Process Driven Access Control and Authorization Approach

John Paul Kasse[1], Lai Xu[1], Paul deVrieze[1], Yuewei Bai[2],
Computing and Informatics, Faculty of Science and Technology,
Bournemouth University,
Poole, BH12 5BB, United Kingdom
{jkasse, lxu, pdvrieze}@bournemouth.ac.uk
[2]Industry Engineering of Engineering College, Shanghai Polytehnic University,
Jinhai Road 2360, Pudong, Shanghai, P.R.China
ywbai@sspu.edu.cn

Abstract. Compliance to regulatory requirements is key to successful collaborative business process execution. The review the EU general data protection regulation (GDPR) brought to the fore the need to comply with data privacy. Access control and authorization mechanisms in workflow management systems based on roles, tasks and attributes do not sufficiently address the current complex and dynamic privacy requirements in collaborative business process environments due to diverse policies. This paper proposes process driven authorization as an alternative approach to data access control and authorization where access is granted based on legitimate need to accomplish a task in the business process. Due to vast sources of regulations, a mechanism to derive and validate a composite set of constraints free of conflicts and contradictions is presented. An extended workflow tree language is also presented to support constraint modeling. An industry case Pick and Pack process is used for illustration.

Keywords: Compliance, Collaborative Business Process, Verification and Validation.

1 Introduction

Compliance requires strict adherence to policies, norms and regulations by an organization's business processes which translate into products and services, e.g. products must meet quality standards, systems must data privacy must be preserved etc. Non-compliance is punishable with monetary fines or litigations. Business processes aim to achieve business objectives, yet compliance objectives provide are a form of controls that constrain the business process and overall operations.

To achieve a balance between objectives and compliance requirements, a compliancy by design approach is adopted where both business and compliance requirements are designed into the process. Data privacy management is a key driver made mandatory by the EU's General Data Protection Regulation (GDPR). It requires privacy by design, which in the business process context impacts the entire engineering process. Separation of Duty (SoD) and Binding of Duty (BoD) [1], [2] are other forms of constraints restricting process behavior from Sarbanes Oxley Act and Basel II.

Business processes are constrained by both company internal and external policies. As policies restrict valid executions of processes (or combinations of processes) these restrictions could lead to deadlocks in the process where the process is incapable of meeting the policy requirements [3]. For example, in a complex process with multiple restrictions the four eyes principle could lead to a problem where there is only 1 authorized person that meets the other restrictions. This makes the need for verification of process behavioral conformance with constraints legitimate.

Existing compliance frameworks do not address conflict checking among regulatory requirements [4], [5]. In a collaborative environment where different policies apply, an illustration of how to achieve a composite policy set and verifying it against contradictions, inconsistency and inaccuracy is desirable.

To address constraint modeling and validation problem in the context of regulatory requirements, an extended workflow tree language with constructs like OR, loops and time is presented. Using a constrained process model, we illustrate process driven authorization as a data access control mechanism with the case study introduced in [6].

The rest of the paper is organized as follows. Section two presents the motivating use case; section 3 presents the proposed language illustrating application of extended constructs while section 4 illustrates how to achieve a composite policy set and its verification. Section 5 discusses how to achieve process driven authorization. Section 6 presents related work and section 7 is conclusion and outlook.

2 Use case

Pick and Pack process is based upon actual industry use. It is collaborative and designed for use in international corporations (Europe and parts of Asia).

To create orders (Fig. 1 and 2) customers register online. Once order is received, the customer and the store are notified. The store staffs check order details, and proceed to pick and pack the order. Before handover, the order is verified to match with order details. For items that may be out of stock, the order is suspended for a period until stock is available or customer is contacted to seek opinion either to proceed without the item, substitute it or cancel the order. Delayed orders can be cancelled by customers; ready ones are picked or delivered by the delivery team. Individual stores may vary the process to fit specific contexts. Consequently, a family of process variants is created with different implications on the control flow, and data resource allocations.

3 Workflow Tree Language

Several formal approaches are used in process modeling with BPMN being a standard from [7]. BPMN limitations like inability to expressively support intuitive and in-depth analysis of business process models involving simulation, validation and verification [8].To support this analysis, models are enhanced with annotations; e.g. security and safety [8]–[10], model verification [11] etc.BPMN may not be the best formalism to model and verify compliance constraints because annotations come with associated complexity.

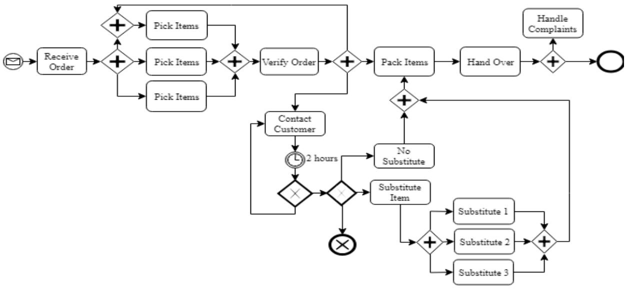

Fig. 1. A BPMN representation of the Pick and Pack business Process

To this effect, a Workflow Tree Language (WTL) is proposed. WTL is a popular approach in process modeling and validation. In Nikovski and Akihiro (2008) WTL is used to represent processes in a way that facilitates process mining in models where parallelism is not explicitly recorded. Crampton and Gutin (2013b) use WTL to express workflow model constraints to facilitate means to extend and solve the workflow satisfiability problem. The study however omits important constructs like the OR, loops and time which relevant current business processes. These constructs form part of our extension as represented by symbols in figure 2 to support the modeling and analysis of processes which are collaborative, adaptive and declarative [12] as well as expression of constraints relating to privacy, SoD, BoD, and need to know.

Table 1. Symbols and their meaning.

Symbol	Name	Symbol	Name
ǁ	Parallel	→	sequence
⊗	XOR	⊕	Inclusive OR
◎	Loop	X	Cancel

Workflow trees provide a natural hierarchical representation of processes. In an ordered tree, the process tasks and functional units are represented by leaves and internal nodes respectively. For instance, ◎T_5 represents a loop back to T_5 in a workflow. The X symbol is a cancel or termination e.g. customer cancels the order due to delay. WTL extension is intended to support verification; 1) among constraints to identify conflicts and inconsistencies, and 2) between model and the constraints. Using compliance attributes in Kasse et al. (2018), we illustrate to achieve of modeling and compliance verification for process models.

3.1 Constraint Expression with WTL

Constraints limit the behavior of the business process in terms of task ordering, resource assignment and data flow. WTL facilitates constraints expression in a manner useful to analyze and identify properties necessary to support their verification. Fig. 3 is a WTL pick and pack process model with SoD ($\neq$) and BoD ($=$) constraints symbols adopted from [13]. Constraints expression over models yields complexity and task redudancy. This neccesitates model verification to guarantee soundness.

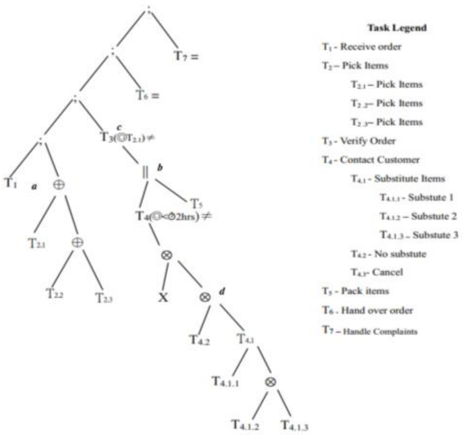

Fig. 2. A Workflow Tree representing a constrained pick and pack business process

In [6] useful compliance attributes in relation to branching and temporal constructs are suggested which we adopt to express compliance constraints over a WTL model. Fig. 2 illustrates expression of serialization (*a*), parallelism (*b*), looping (*c*), XOR (*d*) and choice (*e*) constructs as segments of the use case.

The OR is likely to introduce redundancy in the workflow tree. For example, if three tasks are represented on a single node. The nesting of tasks or use of similar labels for two nodes that have parent/child relationship should be avoided to retain a sound workflow tree. Simply adda child node to the current node. All nodes must have two children or otherwise be eliminated [14], [15]. Time based constraints specify temporal requirements defined as absolute time or relative time e.g. task durations, deadlines, task waiting time, resource availability, data access and authorization schedules. These compound into total process duration, e.g. the total order processing duration is 6 hours from submission time. Delays cause process costs or trigger exception handling tasks, e.g. when customers reject delayed orders it leads to a cancellation.

4 Optimal Policy Derivation and Validation

A mechanism to achieve a composite set of policies from internal and external policies and their validation is described. Policies change overtime directly impact on existing processes. Changes must be propagated to all areas where it has effect.

4.1 Optimal Policy Derivation

Regulations are specified in natural language without implementation specifics. Natural language can be a source of ambiguity. External regulations have a direct influence over internal policies and the two should not contravene, otherwise a violation results in the business processes. Mapping internal and external policies has associated complexity or requires skills not common to compliancy officers. A mechanism to

derive an optimal composite policy set is a step to solve the complexity and facilitate non-expert users.
Internal Policy set: composed of polices to regulate processes behavior. For instance, parties *a, b, c* collaborate on a business process each with individual internal policies. $\mathrm{IP}_{internal.a} = \{\mathrm{P}_{1.a}, \dots \mathrm{P}_{n.a}\}$, $\mathrm{IP}_{internal.b} = \{\mathrm{P}_{1.b}, \dots \mathrm{P}_{n.b}\}$, $\mathrm{IP}_{internal.c} = \{\mathrm{P}_{1.c}, \dots \mathrm{P}_{n.c}\}$
Contractual Policy set: an integration of non-contradicting policies from $\mathrm{IP}_{internal}$ to form a set $P_{contractual}$ binding all parties. If there other relevant policies outside of the $\mathrm{IP}_{internal}$, they are co-opted as P_{other}. Therefore,

$$P_{contractual} = \sum\nolimits_{i=a}^{c} \bigcup\nolimits_{\mathrm{P}_{internal,i}}^{\mathrm{P}_{other}} \quad (1)$$

Global Policies: composed of industry wide policies $P_{global} = \{R_{g1}, R_{g2} \dots R_{g3}\}$

Composite set: composed of global and contractual sets. Therefore

$$P_{Composite} = \sum (P_{contractual}, P_{global}) \quad (2)$$

The composite set should be complete to include all relevant policies.

4.2 Validation of the derived optimal equation

To validate the composite policy set, we define and formalize consistency and completeness equations to support formal reasoning to identify potential errors.
Consistency - equation with at least one solution. The composite set is composed of all non-repeating policies compounded in contractual and global sets.

$$\exists (P_{contractual}, P_{global}) \rightarrow P_{composite} \quad (3)$$

Simple consistency – a composite policy set is consistent if and only if there is no policy ρ in Φ such that a policy ρ and its negation exist in the same set, otherwise Φ is inconsistent. No policy should allow and disallow actions at the same time e.g. no resource assignment can be SoD and BoD at the same task otherwise a deadlock results

$$\exists\ \rho(P_{composite}) \leftrightarrow \nexists \neg\rho(P_{composite}) \quad (4)$$

Maximum consistency - a composite policy set is maximally consistent if and only if for every policy ρ is part of the set.

$$\forall \rho \in (P_{composite}) \leftrightarrow \exists \rho \in (P_{contractual}) \quad (5)$$

Completeness: $P_{composite}$ should include all relevant policies from the internal, contractual and global sets, otherwise it is incomplete.

$$\forall \rho \in (P_{contractual}) \supseteq (P_{composite}) \quad (6)$$

For all policy sets in contractual set are superset of the composite set.
From space limitation it is not possible to illustrate the mechanism with the use case.

5 Towards Process Driven Authorization (PDA)

With a consistent composite policy set, PDA mechanism aims to control access to data based on legitimate and legalized purpose for which it is required in the process. Access is granted with respect to time and history of task executions in the workflow.

5.1 Constraint formalization

Preliminary workflow W definitions are concerned with user – task assignment(u, t), user - role assignment ($UR \in U\ X\ R$) and role – permission assignment ($RP \in R\ X\ P$).
i. SoD constraint δ for two workflow tasks T_1 and T_2 is a tuple expressed as

$$\delta_{t_1 \in T_1, t_2 \in T_2} \rightarrow \neg\exists_{u \in U}\{(u, t_1), (u', t_2)\} \subseteq W \tag{7}$$

δ Constraint is satisfied iff there exists different users assigned to tasks t_1 or t_2 in W
ii. BoD constraint, β a user is assigned to execute two conjoint tasks t_1 and t_2

$$\beta_{t_1, t_2 \neq t_1} \rightarrow \forall_{u \in U}[(u, t_1) \subseteq W \rightarrow ((u, t_2) \subseteq W \wedge \neg\exists_{u' \neq u}[(u', t_2) \in W])] \tag{8}$$

β Constraint is satisfied if there exists a user assigned to execute tasks t_1 and t_2 in W. e.g. tasks 'pack items' and 'verify order' are executed by different users
iii. *Need to Know (N2K) constraint* η assigns special permission to execute task and access necessary data

iv. Authorization policy ℘ over a workflow is a triple composed of constraints SoD, BoD and need to know.

$$\wp \rightarrow (\beta, \delta, \eta) \tag{9}$$

Workflow history includes past executed task instances relevant for future user task assignment (UT). This makes the element of temporal constraint relevant. Temporal constraints assignment applies to the user, object, action to be executed and the intention to allow or deny access i.e.

$$tm\ =\ (\mathrm{I}, \mathrm{U}, \mathrm{A}, +/-\ ) \tag{10}$$

Where I is the period interval, U is the subject or user, A is the action to be taken (e.g. read) and +/ - permissions to allow or deny time based access. These variables fit well with the proposed time-based compliance attributes in Kasse et al. (2018). E.g. AllowBefore, AllowAt, DenyBefore, DenyAt etc. Since the user is already part of the task assignment, it is withdrawn to retain the formula as

$$tm\ =\ (\mathrm{I},\ \mathrm{A}, +/-\ ) \tag{11}$$

Therefore, an authorization policy with temporal constraint is

$$\wp \rightarrow (\beta, \delta, \eta, tm) \tag{12}$$

Additionally, access under PDA is granted with respect history h executions. A valid constrained workflow model is one that satisfies the authorization policy in reference to the execution history. The history is important during execution to check whether a previous user has right to access current task in reference to SOD and BOD. Formally, a constrained workflow model CW with a history is;

$$\mathrm{CW} \rightarrow (\wp, h) \tag{13}$$

Where, h is workflow history. An execution of a workflow model satisfying all constraints is an authorized model under PDA. PDA is achieved as a service at runtime which is contacted whenever a task is to execute. The authorization engine checks the assignments and grants or denies access.

6 Related work

The consistency of task based constraints is addressed in [5] where the authors derive a consistent constrained workflow schema. However, the study did not consider temporal constraints which we have addressed in this paper. Crampton and Gutin address workflow satisfiability problem using constraint expression in [16] and refine it in [17]. Compliance of a workflow to specified constraints is considered a workflow satisfiability problem which they provide solution to. Like the previous study, temporal constraints were ignored. In Basin (2012), an approach for deriving an optimal workflow aware authorization is presented as an NP hard problem and solved as a parameter tractable problem. We did not take that direction though it is a future plan. In [18], [19] a tool is implemented to automate the enforcement of privacy policies and requirements on personal data used in organization systems. The tool disregards other forms of compliance based on business process perspectives. In all, the studies are relevant to the subject of compliancy to regulations. However, none of them specifically supports optimal policy derivation as well as its validation.

7 Conclusion and outlook

This paper presents an explicit mechanism to compose and validate policies that originate from different sources. By presenting a mechanism to integrate, validate and verify different policy sets for consistency and completeness we contribute to the subject. The concept of process driven authorization as an access control mechanism to achieve compliance to data privacy and other regulations has been introduced along with a WTL. Using an industry use case, the concept has been illustrated. Currently we are working on theorem proofs and lemmas to make the concept more concrete.

Acknowledgement*:* This research has been sponsored by EU H2020 FIRST project (Grant No. 734599, FIRST: vF Interoperation suppoRting buSiness innovaTion) and National Key R&D Program of China (2017YFE0118700).

References

[1] E. Bertino, C. Bettini, E. Ferrari, and P. Samarati, “An access control model supporting periodicity constraints and temporal reasoning,” *ACM Trans. Database Syst.*, vol. 23, no. 3, p. 231, 1998.

[2] E. Bertino, E. Ferrari, and V. Atluri, “The specification and enforcement of

authorization constraints in workflow management systems," *ACM Trans. Inf. Syst. Secur.*, vol. 2, no. 1, pp. 65–104, 1999.

[3] G. Karjoth, "Aligning Security and Business Objectives for Process-Aware Information Systems," *Proc. 5th ACM Conf. Data Appl. Secur. Priv. - CODASPY '15*, pp. 243–243, 2015.

[4] S. Sadiq and G. Governatori, "Managing Regulatory Compliance in Business Processes," *Handb. Bus. Process Manag. 2*, vol. 2008, pp. 159–175, 2010.

[5] K. Tan, J. Crampton, and C. A. Gunter, "The Consistency of Task-Based Authorization Constraints in Workflow Systems," *Proc. 17th IEEE Comput. Secur. Found. Work.*, pp. 155–169, 2004.

[6] J. P. Kasse, L. Xu, and P. T. de Vrieze, "The need for Compliance Verification in Collaborative Business Processes," 2018.

[7] O. M. G. OMG, "Business Process Model and Notation (BPMN) Version 2.0," *Business*, vol. 50, no. January, p. 170, 2011.

[8] M. Salnitri, F. Dalpiaz, and P. Giorgini, "Modeling and verifying security policies in business processes," in *Lecture Notes in Business Information Processing*, 2014, vol. 175 LNBIP, pp. 200–214.

[9] G. Monakova, A. D. Brucker, and A. Schaad, "Security and safety of assets in business processes," *Proc. 27th Annu. ACM Symp. Appl. Comput. - SAC '12*, p. 1667, 2012.

[10] J. Müller, "Security Mechanisms for Workflows in Service-Oriented Architectures," 2015.

[11] G. Koliadis, "Verifying Semantic Business Process Models in Verifying Semantic Business Process Models in Inter-operation," 2007.

[12] J. P. Kasse, L. Xu, and P. de Vrieze, *A comparative assessment of collaborative business process verification approaches*, vol. 506. 2017.

[13] D. Basin and E. T. H. Zurich, "Optimal Workflow-Aware Authorizations," *Proc. 17th ACM Symp. Access Control Model. Technol. ACM.*, pp. 93–102, 2012.

[14] A. M. Awad, "A C OMPLIANCE M ANAGEMENT F RAMEWORK FOR B USINESS P ROCESS M ODELS," *PhD Thesis*, 2010.

[15] D. Nikovski and B. Akihiro, "Workflow trees for representation and mining of implicitly concurrent business processes," *ICEIS 2008 - Proc. 10th Int. Conf. Enterp. Inf. Syst.*, vol. 2 ISAS, pp. 30–36, 2008.

[16] J. Crampton and G. Gutin, "Constraint Expressions and Workflow Satisfiability," *Proc. 18th ACM Symp. Access Control Model. Technol. ACM*, pp. 73–84, 2013.

[17] D. R. dos Santos, S. E. Ponta, and S. Ranise, "Modular Synthesis of Enforcement Mechanisms for the Workflow Satisfiability Problem," *Proc. 21st ACM Symp. Access Control Model. Technol. - SACMAT '16*, pp. 89–99, 2016.

[18] M. C. Mont and R. Thyne, "Privacy policy enforcement in enterprises with identity management solutions.," *J. Comput. Secur.*, vol. 16, no. 2, pp. 133–163, 2008.

[19] M. C. Mont and R. Thyne, "A Systemic Approach to Automate Privacy Policy Enforcement in Enterprises information lifecycle management A Systemic Approach to Automate Privacy Policy Enforcement in Enterprises," *Policy*, pp. 118–134, 2006.