Clouds Coalition Detection for Business Processes Outsourcing

Amina Ahmed Nacer, Mohammed Riyadh Abdmeziem and Claude Godart
University of Lorraine, Nancy France

Keywords: Cloud Computing, Decoy Process, Privacy Preserving, Coalition, Know-How Exposure.

Abstract: Companies outsourcing their BP (Business Processes) to the cloud must be sure that sensitive information included in their BP are protected. While there are several existing methods that include splitting the model into a collection of BP fragments before a multi-cloud deployment minimizing therefore the likelihood of a coalition, a risk still remains. We propose in this paper an approach for detecting malicious cloud providers that initiate or participate to a coalition. To do that, we rely on decoy processes having the same structure and number of data as a real process, but with decision strategy making as well as data sets that are completely different (fake) from the real one. Our objective is to detect unexpected exchanges of messages between malicious clouds, that may signify an attempt to initiate or participate to a coalition. The level of reputation of each cloud initiating or joining the coalition will be modified accordingly.

1 INTRODUCTION

The fast development of technologies forces companies to be innovative in order to stay competitive. They must ensure a high degree of efficiency with respect to delivery deadlines. The introduction of cloud computing seems to be the ideal solution as it avoids upfront infrastructure cost, and helps organizations to focus on their core business activities, instead of their system infrastructure.

In this context, companies are willing to outsource their BP (Business Processes) to the cloud. However, as the cloud introduces new security risks related to its shared environment (Network and Agency, 2009; Abdmeziem, 2016; Abdmeziem and Charoy, 2018), they are still reluctant to do so, especially because of the know-how included in BP models which is considered as particularly sensitive.

In our previous work (Goettelmann et al., 2015), we suggested a method for splitting a BP model into BP fragments, so that when these BP fragments are externalized in a multi-cloud setting, a cloud provider cannot understand a crucial part of company know-how.

Although the simple splitting of a process makes such a conspiracy more challenging, this work primarily offers active support against a single malicious cloud at a time and does not explicitly address the risk of a conspiracy involving multiple malicious cloud providers, which could pool their local knowledge of the process model to discover larger critical know-how.

In (Ahmed Nacer et al., 2016) we went one step further in reducing this risk of conspiracy by introducing fake fragments at strategic points in the BP. However, even if this solution reduces considerably the possibility of conspiracy, a risk still remains. In this context, we propose in this paper a solution for detecting clouds initiating or participating to coalitions through the transmission of unexpected messages. Reputation of these clouds will be reduced according to if they initiate or participate to the coalition.

The rest of the paper is organized as follows: Section 2 introduces our motivations and the context of our work. Next section presents our cloud coalition detection approach. Section 4 applies our approach to our motivating example. Section 5 discusses the state of the art and finally section 6 concludes and introduces some future works.

2 MOTIVATIONS AND CONTEXT OF WORK

This section starts with the description of a motivating example, then it introduces the notion of obfuscation based on sensitive fragments separation, and presents a threat scenario on which our approach is based. Finally, it presents two attacker models which illustrate

Ahmed Nacer, A., Abdmeziem, M. and Godart, C.
Clouds Coalition Detection for Business Processes Outsourcing.
DOI: 10.5220/0011963400003464
In *Proceedings of the 18th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2023)*, pages 530-537
ISBN: 978-989-758-647-7; ISSN: 2184-4895

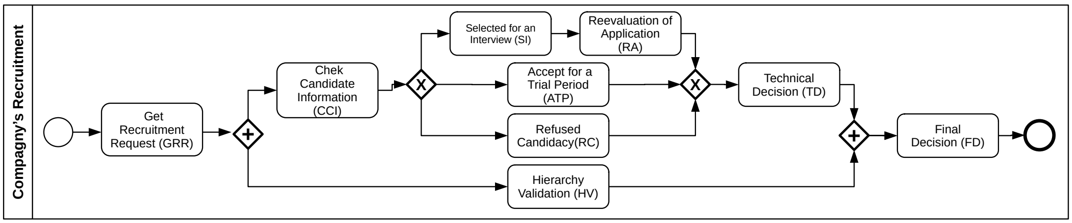
Compagny's Recruitment
Get Recruitment Request (GRR)
Chek Candidate Information (CCI)
Selected for an Interview (SI)
Reevaluation of Application (RA)
Accept for a Trial Period (ATP)
Refused Candidacy(RC)
Hierarchy Validation (HV)
Technical Decision (TD)
Final Decision (FD)
X
X
+
+

Figure 1: Recruitment process.

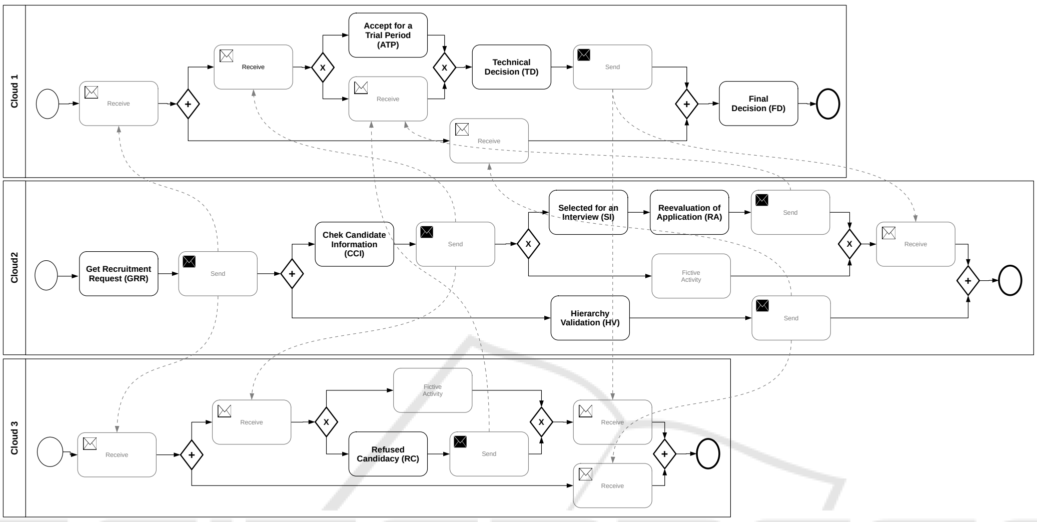

Figure 2: The recruitment process as a collaboration of BP fragments.

how the power of an attacker can be exploited, to extract important BP-know how.

2.1 Motivating Example

Figure 1 depicts[1] a selection process of a company that wants to recruit staff. The objective of this process is to accept or reject a candidacy. Depending on candidate record (professional experience, technical knowledge, police record, ...) the recruitment process is treated in different ways. The candidate can be selected for an interview, accepted for a trial period or directly refused for the job. At any point in the process, the hierarchy can directly intervene. The final decision is taken on the basis of the decision notification combined with hierarchy decision.

The company is prepared to externalize the execution of its business processes to the cloud, but it wants to maintain its method for accepting or rejecting a candidate. In order to achieve this goal, the companie's IT services provider chooses to separate the BP model's logic into a collection of BP fragments.

Figure 2 describes such a collaboration of BP fragments obtained by the splitting algorithm described in (Goettelmann et al., 2015); each fragment being assigned to a different cloud, and the fragments being connected with *send* and *receive* messages.

2.2 BP Obfuscation by Splitting Its Logic Using Sensitive Tasks

The main concept of BP obfuscation is to split a BP model into BP fragments, with each of them being deployed on different clouds. As a result, each cloud provider has just a partial view of the BP model.

While splitting a BP model in several BP fragments is effective for obfuscating a BP model, different splittings are possible and the problem is to choose the best one regarding different QoS parameters. For supporting this objective, one principle is to separate the more sensitive information in different BP fragments.

While this principle remained at the level of best practice in previous work, we have proposed in (Goettelmann et al., 2015) a formalization of the notion of a sensitive activity, and an algorithm for automatically

[1] We use the BPMN (Business Process Modeling Notation (http://www.bpmn.org)) for modeling our BP models. In addition, these process models are supposed to be well structured (Polyvyanyy et al., 2012)

identifying the most sensitive tasks[2] (containing the more BP know-how) and assigning them to different clouds.

Our assumption is that the most sensitive information is located in some specific fragments where important *decisions* and *syntheses* are made.

More precisely, we locate *decisions* in the BP fragments preceding *(x) or-split* gateways triggering alternatives fragments. In addition, we consider that a decision is complemented in the fragment following the (x) or-join gateway closing the opening (x)or-split succeeding the decision fragment. In our motivating example (figure 1), *Check Candidate Information* (CCI) is a decision task and *Technical Decision* (TD) its complement. Both are *sensitive* tasks.

Respectively, we locate *syntheses* in the fragments succeeding *and-join* gateways synchronizing several flows executing in parallel. Also, syntheses are often prepared in the fragment preceding the opening *and-split* gateway corresponding to the closing and-join gateway: we consider that these fragments are also *sensitive* ones. In our motivating example, *Final Decision* (FD) is a synthesis task and *Get Recruitment Request* (GRR) its complement. Both are also *sensitive* tasks.

After identifying those sensitive fragments, the algorithm returns a set of constraints for splitting the centralized process in a business process collaboration deploying sensitive tasks in different clouds. The principle is to separate, as much as possible, a sensitive task and its complement in two different BP fragments assigned to two different clouds (other aspects regarding sensitive tasks are considered in this algorithm but are of no interest for this paper) .

Back to our motivating example, the centralized *resident selection* BP can be split into the BP fragment collaboration depicted in figure 2 which follows the above principles.

2.3 BP Fragments Collaboration

As introduced in the previous section, applying the principle of sensitive complementing activities separation leads to the definition of interacting BP fragments. However the number of BP fragments can depend on other parameters, including other security artifacts and QoS properties. Then these BP fragments are assigned to clouds for execution, but different assignments leading to different cloud configurations are also possible at this step.

In other words, several cloud configurations are possible for deploying a BP and there is a need for selecting the most secure configuration that reduces the risk of a malicious cloud provider initiates a coalition or accepts to be part of it.

Note that regarding the splitting of a BP model into BP fragments and the weaving of these BP fragments, we rely on previous work (Khalaf et al., 2007), including ours (Yildiz and Godart, 2007). This is briefly illustrated in figure 3 where a simple process is split down into three BP fragments connected with *send* and *receive* operations for sending and receiving orchestration messages.

2.4 Threat Scenario

Even if the decomposition of a BP process into BP fragments offers active defense against malicious cloud assaults, a risk still remains. Indeed, the above solution provides active support against one malicious cloud at a time, but does not explicitly address the risk of conspiracy of several malicious cloud providers.

The idea is that for discovering a sensitive information, an attacker has:

- to possess both a sensitive task and the corresponding complementing one: in the context of a BP fragments collaboration, we formalize this risk as the ability for a cloud executing a sensitive task to discover a path between such a sensitive task and its complementing one, following forward the *send* operation(s), and/or backward the *receive* operation(s) of this fragment.
- to collude with clouds on the paths between such sensitive complementing tasks.

2.5 Attacker Model

We consider in this paper the following attacker model:

Constructed Coalitions from Paths. In this case, we assume that a malicious cloud provider deploying a sensitive task initiates a coalition and tries to convince all clouds on the path linking it to the one deploying the complementing task. All paths linking clouds are built from send and receive synchronization messages.

For example, figure 3 represents a collaboration of three fragments assigned to three clouds (C_1, C_2, C_3). C_1 and C_3 own two complementing sensitive tasks (A and A'), but, as there is no direct exchange messages between C_1 and C_3, none of them is aware of the other. However, following the exchange messages (send and receive operations), they can build the path(s) linking them. If C_1 follows the *send*01 message and collude

[2] We use the traditional notion of a task in BP modeling, encompassing atomic and composite process activities.

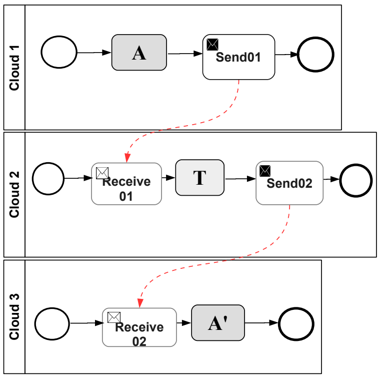

Figure 3: BP fragments with *send* and *receive* operations.

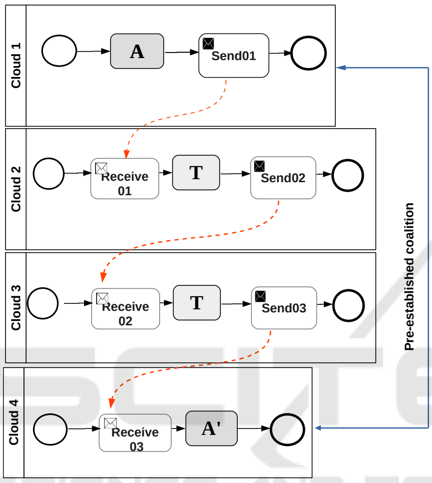

Figure 4: Pre-established coalition.

with C_2, it will be able to see all exchange messages of C_2 (send and receive). Therefore, from C_2 and following its *send*02, C_1 will be aware of C_3 while using the path $C_1 - C_2 - C_3$ (built from the exchange messages). Respectively, using its *receive*02 in a first step, and colluding with C_2, C_3 will be able to reconstruct the path linking it to C_1 (C_3, C_2, C_1).

Pre-Established Coalitions. In this case, a coalition may have already been initiated in a previous execution, thus allowing a direct exchange between two clouds (without convincing all clouds in the path separating them to be part of the coalition), even if no synchronization messages connect them directly or indirectly. As seen in figure 4, Cloud 1 and 4 deploying sensitive tasks are separated by other clouds participating to the collaboration. However, a direct link is connecting them resulting from a previous established coalition.

3 CLOUDS COALITION DETECTION

We present in this section an approach for detecting malicious cloud providers that initiate or participate to a coalition. To do that, we rely on decoy processes having the same structure and number of data as a real process, but with a decision strategy making and values of data that are completely different from the real ones.

Before diving into more details about the approach, we need to consider the following questions

- when to consider deploying decoy processes ??
- how to construct a decoy process?
- how to deploy them in a multi-cloud context?
- how to detect possible coalitions?
- how to reevaluate reputation level of malicious clouds?

3.1 When to Consider Deploying Decoy Processes?

More precisely, the question is: do we deploy this decoy process with the initial BP process ? Or do we consider it before deploying the real process?

Our choice is to consider the decoy process before deploying the real BP process. Indeed, our objective is to detect clouds that initiate or participate to the coalitions, to decrease their level of reputation [3]. More precisely, we consider that a cloud initiating a coalition is less reputable than a cloud that participate to it, which itself is less reputable than a cloud that neither initiates nor participates to the coalition.

After detecting those clouds and reducing their level of reputation, the initial BP process will be deployed following the constraints introduced in (Ahmed Nacer et al., 2016).

3.2 How to Construct a Decoy Process?

More precisely the question is: do we keep the same specifications as the initial process in term of number of activities, data and gateways ?

Our choice is to respect the initial specifications while using fake activities (black box activities) with the same number of input/output data randomly generated for each task on the on hand. On the other hand, we recommend to keep the same number and type of gateways (see figure 5). This choice is motivated by the following reasons:

[3] Reputations are valued between $[0,1]$: 0 for a cloud not being trusted, 1 for the most trusted one.

- We consider that modifying the number of activities and data can raise suspicions about the possibility of use of a decoy process. Therefore, it is important to keep the same number with a content modification (using black box activity with random generated data).
- Modifying the number and type of gateways clearly change the general structure of the process.

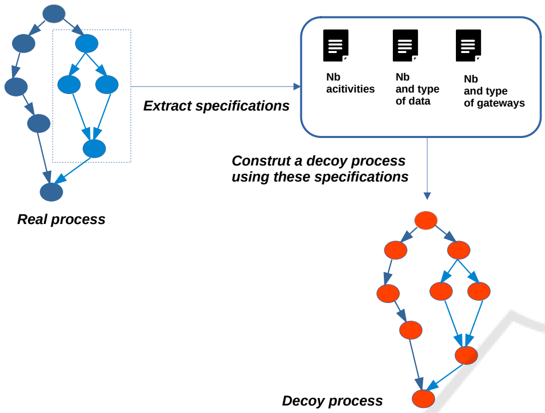

Figure 5: Decoy process construction.

3.3 How to Deploy Them in a Multi-Cloud Context?

More precisely, the question is to to which clouds the fragments of the decoy process are assigned?

We argue that clouds deploying sensitive fragments are the most likely to initiate, or participate to a coalition. As our objective is to detect those clouds, we recommend to :

- Separate fragments that represents sensitive activities according to the structure of the process as follows (Ahmed Nacer et al., 2016)
  - assign the fragment preceding an opening X(OR)-split gateway and the fragment following the corresponding closing X(OR)-join gateway to two different clouds.
  - assign fragment preceding an opening AND-split gateway and the fragment following the corresponding closing AND-join gateway to two different clouds.
  - assign fragments arising from alternative flows following an OR-split gateway to different clouds.
    he alternative flows following an or-split
- In the ideal case (no other functional and nonfunctional constraints), to assign the remaining fragments to different clouds, in order to detect as many malicious clouds as possible.

3.4 How to Detect Possible Coalitions?

After deploying the decoy process, we analyze the exchange messages between the different clouds of the collaboration. We consider that there is a potential attempt to initiate or participate to a coalition if:

- There are messages exchanges between two clouds deploying sensitive fragments without having a direct synchronization messages (*send* or *receive*) linking them in a normal scenario of the process execution.
- There a is a synchronization message *send* from a cloud C_i to a cloud C_j which should have been destined to another cloud C_k in a normal process execution.
- There a is a synchronization message *receive* to a cloud C_i which should have been received by a cloud C_j.

We consider that if one of these scenarios occurs, there is a possibility of a coalition attempt. Therefore, the clouds considered as initiating and/or participating to the coalition are considered as malicious ones. Before deploying the initial BP process, we need to reevaluate their level of reputation before applying our separations constraints introduced in section 2.2.

3.5 How to Reevaluate Reputation Level of Clouds?

As introduced below, we consider that a cloud initiating a coalition is more malicious than a cloud participating to it. Therefore, to reevaluate the reputations' levels, we use the following criteria

- **Attack Type:** if it is a cloud initiating or participating to a coalition.
- **The Number of Exchanged Messages in Each Attack:** we take into consideration the number of exchange messages in an attack scenario. We consider that the more a malicious cloud exchange unexpected messages, the more it is considered as a malicious one, reducing therefore even more its cloud reputation.

Table 1 depicts our cloud reputation reevaluation approach for both criteria.

As seen, each cloud initiating or participating to a coalition will be penalized on the fact of having initiated/participated to the coalition. Moreover, the number of unexpected exchanged messages will be taken into consideration. Indeed, we consider that the more it exchange messages (*send* messages), the less it is reputable, reducing therefore even more its level of reputation.

Table 1: Clouds reputation reevaluation.

Type of attack	Reevaluation based on type	reevaluation based on nb messages
Initiation	$Rep = Rep * 1/2$	$Rep = Rep - (nb * 0.2)$
Participation	$Rep = Rep * 3/4$	$Rep = Rep - (nb * 0.1)$

It must be noted that we consider that if a reputation level of a cloud falls below a minimum threshold, it will be excluded from the list of candidate clouds for the deployment of the initial BP process.

Table 2: Cloud's reputation.

Cloud	Reputation
$cloud0$	0,6
$cloud1$	0,5
$cloud2$	0,4
$cloud3$	0,6
$cloud4$	0,5
$cloud5$	0,7
$cloud6$	0,1
$cloud7$	0,4
$cloud8$	0,2

4 APPLICATION TO THE MOTIVATING EXAMPLE

To illustrate the utility and the applicability of our approach, we apply it to our motivating example (section 2.1) to detect the different clouds participating to the coalition.

The first step is to extract initial BP specifications in terms of

- number of activities: we keep the same number with black box activities.
- number of data : we keep the same number of data while generating them randomly for the decoy process.
- number and type of gateways: we keep the exact number and type of gateway (one opening and closing X(OR) gateway and one opening and closing (AND) gateway).

Using these specifications, the resulted decoy process is depicted in figure 6.

The next step is to deploy the decoy process using the set of clouds of table 2 according to the rules introduced in section 3.3 as follows

- $separate(A, A')$
- $separate(B, B')$
- $separate(C, C')$
- $separate(C, C'')$
- $separate(C', C'')$
- $separate(D, E)$

The resulting deployment that follows the rules introduced in section 3.3 is depicted in figure 7.

As introduced in section 3.4, many scenarios can refer to an attempt to create or participate to a coalition. In this case, we can see that there is unexpected exchange of messages between cloud 1 and cloud 5. These two clouds are deploying sensitive fragments without having a direct message exchange in a normal scenario. However, as seen in figure 7, cloud 5 sends two messages to cloud 1 which responds with a send message. Therefore, we can conclude that cloud 5 initiated a coalition with cloud 1 which accepted to be part of it.

Moreover, we notice that cloud 4 is sending an unexpected message to cloud 2 thus meaning an attempt to initiate a coalition

By applying our approach of clouds reputation reevaluation, the new reputations of cloud 1, cloud 5 and cloud 4 is computed as follows:

- $Rep(cloud1) = (Rep(cloud1) * 3/4) - 0.05 = 0.4$
- $Rep(cloud5) = (Rep(cloud5) * 1/2) - (2 * 0.1) = 0.2$
- $Rep(cloud4) = (Rep(cloud4) * 1/2) - (1 * 0.1) = 0.15$

As we consider that a reputation level of 0.3 is a threshold minimum for accepting a cloud to be part of clouds candidate for BP deployment, the new list of clouds candidate with new reputation levels is depicted in table 3.

Table 3: Clouds candidate after applying our approach.

Cloud	Reputation
$cloud1$	0,4
$cloud2$	0,4
$cloud3$	0,6
$cloud6$	0,8
$cloud7$	0,4
$cloud8$	0,4

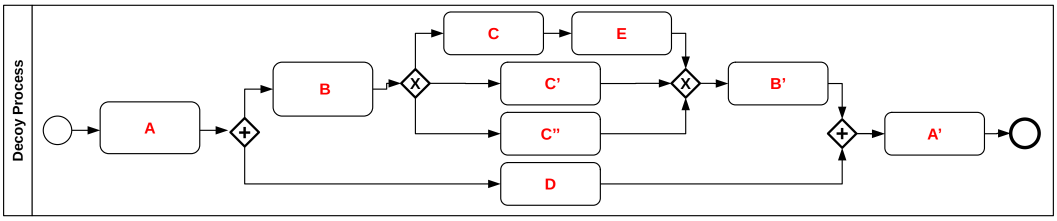

Figure 6: The resulted decoy process.

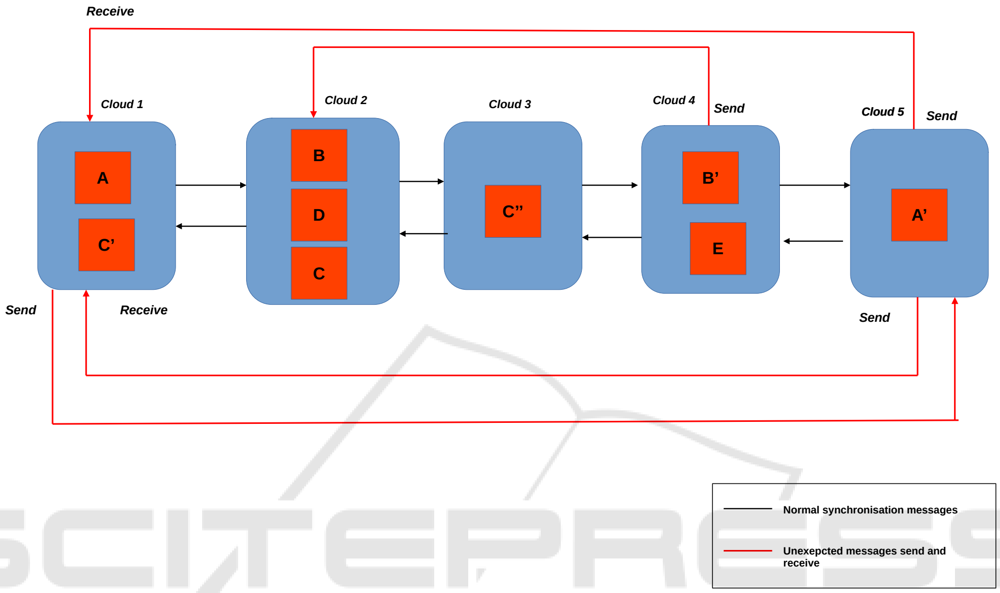

Figure 7: Decoy process deployment.

5 RELATED WORKS

This work is related to Business Process Outsourcing (BPO) which is gaining more and more importance and where many approaches have already been proposed (Ge et al., 2021)(Suresh and Ravichandran, 2022). In this context, Arnold and al. (Arnold, 2000) proposed a model for BP outsourcing based on four elements : (1) the subject making the decision of outsourcing, (2) the objects to be outsourced, (3) the partners which are all possible suppliers for the considered objects, (4) and the design which represents the strategy of outsourcing. In the same vein, Yang and al. (Yang et al., 2007) proposed a methodology for decision outsourcing based on the perspective, the environment and the risk of outsourcing. However, none of them take into account the cloud computing context.

More related to the cloud context, Rekik and al. (Rekik et al., 2015) proposed a model for selecting the tasks to be outsourced to the cloud based on their level of performance degradation and on the cost of deployment. Bentousni and al. (Bentounsi et al., 2012) proposes an anonymization-based approach for preserving the client business activity privacy while sharing process fragments between organizations in the cloud. Zemni and al. (Zemni et al., 2011) propose a framework for Privacy-Preserving Business Process Fragmentation by avoiding sensitive association. However, none of these works consider the risk of knowhow disclosure by a collusion of several clouds.

The idea of a decoy process is directly inspired from useless code for obfuscating a software (Beaucamps and Filiol, 2007), but we are, to our knowledge, the first using it for supporting the privacy of a BP collaboration in a multi cloud context, while taking into consideration the possibility of coalition.

6 CONCLUSION

Companies nowadays are ready to use cloud computing technology to enjoy of its many benefices. However, because of the new security risks introduced, they must be sure that their decision strategy included in their BP is preserved.

Even if the splitting of the process reduces the risk of know how exposure, malicious cloud providers can mine important information by combining their local knowledge.

As a contribution to this topic, we proposed in this paper an approach for detecting clouds initiating or participating to a coalition by using a decoy process. Each cloud participating or initiating a coalition will have its reputation level reduced. The final set of clouds will be used to deploy our initial BP process.

Important cost may be incurred by introducing decoy processes. But on the one hand, this may be the price to pay for know how protection, and on the other this can be minimized by our suggestions, which managed strategically BP deployment.

Regarding future works, we aim to apply our approach to a real case where we aim to use real Cloud resources while analyzing log files, to detect the unexpected messages.

REFERENCES

Abdmeziem, M. R. (2016). *Data confidentiality in the Internet of Things*. PhD thesis.

Abdmeziem, M. R. and Charoy, F. (2018). Fault-tolerant and scalable key management protocol for iot-based collaborative groups. In *Security and Privacy in Communication Networks: SecureComm 2017 International Workshops, ATCS and SePrIoT, Niagara Falls, ON, Canada, October 22–25, 2017, Proceedings 13*, pages 320–338. Springer.

Ahmed Nacer, A., Goettelmann, E., Youcef, S., Tari, A., and Godart, C. (2016). Obfuscating a business process by splitting its logic with fake fragments for securing a multi-coud deployment. In *the IEEE international Congress on Services (SERVICES)*, page 8.

Arnold, U. (2000). New dimensions of outsourcing: a combination of transaction cost economics and the core competencies concept. *European Journal of Purchasing & Supply Management*, 6(1):23–29.

Beaucamps, P. and Filiol, É. (2007). On the possibility of practically obfuscating programs towards a unified perspective of code protection. *Journal in Computer Virology*, 3(1):3–21.

Bentounsi, M., Benbernou, S., Deme, C. S., and Atallah, M. J. (2012). Anonyfrag: an anonymization-based approach for privacy-preserving bpaas. In *1st International Workshop on Cloud Intelligence (colocated with VLDB 2012), Cloud-I '12, Istanbul, Turkey, August 31, 2012*, page 9.

Ge, L., Wang, X., and Yang, Z. (2021). The strategic choice of contract types in business process outsourcing. *Business Process Management Journal*.

Goettelmann, E., Ahmed-Nacer, A., Youcef, S., and Godart, C. (2015). Paving the way towards semi-automatic design-time business process model obfuscation. In *Web Services (ICWS), 2015 IEEE International Conference on*, pages 559–566.

Khalaf, R., Kopp, O., and Leymann, F. (2007). Maintaining data dependencies across BPEL process fragments. In *Service-Oriented Computing - ICSOC 2007, Fifth International Conference, Vienna, Austria, September 17-20, 2007, Proceedings*, pages 207–219.

Network, E. and Agency, I. S. (2009). *Cloud Computing: Benefits, risks and recommendations for information security*. ENISA.

Polyvyanyy, A., García-Bañuelos, L., and Dumas, M. (2012). Structuring acyclic process models. *Information Systems*, 37(6):518 – 538. BPM 2010.

Rekik, M., Boukadi, K., and Ben-Abdallah, H. (2015). Business process outsourcing to the cloud: What activity to outsource? In *Computer Systems and Applications (AICCSA), 2015 IEEE/ACS 12th International Conference of*, pages 1–7. IEEE.

Suresh, S. and Ravichandran, T. (2022). Value gains in business process outsourcing: The vendor perspective. *Information Systems Frontiers*, 24(2):677–690.

Yang, D.-H., Kim, S., Nam, C., and Min, J.-W. (2007). Developing a decision model for business process outsourcing. *Computers & Operations Research*, 34(12):3769–3778.

Yildiz, U. and Godart, C. (2007). Towards decentralized service orchestrations. In *Proceedings of the 2007 ACM Symposium on Applied Computing (SAC), Seoul, Korea, March 11-15, 2007*.

Zemni, M. A., Benbernou, S., and Sahri, S. (2011). Privacypreserving business process fragmentation for reusability.