Security Workflow Model for Database Storage

Hui Yuan, Lei Zheng and Xiangli Peng

Central China Branch of State Grid Corporation of China.

882148@qq.com

Keywords: recovery property, compensation, workflow.

Abstract. The security workflow is proposed according to the transactional workflow discussed in this paper. When it comes to the security workflow, it is necessary to consider the compensation of the attack. In this paper, the recovery properties is introduced in security database workflow: non-vital property, compensable property, retriable property, critical property and replaceable property. With these properties discussed, the paper deeply analyses the abnormal workflow and advises the division of the abnormality. Further, some recovery modes for transaction workflow and security workflow are proposed. Finally, the paper presents the recovery algorithm for database storage. In that algorithm, the paper considers the difference between the master and the slave. The process method of the master database and the slave database is also discussed.

1. Introduction

Workflow is composed of some associated tasks. It is very popular to show the chart of some work. The researchers propose the transactional workflow [2] with the feature of transaction and workflow. The transactional workflow has the property of relaxed atomicity [3]. That is, the transactional workflow or normally executes or compensably withdraws all the committed transaction. Minh Chau Nguyen and Hee Sun Won propose a web-based analytic workflow system to allow end-users to use easily their desired analytic functions to solve specific problems in different domains [4]. Sardar Hussain, Richard O. Sinnott and Ron Poet provide a security-oriented workflow framework that supports secure workflow enactment [5]. Zhenyu Wen, Jacek Cał a, Paul Watson, and Alexander Romanovsky present a novel algorithm to deploy workflow applications on federated clouds[6]. In [7], a security model was proposed to meet the new, multi-criteria requirements.

Security must be an important factor when taking scientific applications into the cloud [8]. Referring to the workflow and model discussed above, the security workflow is proposed. In order to prevent the hacker from attacking, the authors distinguish the security workflow from the transactional workflow. Especially the authors modify the recovery properties and the control structure.

2. Workflow Recovery Properties

In order to discuss the workflow recovery properties, several definitions is shown in the following.

Definition 1(Process) A Process can be defined as a binary group $W=(G,O)$. Note that G is a directed graph, which can be denoted as binary group (A,L). Note that $A=\{a_1,a_2,...,a_{n1}\}$ denotes a set of nodes. $L=\{l_1,l_2,...,l_{n2}\}$ denotes a set of connects. $l_i=<a_j,a_k>$ denotes a link arc from a_j to a_k, $a_j,a_k\in A$, $O=\{o_1,o_2,...,o_{n3}\}$. Note that O denotes the set of the abstract data objects visited by the nodes.

Definition 2(Subprocess) A subprocess can be defined as a $P=(G^{'},O^{'})$. Note that $G^{'}=(A^{'},L^{'})$ denotes a directed sub-graph, $A^{'}\subseteq A$, $L^{'}\subseteq L$. Note that $O^{'}$ denotes the set of the abstract data objects visited by $A^{'}$, $O^{'}\subseteq O$. The subprocess is also a workflow.

Definition 3(Activity) An activity can be defined if only $G^{'}$ is a Trivial graph, the subprocess P cannot be subdivided.

Definition 4(Transfer) $l=<a_j,a_k>$ is a link are from a_j to a_k ,$a_j,a_k \in A$. l is called a transfer. a_j is the precedence of a_k. a_k is the subcequence of a_j.

Definition 5(Transaction Recovery Properties) Recovery Properties is defined as the properties of the recovery characteristic of activities. Recovery properties have five kinds: non-vital transaction property, compensable transaction property, retriable transaction property, critical transaction property and replaceable transaction property. A Denotes the activity of a process definition. A is denoted as an activity a when the process is in running process. $INA=\{a_1,a_2,...,a_n\}$ Denotes the set of activity instances. The instances running sequence denoted as $<a_1,a_2,...,a_m>(m \le n)$. Note that a^* denotes the repeat of activities a and that a^i denotes the ith execution of the activity a.

Definition 6(Non-vital Transaction Property) In transactional workflow, a is non-vital only if the execution result between the execution sequence $<a_1,a_2,...,a_{i-1}a,a_i,...,a_n>$ $(i \le n)$ and $<a_1,a_2,...,a_{i-1}a_i,...,a_n>$ $(i \le n)$ is same, $a \in INA$.

Definition 7(Compensable Transaction Property) If there exists an activity a^{-1}, which makes the sequence $\delta=<a,a^{-1}>$ non-vital, $a \in INA$. Then a^{-1} is the compensation of a and a is compensable.

Definition 8(Retriable Transaction Property)$\exists m \in N$, $\forall k$, $1 \le k < m$, if a^k is abandoned, but a^m can be sure to be committed, $a \in INA$. Then a is retriable.

Definition 9(Replaceable Transaction Property) a is replaceable only if there exists an activity $a^{'} \ne a$ and the semantics function of a is same to that of $a^{'}$. $a^{'}$ is the replaced activity of activity a, $a \in INA$.

Definition 10(Critical Transaction Property) If a is not compensable, a is not non-vital and a is not replaceable, then a is critical, $a \in INA$.

Definition 11(Reduced Transaction Execution) If $\exists(a,a^{-1}) \in INA$ and $a << a^{-1}$, then a^{-1} and a can be removed, denoted as $reduced(INA)$. If a is a transaction activity , then the execution is reduced transaction execution.

Definition 12(Security Recovery Properties) Security Recovery Properties is defined as the properties of the security recovery characteristic of activities. Security recovery properties have five kinds: security property, compensable property, retriable property, critical property and replaceable property. A denotes the activity of a process definition. A is denoted as an activity a when the process is in running process. $INA=\{a_1,a_2,...,a_n\}$ denotes the set of activity instances. The instances running sequence denoted as $<a_1,a_2,...,a_m>(m \le n)$. Note that a^* denotes the repeat of activities a and that a^i denotes the ith execution of the activity a.

Definition 13 (Non-vital Security Property) If $<a_1,a_2,...,a_{i-1},a_i,...,a_n>(i \le n)$ is safe, then activity a is non-vital only if the execution sequence $<a_1,a_2,...,a_{i-1}a,a_i,...,a_n>$ $(i \le n)$ is safe, $a \in INA$.

Definition 14(Compensable Security Property) If there exists an activity a^{-1}, which makes the sequence $\delta=<a,a^{-1}>$ safe, $a \in INA$. Then a^{-1} is the compensation of a and a is compensable.

In the security workflow, a^{-1} is the workflow which can prevent the hacker from attacking the system. Once the attack a happened, a^{-1} should be run in the safe time limitation. The execution result of a^{-1} should control the effect so that the final result can be accepted.

Definition 15 (Retriable Security Property) In the security workflow, a^{-1} is the compensable workflow which can prevent the hacker from attacking the system. a^{-i} denotes the ith time execution of a^{-1}. $\exists m \in N, \forall k, 1 \le k < m$, If a^{-k} is abandoned, but a^{-m} can be sure to be committed to prevent the hacker from attacking, $a \in INA$. Then a^{-1} is retriable.

Definition 16(Replaceable Security Property) In security workflow, a^{-1} is replaceable only if there exists an activity $a^{'-1} \neq a^{-1}$ and the anti-attack semantics function of a^{-1} is same to that of $a^{'-1}$. $a^{'-1}$ is the replaced activity of activity a^{-1}, $a \in INA$.

Definition 17(Critical Security Property) If activity a is not compensable and not safe, then a is critical, $a \in INA$.

Definition 18(Reduced Security Execution) If $\exists(a, a^{-1}) \in INA$ and $a << a^{-1}$, then a^{-1} and a can be removed, denoted as $reduced(INA)$. If a is a dangerous activity , then the execution is reduced security execution.

3. Recovery Strategy

3.1 Analysis of Abnormal Workflow

According to the abnormal workflow discussed above, the division of the abnormal workflow is shown in figure 1. The abnormity maybe happen because of the transaction abnormity, maybe happen because of the intrusion.

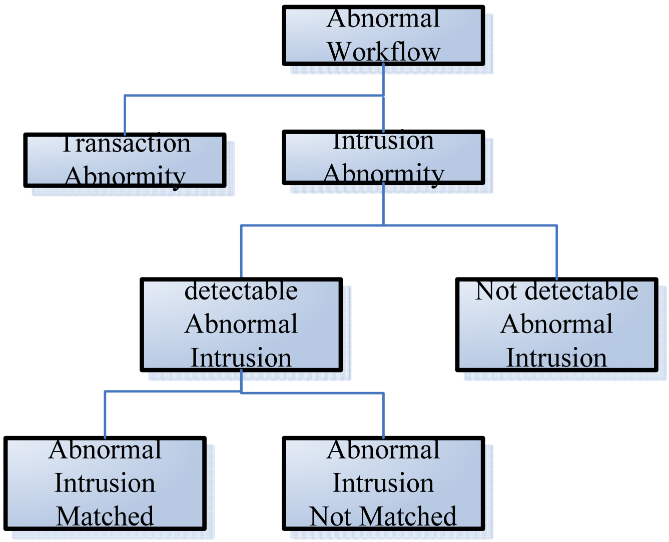
Abnormal Workflow
Transaction Abnormity
Intrusion Abnormity
detectable Abnormal Intrusion
Not detectable Abnormal Intrusion
Abnormal Intrusion Matched
Abnormal Intrusion Not Matched

Fig. 1 Abnormal Workflow

As a transaction workflow, the compensation is designed in the design stage. Whenever the transaction is aborted, the compensation workflow is scheduled. So if the abnormal workflow is a transaction abnormity, it must have its compensation to commit the transaction. It is not necessary to consider the abnormity whether the abnormity transaction can be detected or not.

However, as an intrusion abnormity, the intrusion may not be detected because of the new intrusion that is never found before. So the intrusion should be divided into two types: detectable abnormal intrusion and not detectable abnormal intrusion. Moreover, the detectable abnormal intrusion is forwardly divided into two types: abnormal intrusion that can be matched and the abnormal intrusion that can not be matched.

3.2 The History of the Process Execution

Definition 19(Process Execution History) Process execution history Refers to the path of a process instance from the beginning to the end. It is denoted as $H = (INA, <<)$. INA denotes a sub-flow of a workflow. $<<$ denotes a partial order.

Definition 20(Recovery Of Reduction Transaction) If INA is a reduced transaction execution, then

$$re\cos very(INA) = INA + reduced(INA)^{-1}.$$

$reduced(INA)^{-1}$ denotes the compensation of $reduced(INA)$.

Definition 21(Recovery Of Reduction Security) If INA is a reduced security execution, then $recovery(INA) = INA + reduced(INA)^{-1}$.

$reduced(INA)^{-1}$ denotes the compensation of $reduced(INA)$.

3.3 Transaction Recovery Mode

To design the transaction recovery for database, it is necessary to know the recovery mode. For transaction recovery, there are several modes:

(1)**No compensation mode:** Only for the activity with non-vital transaction property.

(2)**Retriable execution without compensation mode:** Only for the activity with retriable transaction property.

(3)**Replaceable execution mode:** Only for the activity with replaceable transaction property.

(4)**Reduction Recovery mode:** Only for compensable activity and its compensation, which can be both reduced when the transaction recovery workflow is executed.

(5)Compensation recovery mode: Only for compensable activity, executed backward or executed forward. This mode should be adopted when the above modes can not be chosen.

3.4 Security Recovery Mode

To design the security recovery for database, it is necessary to know the recovery mode. The modes are similar to that of transaction recovery. However, considering that the attack and its anti should not coexist, the reduction recovery mode for security workflow should not be proposed. The attack must be left out. For security recovery, there are several modes, too:

(1)**No compensation mode:** Only for the activity with non-vital security property.

(2)**Retriable execution mode:** Only for the activity with retriable security property.

(3)**Replaceable execution mode:** Only for the activity with replaceable security property.

(4)Compensation recovery mode: Only for compensable activity, executed backward or executed forward. This mode should be adopted when the above modes can not be chosen.

The following modes is adopted for the abnormal intrusion not matched and the abnormal intrusion not detected.

(5)**No Matched mode:** Only for the abnormal intrusion not matched. If the intrusion workflow is detected but not matched, the only way is stop the workflow when the database is slave.

(6)**No detected mode:** Only for the abnormal intrusion not detected. If the intrusion workflow is not detected, intelligent method, such as Genetic Algorithm, Ant Colony Optimization, Artificial Immune Algorithm and other Evolutionary Algorithm, should be adopted for such abnormal intrusion.

4. The Recovery of Database Storage

In the Internet, the users modify the data of the database through the workflow sent from the users software, maybe web or client software. Usually, the master database has its slave database server, which can recover the master server data when the master server is collapsed. As is shown in figure 2.

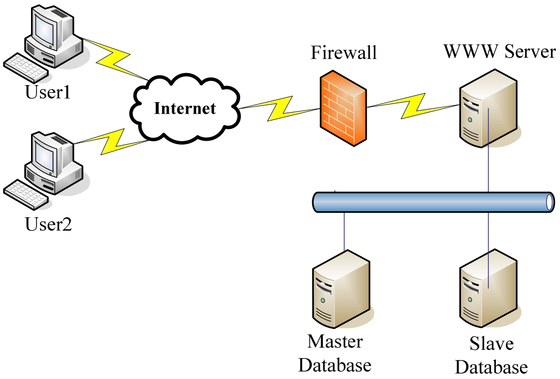

Fig. 2 Database in the Internet

According to the recovery strategy discussed in Section III, once the users submit the new data or modify the database, the following steps should be adopted:

Step 1 The system differentiates the abnormal workflow. If it is a transaction abnormal workflow, the transaction recovery mode should be enabled. As for transaction workflow, there are pre emergency measures to solve it.

Step 1.1 If the transaction activity is not activity with compensable transaction property, it is necessary to execute as the following.

Step 1.1.1 If it is activity with non-vital transaction property, it is not necessary to focus it and it is to leave it out.

The slave's execution is same to that of the master.

Step 1.1.2 If it is activity with retriable transaction property, it is not necessary to compensate it, only redo the activity until the workflow is commit successfully.

The slave's execution is same to that of the master.

Step 1.1.3 If it is activity with replaceable transaction property, it is possible to adopt its replaceable activity at the consistent point.

The slave can execute the replaceable activity.

Step 1.2 If the workflow is activity with compensable transaction property, it is necessary to compensate the activity.

Step 1.2.1 If the transaction activity and its compensation are together in such workflow, if the activity has been executed without executing its compensation, the master database should finish its compensation. If the activity has not been executed, the master should leave the activity and its compensation out.

However, For the slave database, it is necessary to leave them out during the period of backup.

Step 1.2.2 If the transaction activity's compensation isn't in such workflow and it is reverse activity, it is necessary to reverse the execution to consensus. Having finished reversing, the remain work is to redo the activity.

For the slave database, it is not necessary to execute reverse workflow.

Step 1.2.3 If the transaction activity's compensation isn't in such workflow and it is forward activity, it is necessary to forward the execution from the type of activity that was first executed. Having finished forward execution, the remain work is to execute the activities before the fail point.

The slave's execution is same to that of the master.

Step 2 For the detected attacks, the corresponding matched strategy is searched.

Step 2.1 If the match is successful, the main server is directly compensated, and the compensation process is divided into the following situations:

Step 2.1.1 If it is activity with non-vital security property, it is not necessary to focus it. It is necessary to decide that whether it is a transaction workflow. It is necessary to stop such execution at any time without compensating.

The slave's execution bypasses such operation directly without the database modified.

Step 2.1.2 If the workflow is compensable attacks, it is necessary to compensate the attack.

The slave's execution bypasses such operation directly without the database modified.

Step 2.1.3 If it is compensation with retriable security property, it is necessary to repeat the compensation until it is committed successfully.

The slave's execution bypasses such operation directly without the database modified.

Step 2.1.4 If it is activity with replaceable transaction property, it is possible to adopt its replaceable activity at the consistent point.

The slave's execution bypasses such operation directly without the database modified.

Step 2.1.5 If the activity's compensation isn't in such workflow and it is reverse activity, it is necessary to reverse the execution to consensus. Having finished reversing, the remain work is to redo the activity.

The slave's execution bypasses such operation directly without the database modified.

Step 2.1.6 If the activity's compensation isn't in such workflow and it is forward activity, it is necessary to forward the execution from the type of activity that was first executed. Having finished forward execution, the remain work is to execute the activities before the fail point.

The slave's execution bypasses such operation directly without the database modified.

Step 2.2 If the match is not successful, the master database cannot perform the method to block the attack. At this point, the operation should be reversed and returned to the point of agreement.

When the slave database backup the master, you should recognize the attack directly and bypass the attack.

Step 3 for suspected attacks, the system cannot detect whether it is an attack workflow. In order to test whether the workflow is an attack workflow, some intelligent computing methods can be used to induce workflow into the simulation environment, such as honeypot environment.

Step 3.1 The system introduces this workflow into the simulation environment.

Step 3.2 The simulation environment learns the workflow and captures relevant features.

Step 3.3 The system analyses the relevant features and formulates relevant strategies.

Step 3.4 The master database interrupts the workflow, returns to the consistent point and executes the subsequent workflow. It waits for the simulation to determine whether the workflow should be executed according to the results of the decision.

Step 3.5 the slave's execution bypasses such operation directly without the database modified. The subsequent workflow is executed by the slave database. It also waits for the simulation to determine whether the workflow should be executed according to the results of the decision.

5. Summary

The paper proposes the security workflow according to the transactional workflow. In this paper, the recovery properties is introduced in security database workflow: non-vital property, compensable property, retriable property, critical property and replaceable property. With these properties discussed, the paper deeply analyses the abnormal workflow and advises the division of the abnormality. And it also proposes a series of recovery modes for transaction workflow and security workflow. Finally, the paper presents the recovery algorithm for database storage.

References

[1]. C. Wang, K. Ren, and J. Wang, "Secure and practical outsourcing of linear programming in cloud computing," in Proceedings of the 2011 INFOCOM. IEEE, 2011, pp. 820–828.

[2]. Alonso G, Agrawal D, Abbadi A et al. Advanced transaction models in Workflow contexts. In: Proceedings of International Conference on Data Engineering New Orleans, 1996. 574-581.

[3]. Grefen P, Vonk J, Boertjes E, Apers P. Semantics and architecture of global transaction support in Workflow environments. In: Proceedings of the 4th IFCIS International Conference on Cooperative Information Systems Edinburgh Scotland, 1999. 348-359.

[4]. Minh Chau Nguyen, Hee Sun Won. A Case Study on Web-based Analytic Workflow in Big Data Platform. In: Proceedings of 2016 International Conference on Computational Science and Computational Intelligence. 2016. 421-430.

[5]. Sardar Hussain, Richard O. Sinnott and Ron Poet. A Security-oriented Workflow Framework for Collaborative Environments. In: Proceedings of 2016 IEEE TrustCom/BigDataSE/ISPA. 2016. 707-714.

[6]. Zhenyu Wen, Jacek Cała, Paul Watson, and Alexander Romanovsky. Cost Effective, Reliable and Secure Workflow Deployment over Federated Clouds. In: IEEE Transactions on Services Computing. VOL. 13, NO. 9, SEPTEMBER 2014. 1-13.

[7]. P. Watson, "A multi-level security model for partitioning workflows over federated clouds," Journal of Cloud Computing, vol. 1, no. 1, pp. 1–15, 2012. [Online]. Available: http: //dx.doi.org/10.1186/2192-113X-1-15.

[8]. I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in Grid Computing Environments Workshop, 2008. GCE'08. IEEE, 2008, pp. 1–10.