On Mitigating Covert Channels in RFID-Enabled Supply Chains

Kirti Chawla[1], Gabriel Robins, and Westley Weimer

Department of Computer Science
University of Virginia, Charlottesville, VA 22904, USA
{kirti, robins, weimer}@cs.virginia.edu

Abstract. In a competitive business environment, RFID technology can help a business to optimize its supply chain. However, it may also enable an adversary using covert channels to surreptitiously learn sensitive information about the supply chain of a target business. We argue that the tracking of tags and the compromising of readers can create covert channels in the supply chain and cause detrimental market effects. To mitigate such attacks, we propose a framework that enables a business to monitor its supply chain in a fine-grained manner. We model the supply chain as a network flow graph, select key nodes to verify the tag flow, and actively search for covert channels. We note that optimal checkpoint node selection is NP-Complete, propose node selection and flow verification heuristics with various tradeoffs, and discuss appropriate countermeasures against covert channels detected in the supply chain. These practical methods can be implemented economically using current RFID technology.

Keywords. RFID, Covert Channel, Supply Chain, Network Flow

1. Introduction

Radio Frequency Identification (RFID) enables items to be tracked via attached tags, which respond to radio fields emitted by readers in their vicinity. A business can use this technology to make its internal processes more efficient and optimize its supply chain. RFID technology can streamline all phases of the production cycle, including pre-production activities, asset management, inventory control, production tracking, shipping, recalls and warranty authorization [1, 2].

However, the pervasive nature of RFID technology can also help adversaries glean sensitive information about the internal processes of a target business [6]. An adversary can track and/or modify existing tags, inject duplicate tags into an existing item flow, and compromise RFID readers in the supply chain of a target business. We say that such activities "*taint*" the flow of the information, and constitute covert channels in the supply chain of a target business [8]. These covert channels can surreptitiously reveal item flow patterns, including segregation, assimilation sites, site-specific inventory, delivery schedules, and other valuable sensitive information. An adversary can use this illicitly obtained information to gain an unfair (and not necessarily even illegal)

[1] Corresponding Author. This research is supported in part by a U.S. National Science Foundation grant CNS-0716635 (PI: Gabriel Robins).

marketplace advantage over a target business. Given such possible threats, it is important for a target business to verify the information flow in a fine-grained manner in order to detect the presence of covert channels and mitigate their effect.

Our contributions towards these goals are as follows. We analyze the threat sources in an RFID-enabled supply chain by enumerating four representative (but not exhaustive) attacks which an adversary can use to track the supply chain of a target business. We consider, both qualitatively and also using simulations, the ability of such attacks to affect market change. We model supply chains using network flow graphs, where nodes represent the sites and edges model the flow of items among sites. We select key nodes of the supply chain flow graph to verify the information flow. We call these selected nodes the "*taint checkpoints*", and refer to the process of their optimal selection as the "*taint-check cover*" problem, which we note is NP-Complete.

We propose taint-check cover heuristics based on various tradeoffs, such as the number of desired taint checkpoints. We propose verification algorithms that verify the flow of information, both locally and globally, in the supply chain. Our algorithms provide user-controlled tradeoffs between the strength of the verification results versus the time required to compute them. This enables post-detection actions to be taken by the target business either at a local site or along global paths. Finally, we evaluate our algorithms using a supply chain simulator, and provide a set of remedies that a target business can utilize to mitigate the effect of the discovered covert channels.

This paper is organized as follows. In section 2 we present the threat model chosen to analyze the RFID-enabled supply chain, and enumerate four possible attacks on such supply chains. In section 3 we describe some likely market change scenarios as a direct outcome of possible attacks. We discuss potential candidate models for supply chains and propose using network flow graphs in section 4. In section 5 we show that determining the optimal taint-check cover is NP-Complete, present taint-check cover heuristics, and describe verification algorithms to detect the presence of covert channels in a supply chain. We evaluate our algorithms by developing a supply chain simulator, as described in section 6. We detail possible mitigating contingencies in section 7, and conclude with future directions in section 8.

2. Threat Perception in RFID-enabled Supply Chain

In this section, we discuss the threat model chosen to analyze RFID-enabled supply chains, and present four possible supply chain attacks.

2.1.Threat Model

We present a motivating example to highlight the underlying assumptions used in the proposed threat model. Consider two competing businesses, each developing largely-interchangeable products, such as cellular phones. These businesses differentiate their products via competitive pricing and/or features, and are subject to user preferences and brand loyalty. For the sake of simplicity, assume that both businesses are competing in the same markets and target the same consumer base.

To remain competitive, these two businesses strive to optimize their internal processes to make their supply chains more efficient. When the target business invests in a new technology such as RFID, it examines the associated costs and benefits. While the benefits may be obvious in terms of efficient inventory control, production tracking, warranty authorization, etc., the cost of such a technology may involve more than just the direct cost of RFID equipment and processes.

The adversary business may also adopt RFID technology to remain competitive with respect to the target business. However, the adversary can also exploit the pervasive nature of this technology to clandestinely learn patterns of item flow in the supply chain of the target business. This can be construed as a form of industrial or economic espionage, wherein an adversary can use such discovered patterns in time-sensitive way, to provide lower consumer prices, or flood its products into selected regions or stores while the products of the target business become scarcer. Such practices can significantly reduce the profitability of the target business. The resulting profit drop can be viewed as hidden cost, which the target business would find difficult to anticipate, or even to correctly identify. Recent advances in RFID technology and the proliferation of its usage can thus give an adversary selective "*insider access*" to the target business supply chain, without direct access to target business' physical premises.

2.2.Attacks

A target business supply chain can inadvertently reveal its item flow to an adversary in a number of conceivable ways. We enumerate four representative (although not exhaustive) possible attacks, some of which have already received attention from the security research community [6, 14]. We explain the significance of these attacks when applied to a supply chain scenario, discuss the potential implications, and present possible ways to mitigate such attacks. Although such attacks are not dependent on any given RFID standard, for the sake of concreteness this paper assumes the EPC Gen2 standard [11, 12].

Tag tracking: In this attack, the adversary tracks the existing tags over the supply chain of a target business. We note that tags can be applied at the item-level or case-level. We assume that a target business assembles the finished product at its factory, attaches the tag at the case-level, and then ships them to geographically-separate warehouses. Upon arrival, these cases are organized into different batches and delivered to various retail stores. An adversary can learn the item-flow by copying the information stored in some case-level tags, and then querying them at different places in the supply chain. We note that such copied case-level tags constitute a covert channel, as they leak item-flow information from the target business to the adversary, while traveling unobtrusively through the supply chain of the target business.

Tag duplication: In this attack, an adversary copies the information stored in an existing tag and constructs a duplicate tag. Consequently, the adversary attaches this duplicate tag to a different case, enabling it to become part of the supply chain of the target business and thus a covert channel source. The adversary then queries the cases at different points of the supply chain to determine the item flow (e.g., if the adversary sees both the duplicate case-level tags at a warehouse then they aggregate at that

warehouse starting from different locations). This attack scenario is stronger than the previous tracking-only attack, since here the adversary is required to inject duplicate tags into the supply chain. Tag duplication hardware is relatively inexpensive and easily available; thus, an adversary can mount such an attack with modest effort [18].

Tag modification: In an EPC Gen2-compliant tag, there are four memory banks – Reserved, EPC, TID and User. The inventory process in a target business supply chain primarily uses the EPC portion of a tag's memory, and typically ignores the contents of the other memory banks. Therefore, an adversary can modify the information in the writable portions of other memory banks, which can then serve as a covert channel source. Independently, it has been suggested that the unused portion of memory of a tag can be utilized to conceal information [6, 17]. Such a vulnerability can be an attractive target to an adversary, due to its potentially large payoff versus relatively low effort to exploit.

Reader compromise: With rapid advances in RFID technology, various types of RFID readers are available in a wide variety of form-factors, hardware/software combinations, and use-case scenarios (i.e., handheld, rack-mountable, battery-powered, etc.). Many of these readers are deployed in supply chains in a manner that enables an adversary to compromise them (e.g., snooping on a wireless reader transmission, compromising the on-board software of a mobile reader, etc.). We differentiate two variations of this attack, as described in Figure 1.

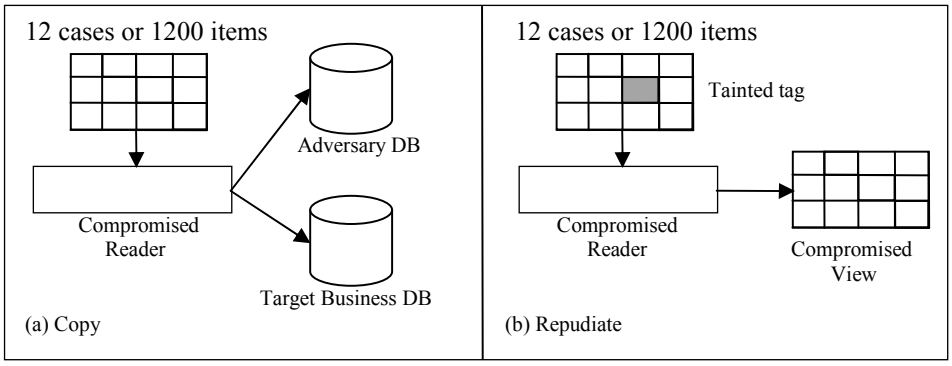


Figure 1: Reader compromise attack: (a) a compromised reader makes a copy of case-level tags; and (b) a compromised reader repudiates presence of covert channel.

In Figure 1(a), a compromised reader copies a limited number of case-level tags, and provides its information to an adversary. In Figure 1(b), an adversarial compromised reader selectively ignores the presence of any duplicate or modified tags. The reader's compromised view enables a covert channel to exist unobtrusively in the supply chain of the target business. We note that such a reader-compromise attack subsumes the tag duplication and modification attacks in terms of potential risks to the target business. From the adversary's perspective, in order to ensure a successful attack, while at least one compromised tag at the case-level is necessary, it may not be sufficient, since that tag may fail or become undetectable. Thus several compromised (i.e., duplicated and/or modified) tags should be used at the case-level (e.g., three tags per 100). On the other hand, if an adversary deploys too many compromised tags (e.g.,

half of the total), the adversary's exposure risk also increases dramatically. Moreover, an adversary may not need to track item-flow information at the item-level, since case-level tracking is sufficient for that purpose. The above types of attacks create covert channels that are said to "*taint*" the supply chain of a target business.

3. Projections for Market Change Scenario

In this section we examine two possible market scenarios to illustrate the potential impact of the RFID tag attacks described above. We note that a supply chain involves business-related variables such as stock levels at factories and retailers, delivery schedules from raw-material site to warehouses, numbers of back-orders, etc. Such strategic business information leak can occur as a result of attacking the supply chain, which can enable an adversary to engage in unfair competitive practices. Furthermore, an adversary can affect negative market changes by knowing the business practices of its competitors. We used the Anylogic supply chain model simulator [13] to obtain sample projections which, while simplistic, can still qualitatively describe possible outcomes of such attacks.

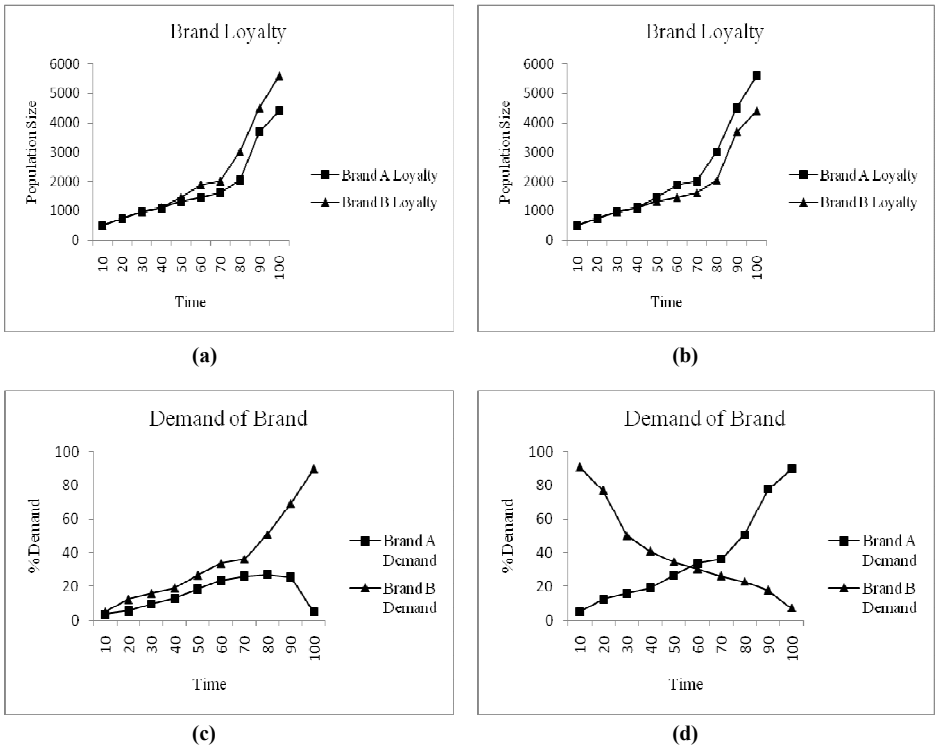


Figure 2: Market change projections using the Anylogic supply chain simulator [13]: (a) Consumers prefer brand B over brand A; (b) Consumer switch from brand B to brand A; (c) Brand B enjoys more demand than brand A; and (d) Brand A demand increases, while brand B demand decreases.

3.1. Brand Loyalty Switch

In the first scenario (Figure 2(a) and 2(b)), we consider two businesses serving a population of 10,000 consumers with brands A and B, respectively. We assume the two brands to be interchangeable, and have the same retail price. The business with brand B is the target business, while the business with brand A is the adversary. Consumers must purchase either brand A or brand B every time unit (i.e., the product is a staple item). In Figure 2(a), consumers are projected to prefer brand B to brand A by 55% to 45% (i.e., whenever a consumer arrives at a store, he chooses a product at random from the set of available equivalent products, preferring brand B slightly over brand A). However, by carefully timing its production so that more brand A products are available at a time when few brand B products are stocked or available, the adversary can induce consumers to switch brands. In Figure 2(b), the adversary has succeeded in inducing the consumers to switch brands, now favoring A over B by 57% to 43%.

3.2. Brand aversion

In the second scenario (Figure 2(c) and 2(d)), we consider a neighborhood store served by two businesses A and B, as before. Stores often stock products that enjoy consistent demand, in order to maintain profitability. Initially, the store stocks both items in equal amount. However, at a later point, as shown in Figure 2(c), brand B (i.e., the target business' product) is projected to have a higher demand than brand A by 89% to 5%. There is typically a demand threshold below which it will become non-profitable to stock a brand (i.e., "*brand aversion*"). An adversary aiming to bolster its own shelf presence may resort to illegitimately acquiring sensitive supply chain information of the competitor's business. Figure 2(d) projects such a scenario, when the adversary engages in supply chain attacks to obtain time-sensitive information about a target business, and use it to manipulate the market.

3.3. A Note on the Projected Market Change Scenarios

Enabling a supply chain with RFID technology entails attaching RFID tags at the item-level or case-level and tracking them throughput the supply chain using RFID readers. The target business keeps track of items starting from the purchase phase (e.g., in raw material form) through the distribution phase (i.e., in finished product form, stored at different warehouses or retail outlets). An adversary can use the possible attacks described above in order to learn vital strategic information, resulting in the projected market scenarios, which are detrimental to the target business.

If the benefits to an attacker are higher than its incurred costs, the adversary has strong motivation (i.e., economic incentive) for perpetrating such attacks. We believe that such attacks are viable in an RFID-enabled supply chain, given the potentially high payoff to an adversary, although specific occurrences of such attacks seem to have not yet been publicly reported. While we have argued that the exposure of only a few business variables to an adversary can result in an unfair (and not necessarily even illegal) marketplace advantage, it would be interesting to study more elaborate and detailed marketplace scenarios and projections. Such "*what-if*" scenarios can stimulate further discussions regarding the associated risks as well as the effectiveness of possible solutions in RFID-enabled supply chains.

4. Supply Chain Model

In this section, we focus on the problem of modeling a supply chain, towards the goals of preventing an attack or mitigating its effects. A supply chain typically spans multiple geographically dispersed sites and involves numerous phases that include the sourcing of raw-materials, processing and storing the end-product, and delivering the product to markets and consumers [6]. Supply chain models can be categorized as deterministic models, stochastic models, hybrid models, economic models, and IT-driven models [1, 2]. While these models intend to capture many aspects of a supply chain in great detail, our aim is to construct a simpler model that enables us to focus on the fundamentals and roots of potential attacks.

In any supply chain, there are item-flows between sites (e.g., raw materials moving among various locations), however in a RFID-enabled supply chain, item-flow between sites is analogous to "*tag-flow*", since RFID tags are attached to each item. The supply chain consists of multiple phases, wherein each phase is a collection of sites. Furthermore, to detect the presence of duplicate tags, modified tags, and compromised readers, we need mechanisms to track item-flows between supply chain phases. With these three key observations in mind, we have developed a model based on network flow graphs [10], which we call "*supply chain flow graphs*".

4.1. Phases

A supply chain can be broadly divided into three phases: the purchase phase, the production phase, and the distribution phase (e.g., sites associated with the production phase are involved primarily in manufacturing a product). Each phase of the supply chain is a collection of interconnected sites with an item-flow among them. We define the supply chain flow graph $G = (V, E)$ as a directed connected graph, where a node p corresponds to a site and an edge (p, q) models a connection between the two sites. Each edge $(p, q) \in E$ has a positive item-flow capacity $C(p, q) > 0$, while "*non-edges*" have 0-capacity: $\forall (p, q) \notin E.\ C(p, q) = 0,$. There are two special nodes called the "*source node*" (S) and the "*sink node*" (T). We partition the supply chain flow graph into three sub-graphs, corresponding to the purchase phase, production phase, and distribution phase, respectively.

Network flows are subject to the usual constraints on edge capacity and flow conservation at nodes [10]. We propose an additional property, namely the node maximal outgoing flow, which will enable us to address issues related to attacks. There are typically multiple paths for item-flow in a supply chain. A "*critical node*" or "*critical edge*" may experience more item-flow than other paths. To model this characteristic in the supply chain, each node keeps track of its maximum outgoing flow. If two nodes have the same maximal outgoing flow, we resolve the tie by giving precedence to the node having a higher flow value predecessor. Supply chain flow graphs with such criticality labels facilitate reasoning about issues related to possible attacks and item-flow inspections.

4.2. Taint Checkpoints

A direct approach for detecting covert channel attacks can entail inspecting for tainted RFID tags at every node of the supply chain. However, this would be prohibitively expensive and time consuming. Instead, we propose to select a subset of nodes, called "*taint checkpoints*", verify the item-flow at these selected locations, and report the presence of any discovered covert channels in the supply chain flow graph. When RFID tags are attached to items by the target business in the early phases of the supply chain, the information present on them is recorded in order to track inventory. In subsequent phases of the supply chain, this information is available to taint checkpoints for the purpose of inspection and verification. This verification process involves comparing the information present on the currently viewable RFID tag with trusted, stored information. Any mismatch may indicate the presence of covert channels or other tampering. Figure 3 illustrates a supply chain flow graph, including several taint checkpoints where item-flow is inspected.

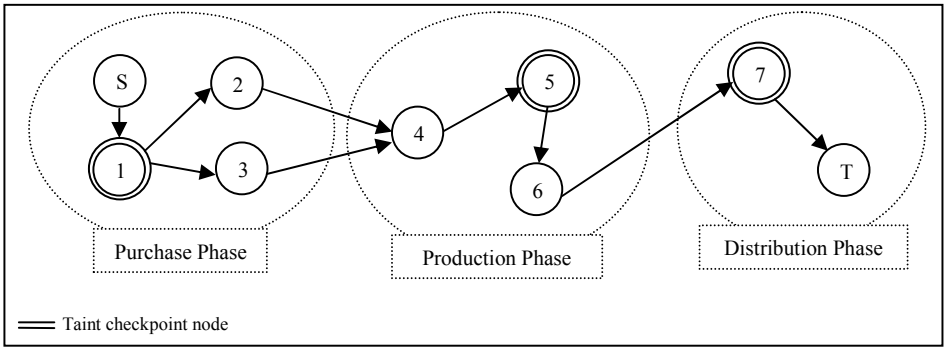


Figure 3: A supply chain flow graph with three taint checkpoints.

5. Taint Check Cover Generation and Verification Algorithm

In this section, we formulate the problem of optimal selection of taint checkpoints in the supply chain flow graph, observe that it is NP-Complete, and suggest heuristics to generate good approximate solutions.

5.1. Taint Check Cover Problem Statement

To ensure the absence of covert channels in the supply chain, the taint checkpoints should provide broad coverage for the entire graph. The associated optimization problem is to select as few taint checkpoints as possible, while providing broad coverage for the entire supply chain flow graph. Thus we seek a "*taint check cover*" V' of the supply chain flow graph $G_U = (V, E)$, where $V' \subseteq V$ and such that every edge of E has at least one of its end points in V'. Note that we may choose to only cover some critical subset of the flow graph's nodes, rather than the entire graph. Either way, this objective corresponds to the classical graph vertex cover problem, which is known to be NP-complete [10].

5.2. Heuristic Taint Check Cover Generation

There is a simple efficient heuristic for vertex cover that produces solutions of size no worse than twice the optimal [10]. This heuristic selects an arbitrary graph edge, adds its two endpoints to the growing vertex cover solution, eliminates this edge and its endpoints from the graph, and iterates until the graph is exhausted. To see that this scheme produces a 2·OPT solution, we observe that one of the two nodes of each removed edge must be present in any optimal solution. Given the high degree of freedom in how edges (and thus nodes) are selected in constructing such a heuristic taint check cover solution, a target business may introduce different selection criteria, based on practical, economic, or strategic considerations.

Parameters: A target business may wish to limit the number of taint checkpoints, seek tradeoffs between the efficiency of its supply chain versus the coverage provided by taint checkpoints, consider checkpoint selection criteria based on the specific structure of the supply chain, etc. To address these considerations, we introduce two parameters:

1. **Taint checkpoint to nodes ratio (or *TNR*):** This is defined as ratio of taint checkpoints to graph nodes in the supply chain flow graph, and enables the target business to control the number of taint checkpoints:

$$TNR = \frac{|V'|}{|V|} \qquad \text{where, } |V| \neq 0 \qquad (1)$$

2. **Coverage to efficiency ratio (or *CER*):** The ratio ε of coverage and efficiency provides a tradeoff to balance the quality of item-flow inspection against the overall operational efficiency of the supply chain:

$$CER = \varepsilon \qquad \text{where, } \varepsilon > 0 \qquad (2)$$

Heuristic template: When determining a taint check cover, the target business may choose from a continuous tradeoff between efficiency and coverage. This can be achieved by using the parameters *TNR* and *CER* to determine from which subset of the flow graph nodes (i.e., V') a taint check cover will be selected (using, e.g., the 2·OPT node cover heuristic [10], the techniques described in [20], or any other node cover heuristic). We can also presort the node selection pool by increasing node maximal outgoing flow values, in order to give higher priority to high-flow nodes. Alternatively, the nodes can be permuted in some other manner (e.g., by aggregate product value, time-criticality, or even randomly), in order to capture topological or economic considerations during the construction of a taint check cover. In summary, our template is quite general in that it can utilize (based on the above parameters) any reasonable criteria to determine which node subset will be used from which to select a taint check cover (using an arbitrary node cover heuristic).

We note that not nearly every node and/or edge in the flow graph must necessarily be covered (i.e., imbued with taint-checking capability). This is because tainted tags that are missed at some points along the graph will likely be discovered at subsequent locations downstream. On the other hand, including any flow graph "*cut*" in the taint check cover can ensure that every tainted tag will be discovered in at least one location.

An alternative taint check cover can therefore entail selecting a small (but somewhat redundant) set of cuts across the flow graph. This can insure at relatively low infrastructural cost that any tainted tags moving in the graph will eventually be detected. The taint checkpoints chosen in Figure 3 demonstrate such a cut-based cover. Choosing low-cost (or even optimal) graph cuts can be accomplished using well-known min-cut algorithms [19].

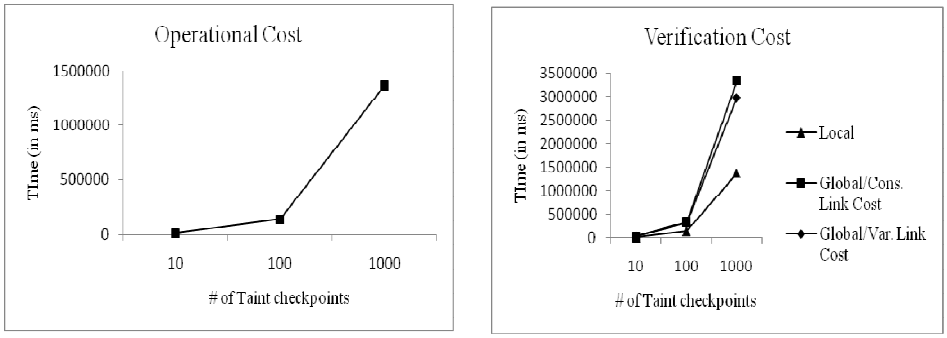
5.3. Verification Algorithm

Each node in the taint check cover (i.e., each taint checkpoint) is responsible for inspecting and verifying the item-flow passing through it. Each item in this flow has a unique RFID tag ID. If a taint checkpoint reads multiple counts of the same tag ID, or the system detects the same tag ID at two different places simultaneously, then a duplicate tag has been detected. By comparing the information present on each viewable tag with data stored a priori in a trusted tag database, modifications to tags can be detected at taint checkpoints. Item-flow verification can be performed "*locally*" at a given taint checkpoint or "*globally*" across a given path or cut, as checkpoints accumulate, exchange, and compare tag information.

6. Evaluation

We used simulations to evaluate our proposed approaches. We assume a base supply chain flow graph configuration of 2000 nodes, and selected between 10 and 1000 nodes to be taint checkpoints. Each checkpoint verifies 1000 cases of 100 items at each time interval. We assume each checkpoint has direct access to a trusted database implementing a tag lookup service. In our first simulation, we measure the relationship between the number of taint checkpoints and cumulative time required to perform local verification. Figure 4(a) shows that as the number of taint checkpoints increase, there is a corresponding increase in the time to locally verify the item-flow.

Our second simulation evaluates our global verification algorithm, which collects local verification results from taint checkpoints. The cost of the collection process depends on the underlying speeds of the network links.



(a) **(b)**

Figure 4: (a) cumulative local verification time as a function of the number of taint checkpoints; and (b) local and global verification costs as a function of the number of taint checkpoints.

Figure 4(b) shows the simulated verification cost when the link cost is either a constant (500 ms) or a variable time window (ranging from 2 to 1000 ms), based on the node's geographical distance from the central database server. We thus explored the verification communication cost as the number of taint checkpoints increases. We note that the communication cost can grow rapidly in the more realistic scenario where taint checkpoints are at large variable distances from the central node.

7. Responses to Covert Channels

In this section, we enumerate some possible response actions available to the target business when the covert channels are detected in its supply chain. Note that the presence of covert channels in the supply chain can never be completely ruled out, even when privacy-preserving algorithms are used in the underlying RFID technology [4, 5].

Passwords: According to EPC Gen2 standard, an RFID tag is required to support password protection for read or write access to the tag. The systematic use of passwords can mitigate tag tracking, tag duplication, and tag modification attacks. However, this requires that all RFID hardware in the supply chain support and conform to the same password scheme.

Pseudonyms: An RFID tag using pseudonyms transmits a slightly different ID each time it is queried [3]. This can prevent the adversary from discovering patterns in a supply chain, but requires the target business to accommodate the pseudonym scheme in its tracking logic. Burmester *et al.* describe an unlinking technique that can also be used to prevent tag tracking attacks [15].

Re-encryptions: The use of encryption to conceal the tag data still allows the adversary to track the static encrypted tag over the supply chain. To defeat such an attack, the tags can be re-encrypted after each phase of the supply chain, in order to prevent the adversary from modifying or tracking the tags.

Direct Mitigation: Rieback *et al.* describe a device that can be used for sweeping and preventing reader compromise attacks [7]. When a covert channel source is discovered, we can physically clear the operating environment while temporarily altering the flow of items. Oua *et al.* present a path checking technique that can trace tags following the altered route [16].

Physically Unclonable Functions (or *PUF*s): PUFs are hardware random number generators that rely on inherent wire-delays and process variations [9]. PUF-based privacy-preserving algorithms provide a way to build message authentication codes to ensure data integrity and aid in preventing tag modification attacks.

8. Conclusion

In this paper, we discussed and analyzed vulnerabilites in RFID-enabled supply chains, and enumerated possible attacks that can be mounted with relatively modest effort. We have shown that an adversary can learn item-flow patterns in the RFID-enabled supply chain of a target business, which may result in harmful market change scenarios. We proposed a concise model for reasoning about supply chain flow and

RFID attack mitigation, and demonstrated that attacks can be detected and addressed at a few select nodes in the supply chain. For the NP-complete problem of checkpoint selection we presented a practical heuristic template that can trade off attack coverage for efficiency. We simulated and analyzed these algorithms, and enumerated possible responses by a target business to covert channels. While this work is preliminary, we view it as an important step toward the analysis and mitigation of attacks on RFID-enabled supply chains. Possible future research directions include extending the basic model to include additional practical considerations, fine-tuning the heuristics to take these additional practical considerations into account, and further study the tradeoffs between coverage and efficiency.

References

[1] H. Min and G. Zhou, Supply Chain Modeling: Past, Present and Future, *Journal of Computer and Industrial Engineering*, Elsevier Science Direct, Volume 43, Issue 1-2, pp. 231-249, July 2002.

[2] R. Angeles, RFID Technologies: Supply-Chain Applications and Implementation Issues, *Information Systems Management*, 22:1, pp. 51-65, 2005.

[3] D. Molnar, A. Soppera and D. Wagner, A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags, *Selected Areas in Cryptography*, Ontario, Canada, 2005.

[4] D. V. Bailey, D. Boneh, E. Goh and A. Juels, Covert Channels in Privacy-Preserving Identification Systems, *14th ACM International Conference on Computer and Communication Security*, Alexandria, Virginia,pp. 297-306, 2007.

[5] S. L. Garfinkel, A. Juels and R. Pappu, RFID Privacy: An Overview of Problems and Proposed Solutions. *IEEE Security and Privacy*, Volume 3, Issue 3, pp. 34-43, May 2005.

[6] A. Mitrokotsa, M. R. Rieback and A. S. Tanenbaum, Classification of RFID Attacks. *International Workshop on RFID Technology*, Barcelona, Spain, pp. 73-86, June 2008.

[7] M. R. Rieback, B. Crispo and A. S. Tanenbaum, RFID Guardian: A Battery-Powered Mobile Device for RFID Privacy Management, *Lecture Notes in Computer Science*, Springer, Volume 3574, pp. 184-194, July 2005.

[8] I. S. Moskowitz and M. H. Kang, Covert Channels – Here to Stay, *9th IEEE International Conference on Computer Assurance*, pp. 235-243, July 1994.

[9] L. Bolotnyy and G. Robins, Physically Unclonable Function-Based Security and Privacy in RFID System, *5th International Conference on Pervasive Computing and Communications*, New York, USA, pp. 211-128, March 2007.

[10] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms, Third Edition*, MIT Press, Cambridge, 2009.

[11] EPCGlobal, UHF C1 G2 Air Interface Protocol Standard http://www.epcglobalinc.org/standards/uhfc1g2/uhfc1g2_1_1_0-standard-20071017.pdf

[12] EPCGlobal, Tag Data Standards Version 1.4, Revision June 11, 2008 http://www.epcglobalinc.org/standards/tds/tds_1_4-standard- 20080611.pdf

[13] Anylogic Professional 6. AB-SD Supply Chain Model Simulator, http://www.xjtek.com

[14] G. Avoine, C. Lauradoux, and T. Martin, When Compromised Readers Meet RFID, *Workshop on RFID Security*, Leuven, Belgium, 2009.

[15] M. Burmester and J. Munilla, A Flyweight RFID Authentication Protocol, *Workshop on RFID Security*, Leuven, Belgium, 2009.

[16] K. Oua and S. Vaudenay, Pathchecker: An RFID Application for Tracing Products in Supply-Chains, *Workshop on RFID Security*, Leuven, Belgium, 2009.

[17] A. Karygiannis, T. Phillips, and A. Tsibertzopoulos, RFID Security: A Taxonomy of Risks, *Conference on Communications and Networking in China,* Beijing, China, pp. 1-8, 2006.

[18] J. Mandel, A. Roach, and K. Winstein, MIT Proximity Card Vulnerabilities, *Technical report*, MIT, March 2004. http://web.mit.edu/keithw/Public/MIT-Card-Vulnerabilities-March31.pdf

[19] M. Stoer and F. Wagner, A Simple Min-Cut Algorithm, *Journal of the ACM*, Vol. 44, Issue 4, pp. 585-591, July 1997.

[20] J. Chen, I. A. Kanj, W. Jia, Vertex Cover: Further Observations and Further Improvements, *Journal of Algorithms*, Publisher: Elsevier, Vol. 41, Issue 2, pp. 280-301, November 2001.