Advanced Database Update and Security Using Markov Decision Policy

Prof. Reshma Totare, Anjali Singh, Vishakha Swamy, Tejaswini Yadav.

Dept. of Info. Technology, AISSMS's IOIT, Maharashtra, India.

***Abstract*— In this work, an advanced database security system using Markov based policy is proposed. In large databases, it's a difficult task to keep the data up to date where huge data of dynamic nature occurs. In case of e-commerce sites, large data might come and the database might not get updated. This causes data staleness and also if the database undergoes updation, it might not be available for access during that time. Sometimes, updation is not possible without human intervention, hence Markov based policy is used which would perform fixed interval updates effectively and lead to cost savings. In addition to updation, this policy can be used for the security of the system; the proposed system has a middleware that will restore the database using the Markov updation policy in case of attack. The middleware will process the changes in the data and display the details of the attacker and also the changes done by the attacker. Thus, this database system can also be secured using the Markov policy using these three layers.**

Keywords: DATABASE SECURITY, DATABASE UPDATION, MARKOV POLICY.Keywords—component; formatting; style; styling; insert (key words).

I. Introduction

In recent times, people have largely started shopping online, banking online and many other processes are digitalized.E-commerce sites are growing largely day by day. These sites are exposed a large amount of public globally and hence it is very important to keep them secure. Large amount of data of different nature occurs on these sites and should be frequently updated. Also during updation there is a chance of the database being unavailable for access. The data cannot be kept constant nor can the security be compromised.

It has hence become of vital importance to provide such databases which get frequently updated and the same time is secure too. In this era it is experienced that the data occurs from different sources and is of dynamic nature. Therefore, it is needed to have a vigilant eye on eavesdroppers and attackers who can cause severe damage to the system and huge losses to the people involved in online transactions. It can become very difficult to track this attacker and the modification done by the attacker if the database is not frequently updated or kept a track of properly. Also it is very expensive to include human intervention each time such unethical activity is observed to revert the data and to obtain details of the attacker. Hence such a system is needed in these times which will help to update the data frequently and also keep it available for use.

II. Related Work

Wei Zong et al.[1], discussed the data timeliness and the data updation policy which will be used for the rollback and data updation in the project. This paper has developed Markov decision process model, solved by dynamic programming method, to obtain the optimal update policy that minimizes the sum of data staleness cost and update cost, it however does not guarantee a strong security during the updation process.

O¨ zgeCepheli et al.[2] introduces an encryption-aware physical layer security framework. This framework allocates resources according to the distinct requirements, enabling to tune the overall power consumption and the secrecy levels.

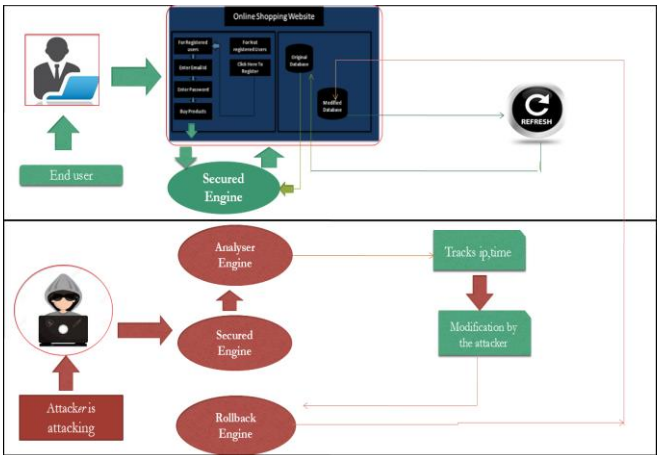
Qi Han, NaliniVenkatasubramanian [3] Addressed the Tradeoffs in Information Collection for Dynamic Environmentsto accommodate the diverse characteristicsof information sources and varying requirementsfrom information consumers.

Il-Min Kim, Byoung-HoonKim [4] designed a BER-Based Physical Layer Security with Finite Code length to achieve a sufficient level of physical layer security, while ensuring reliable reception quality for legal users..

III. Proposed System

The system objective is to develop a secured online shopping website with a strong and secured database which allows consumers to directly buy goods or services from a seller over the Internet using a web browser. The main motto of this application is to protect our banking system from various attacks. Another one is to develop the 3-layer system such as Analyzer Engine, Secured Engine and a Rollback Engine. In this analyzer engine will trace the database & changes made by an attacker will be visible on a front end. Id, password, IP address of attacker will be displayed on the Screen. Then Secured layer act as a middleware, where a request made by user or attacker will go this layer. The last one is Rollback Engine will revert all modification & give original data in case of an attack.

The following figure shows the architectural composition of the system:



Algorithm:

A. System Analysis

Let *dm* be the update decision for the database.

If *dm*=0, the database is not updated and if dm=1, then the database is updated at that respective point of time *tm*=0.

Let *sm* be the system state.

Then, at time *tm+1*;

If dm=0; *sm+1=sm+Im*

Where *Im* is the new data accumulated between time *tm* and *tm+1.*

If dm=1; *sm+1=Im*

B. System cost analysis

Let *cm* be the system cost. It is a function of system state and the update decision given by *cm(sm,dm).*

If dm=0; *cm(sm, dm) =cs(sm)*

where *cs(sm)* is the staleness cost that is incurred between time tm and tm+1.

If dm=1; *cm(sm, dm)=cu*

Where *cu* is the incurred update cost.

Based on the update decision and the corresponding cost at system time *tm*, we can formulate the objective function of the purchase data update problem as

$C = \min E\,(c1\,(s1,\, d1) + c2\,(s2,\, d2) + \cdots + cM(sM\,,dM))$

Based on the above equation, we have

$V1=E(c1(s1,d1)+ c2(s2,d2) +c3(s3,d3)+\ldots+.cM(sM,dM)\,)$

Where v1 represents the total expected system cost.

Same manner,

$V2=E(c2\,(s2,d2) +c3(s3,d3)+\ldots+.cM(sM,dM)\,)$

Therefore,

$V1=c1\,(s1,\, d1)+V2$

Generally

$VM=cM(sM,\, dM)$

The probability distribution *sm* transiting to *sm+1* is included so that all the values of the accumulated data are considered to get the equation:

$Vm=cm(sm,\, dm)+ \quad \sum_{sm+:}.(Vm+1)$

Where Vm represents the total expected system cost between system time intervals *tm* to *tm+1.*

We must find an optimal decision for update such that

m= (dm, dm+1,…, dM) and m=1,2,…M.

The expected optimal total cost from *tm* to *tM* is given by

$V^{*}m=\min\,(cm(sm,dm)+ \quad \sum_{sm+:}.(V^{*}m+1))$

To obtain optimal update policy from time *tm* to *tM* , *Vm and Vm**have to be analyzed.

Therefore from the above equation,

$R(sm,\, dm) =cm(sm,\, dm)+ \quad \sum_{sm+:}.(V^{*}m+1)$

Or simply,

$V^{*}m=\min R(sm,dm)$

Hence, the optimal system cost is dependent on *dm.*

C. Analysis of optimal update policy

As considered *dm* takes value either 0 or 1. Therefore, *d***m* increases from 0 to 1.

Therefore,that is to say there is a control limit *lm* for system state *sm* at each decision point which determines the optimal decision policy 0 or 1.

Therefore,

$d^{*}m=0$ *if* $sm<lm$

$d^{*}m=1$ *if* $sm>lm$

The objective hence, is to find the value for control limit *lm.* Backward induction is an efficient method for solving markov based decision problem. The control limit has to be found out at every decision point, which is the minimum quantity of data that satisfies the condition that *R(sm,* *dm=0)>=R(sm, dm=1)* for *m=1, 2, 3…M,* because at each step $V^{*}m+1$ for all possible system states has to be computed. But *d*m*increases from 0 to 1 for system states *sm*, therecords for *R(sm, dm=0) and R(sm, dm=1)* can be recorded and compared at each step till the minimal *sm* makes *R(sm,dm=0)* exceed or equal *R(sm,dm=1).*This minimal *sm* that makes *R(sm,dm=0)*exceed or equal *R(sm,dm=1)* is the control limit *lm.*

Notation Summary:

dm	Update decision
sm	System state
tm	Time instant
Im	New arrived data from *tm* to *tm+1*
cs(sm)	Staleness cost between *tm* to *tm+1* intervals *tm* to *tm+1*
R(sm,dm)	Optimal update policy

Expected Result:

The proposed system will give a protection against attack, since the unethical changes will not be retained in the database, and also the hacker can be caught since all the details of the attacker, details of the modification done will all be displayed on the front end. Also, it is expected that the system will improve the database performance by reducing the time required for database updation and also it will avoid data getting stale and make data more fresh and handy, even in case of data occurring in dynamic nature

Conclusion

Contained in this paper is the study of various methods for addressing data timeliness, data accuracy and data tradeoff costs. It contains a middleware as a solution to the security and integrity threats of the databases, also explained along is the architecture of the system that uses the markov based update policy for the updation and security of data. Algorithms that would be used for the system implementation are explained in detail

References:

.

[1] [1] Wei Zong, Feng Wu, and Zhengrui Jiang “A Markov-Based Update Policy for Constantly Changing Database Systems”, 2017.

[2] [2] O¨ zgeCepheli, Guido Dartmannz, Gu¨nes¸ Karabulut Kurt_, GerdAscheid, “An Encryption Aware Physical Layer Security System” 2017

[3] [3] Qi Han, Nalini Venkatasubramanian, “Addressing Timeliness/Accuracy/Cost Tradeoffs in Information Collection for Dynamic Environments”, IEEE 2003

[4] [4] Il-Min Kim, Byoung-Hoon Kim, “BER-Based Physical Layer Security with Finite

[5] Code length”, 2016