CHAPTER 3

SYSTEM ARCHITECTURE

In this thesis, a novel framework for effective temporal mining has been proposed and its architecture is shown in Figure 3.1. This system consists of seven major components namely ETMSQL Interface, User Interface, Temporal Database Manager, Security Decision Manager, Temporal Database, Temporal Reasoning Subsystem and Temporal Security Subsystem.

Figure 3.1 Overall System Architecture
The user query is given to the Temporal Database Manager through User Interface or ETMSQL Interface. The Temporal Database Manager has the overall control over the entire system. Any access to the temporal database can be carried out only through the Temporal Database Manager. The Temporal Database manager accepts data requests from the User Interface or ETMSQL Interface and handles temporal data management activities such as creation, insertion, deletion and retrieval of temporal tuples in coordination with the Rule Manager. It also executes the interval value functions and performs the temporal reasoning tasks namely Forecasting, Analysis of past, Planning and Learning and in addition it carries out the Mining tasks. The rules have been validated with the aid of human experts and then they are stored in the Knowledge Base along with the existing facts. Rules are stored in the form of IF..THEN conditions and are fired by the Rule Manager.

3.1 ETMSQL INTERFACE

The query language ETMSQL proposed, designed and implemented in this thesis is used as an interface to the temporal database manager so that a user can define and manipulate database objects interactively. This language has been designed by integrating and extending the query languages ETSQL and TQML. Application development with temporal database needs the use of ETMSQL interface which not only provides the features of the commercial query language SQL but also the features for temporal reasoning and mining. The query processor can understand only ETMSQL commands and hence any user request is presented to temporal database manager either using ETMSQL interface or through user interface which generates ETMSQL queries.
3.2 USER INTERFACE

The user interface accepts the input queries and forwards them to the Temporal Database Manager. In order to perform this, the user interface provides menus and forms that can be used effectively by users to interact with the temporal database manager. The user interface is also responsible for displaying the query results to the user.

3.3 TEMPORAL DATABASE MANAGER

The overall control of the system is with temporal database manager. Any access to the temporal database can be carried out only through the temporal database manager. The temporal database manager handles temporal data management activities such as data definition, data manipulation and data control with respect to temporal data in coordination with the rule manager. It also executes the interval value functions and performs the temporal reasoning tasks such as Forecasting, Analysis of Past, Planning, Learning and Mining. This module is also responsible for query processing, null handling, data dictionary maintenance and log maintenance in order to maintain integrity, security and consistency. In this system, the temporal information management tasks are initiated by the query language ETMSQL and are carried out by the temporal database manager.

3.4 TEMPORAL DATABASE

The physical storage called temporal database is designed to represent the temporal data effectively. It uses a temporal index structure namely time-split tree to organize the data in an optimal way. Moreover, it uses instant stamping of tuples for transaction time and interval stamping of tuples for valid time.
3.5 SECURITY DECISION MANAGER

The security Decision Manager has two sub-components namely access control manager and intrusion detection manager. The access control manager is responsible for interacting with rule subsystem when access privileges are checked based on temporal and normal constraints. On the other hand, intrusion detection manager interacts with intrusion detection system agents and rule subsystem actively using special kinds of rules.

3.6 TEMPORAL REASONING SUBSYSTEM

The temporal reasoning subsystem is used for effective temporal reasoning with the coordination of the rule system. It consists of two components namely Rule Manager and Knowledge Base.

3.6.1 Rule Manager

The purpose of the rule manager is to provide various database services and to support temporal reasoning. Whenever a user command is given through the ETMSQL interface to the temporal database manager, the temporal database manager checks the data dictionary and find out whether it is constrained by rules. If it is constrained by rules, it sends the tuples which are needed by the rule manager to the discrimination network module of the Rule Manager and also sends a message to the rule events manager. The rule events manager triggers the necessary rules by retrieving them from the rule base. This together with temporal rules are given to the rule execution monitor for firing all possible rules. The execution monitor assigns priority to each rule and transfers them to the scheduler which in turn fires one rule at a time and sends them to the query processor. The Rule Manager evaluates the
condition part of a rule and then the action part of the rule is executed to check the constraints.

3.6.2 Knowledge Base

Using the knowledge gained from the experts after validation, the knowledge is coded into the knowledge base in the form of facts and rules. Database systems provide indexing and hashing techniques to organize the data effectively in the physical database. On the other hand in this system, the entire set of rules and facts are stored in Knowledge Base using a Frame-based Knowledge representation technique for effective creation and maintenance of the Knowledge Base. The major advantage of using this knowledge representation helps for effective retrieval in order to reduce the cost of query processing. The Knowledge Base system uses a slot-filler format to store facts and rules using two fields respectively for storing the Condition and Action parts.

3.7 TEMPORAL SECURITY SUBSYSTEM

This subsystem provides a secure way to access the data and also helps to detect unauthorized intruders from the entire system. The temporal security subsystem consists of two sub components in which the first one is constructed for providing effective access control and the second subcomponent is responsible for intrusion detection in order to provide enhanced security. Moreover, a new type of Agent-Based Intelligent Intrusion Detection Module proposed here is highly accurate since it has very low false alarms. Moreover, we propose an algorithm called Temporal Role Based Access Control using intelligent agents which enables the access privileges of all users by validating every user for the specified duration. A Modified Bayesian Classification algorithm has been proposed and implemented for
performing effective intrusion detection which is an extension of Naïve Bayes Classification Algorithm with additional constructs for handling temporal information.

3.7.1 Access Control Module

The access control module provides security using roles and agents. In order to provide security through roles, this system uses temporal roles and suitable agents for processing role based access control rules by making use of the rules stored in the Rule Base.

3.7.2 Intrusion Detection Module

In this work, a new type of Agent-Based Intelligent Intrusion Detection Module has been proposed and implemented using intelligent agents. For this purpose, five agents have been proposed and used in our system namely Monitoring Agent, Detection Agent, Classification Agent, Action Agent and Communication Agent. The Monitor Agent identifies anomalous network activities (represented by different variables) and use them to detect known attacks and events of interest. The Detection Agent detects one of the known attacks which receives its inputs from various Monitor Agents. These input values are fuzzified by membership functions that partition each input variable. Using the information provided by Monitor Agents, the Detection Agents can detect the status of attacks by using attack definition stored in a Rule Base. The Classification Agent identifies the type of Intrusion Detection that has occurred using temporal constraints and makes the decision with the help of rules. After identifying the type of intrusion, the Action Agent takes necessary steps for further process. All these agents use the Communication Agent in order to make effective communication with the Rule Manager.
3.8 THESIS CONTRIBUTIONS ON ARCHITECTURE

The major contributions of this research work with respect to the architectural framework are the proposal of the Temporal Reasoning Subsystem and Temporal Security Subsystem. The Temporal Reasoning Subsystem uses various algorithms for effective query processing. With the help of Temporal Access Control techniques and Agent Based Intelligent Intrusion Detection techniques, the Temporal Security Subsystem provides effective and secured services.