
Joe Simons and Luis Zaman

FACULTY MENTOR: Dr. Edwin Hou

Abstract

Intrusion Detection Systems (IDS) are notorious for generating a flood of alerts. Several methods for alert aggregation and correlation aim to alleviate this issue, however, there has been little work done to facilitate the development of new aggregation and correlation methods. Our approach uses the extensibility and simplicity of Petri Nets as a framework for modeling intrusion detection systems. We propose a novel tool for Rapid Analysis and Prototyping of Intrusion Detection Systems (RAPIDS) using extensible Petri Nets and a graphical user interface.

Introduction

Aggregation and correlation of IDS alerts address shortcomings in traditional IDS systems such as their tendency to flood the system operator with information that is often not relevant, the inability to group alerts meaningfully, and the high number of false alerts that IDS systems generate [3]. The methods of aggregating and correlating IDS alerts can be categorized into two fundamental groups: misuse detection and anomaly detection. Misuse detection-based systems attempt to match system events to an intrusive behavior while anomaly based systems attempt to detect divergence from normal behavior. Actual instances of these two groups can take on a multiplicity of forms from rule-based systems to state-transition models [2, 5, 7, 9, 10].

Our goal is to develop a tool that allows for rapid prototyping of various IDS systems as well as IDS alert aggregation and correlation systems. We believe that Petri Nets are the best representation for this task and implement a graphical tool which allows the user to quickly and intuitively create Petri Net models.

Our Approach

Why use Petri nets

Petri Nets have several advantages for rapidly prototyping an IDS or an intrusion alert aggregation and correlation system.

Graphical Representation. A Petri Net can encode all essential system information into a single graphical display, and in the context of a toolkit, more information can be retrieved and analyzed as needed.

Explicit Description of States and Actions. This allows the system administrator to analyze system state—the current state as well as past and possible future states—consider the events that will lead to state changes. The ability to analyze either states or events, or both states and events simultaneously gives great flexibility to the model.

Partial Ordering of Events. It is advantageous to model multiple attack scenarios where certain events must occur in sequence and other events can occur in any order or even concurrently.

Hierarchical Descriptions. A large model can be divided into subsets, which can be analyzed individually or as a part of the whole. This modularization allows large systems to be simulated more quickly. Also, it allows the modeler to hide unwanted details when working at a system-wide level.

Petri Nets have many other advantages which are not included here due to space restrictions. For more on Petri Nets, see [2, 9, 10].

The RAPIDS Toolkit

RAPIDS stands for Rapid Analysis and Prototyping of Intrusion Detection Systems. The toolkit is a visual editor and simulator for Extensible Petri Nets.

RAPIDS Features:

• Platform independent (Java based implementation)
• Open Source
• User Friendly GUI
• Load/Save to XML and export to image
• Supports Generalized Stochastic Petri Nets
• Supports modified Colored Petri Nets
• Executes Alert Streams

Petri Nets have two types of nodes, places (circles) and transitions (rectangles). Places are connected to transitions and transitions are connected to places via directed arcs.

Tokens (dots) represent a stage in the execution, and the overall state of the machine is called the marking of the Petri Net.

Transitions fire when all the input arcs are satisfied; that is, when all the places connected to the input arcs of the transition contain a token(s).

In this example, the Start state and state P2 contain tokens so the Petri Net is marked. Since the input arc to transition T0 is satisfied (there is a token in the Start state), T0 is enabled.

Informal Definition

Our Extensible Petri Net model is formally a 10-tuple \((P, T, A, M, R, W, D, F, G, S)\) where informally:

- \(P\) The set of places
- \(T\) The set of transitions
- \(A\) The arcs between places and transitions and between transitions and places
- \(S\) The set of colors
- \(M\) The marking distribution
- \(R\) The set of firing rates for the transitions
- \(W\) The weight function of the arcs
- \(D\) The guard functions on the transitions, a Boolean function that is evaluated on a transition and a unified token (color) set.
- \(F\) The set of observable items
- \(S\) The probability mapping on observations and transitions of a transition becoming active.

This model implements aspects of colored Petri Nets which add guard functions. This gives a sub-procedural aspect to the execution of the Petri Net. It also allows for tokens to collect information as they move.

- The generalized stochastic Petri Net aspects allow for both timed and untimed transitions. That is, transitions that fire after a randomly distributed time and those that will fire instantly.

- The model also includes a distinction between observation and action which we call hidden Petri Nets. Observations correspond to probabilities that a transition will fire, so a probabilistic distribution can characterize the system, and a confidence value can be calculated during each stage of the execution.

Future Work

- Add full support for Hidden Colored Petri Nets (i.e. the ability to train the Petri Net, and the recalculating of probability distributions.)
- Support the automatic generation of a Petri Net given another knowledge base. For example, the TIAA toolkit’s Hyper Alert file [5].
- Add support for Inhibitor Arcs.
- Refine the original implementation of timed transitions.
- Expand Colored Petri Net support to include complex color sets.
- Implement the guard functions.
- Investigate the usefulness of pre-condition, post-condition, and invariant expressions.
- Add more analysis modules to RAPIDS.

References