Chapter 4

SYSTEM DESIGN

System design is the process of defining the architecture, modules, components, interfaces and the data for the system to satisfy specified requirements. It pertains to the abstract representation of the data flows, inputs and outputs of the system.

4.1 Design Considerations

In this section, we consider the different modules under consideration, use-case diagram, sequence diagram, data flow diagram and state diagram.

4.1.1 Modules

The system is comprised of five modules. They are -

- Control Unit (CU)
- Traffic Capture Module (TCM)
- Learning Module (LM)
- Detection Module (DM)
- Mitigation Module (MM)

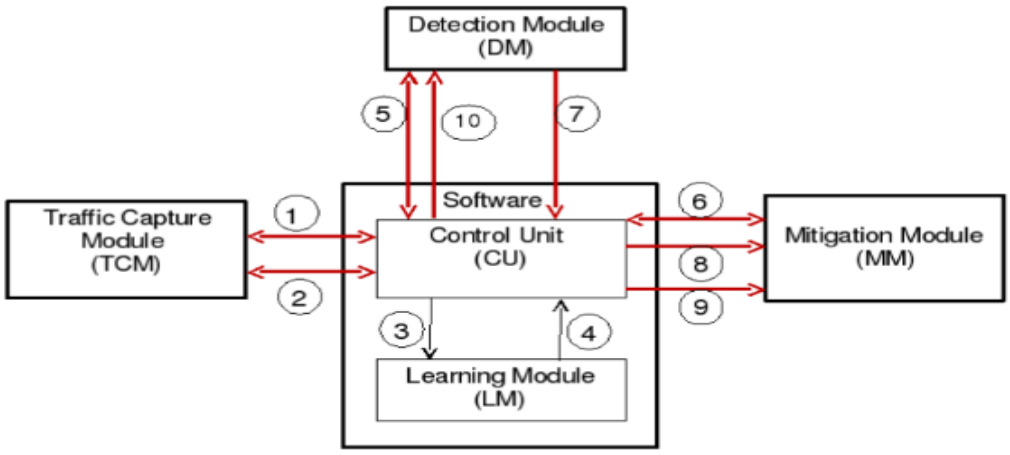

Figure 4.1 Overall System Design

Control Unit (CU):

The Control Unit is a software module, which centrally manages communication between the rest of the modules. CU is resident on the computer, hosting the Learning Module (LM), and interacts with the user (through a Graphical User Interface) to obtain inputs for the above modules. The CU also is responsible for information exchange and dissemination between the modules, and establishes a communication network between each of these modules for this purpose. The CU runs different protocols with the various modules for the task. This is a software module.

Traffic Capture Module (TCM):

The Traffic Capture Module (TCM) is a hardware module, which captures traffic required for learning phase, and computes traffic statistics as required for the Learning Module. The TCM communicates with the Control Unit (CU) to obtain it's inputs, namely stream information. Based on triggers, the values of stream configuration are loaded (by CU), traffic captured and statistics supplied back to the Control Unit.

Learning Module (LM):

The Learning Module (LM) is a software module, which resides in the host computer which is connected to the rest of the hardware modules. The LM accepts configuration details of the server targets, and learning traffic statistics (as captured by the TCM) from the CU, and arrives at model probabilities which will later be used by the Detection Module (DM) for classification of normal traffic/anomalous traffic. It is into this module that the ideas developed as a result of this research work are implemented.

Detection Module (DM):

The Detection Module (DM) is a hardware module, which on input the set of model probabilities of normal traffic per stream (output by LM, and supplied by CU), determines the state of the network (normal or abnormal) in real-time. This module is the operations module for the system. The DM talks with the CU to obtain it's inputs and interrupts the CU in case an attack flag is asserted. This will cause further action by the CU on the Mitigation Module (MM).

Mitigation Module (MM):

The Mitigation Module is a hardware module, which allows input traffic unaltered when attack flag is not set in any stream, and when attack flag is set in one or more streams, it filters traffic for the stream (s) on the basis of a white list of source IPs corresponding to the stream.

4.1.2 Use Case Diagram

Use case diagram represents a user's interaction with the system. It represents the dynamic aspect of the system.

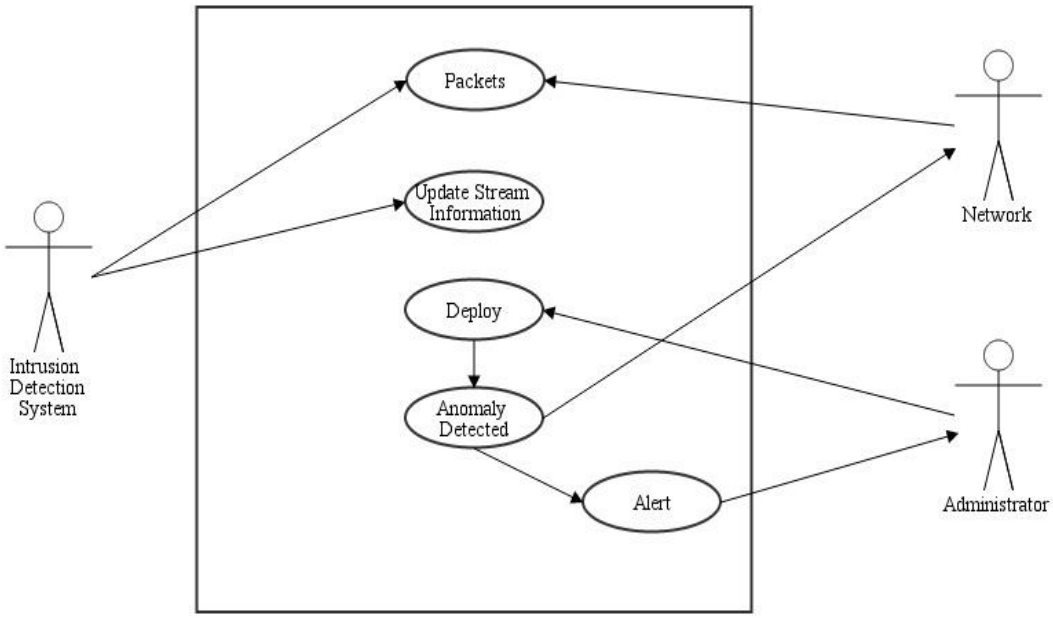

Figure 4.2 Use Case Diagram

The three actors of the system are -

- Intrusion Detection System
- Network
- Administrator

The packets are sent across the network which are accessed by the IDS to update the Stream Information. The updated stream information is used in the testing phase where, based on the incoming traffic, anomalies are detected and are notified to the administrator.

4.1.3 Sequence Diagram

Sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It shows object interactions arranged in time sequence.

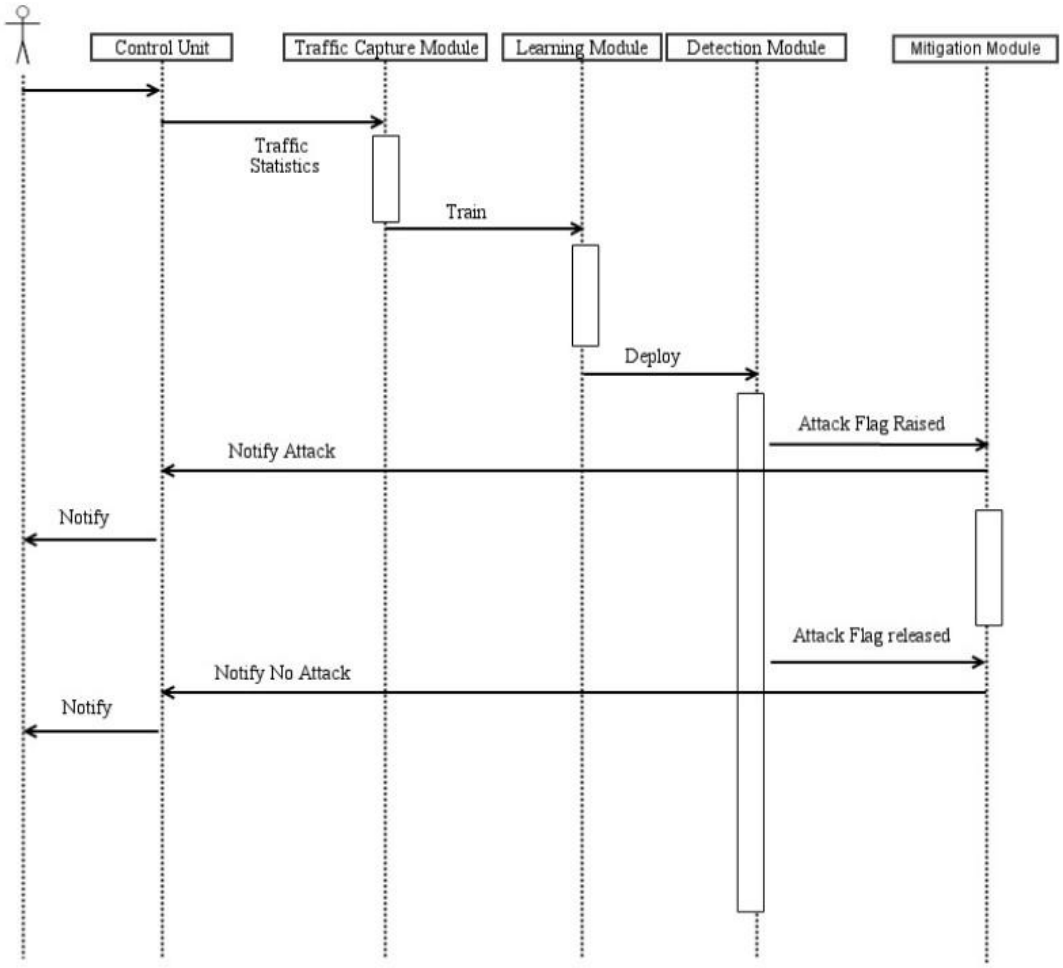

Figure 4.3 Sequence Diagram

The different objects in the system are administrator, CU, TCM, LM, DM and MM. The interaction between each of these objects are denoted by horizontal arrows. The sequence of operation is as follows:

1. The administrator is interacting with the CU to provide input.
2. The input in the form of traffic statistics is given to the TCM.
3. Parameters from the TCM are filtered and provided to the LM to train the model. This phase is the learning phase.

4. After the learning phase, the model is deployed. The incoming traffic is analyzed by the DM. If any anomaly is detected, an attack flag is raised and sent to the MM.
5. A notification is sent to the CU which then notifies the admin indicating an attack.
6. The MM whitelists the IPs and once the attack flag is released by the DM, the CU is notified that the attack is curbed and in turn informs the system admininistrator.

4.1.4 Data Flow Diagram

Data Flow Diagram (DFD) is a graphical representation of the flow of data through the system. A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored.

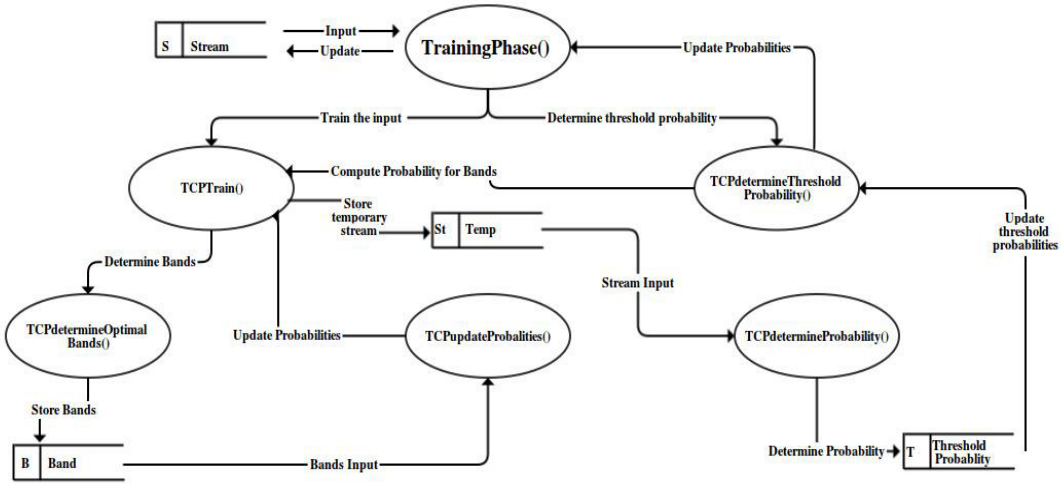

Figure 4.4 Data Flow Diagram

Flow of data in the system is explained as follows. The stream S is passed as an input to the training phase. The input is used to train the model by determining bands. The bands are stored and passed as input to update the probabilities. The updated probabilities are used to determine the threshold probabilities. The threshold probabilities are updated in the stream. Once the training is done, the model is deployed and is used to identify and classify traffic as Normal or Anomalous traffic.

4.1.5 State Diagram

State diagram defines the behavior of the system. It gives an abstract description of the behavior of the system.

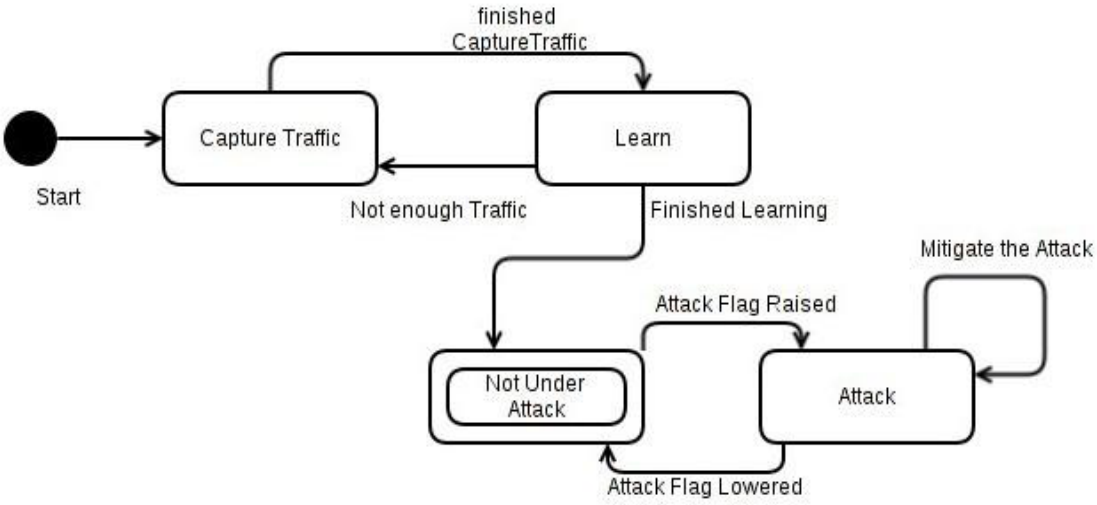

Figure 4.5 State Diagram

The different states are Capture Traffic, Learn, Not Under Attack and Attack. The behavior of the system is explained as follows -

1. Initially, the system is in the **Traffic Capture** state, where it captures the network traffic.
2. Once enough traffic is captured the system moves to the **Learning** state. Here, it learns and determines the normal profile of the network.
3. When the learning process is over, the state is changed to **Not Under Attack**.
4. When an anomaly is detected, the attack flag is raised and the state changes to **Attack** state.
5. While in the Attack state, the system tries to mitigates the attack.
6. After successful mitigation, the attack flag is lowered, and the state is restored to **Not Under Attack**.