

Monitoring Security and Safety of Assets in Supply Chains

Ganna Monakova¹ and Cristina Severin² and Achim D. Brucker¹ and Ulrich Flegel³, and Andreas Schaad¹

¹ SAP Research, Vincenz-Priessnitz-Str. 1, 76131 Karlsruhe, Germany
{ganna.monakova, achim.brucker, andreas.schaad}@sap.com

² esciris gmbh, Max-Eyth-Str. 38, 71088 Holzgerlingen
cristina.severin@esciris.de

³ HFT Stuttgart, Schellingstr. 24, 70174 Stuttgart, Germany
ulrich.flegel@hft-stuttgart.de

Abstract In the today's world of the global economy supply chains become more and more complicated. Widely distributed supply chains open more possibilities for attacks on both IT as well physical level. The potential threats can span over multiple supply chains. For example, if the same truck is used to transport chemicals and then the same truck is used to transport food, a contamination threat arises that neither of the supply chains can detect when analysed independently. In this paper, we present a tool-supported framework that extends modelling and execution of supply chains processes with specification, execution and monitoring of the security and safety constraints that are used to protect supply chain assets. The tool allows to detect not only threats scoped to a single supply chain, but cross-cutting threats that can only be detected through analysis of the whole system.

Keywords: Supply Chain Security, Monitoring, Resource modelling

1 Introduction

Security, safety, and compliance of a supply chain process and involved assets are critical to any organisation. Especially all supply chain participants want to be sure that assets sent to another party are treated correctly and that assets received from another party can be trusted. For example, supply chain partners not only want to ensure that the purchase order data and the payment data are correct, but also that the ordered goods have been treated according to various requirements. To obtain such an assurance, partners need to get a proof that the asset will be treated in a certain way, e.g., it will be ensured that the temperature and packaging of the ordered goods is correct. As many enterprises need to comply to regulations such as the European Food Safety regulations (e.g., see Regulation (EC) No. 882/2004 of the European Parliament and of the Council of 29 April 2004 and related documents), there is a strong demand to specify and communicate security and safety requirements on the level of the supply chain models. In this way, each of the supply chain participants will be



able to explicitly specify taken security and safety measures to other partners. Documenting such requirements as part of the supply chain model can also be used as contract specification between supply chain participants. During the execution of a supply chain, the compliance with the specified security and safety requirements needs to be monitored and certified.

In addition to the threats related to a single supply chain, there are a number of threats that occur when different supply chains come together. For example, product cross-contamination can occur when incompatible goods belonging to different supply chains are stored in the same area. This means, that in addition to the threats scoped to a single supply chain, threats that span across multiple supply chains must be considered. We refer to such threats as *contextual* threats, emphasising that a specific issue only becomes a threat when the environmental and logical contexts of a supply chain execution are considered.

Together with retailers, freight carriers, and food manufactures, we identified the following objectives that must be supported by a framework for business level specification of the security and safety constraints:

- *Security and Safety Awareness*: a supply chain participant should be aware of security and safety threats for the assets used in a supply chain.
- *Security and Safety Visibility*: a supply chain partner should be able to communicate security and safety requirements as well as taken measures through visual representations or annotations.
- *Security and Safety Consistency*: requirements should be fulfilled in a consistent way.
- *Security and Safety Provability*: it should be possible to prove fulfillment of specified requirements.

To address these requirements, we developed an approach for modelling and monitoring security and safety requirements. In contrast to previous work [7] that presented the general approach, in this paper we concentrate on the modelling and detection of security and safety threats scoped to a single supply chain, as well threats related to the relationships between different supply chains.

2 Motivating Example

According to [4], approximately one-third of all fresh fruit and vegetables produced worldwide is lost before it reaches consumers. In [5] the authors state that sometimes the losses and wastage of the food may even reach 50 percent between field and fork. Incorrect harvesting, transport, storage and packaging play an important role in these losses.

To demonstrate the developed approach we consider two initially independent supply chains, one that orders and delivers ice cream, and another one that orders and delivers toxic chemicals. Figure 1 presents the simplified model of the ice cream supply chain involving three parties: *Retailer* sends an order to *Production* in the *Order* activity; Producer dispatches the required amount of the product (*Dispatch* activity) and uses *Logistic* partner to deliver it (*Transport* sub-process) to *Retailer*. There are two data objects modeled in the process: *Pur-*

chaseOrder object contains all information required to make an order, including required amount of the product and the delivery destination; *IceCream* data object represents the actual physical good that is passed between the supply chain participants.

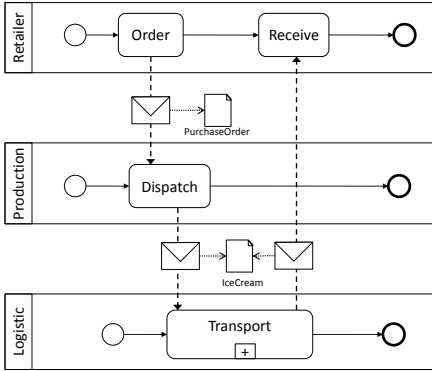


Figure 1. A supply chain process

the same basic structure: it has a *Retailer* who orders chemicals from a *Production* unit. The chemicals are delivered through a *Logistic* partner. Similar to the first supply chain it contains two assets: *PurchaseOrder* and *Chemical*. While *PurchaseOrder* in this case has the same threats as in the previous supply chain, *Chemical* has a threat of being exposed, e.g., through evaporation.

In addition, when considering both of the supply chains, there is a potential threat of ice cream contamination through the toxic chemicals if at any point in time the ice cream will either be stored too close to the chemicals, or the same truck will be used first for the transportation of the chemicals, and then for the ice cream.

With respect to the four objectives we identified, the desired outcome in our example is as follows: support during supply chain design with identification of the potential threats that span over multiple supply chains for the *PurchaseOrder*, *IceCream* and *Chemical* assets to achieve *security awareness*; automated help in identification of countermeasures for the identified threats to achieve *consistency* in applying security controls; suitable tools for visualisation of the security measures taken to protect assets for *security visibility* and *provability* on the design level; extension of the execution environment to allow monitoring and execution of security measures to achieve *provability* of the taken measures during runtime.

3 Proposed Approach

Ensuring the safety and security of assets in supply chains requires special support both at design and at runtime (see Figure 2). At design time, we need to specify the security requirements and, moreover, we need to apply and config-

PurchaseOrder contains sensitive information and must be protected against tampering. For instance, the *Production* wants to be sure that the requested amount and the destination address have not been changed by an unauthorised party. *Retailer* wants to be sure that *IceCream* has been handled in a correct manner, e.g., temperature of the product was always in the region of -26°C to -25°C and that there was no unauthorised access to the product during transportation to ensure that product was not deliberately contaminated.

The second supply chain has the

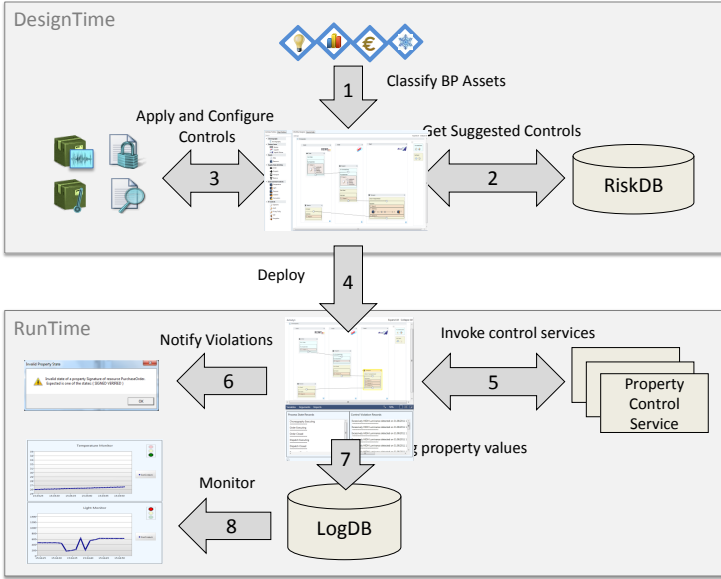






















Figure 2. Approach in prototypical implementation

ure the necessary controls. The selection of necessary controls is supported by a specific risk database (RiskDB). At runtime, the compliance to the specified security and safety requirements is monitored and violations are reported. On the modelling level, the three main concepts of our approach are *Asset*, *Threat*, and *Control*. An asset has potential threats, while controls can countermeasure these threats. The role of the rest of the model is to help identify which threats are applicable to which asset and which controls can be used to countermeasure identified threats. The following describes the main steps of our approach:

1. *Asset identification*: In this step we analyse what are the assets used in a supply chain that we want to protect. We identified two types of assets: a *logical asset* is data that contains certain sensitive information, such as purchase order details or credit card number, while a *physical asset* is a real world object that is used in the supply chain. Any asset can be described by a set of *properties* it possesses. Thereby any logical asset can be described by a set of the same properties, such as *signature*, *content* and *encryption* properties. Similar, any physical asset can be described by the set of the same properties, such as *temperature*, *location* and *size*.
2. *Threat modelling*: Different threats are applicable to different assets depending on asset classification. Thereby it is not sufficient to only distinguish between logical and physical assets. For example, two logical assets can have different threats: the first logical asset might contain *private* information about a customer with a threat of information disclosure, while another log-

Physical Asset		Logical Asset	
Deep-frozen		Private	
Light-sensitive		Financial	
Explosive		Legal	
Poisonous		Confidential	
Radioactive		Audit-relevant	

Physical Asset		Logical Asset	
Light		Signature	
Temperature		Encryption	
Concussion		Audit-Controls	
Pressure		Privacy-Policies	
Location		Separation of Duties	

(a) Tags for assets

(b) Controls for assets

Figure 3. Tags and controls for logical and physical assets

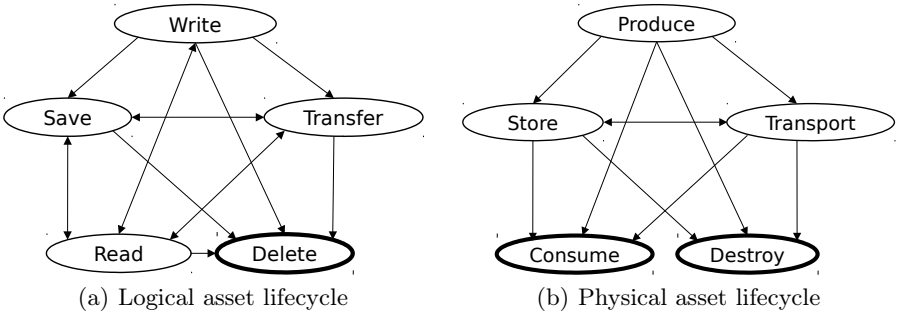


Figure 4. Action lifecycles for physical and logical assets

ical asset might contain *financial* data, which has a threat of unauthorised modification. Similar, a *frozen* physical asset might have a threat of being defrozen, while a *fragile* physical asset has a threat of being broken. To allow supply chain designer to classify different assets, a concept of *tag* has been introduced. A tag attached to an asset identifies a certain characteristic or classification of this asset. 3(a) shows an example set of tags that can be used to classify logical and physical assets. An asset can be a subject to a number of actions. Figure 4 shows the action based lifecycles for logical and physical assets where bold lines denote the final actions performed on an asset⁴. Each action in the asset lifecycle bears different threats depending on asset classification. For example, the *Read* action on *financial* data bears a threat of consuming incorrect or maliciously modified information, while the *Consume* action on *food* bears a threat of consuming contaminated products. RiskDB (see Figure 2) stores a set of rules that relates asset classifications

⁴ We assume that a data object can only be written once: if a data object is modified then the old data object is deleted and a new data object is created.

defined by the asset tags to the potential threats depending on the actions performed on this asset.

3. *Control identification*: After the potential threats for each activity and each asset have been identified, we need to specify countermeasures to protect assets from these threats. The RiskDB contains rules that map different threats to the possible countermeasures, also called controls, that can be applied to prevent, detect or react to the threat occurrences. For example, if an activity writes a data object, then it might need to sign it to protect the data from unauthorised modification and to ensure non-repudiation of the *financial* data. If an activity stores data, then it might need to encrypt the data before saving it to protect the *confidential* data. Similar, if an activity transports a physical asset, it has to apply controls with regard to the transportation regulations for the given object depending on the asset classification defined by the tags, e.g. *flammable*, *explosive*, or *deep-frozen* goods.

Controls can be divided into three main categories based on their execution point with respect to the threat occurrence. We will use the threat of food contamination to demonstrate the difference between these categories:

- *Preventive controls* are controls that are applied to prevent a certain threat. For example, ensuring correct storage conditions, such as low temperature and clean storage areas, are some of the preventive measures that can be taken to avoid food contamination.
- *Detective controls* are used to identify occurrence of a threat. For example, laboratory examination can detect contamination of a product, while evaluation of the temperature sensor data allows to detect incorrect storage conditions.
- *Reactive controls* are used to recover from a violation. For example, if contamination of a product has been detected, then product recall must be initiated. In case of a high temperature detection an emergency refrigerator can be started.

When a threat is scoped to a single supply chain, such as for example melting of the ice cream, all of the control types can be realised locally to the execution of the supply chain. However, when we consider contextual threats, such as product cross-contamination, detective controls become more complicated. In general, there are four stages to a detective control:

- (a) *Signalling*. At this stage all events related to the threat are collected. For example, to detect cross-contamination we require location information, such as GPS data, related to the position of ice cream and toxic chemical.
- (b) *Evaluation*. At this stage the collected events are evaluated according to the rules that identify threat occurrence. In our example we would see that the goods are located too close to each other.
- (c) *Notification*. After a violation has been detected, the responsible party is notified at this stage. In our example the retailer who ordered the ice cream must be notified. In addition other participants and legal authorities can be notified as well.

- (d) *Reaction*. After a violation notification has been received, reactive actions can be taken to recover the fault. If cross-contamination has been detected before the ice cream reached the retailer, the retailer might decline the order. Otherwise all contaminated ice cream must be destroyed. In case of contextual threats signalling is implemented locally on each supply chain, while evaluation of the produced events must happen on the external component that can combine events coming from different supply chains.

4 Implementation

Our approach is supported by a prototype: To demonstrate design concepts we extended Windows Workflow Foundation (WF 4.0) with the supply chain modelling capabilities. We introduced assets, tags and control modelling blocks that enable specification of security and safety in the supply chains.

4.1 Workflow Design

Windows Workflow Foundation uses variables to represent data used in a business process. The variables are defined in a variable tab and are not visible in the designer. To advocate security awareness, we extended existing workflow modelling constructs with two visual elements for logical and physical assets. We added an asset (or variable) panel to the business process, which contains all assets used in the process. To add a new asset (variable) to the process, the user needs to drag & drop the corresponding visual element into the asset panel of the workflow.

To enable asset classification we provide a tag toolbar: the user can drag & drop the corresponding tag from the toolbar onto the visual asset specification being present in the asset panel. By combining different tags, a user can specify different characteristics of an asset.

Figure 5 shows a screenshot of the ice cream supply chain process modeled using our tool. It contains two variables that can be seen in the right panel: an *IceCream* variable annotated with a *DeepFrozen* and *LighSensitive* tags, and a *PurchaseOrder* variable annotated with *Financial* and *AuditRelevant* tags.

Figure 5 shows four activities: *Order*, *Dispatch*, *Transport*, and *Receive*. The *Order* activity outputs *PurchaseOrder*, which is then passed as an input argument to the *Dispatch* activity. The *Dispatch* activity then outputs *IceCream*, which is passed to the *Transport* activity and then through the *Transport* activity to the *Receive* activity. Depending on the argument type (In, Out or InOut), we can see different types of control points available for each asset in each activity. This allows the user to define input state controls on the incoming asset states (*PurchaseOrder* in *Dispatch* activity) output controls on outgoing asset states (*PurchaseOrder* in *Order* activity), and internal controls on data that exists all the way through activity execution (*IceCream* in *Transport* activity).

To identify controls required to countermeasure potential threats, we developed a Risk Database (RiskDB). The RiskDB stores relations between asset

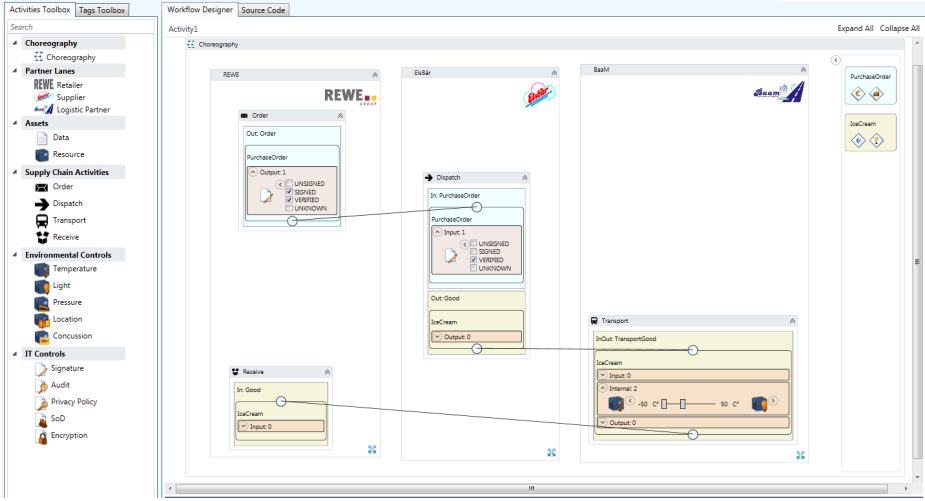


Figure 5. Design of the supply chain process

tags, threats these tags imply for different activities, and controls that should be applied to such assets in each activity. When a user annotates an asset with a new tag, a query is sent to the RiskDB that selects the potential threats for the current asset classification and each activity that uses this asset, as well as applicable protection measurements (controls) for each identified threat. After this the tool checks if the controls are already present in the model and if not, shows an error with the information about missing controls. This enforces the user to model secure processes with respect to the rules stored in the RiskDB. The rules in RiskDB reflect expert knowledge with respect to each asset classification.

To enable control specification, we provide a control toolbar. To identify at which point of activity execution a control must be applied, the user needs to drop a control into the corresponding container. In Figure 5 we can see an output signature control applied to the *PurchaseOrder* variable in *Order* activity. This control specifies that the data must be signed when it leaves this activity. In the *Dispatch* activity we can see an example incoming state control, that states that the *PurchaseOrder* signature property must be in state *verified* to be used by this activity. In the *Transport* activity the internal temperature and light controls are applied. The controls specify that the *IceCream* temperature must be between -50°C and -25°C and the light must be under 200lm . Additional controls could be added as input and output controls. In general, any number of controls can be applied to each asset in each activity at any control point.

Design time extensions of the workflow foundation provides security visibility and awareness by providing tag and control toolbars, security awareness and consistency through connection to the RiskDB that consistently applies the same rules in similar situations and notifies violations if any controls are missing, and

security provability on design level by showing that there are no missing controls in the model with respect to the RiskDB rules.

4.2 Workflow Execution and Monitoring

To enable execution of the extensions, the visual assets have been mapped to the variables and passed as arguments into the corresponding activities. Incoming state controls and outgoing state controls are enforced by the workflow engine—it invokes property related control services to verify that the asset properties are in a correct states. If a violation is detected it either suspends the workflow, reschedules the failed activity, or executes any other specifically defined reactive process. The internal controls on the other side can be viewed as the requirements on the activity implementation with regard to the asset handling. Each control knows the property it targets. When a control is scheduled, it invokes the corresponding property control service. Such a service can be an internal implementation, such as automatic signature implementation, but can also be a remote service, such as sensor control that monitors resource temperature. In general, all property-specific actions are done by the property control services. This allows for a general model of the controls in the business process: a business process control knows the asset it needs to control, the property it targets, the service that can evaluate the state of this property, the point in time when the state needs to be evaluated, and the states that are allowed at the evaluation time point. The property control service knows how to determine the current state of the resource and how to modify the state, but it is unaware of the business related semantics or the valid states of this property. At the specified execution point, the control asks a property service to evaluate the current state of an asset, logs results into the *LogDB* (see Figure 2), compares it with the set of valid states and notifies the user, if the state is invalid.

All input controls scoped to an activity are evaluated before this activity starts its execution. If any violations are detected during these checks, the process terminates. After each activity execution, all output controls scoped to these activity are evaluated, and, if any violations are detected, the activity is reiterated⁵ until all assets have the valid property states. For the internal controls, monitors are triggered at the beginning of the activity execution and stopped when activity completes. Monitors observe, evaluate and log the states of the corresponding properties during the activity execution with the specified frequency. Collected evidence can then be used to prove fulfillment of the specified restrictions. If a violation of an internal control is detected during activity execution, business process partners are notified.

Figure 6 shows an example monitoring screenshot taken during the simulation of the *Transport* activity in the supply chain process. On the right side we can see a chart representing the values logged by the internal temperature and light

⁵ Process termination and activity reiteration have been implemented as examples of the possible reactions to the control violations. In general, any customer-defined actions can be used as reactions to the detected violations.

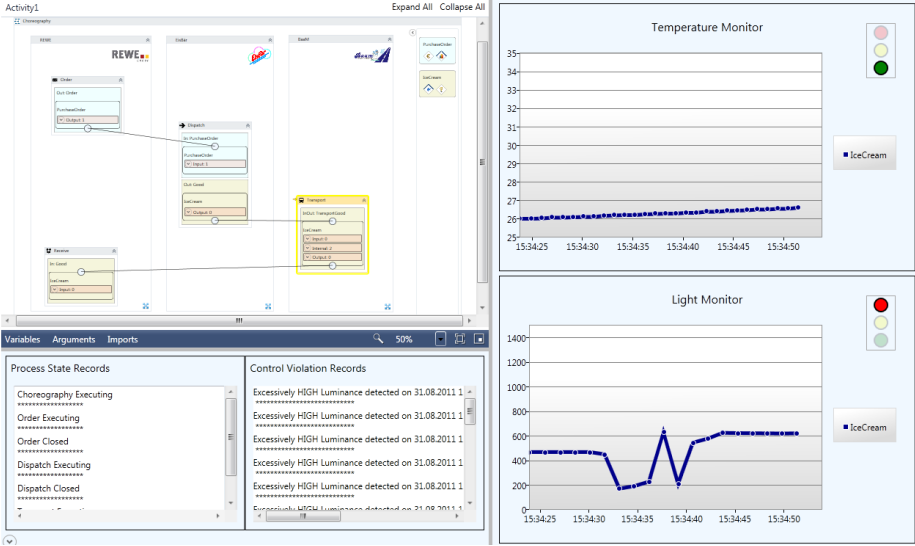


Figure 6. Execution of the supply chain process

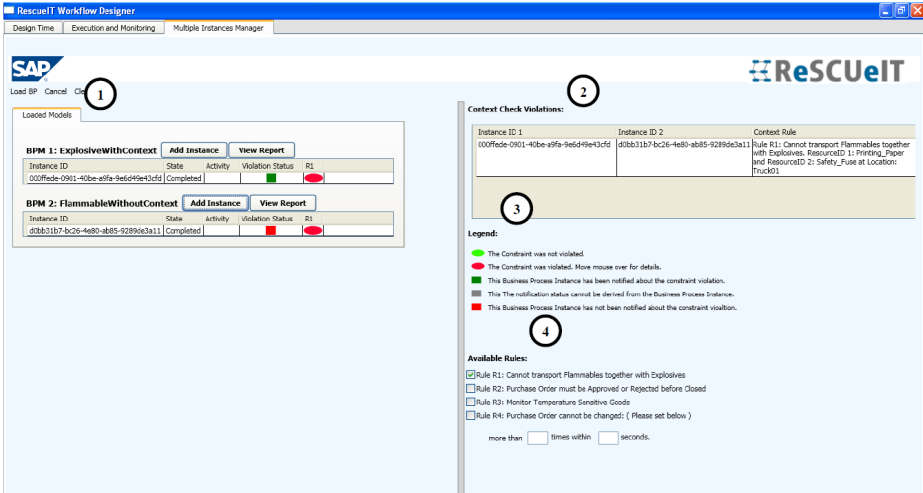


Figure 7. Detection of contextual threat occurrences

controls. Traffic light symbol in the top right corner of the light monitor is red (top circle of the traffic light), signalling that a violation has been detected. To compare, the traffic light of the temperature monitor is green (bottom circle of the traffic light), which shows that the temperature is in the valid region. At the bottom left of the screen we can see the tracking information about the current

state of the process execution and logged violations, while the currently active activity is highlighted on the top left part of the screen.

Figure 7 shows a screenshot of the contextual dashboard. To detect contextual threats we evaluate events coming from different supply chains using a complex event processing engine. The screenshot in Figure 7 displays the state of two running process instances in (1) with respect to the rules activated in (4). In the top left corner (1) it can be seen that rule R_1 that forbids transportation of incompatible goods has been violated, which in our implementation is signalled through the red circle in R_1 column. The *Violation Status* column visualises whether the corresponding supply chain instance has been notified about the violation (status green) or not (status red). In our scenario only the first supply chain has been notified, while the second one does not know about it at the current point in time. The table at (2) gives details about the rule violations and impacted supply chains, while (3) explains the legend.

5 Related Work

There is a large body of literature, e.g., [10,1,6] that motivate the need for improving the process visualisation to enhance the understanding of the processes in general. It is well known that workflows and business processes are security critical. For example, in [3] the authors present workflow related security goals and study their possible assignment to main categories of business process elements such as agents, roles, artifacts, and activities. Consequently, there are several work, e.g., [11,12,2,9] that suggest domain-specific extensions of a process modelling language for expressing safety or security properties, only a few, namely [12,2], use these extensions for monitoring or enforcing the specified properties at runtime. From those, [2] is the closest to our work: the authors of [2] present a tool-supported approach for modelling security properties on the business process level and to generate both security configurations for standardised security infrastructures as well as specific security controls for a business process execution engine. Still, this work does not discuss physical assets and, moreover, does not integrate a risk database.

6 Conclusion and Future Work

We presented an approach allowing business users to easily specify security and safety requirements of supply chains. The compliance to these requirements is monitored during the execution of the processes. Overall, this transfers the well known model-driven software development paradigm to workflow management systems that can execute the abstract process models directly.

Our prototype has been developed in the context of a German funded project RescueIT that develops techniques for security- and safety-critical supply chains. This prototype has been showcased at various trade fairs and received positive feedback from the different parties involved in such supply chains. We found that even a non IT audience easily understands the visualisation of security

constraints (e.g., a signature symbol on a purchase order) as well as safety constraints (e.g., a temperature symbol on the purchased good).

Further work also includes the integration of business process constraint visualisation and analysis techniques. For example, [8] presents 3D visualisation approach that allows the analysis of business process constraints and dependencies between different process dimensions. Integrating such analysis and visualisation frameworks into our prototype would provide an integrated toolchain for business experts for modelling, analysing, and executing security-critical and safety-critical supply chains or business processes in a way that guarantees the monitoring and enforcement of the security and safety requirements.

Acknowledgments. The research leading to these results has received funding from the German “Federal Ministry of Education and Research” in the context of the project “RescueIT”

References

1. R. Bobrik, T. Bauer, and M. Reichert. Proviado - personalized and configurable visualizations of business processes. In *EC-Web*, pages 61–71, 2006.
2. A. D. Brucker, I. Hang, G. Lückemeyer, and R. Ruparel. SecureBPMN: Modeling and enforcing access control requirements in business processes. In *SACMAT*. ACM Press, 2012.
3. P. Herrmann and G. Herrmann. Security requirement analysis of business processes. 6:305–335, October 2006.
4. A. Kader. Increasing food availability by reducing postharvest losses of fresh produce. In *V International Postharvest Symposium, International Society for Horticultural Science*, 2005.
5. J. Lundqvist, C. de Fraiture, and D. Molden. Saving water: From field to fork – curbing losses and wastage in the food chain. In *SIWI Policy Brief*, 2008.
6. J. Mendling and J. Recker. Towards systematic usage of labels and icons in business process models. In *13th International Workshop on Exploring Modeling Methods for Systems Analysis and Design*, 2008.
7. G. Monakova, A. D. Brucker, and A. Schaad. Security and safety of assets in business processes. In *ACM Symposium on Applied Computing (SAC)*. ACM Press, 2012.
8. G. Monakova and F. Leymann. Workflow art: a framework for multidimensional workflow analysis. *Enterprise Information Systems*, 2012.
9. J. Mülle, S. von Stackelberg, and K. Böhm. A security language for BPMN process models. Technical report, University Karlsruhe (KIT), 2011.
10. S. Rinderle, R. Bobrik, M. Reichert, and T. Bauer. Business process visualization - use cases, challenges, solutions. In *ICEIS (3)*, pages 204–211, 2006.
11. A. Rodríguez, E. Fernández-Medina, and M. Piattini. A bpmn extension for the modeling of security requirements in business processes. *IEICE - Trans. Inf. Syst.*, E90-D:745–752, March 2007.
12. C. Wolter, M. Menzel, A. Schaad, P. Miseldine, and C. Meinel. Model-driven business process security requirement specification. *Journal of Systems Architecture*, 55(4):211–223, 2009.