August, 2003

Network Security Risk Assessment Modeling Tools for Critical Infrastructure Assessment

George H Baker, *James Madison University*
Samuel Redwine, *James Madison University*
Joseph Blandino, *Virginia Military Institute*

Network Security Risk Assessment Modeling (NSRAM) Tools for Critical Infrastructure Assessment

G. Baker, S. Redwine, P. Riley, J. Blandino
College of Integrated Science and Technology
James Madison University

Abstract

The James Madison University (JMU) CIPP research team is developing Network Security Risk Assessment Modeling (NSRAM) tools that will enable the assessment of both cyber and physical infrastructure security risks. The effort is driven by the need to predict and compute the probability of adverse effects stemming from system attacks and malfunctions, to understand their consequences, and to improve existing systems to minimize these consequences.

The tools are targeted at systems supporting critical infrastructures varying from individual systems to organization-wide systems, to systems covering entire geographical regions. Early work emphasizes computing systems, but systems sharing the network nature of computing systems, such as electrical and water supply systems are potential targets.

Input consisting of network topologies and interdependencies, recovery and repair capabilities, attack scenarios, and traffic analysis data, will enable the NSRAM tools to evaluate critical dependability issues including potential outage longevity and costs, data loss and such risks as are associated with network problems for user specified scenarios. Decision-making analysis support will be included such that it will provide modeling to support design, operation, maintenance, continuity, and recovery of these systems.

It is expected that the initial products will be somewhat technical in nature, for the use of JMU consultant-level experts, with the immediate future development work concentrating on modeling computer security phenomena and user interface refinements to increase accessibility.

1.0 Introduction

In 2003, the Department of Homeland Security issued national strategy documents for the protection of physical and cyber infrastructures that call for vulnerability assessments of critical infrastructure systems.[1,2] Modeling tools for simulation of network security and risk assessment will be an important part of such assessments.

Critical infrastructure systems and facilities are subject to many different failure modes. It is important to anticipate the possible modes, the likelihood of their occurrence, and the relative seriousness of their consequences. Failures may be due to many causes, intentional and non-intentional, including cyber attacks, accidents, aging or sabotage from insiders or external malefactors. Failures can propagate such that seemingly minor problems may lead to complete functional failure. Some serious failure modes may be counter-intuitive. Of particular concern is the presence of "single point failure" locations known to exist in many existing critical facilities. Assessments provide an important basis for determining the most serious failure modes, implementing cost-effective countermeasures, and planning for reconstitution.

To facilitate balanced assessments of both physical and cyber security problems, we are pursuing two approaches which extend probabilistic risk assessment into the time domain.

a. An approach oriented to physical infrastructure assessment involving system fault trees
b. An approach oriented to information infrastructure assessment involving simulation of network flows

2.0 Time Domain Fault Tree Technique

A standard technique for assessing system failure modes and their respective likelihoods is probabilistic risk assessment (PRA). One approach to PRA modeling is accomplished by building fault trees that link together the hierarchy of systems, subsystems, and components necessary for a facility, system, or system network to perform its mission. To analyze and quantify survivability, conventional probabilistic risk assessment methods provide a snapshot of potential failure modes at a single point in time for certain initiating conditions.

We are perfecting an approach that improves upon normal PRA by adding the time dimension to the evaluation of failure modes of interdependent systems. The rudiments of the analytical method were first developed by the Department of Defense for target

[1] The National Strategy for the Physical Protection of Critical Infrastructures and Key Assets, U.S. Dept of Homeland Security, February 2003.
[2] The National Strategy to Secure Cyberspace, U.S. Dept of Homeland Security, February 2003

assessment. With straightforward modifications, that approach is well suited for defensive analysis of critical infrastructure failure modes and protection measures.[3]

The approach involves computing the evolution of overall system functionality in time by evaluating initial failure probabilities, effects onset times, and system repair-reconstitution times for single or combinations of critical systems. The model provides a measure of incident seriousness in terms of likelihood, outage longevity, and seriousness of consequences. Using this technique, an assessment team can determine which types of failures have the highest probability of putting a critical system off-line the longest.

The technique gives the infrastructure owner/provider metrics to assist in decisions concerning investment strategies to improve infrastructure protection and reconstitution. The technique provides information useful in weighing the advantages of buying protection to reduce initial failure probabilities (often a costly proposition) or accepting high initial failure probabilities and relying on emergency response contingencies. The model is unique in that the time evolution of the probability of effects on mission is built in using system fault-tree analysis, individual functional damage probability distributions, and time constants governing effects onset and reconstitution/repair.

2.1. LabView® Coding

The technique is being prototyped using the LabView® graphical programming language. LabView® allows the development of visually appealing and intuitive user interfaces. These interfaces can be developed with minimal programming experience. For probabilistic modeling, each system component is described by a separate subroutine. Individual component status indicators (figure 1) can be displayed and changed according to the scenario of interest.

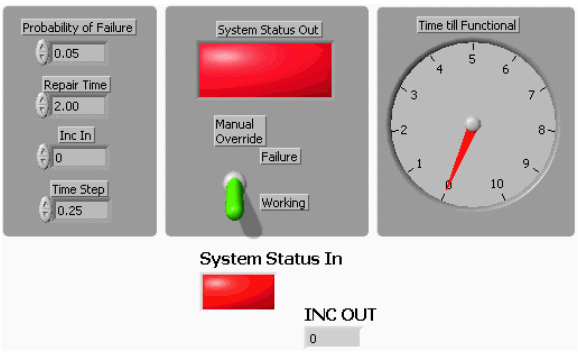

Figure 1. Individual Component Status Display

[3] G. Baker, C. Mo, K. Calahan, " Functional Survivability Modeling Tool for Complex Facilities," EUROEM Conference, Edinburgh, Scotland, May 2000.

Once developed, these individual component subroutines can be easily connected or "wired" together through logic icons. As an initial test case, we are using the hypothetical communications center programmed in Figure 2.

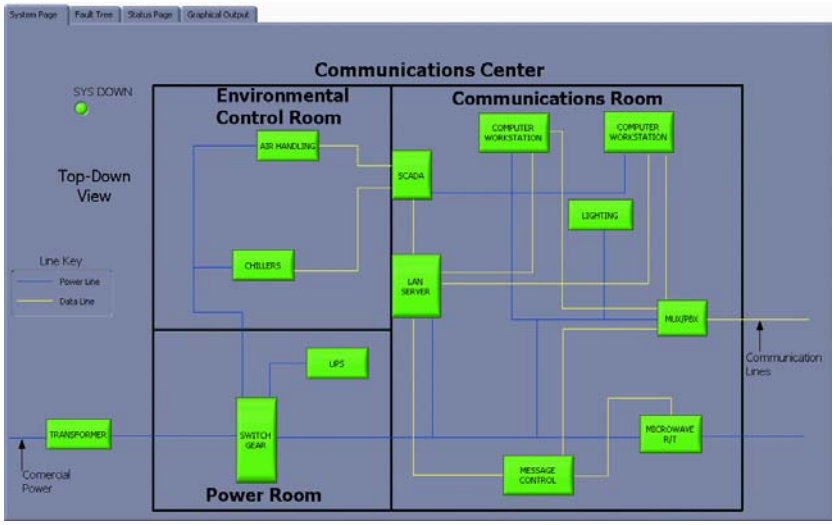

Figure 2. Communication Center System Interconnection Diagram

Based on the system functional diagram, a fault tree is derived that contains the Boolean relationships among the system elements (Figure 3). The element states are green or red depending upon whether the element is functioning or not. The initial states of the components are programmed based on the scenarios of interest. The fault tree determines how effects on single or combinations of system elements propagate and ultimately indicates whether the total system can perform its mission. The code is being developed to compute system reconstitution times based on repair times for individual elements and repair sequences (e.g. in many cases it is necessary to reconstitute electric power before other systems can be serviced).

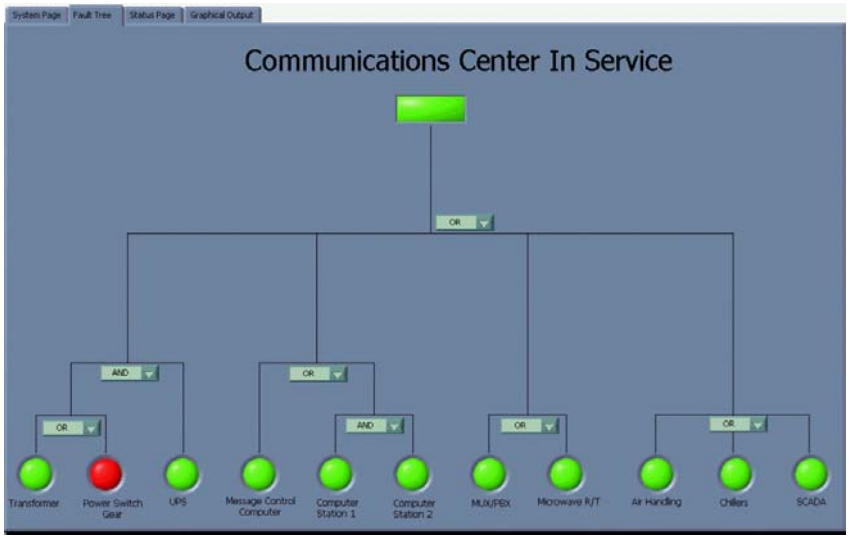

Figure 3. Communication Center System Fault Tree

The software advances the clock in discrete time steps. Using a basic Monte Carlo simulation the code will produce a graph of $P_o(t)$, the probability that the system is out of service up to time t based on the specified scenario. By convolving the code output with outage cost values, $C(t)$, one can estimate the probable value of lost services:

$$Loss\ Value = \int_0^t P_o(t)C(t)dt$$

The code is oriented to simple, top-level risk assessments of physical infrastructure systems. We have demonstrated proof-of-principle for the time domain risk assessment technique and are now pursuing applications to real system problems.

3. Network Flow Simulation Technique

Bits flow through fiber optic cable; water flows through water mains; electric power flows from generation to use; traffic flows over highways, railways, and waterways; and money flows through the banking network. The network flow approach mimics this dynamic behavior of critical infrastructure networks and overcomes the need for a human analyst to construct a fault tree – and the possibility of human error in its construction. A modeler finds it more natural and more easily validated to directly model the system.

On the other hand, this technique shares concern for all the temporal phenomena discussed in Section 2 including PRA. The phenomena, however, can be modeled with more gradation and subtlety. For example, each element in the model can have multiple attributes, and degradation is possible – not just failure.

Central to the modeling framework of the NSRAM Tool are the concepts of elements and flows among elements. In its full generality, elements flow themselves as well as receive, store, transform, compute implications of, and send flows. Flows can be discrete or continuous. Continuous flows, however, are represented by discrete chunks (normally for each time interval). Elements have points of connection or ports for flows. An element whose only function it is to convey a flow is called a conduit.

The flow model represents element interdependency as a system of "flows" between elements. Each element puts out various flows that other elements may depend on for correct functioning. For instance: a generator produces a flow of electricity that electrical devices depend on. Less obviously, one may also model a cooling system as producing a flow of "coolness" that machinery and electronic devices depend on or a controller unit as producing a flow of commands that a device depends on.

Among the key functionalities required in addition to the basic simulation engine are the abilities to do

- Varying scenarios
- Dynamic network reconfiguration
- Calculation of measures of merit
- Comparison and analyses of results
- Extensions of previously performed simulations
- Maintenance of history of previous simulations and analyses
- A simulation in parallel on multiple machines
- Statistical analysis of results
- Sensitivity analysis
- Visualization
- Repeatable runs
- Step-by-step as well as complete runs
- Observe behavior in detail

The last three are necessary developmental needs as well as needs of users.

The simulation approach is a Monte Carlo one with many runs being done and the resulting distributions of behavior and measures of merit over time being the basic results. The approach is essentially a "computational experiments" one with each local aspect's behavior being modeled and the effects on services and system emerging from the computations.

In this dynamic environment, the question arises of whether some useful static cause-and-effect or dependence diagram could also be derived. That this will become a research question shows, in part, the difficulties of static approaches such as fault trees in highly interactive and adaptive systems.

How to best do simulations of computer-security-related phenomena is also a research question – or a set of questions for different phenomena and purposes. While the NSRAM Tool may be innovative in some of its approaches to modeling networks and distributed systems, this is a better understood territory than the area of modeling computer and network security-related phenomena.

The figures show views in an early prototype tool interface of a multi-star network. As can be seen, even a small computer network such as this begins to become hard to see in detail all at once on a workstation screen. Therefore, the interface provides zoom capabilities of several kinds including by selecting a portion of the screen to zoom in on.

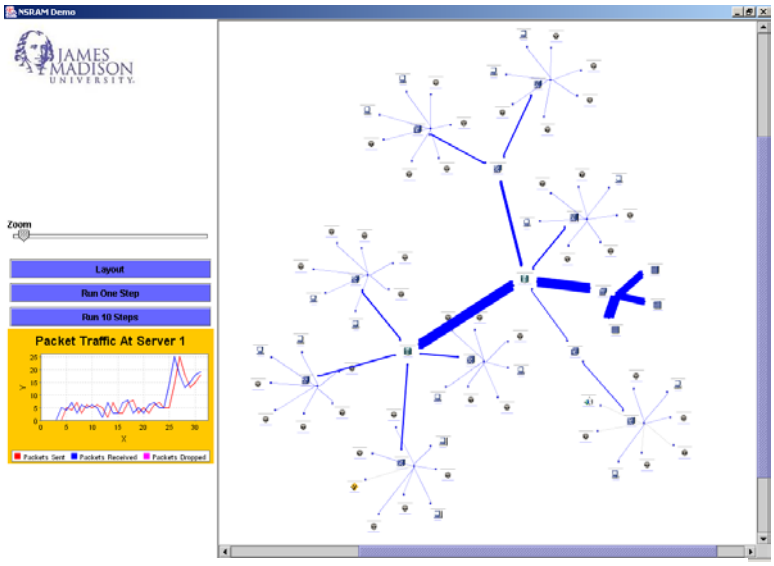

Figure 4 – NSRAM Tool Zoomed Out

The interface animates the thickness of connecting lines to indicate the volume of traffic flowing. To provide immediate feedback to the user, the user can highlight some node in the network and obtain data on its behavior. Of course, full analysis of a Monte Carlo set of runs must wait until the tool has performed all the runs in the set.

This tool is early in its development. We have performed an initial needs analysis, and postulated a list of features a fully developed tool would have. We recognize, however, that the need for a significant number of new features and variations on features will arise during the early uses of the tool as it models real world problems.

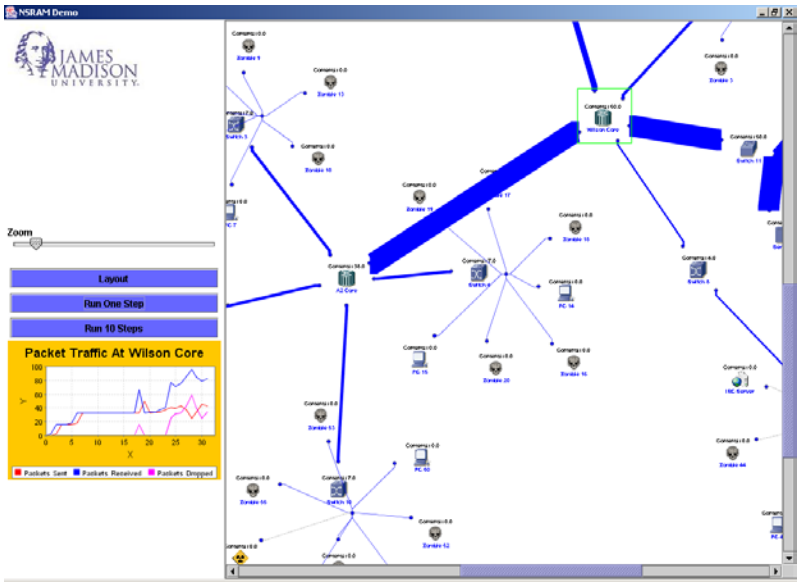

Figure 5 – NSRAM Tool with Node Highlighted

While many questions can be addressed relatively straightforwardly, significant effort will be required to best formulate useful advanced computer network security questions in ways that can be insightfully addressed by the tool (or any tool). The issues are not all technical. For example, the impacts of contaminated data and its propagation are quite dependent on the uses of the data.

As mentioned in the prior section, ultimately, the stakeholders' utilities for various system conditions over time must be factored in if one wants to quantify risk. Some quantification of risk is often necessary to make rational decisions on investing in upgrades to increase dependability including increasing security. This part of the JMU

effort aims at providing sophisticated modeling support to aid risk and value determinations involving complex network-oriented systems and phenomena.

4. Conclusion

We are developing risk analysis modeling tools that address physical and cyber infrastructures. The tools extend probabilistic risk assessment into the time domain and include a time-domain fault tree technique and a network flow simulation-based technique. The fault tree technique provides a simple, top-level calculation of overall system mission functionality vs time. The network flow-based technique provides detailed system service performance, security and risk metrics vs time.

For critical infrastructure networks the JMU models/tools will provide insights into failure and degradation including possibilities, probabilities, modes, and durations. The tools are particularly attractive for modeling interdependencies and cascading failures. They may be used to gain insight into most probable failure points and cost-effective protection and/or upgrade options. The models enable estimates of the cost of service outages.

The tools are still in the developmental stage. It is expected that the initial products will be somewhat technical in nature, for the use of JMU consultant-level experts, with the immediate future development work concentrating on modeling computer security phenomena and user interface refinements to increase accessibility.

Successful application of these tools requires that they be used as part of a well-defined risk assessment methodology and that system subject matter experts be involved in defining input parameters to provide reliable results.

November 2003