the* $\mathcal{R}_{\blacktriangle}$*—tree*. The subtree rooted under the grandparent is the *second* $\mathcal{R}_{\blacktriangledown}$*—subtree of the* $\mathcal{R}_{\blacktriangle}$*—tree*. With this numbering scheme in place, the total number of replacement solutions at the i^{th} $\mathcal{R}_{\blacktriangledown}$—subtree of the $\mathcal{R}_{\blacktriangle}$—tree and beyond can be expressed as follows:

$$upstream_{sol_i} = sol_i + sol_i * upstream_{sol_{i+1}} \quad (3)$$

Here, sol_i denotes the number of replacement solutions in the i^{th} $\mathcal{R}_{\blacktriangledown}$—subtree of the $\mathcal{R}_{\blacktriangle}$—tree. The general form of equation to calculate the number of replacement solutions in the $\mathcal{R}_{\blacktriangle}$—tree can be expressed as follows:

$$upstream_{sol} = sol_1 + sol_1 * (sol_2 + sol_2 * (sol_3 + sol_3 * (\ldots)) \ldots))) \quad (4)$$

V. Replacement Solutions Algorithm

The above concept can be extended to develop a cyber algorithm to calculate alternative $\mathcal{D}_H$ event placements for a given initial placement. This is listed in Algorithm 1. The outcome of this algorithm can be integrated with the overall control algorithm of the system to provide Nondeducibility secure $\mathcal{D}_H$ events.

Starting with the initial placement, which is the first candidate solution, two replacement solutions, $\mathcal{R}_{\blacktriangle}$ and $\mathcal{R}_{\blacktriangledown}$ are calculated, if they lie within the max and min depths. Functions *ReplaceUpstream*() and *ReplaceDownstream*() are used for this purpose respectively; These two functions behave the same way in terms of recursion, but traverse opposite directions from the initial placement. Specifying a max and min placement depths makes the solution search space smaller.

Algorithm 1: Replacement Solution Finder Algorithm: $\text{FindRepSolSet}(\mathcal{N}, \mathcal{S}, E_C, E_S, Min_d, Max_d)$

input: A directed weighted flow network $\mathcal{N} = (V, E)$ where V is a set of vertices and E is a set of edges, a solution list container $\mathcal{S}$, a list of candidate edges E_C, a list of placement avoidance edges E_S, minimum placement depth Min_d, maximum placement depth Max_d

output: A solution set of device placements $\mathcal{S}$

```
begin
    for each e_c ∈ E_C do
        if GetDepth(e_c) > Min_d then
            result ←
            ReplaceUpstream(N, e_c, E_C, E_S)
            if result ≠ ∅ & result ∉ S then
                S ← FindRepSolSet(N, S + result, result, E_S + e_c, Min_d, Max_d)
            end
        end
        if GetDepth(e_c) > Max_d then
            result ←
            ReplaceDownstream(N, e_c, E_C, E_S)
            if result ≠ ∅ & result ∉ S then
                S ← FindRepSolSet(N, S + result, result, E_S + e_c, Min_d, Max_d)
            end
        end
    end
    return S
end
```

After the recursive step returns, all the remaining edges in the candidate list of the previous call continue to follow the same procedure. The solution set that returns once the all the candidates in the first set have been considered consists of all possible solutions under the constraints of the maximum-minimum depth and placement avoidance set.

Theorem 2: The cost of calculating all replacement solutions for a given initial placement in a directed weighted flow network $\mathcal{N} = (V, E)$ where V is a set of vertices and E is a set of edges is $O(|E^2|)$.

Proof: Calculating all possible replacement solutions for a given initial placement is naturally a recursive process. Thus, the general concept behind of the proof of this theorem is to perform a recurrence over the levels of the structure detailed in [11]. ■

Function ReplaceUpstream$(\mathcal{N}, \mathcal{T}, E_C, E_S)$

input: A directed weighted flow network $\mathcal{N} = (V, E)$, a target device to replace $\mathcal{T}$, a list of candidate edges E_C, a list of placement avoidance edges E_S
output: An $\mathcal{R}_{\blacktriangle}$ solution candidate
begin
  result $\leftarrow E_C$
  for each $\mathcal{T}.parent$ **do**
    if $\mathcal{T}.parent \notin E_S$ **then**
      result $\leftarrow$ result $+ \mathcal{T}.parent$
    end
  end
  for each $\mathcal{T}.sibling$ **do**
    if $\mathcal{T}.sibling \notin E_S$ **then**
      result $\leftarrow$ result $+ \mathcal{T}.sibling$
    end
  end
  if result $\neq E_C$ **then**
    result $\leftarrow$ result $- \mathcal{T}$
  else
    result $\leftarrow \emptyset$
  else
  return result
end

Function ReplaceDownstream$(\mathcal{N}, \mathcal{T}, E_C, E_S)$

input: A directed weighted flow network $\mathcal{N} = (V, E)$, a target device to replace $\mathcal{T}$, a list of candidate edges E_C, a list of placement avoidance edges E_S
output A $\mathcal{R}_{\blacktriangledown}$ solution candidate
begin
  result $\leftarrow E_C$
  for each $\mathcal{T}.child$ **do**
    if $\mathcal{T}.child \notin E_S$ **do**
      result $\leftarrow$ result $+ \mathcal{T}.child$
    end
  end
  if result $\neq E_C$ **then**
    result $\leftarrow$ result $- \mathcal{T}$
  else
    result $\leftarrow \emptyset$
  end
  return result
end

Corollary 1: Algorithm 1 produces Nondeducibility secure solutions.

Algorithm 1 is capable of generating replacement solutions for a given original $\mathcal{D}_H$ event. From Theorem 1, replacement solutions are Nondeducibility secure. As a consequence, Algorithm 1 Nondeducibility secure solutions.

VI. Applicability to Structured Topologies

Consider the physical network of an experimental 32-bus power distribution smart grid[2] depicted in Fig. 4. Here, bus_0 acts as the swing bus of the system. Lines colored in green (with an arrow pointing to it) represent intelligent cyber controllers [12] enforcing $\mathcal{D}_H$ cyber events, where the corresponding $\mathcal{D}_L$ observations are represented in blue (labeled "U") and orange (labeled "D") colored lines to denote increase and decrease in flow respectively. The primary assumption is that, *all the power lines are under the control of power electronic cyber controllers*. For example, the green colored line l_{1-3} in inset 4(a) represents a particular $\mathcal{D}_H$ event that results in observable flow changes throughout the network as shown.[3] The dotted line l_{3-4} represents a line contingency, which prompted the cyber control algorithm to cause the above mentioned $\mathcal{D}_H$ event. The corresponding $\mathcal{R}_{\blacktriangledown}$ and $\mathcal{R}_{\blacktriangle}$ are shown in insets 4(b) and 4(c) respectively, both resulting in the same $\mathcal{D}_L$ projection as 4(a).

The initial $\mathcal{D}_H$ event placement for this structured topology is on l_{1-3}. The subnetwork rooted at bus_3 is a two-level $\mathcal{R}_{\blacktriangledown}$—tree equivalent to Fig. 3. Thus, $downstream_{sol} = 4$. The first $\mathcal{R}_{\blacktriangledown}$—subtree of the $\mathcal{R}_{\blacktriangle}$—tree consists of the series parent line l_{0-1}. The second $\mathcal{R}_{\blacktriangledown}$—subtree of the $\mathcal{R}_{\blacktriangle}$—tree is rooted at bus_0, and hosts a single three-level $\mathcal{R}_{\blacktriangledown}$—tree. bus_2 hosts two two-level $\mathcal{R}_{\blacktriangledown}$—trees each having 5 replacement solutions. Thus, the total number of replacement solutions in the subnetwork under bus_2 equals $5 * 5 = 25$. Since bus_0 and bus_2 are in series, other possible replacement is on line l_{0-2}, which makes the total number of replacement solutions for the subnetwork under bus_0 equal to 26. Thus:

$$\begin{aligned} upstream_{sol} &= (sol_1 + sol_1 * (upstream_{sol_2})) \\ &= (1 + 1 * (26) = 27 \end{aligned}$$

Thus, for the 32-bus experimental smart grid network in Fig. 4 with the initial $\mathcal{D}_H$ event placement on l_{1-3}, the total number of all possible replacement solutions equals:

$$\begin{aligned} total_{sol} &= downstream_{sol} + upstream_{sol} + original_{sol} \\ &= 4 + 27 + 1 = 32 \end{aligned}$$

The same formulae can be used to calculate the total number of replacement solutions for a given placement at any level in any given structured network. The number of subnetworks in a branch and the number of replacement solutions in each branch will affect the total number of solutions. Care must be taken to consider all possible permutations and combinations when dealing with trees that are not balanced or complete. For example, the total number of replacement solutions for an initial placement in each of the 3 levels in the 32-bus experimental smart grid network is listed under Table I.

[2]Not a standard IEEE bus system

[3]The flow change comparison is done w.r.t. the initial state of the system before the $\mathcal{D}_H$ event on l_{1-3}

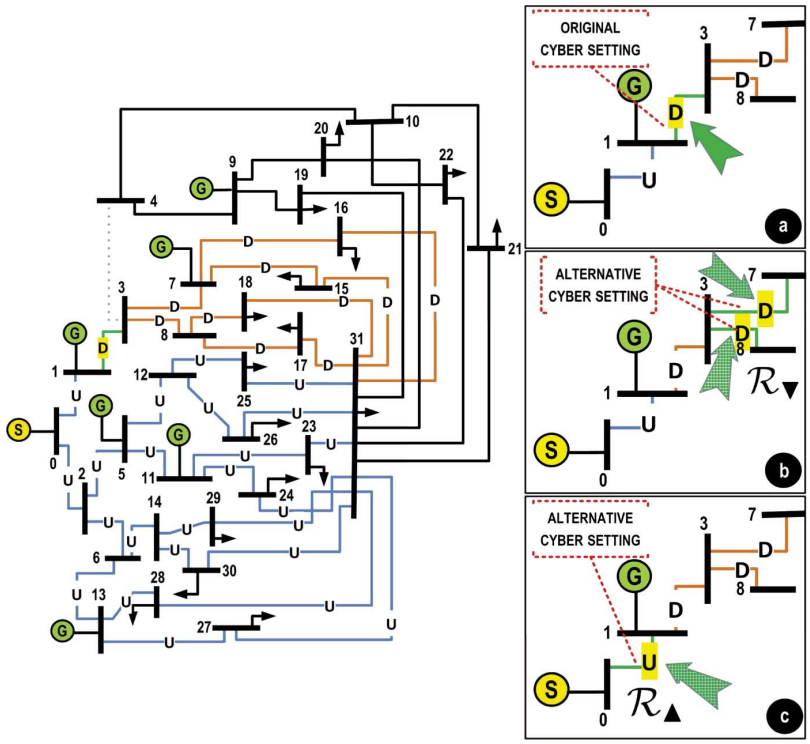

Fig. 4. The physical network of an experimental 32-bus smart grid with three $\mathcal{D}_H$ events with the same $\mathcal{D}_L$ events.

TABLE I
SUMMARY OF INITIAL PLACEMENT SOLUTIONS FOR THE 32-BUS EXPERIMENTAL SMART GRID

Initial Placement Level	$\text{downstream}_{\text{sol}}$	$\text{upstream}_{\text{sol}}$	$\textbf{total}_{\textbf{sol}}$
1	4	27	**32**
2	1	136	**138**
3	0	272	**274**

VII. APPLICABILITY TO UNSTRUCTURED TOPOLOGIES

The Replacement Solutions scheme was applied on several well studied cascading failure scenarios in the standard IEEE 118-bus test system[4] shown in Fig. 5. A thorough analysis on controlling cascading failures using Flexible AC Transmission System (FACTS) devices (a type of intelligent power electronic controllers) can be found in [12], [13], of which, results from test case l_{4-5} is listed in this work to show the applicability of Replacement Solutions in unstructured topologies.

A. Line Outage on l_{4-5}

Fig. 6 shows the sequence of state transitions the system goes through following a line outage on l_{4-5}. Fig. 6(a) is the outage state with l_{5-11} in red indicating a line overload. A greedy FACTS placement [12] on the most overloaded line mitigates this overload and prevents the possibility of a cascading failure. This in fact is the $\mathcal{D}_H$ event which results in a set of $\mathcal{D}_L$ observations as illustrated in Fig. 6(b). The comparison is done w.r.t. the outage state.

[4]Power Systems Test Case Archive http://www.ee.washington.edu/research/pstca/

The $\mathcal{D}_L$ observation on l_{5-11} flips for the sibling l_{5-6} and replicates for l_{5-8} thus, follows the observation inheritance rules defined in [3]. The main source of power in this case is flowing from the generator at bus_{10}, which does not change between the outage state and the $\mathcal{D}_H$ event state. As a result, flow in l_{5-8} decreases while flow in l_{8-30} increases. This excess power flow to the load on bus_{11} through the path $bus_8 \rightarrow bus_{30} \rightarrow bus_{17} \rightarrow bus_{16} \rightarrow bus_{12} \rightarrow bus_{11}$ to meet its demand. Note that the irregular connectivity of the system prevents $\mathcal{D}_L$ observations of the $\mathcal{D}_H$ event from propagating through the whole network in the same manner as structured networks.

Fig. 6(c) shows the first $\mathcal{R}_{\blacktriangle}$ of the $\mathcal{D}_H$ event on l_{5-11}. In comparison to the $\mathcal{D}_H$ event state in Fig. 6(b), there are no $\mathcal{D}_L$ observable changes in this state. What this means is that the $\mathcal{R}_{\blacktriangle}$ and $\mathcal{D}_H$ event are Nondeducibility secure since both the original placement and its $\mathcal{R}_{\blacktriangle}$ produce the same $\mathcal{D}_L$ projection.

Not all Replacement Solutions possible. For example, Fig. 6(d) is the $\mathcal{R}_{\blacktriangledown}$ for of the same $\mathcal{D}_H$ event. As seen here, the $\mathcal{R}_{\blacktriangledown}$ was not as successful in obfuscating the initial $\mathcal{D}_H$ event. Placing a FACTS device on l_{4-11} is infeasible because it would restrict the load at bus_4 from meeting its demand. This leaves l_{11-13} as the lone $\mathcal{R}_{\blacktriangledown}$ eligible power line, which unfortunately cause the initial overload to reappear. Not only that, there are a significant number of $\mathcal{D}_L$ observation differences between the initial $\mathcal{D}_H$ event and the $\mathcal{R}_{\blacktriangledown}$.

VIII. CONCLUSION

This paper presents the specifics of the replacement solutions driven Nondeducibility algorithm for smart grids including its applicability. Smart grid infrastructures with significant cyber-physical integrations are prone to external observation

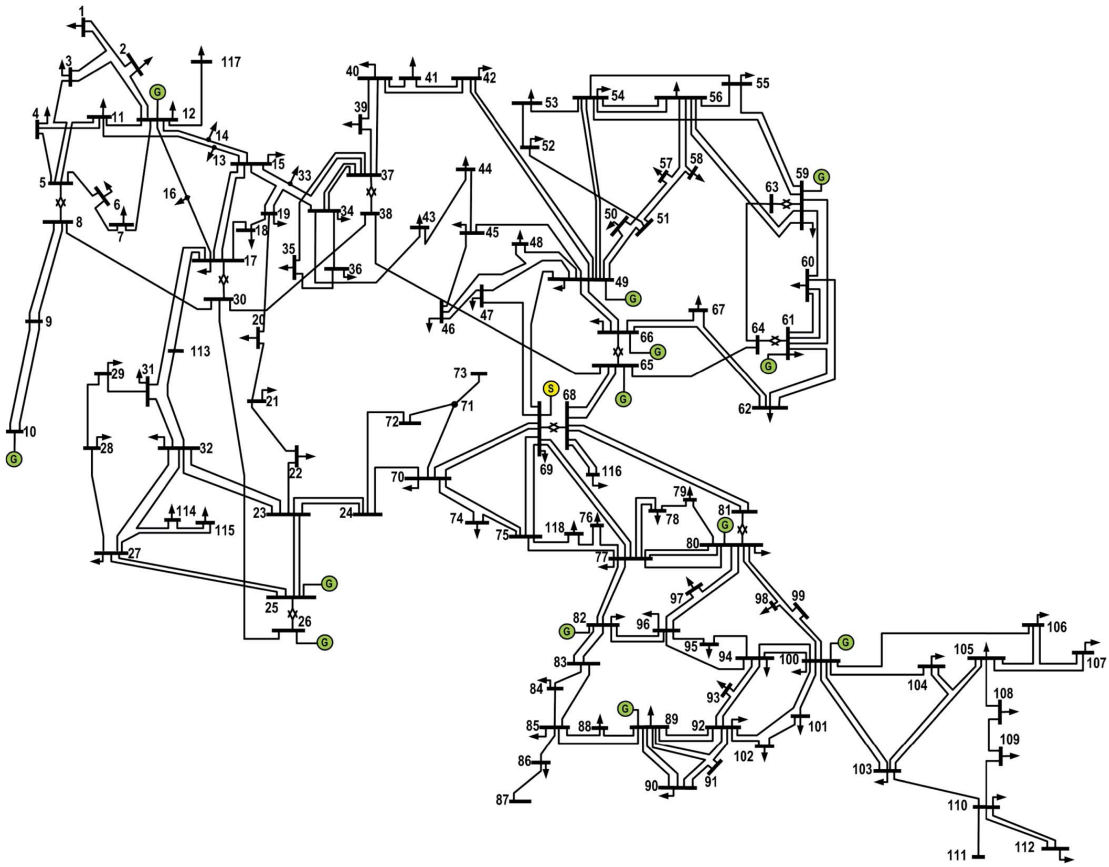

Fig. 5. The IEEE 118-bus test system.

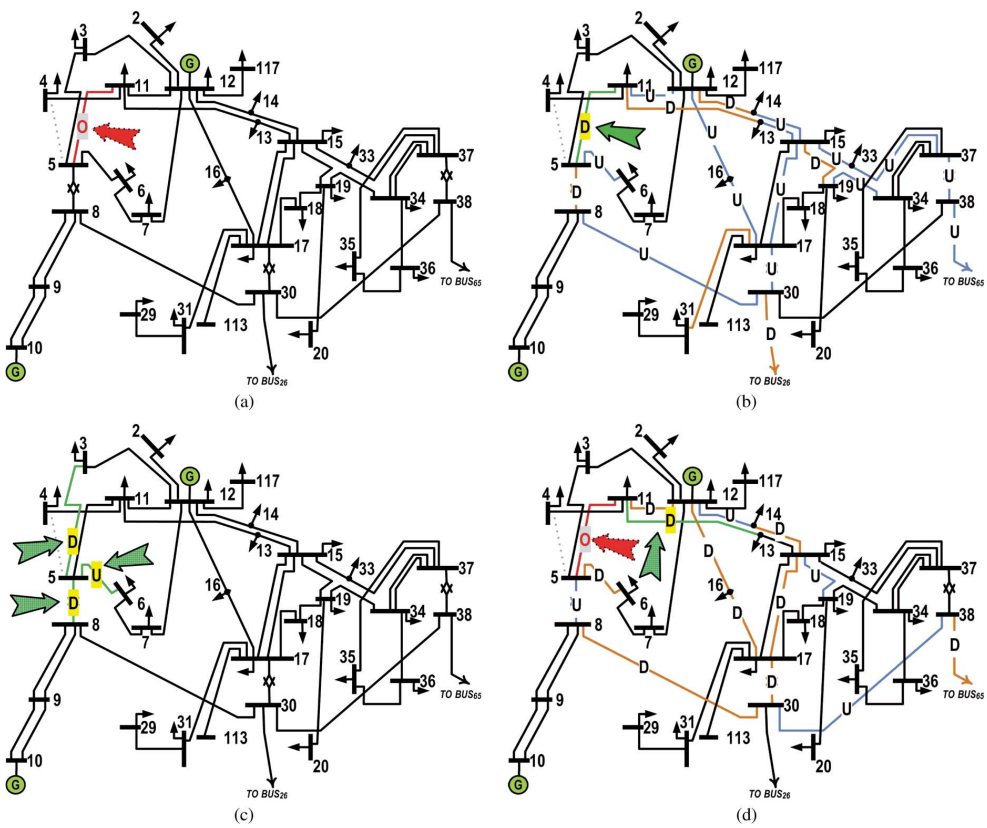

Fig. 6. Line outage l_{4-5} states. (a) Outage state; (b) $\mathcal{D}_H$ event state; (c) $\mathcal{R}_{\blacktriangle}$; (d) $\mathcal{R}_{\blacktriangledown}$.

based confidentiality violations. This work proposed an information flow security properties (IFP) based confidentiality model that can mitigate such violations. While this work lays out the necessary ground work to analyze complex cyber-phys-

ical system (CPS) dynamics in a semantically integral way, more work needs to be done to build a theory of composition under theoretical (irregularity in connectivity), operational (resource availability), and framework (computational overhead) constraints. In that respect, the significance of the work presented here is the extension of previously developed theories into implementations that can be readily used in smart grids and, more generally, in cyber-physical systems.

Having a higher number of replacement solutions is a positive indication of Nondeducibility since this ensures a greater degree of $\mathcal{D}_H$ event confidentiality. The number of controllers, however, involved in replacement solutions increase with its distance from the initial placement; For practical purposes, it's infeasible to place a controller on each line. Thus, most of these are theoretical solutions. A more feasible option would be to bound the solution space to a certain number of hops from the original location and only consider solutions that comprise of an agreeable number of controllers.

REFERENCES

[1] National Institute of Standards and Technology, "NISTIR 7628 guidelines for smart grid cyber security: Vol. 2, privacy and the smart grid," The Smart Grid Interoperability Panel—Cyber Security Working Group, Aug. 2010 [Online]. Available: http://goo.gl/uWCt2

[2] North American Electric Reliability Corporation, "Reliability standards for the bulk electric systems of North America," pp. 66–224, Jun. 2012 [Online]. Available: http://goo.gl/YQcww

[3] T. T. Gamage, T. P. Roth, and B. M. McMillin, "Confidentiality preserving security properties for cyber-physical systems," in *Proc. IEEE 35th Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Jul. 2011.

[4] National Institute of Standards and Technology, "NISTIR 7628 guidelines for smart grid cyber security: Vol. 1, smart grid cyber security strategy, architecture, and high-level requirements," The Smart Grid Interoperability Panel—Cyber SecurityWorking Group Sep. 2010 [Online]. Available: http://goo.gl/h3UrG

[5] D. E. Bell and L. J. Lapadula, "Secure computer system: Unified exposition and multics interpretation," MITRE Corp., Tech. Rep. ESD-TR-75-306, 1976.

[6] S. Sridhar, A. Hahn, and M. Govindarasu, "Cyber-physical system security for the electric power grid," *Proc. IEEE*, vol. 100, no. 1, pp. 210–224, Jan. 2012.

[7] F. B. Schneider, "Enforceable security policies," *ACM Trans. Inf. Syst. Security*, vol. 3, no. 1, pp. 30–50, 2000.

[8] J. Ligatti, L. Bauer, and D. Walker, "Edit automata: Enforcement mechanisms for run-time security policies," *Int. J. Inf. Security*, vol. 4, no. 1-2, pp. 2–16, Feb. 2005.

[9] N. Nagatou and T. Watanabe, "Run-time detection of covert channels," in *Proc. 1st Int. Conf. Availability, Rel., Security (ARES)*, 2006, pp. 577–584.

[10] D. Sutherland, "A model of information," in *Proc. 9th Natl. Comput. Security Conf.*, Baltimore, MD, USA, Sep. 1986, pp. 175–183.

[11] T. T. Gamage, "CEEME: Compensating events-based execution monitoring enforcement for cyber-physical systems" Ph.D. dissertation, Missouri Univ. Sci. Technol., Rolla, MO, USA, Dec. 2011 [Online]. Available: http://scholarsmine.mst.edu/thesis/pdf/Gamage09007dcc809510e1.pdf

[12] A. Lininger, B. McMillin, M. Crow, and B. Chowdhury, "Use of max-flow on FACTS devices," in *Proc. 39th North Amer. Power Symp. (NAPS)*, Oct. 2007, pp. 288–294.

[13] B. Chowdhury and S. Baravc, "Creating cascading failure scenarios in interconnected power systems," in *Proc. IEEE Power Eng. Soc. Gen. Meet. 2006*, pp. 18–22.

Thoshitha T. Gamage received his B.S.E. in computer engineering (2006) from the University of Peradeniya, Sri Lanka, the M.S. in computer science (2008) from St. Cloud State University, St Cloud, MN, USA, and the Ph.D. in computer science (2011) from the Missouri University of Science and Technology, Rolla, MO, USA.

He is currently a Research Assistant Professor at Washington State University, Pullman, WA, USA. Dr. Gamage's primary research interests are in computer security and its intersections with formal methods, distributed systems, and cross-disciplinary architectures, in particular, cyber-physical systems.

Thomas M. Roth received his B.S. degree in computer science from the Missouri University of Science and Technology, Rolla, MO, USA, in 2011. He is currently working towards the Ph.D. degree in computer science from the same university. His research interests are in the detection of dishonest peers in distributed cyber-physical systems.

Bruce L. McMillin is currently a Professor of Computer Science and Director of the Center for Information Assurance at the Missouri University of Science and Technology. He leads and participates in interdisciplinary teams in formal methods for fault tolerance and security in distributed embedded systems with an eye towards critical infrastructure protection. His current work focuses on protection for advanced power grid control.

Dr. McMillin has authored over 90 refereed papers in international conferences and journals. He served as program co-chair for the 2007 IEEE Computers, Software, and Applications Conference and its 2011 general chair. He is leading the distributed grid intelligence project of the Future Renewables Engineering Research Center.

Mariesa Crow (F'10) received her B.S.E. in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, and her Ph.D. in electrical engineering from the University of Illinois, Urbana-Champaign, IL, USA.

She is the Fred Finley Distinguished Professor of Electrical Engineering at the Missouri University of Science and Technology. Dr. Crow's area of professional interest is computational methods and power electronics applications to renewable energy systems. She is a Registered Professional Engineer in the State of Missouri.