Time as a Metric for Defence in Survivable Networks

Kalle Burbeck, Sara Garcia Andres
and Simin Nadjm-Tehrani
Dept. of Computer and Information Science
Linköping University, Sweden
kalbu@ida.liu.se

Michael Semling and Thomas Dagonnier
Innovations, Security & Service Management
Swisscom Ltd, Switzerland

***Abstract*—Critical infrastructures of today's society are built over networks that require a degree of survivability not foreseen when they were built. This paper reports on work in progress in a European project that aims to safeguard critical infrastructures such as electricity and telecom networks. It assumes that there will be accidents, attacks, and failures in parts of a network. The goal of safeguard is to enable delivery of the essential services despite these. Hence, we define a metric for network level survivability in terms of a continuous function of critical components' availability and integrity. We further go on to measure the survivability of the system in terms of the *time* taken to breach of survivability. In a system where the implemented defence/recovery mechanisms are not adequate, this time is finite. In a system that implements self-healing, the presence of attacks and failures is continuously compensated by defence and recovery mechanisms. Again, a measure of time to recover from component failures is a key to increased network survivability. The paper presents a preliminary study of defence mechanisms in a telecom management network, and illustrates how simulations of the network and harmful data can be used to identify trade-offs that are central to increased survivability.**

***Keywords*—Survivability, Simulation, Intrusion Tolerance, Dependability, Timely defence.**

I. Introduction

THE economy and security of Europe is increasingly dependent on a spectrum of critical infrastructures such as energy distribution and telecommunication networks. Protecting these infrastructures requires an understanding of the multiplicity of vulnerabilities that exist in every layer of the network, from the physical layer up to the network and service layers as well as the organisational layer that supports the complex operation of these networks [1]. A growing proportion of the vulnerabilities are due to the unbounded nature of today's networks and the merger with global information systems (IP-based networks). There is a growing awareness that centralised solutions against security breaches are not sufficient and a distributed system of safeguards with defence mechanisms at all levels of the network are required for self-healing and continuous delivery of critical services.

The aim of Safeguard project is to build demonstrators that exhibit a positive impact on survivability in both electricity and telecom domains. In this paper, we present initial studies in a simulation environment for deriving the suitable mix of defences that allow a demonstrable improvement to survivability. The studies indicate the usefulness of time to breakdown as a metric for adequacy of a given range of defence mechanisms. The paper shows the application of the simulation platform to a model of a management network using data streams from the main Swiss telecom operator company Swisscom. Further work in the project will use the basic insights gained in these studies to design agents that monitor and act on anomalous and harmful data and assure predefined levels of critical service. These will be tested in an emulated environment.

The main contributions of the paper are as follows:

- Providing a set up in which critical service levels can be formally defined and their relationship to the performance of defence mechanisms studied;
- The use of *time to break down* as a prime metric for increased survivability, and its relation to critical service levels and time to recovery in attacked nodes;
- Illustrating the need for trade-off studies with respect to different dependability attributes, by analysis of the gains in integrity in the light of number of false positives and false negative rates;
- Applying novel simulation studies for survivability analyses in the context of a real telecom environment.

II. Modelling survivability

Network survivability has been the subject of extensive studies in the light of alarms generated by the commissioned reports in the 90's, specially in the US [2][3]. Several articles attempt to define, delimit, and survey the approaches for achieving survivability, some of which provide a comprehensive introduction to the topic, and others propose specific approaches (see references in [4]). On a European scale, an early initiative was a study of the problem area by the EU Joint Research Centre [5] and the follow-up European dependability initiative.

A. The basic notions

Ellison et al. define survivability as the capability of a system to fulfil its *mission*, in a *timely* manner, in presence of attacks, failures and accidents [3]. We adopt the basic definition above, and instantiate the "mission" as delivery of *critical services* in the management network under consideration. In our approach we quantify the critical service level for a network in terms of a formula over individual services provided in the network. Thus, there is a direct link between availability of the individual services provided by the network and the overall survivability of the network (in terms of providing a weighted minimum accepted combination of these services).

Supported by the European project Safeguard IST-2001-32685.

Noteworthy in the context is the distinction of survivability from other dependable computing concepts since it focuses on the system behaviour *after* attacks have taken place, whereas typical attributes for security attempt to *prevent* attacks taking place, as seen by some authors (see references in [4]). Other authors would classify high-level notions such as survivability, dependability, and trustworthiness as the same essential properties that assure protection [6]. The scenarios we present are representative of a network's behaviour despite the success of a number of attacks, and our simulation platform covers the potential worst-case scenario, with a total break down of delivery of critical services. Our model of a network also represents mechanisms for recognition, resistance, reaction and recovery in presence of external attacks.

B. The modelling approach

The basic idea in this work is to model the machines that build up a network and the services they are capable of providing, represent their vulnerabilities to external attacks, and measure the time that the network would survive if fed with realistic packet streams, with no defences in place. The survival of the network is defined in terms of survival of a predefined level of service (relating to the notion of mission above). This is typically a function of number of (and/or % of) machines or services of certain types. Later, the simulated network is extended with various defence mechanisms and exposed to the same data sets. The time measured, this time in presence of various defence mechanisms, is expected to be longer than the original time to break down, and the extension in time gives an indication of which defence types are more appropriate in which circumstances. The experiments also expose the interplay between the interesting parameters.

A modelled network includes examples of machine/service types and it represents the operating system and the attack types each service is vulnerable against. In our studies we have used the following machine/service types: Workstation, Router, Web server, Print server, Mail Server and Ftp server.

The model also includes a critical service level expressed as a function of Simulated Machines (SM). This parameter can be modularly changed and experimented with. An example of a service level definition is shown below, where *i* ranges over the type of service:

$$f(\ldots) = \sum_i \left(\frac{100}{\alpha_i} \cdot \frac{|InfectedSM_i|}{|SM_i|} + \theta((|SM_i| - |InfectedSM_i|) - MinSM_i) \right)$$

where $\theta(z) = \begin{cases} 0 \; if \; z \geq 0 \\ 1 \; if \; z < 0 \end{cases}$.

The system is considered to suffer a complete break down when it no longer provides its critical service, i.e. when the value of f exceeds 1. Note that the first term in the expression above (left operand of the addition inside the summation term) describes the relative importance of different types of machines/servers in the essential services provided by the network, weighted by α_i in the formula. This parameter describes the relative importance of various service types; the higher the value of α_i the lower the impact of infections in that service type on the overall function f approaching 1. The second term, described by the result of the θ function, indicates the minimum number of machines of each type needed for network survival. θ is typically chosen as a step function.

The data set that the network is subjected to consists of packet streams that have a built-in characterization of *good* and *bad* packets. How these characterizations may look like in a specific case will be detailed in section III. In the absence of any defences, the bad packets destined for a given machine, if they match the vulnerability of that machine, will compromise the services on that machine within a certain interval of time. This interval is modelled by a random duration to reflect the uncertainties present in realistic scenarios. Each time a machine is compromised, an observer that checks the availability of the critical service level is notified and the time for this event is recorded. Once there are enough services affected by the attacking packets, the observer notices the break down of the system and the measured time at that point from the start of the experiment gives the time to break down (see Figure 1).

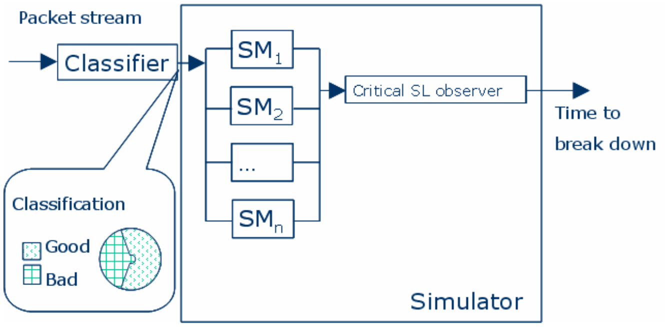

Figure 1: Classification of packets in good or bad

To model defence mechanisms, a number of recognition, reaction and recovery strategies are modelled. The system may also include a measure of resistance, e.g. by having replicated services, even in the absence of any other defences. To model *recognition and reaction* (R & R) in the simulator, we create a representation of a network intrusion detection mechanism that classifies packets as *Red* and *Green* respectively, together with a per-packet strategy of what to do with each Red packet. Green packets should obviously run through the system and not affect survivability. A system recognition and reaction mechanism may, however, decide to remove some proportion of the Red packets.

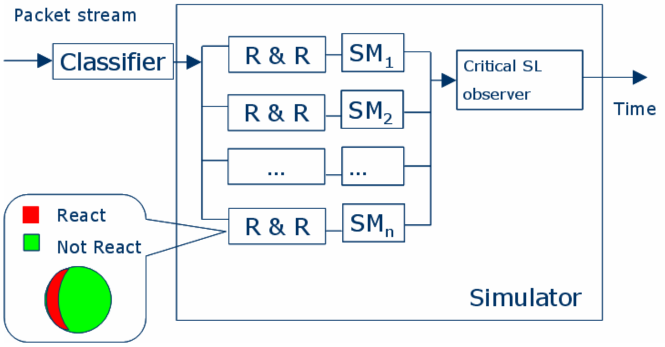

Figure 2: Recognition and response

Figure 2 shows the addition of the above defence mechanisms, and the measurement of the time to break down on same data sets as before.

Next, the network can be extended with recovery mechanisms, the simplest of which would be a unit that notices a crashed service and restarts the service/machine. Again, to reflect the variations and uncertainties present in real systems we model the recovery for each service to take some random duration within a given maximum recovery interval. Adding this dimension to the network and measuring time to break down is expected to increase the survivability of the system given the same critical service level. Figure 3 shows the complete simulated environment that has been instantiated for a telecom-based application further described in section III.

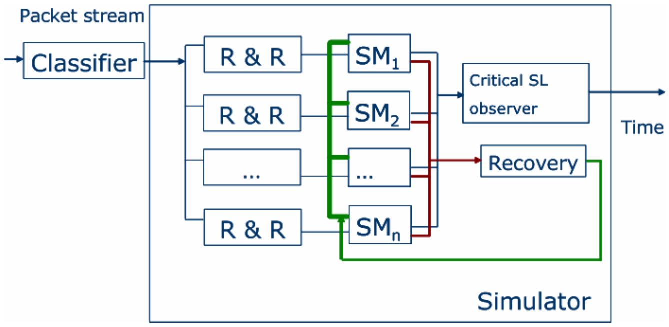

Figure 3: Recovery in combination with recognition and response

III. THE EXPERIMENTAL SET-UP

In this section we present the details of how the simulation runs were designed using a model of Swisscom management network data and topology.

A. Physical structure and data set

The modelled network consists of 108 machines with various services, each with some vulnerability. The topology of the network (choice of IP addresses, proportion of machines of each type) is provided by Swisscom. Inputs are taken from a real Swisscom network packet stream. The packet dumps are classified and the relevant information is collected as XML descriptions fed into the simulator. The information in the packet description includes the destination address, a classification of the packet as a potential attack and the vulnerability it targets.

B. Packet classification

For these experiments Snort [7], a popular open source network intrusion detection system, was selected as the packet classifier. However, the packets considered suspicious by Snort do not necessarily successfully compromise the system. To model the existing resistance mechanisms in a network (organisational, technical, e.g. anti virus), we model some of the suspicions as successful and some as not successful (Bad/Good), even in the absence of explicit defence mechanisms. The choice between Bad/Good classifications was implemented as a random selection based on a % of packets in each Snort category (priority 1 for very harmful packets, 2 for less harmful packets, and so on). The classifier can be changed in different experiments and for different networks.

The next aspect to model was “how long should a packet take to bring down a service (compromise/infect a machine)?” In real systems there is an obvious element of uncertainty in this behaviour. For some attack types, if a packet arrives at a machine but its pay-load is not deployed for a long time, there will be no threat for that duration of time. For others, the threat is more imminent. In order to reflect this aspect of reality we have devised a *maximum time to compromise*, and each packet is allocated a random time to compromise within this interval. This part of our model is similar to the models used for simulating viral infection processes reported earlier in the literature (see reference in [4]). The modelled networks were tested against three data sets. A *massive* attack scenario from Swisscom in which all types of services are targeted, a (temporally) modified version of this scenario to represent a *sparse* attack, and a third synthetic attack concentrating on one service type making the attack *systematic*.

C. Recognition and Reaction

The choice of reactive defence mechanism follows the same style as the classification (see above). The defence mechanism has to incorporate a number of rules for recognition, as well as a decision what to do with a harmful packet. Whereas the goal of the first classification (Bad/Good) was to determine the effect of each packet’s arrival at a simulated machine, the idea with the Red/Green classification is to decide what to do with a packet as seen from a defender’s perspective. For the latter purpose, we model a recognition technique that follows the misuse detection implemented by Snort. However, as detecting real attacks is very difficult on a per-packet basis, if all suspicious packets are removed from a network, the chances are that there is a serious reduction in system availability (removing a high proportion of good packets too). That is, a tight set of rules would reject many harmful packets and a few harmless ones, and a less rigid set of rules would do the opposite. We tested two different recognition and removal mechanisms as follows:

- **Mechanism R1**: Remove 100% of packets classified with priority 1, 20% of those with priority 2, 10% of priority 3 and 0 % of priority 4.
- **Mechanism R2**: Remove 100% of packets with priority 1, 70% of those with priority 2, 30% of priority 3 and 0% of priority 4.

D. Recovery mechanism

Every compromised machine will be restarted within a given *maximum recovery time* from the time of infection/compromise for that machine (using a random distribution). In these experiments the idea is to show the dependency of the survivability of a network on the *time to recovery* for each node.

IV. THE SIMULATION TOOL

As a basis for the simulation tool the Swarm platform [8] was used. Swarm was chosen because of its extensive relevant functionality and since it is well known and well documented. Other simulation platforms were also considered but ruled out in early evaluations (for example, the Easel tool was only available on a Macintosh platform when we started the experiments).

Figure 4 shows one output of the tool: a graph showing the value of the service level function. In this case the network has so far survived because the service level function has not reached the value one. The graph indicates that recovery is used since the value sometimes decreases.

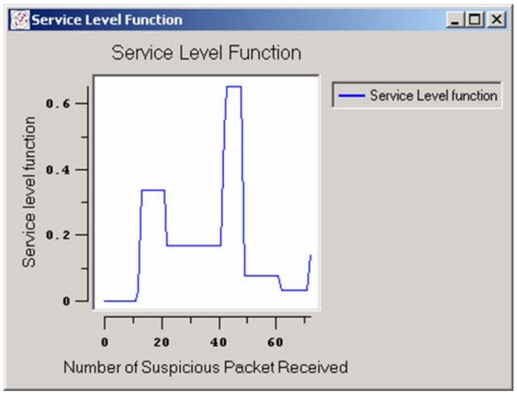

Figure 4: GUI of the tool showing the value of the service level function.

With 108 machines and roughly 160 suspected attacks, memory usage is negligible and the complete time for an experiment is a few seconds using a Pentium 1800 MHz windows workstation

V. Evaluation results

Our work brings insights into the relation between network and traffic characteristics and defence regimes. In particular:

- How do different defence mechanisms compare in increasing the level of survivability? Can various defence types be combined with an added effect?
- How is the network survivability affected by the chosen definition of critical services?
- How can the trade-offs between integrity and availability be used to evaluate the defence architecture?
- How do other factors affect the above analyses, e.g. network size, attack density, maximum recovery time?

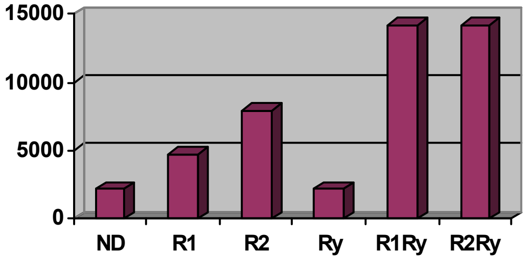

Figure 5: Time to breakdown (in seconds) for different defence mechanisms

This section presents a few results from our experiments, where **Ry** is short for recovery, maximum time to compromise is set to 1500 seconds, and maximum recovery time to 1800 seconds. In the conference presentation other results can be illustrated that are omitted due to space restrictions. Figure 5 shows time to breakdown for different defence mechanisms.

We see that in absence of recovery, R2 is more effective than R1. However, recovery without response is just as bad as no defence (ND). This may be explained by the high rate of infection/compromise without response. There is simply not enough time for machines to recover before the network breakdown.

Further insights can be summarized as follows. (1) Due to the inherent difficulty in distinguishing between harmful and harmless packets, no response is perfect. We pay for our protection with false positives (Good packets classified as Red). R2 gives better protection (but higher false positives) compared to R1. Removing Good packets means decreasing our availability since those lost good packets need to be sent again, increasing communication time, and wasting network resources. (2) The definition of service level is a very important parameter. Accepting a lower service level means that time to breakdown increases. Experiments indicate that to achieve a certain low service level, recovery alone is enough. This can be contrasted with a high service level, for which recovery alone was no better than no defence at all. (3) Comparing the sparse attack with the massive attack using a low service level, it is clear that time is a very important factor. A massive attack quickly causes a breakdown in the network, even when using defence mechanisms that maintained the network survival under the sparse attack. (4) Our base-line experiments indicate that using recovery alone is typically not sufficient to protect the network. However, if recovery is made fast enough, this need not be true. With a maximum recovery time of 300 seconds (low service level, massive attack) the network survives with no additional response (neither R1 nor R2 in place). The full details of these experiments can be found in a longer version of this paper [4]. It further illustrates the effects of network size and the degree of replication on the survival of the network in presence of systematic attacks.

References

[1] S. Nadjm-Tehrani (Ed.), “Interim Report on Survivability Modelling”, April 2002, Safeguard project Deliverable 1. Short version available at http://www.ist-safeguard.org, Acc. 29th November 2002.

[2] Presidential Commission on Critical Infrastructure Protection, The PCCIP Report, October 1997.

[3] B. Ellison, D. A. Fisher, R. C. Linger, H. F. Lipson, T. Longstaff, and N. R. Mead, “Survivable Network Systems: An Emerging Discipline”, Technical Report, Carnegie Mellon University, Software Engineering Institute, CMU/SEI-97-TR-013, November 1997.

[4] K. Burbeck, S. G. Andres, S. Nadjm-Tehrani, M. Semling and T. Daggonier, Evaluation of Defence Mechanisms in Survivable Networks, Technical Report, Linköping University, March 2003, http://www.ist-safeguard.org

[5] M. Wilikens, T. Jackson, “Survivability of Networked Information Systems and Infrastructures”, A State-of-the-art Study, European Joint Research Centre (JRC), Ispra, VA, Italy, 1997.

[6] M. A. Avizienis, J. C. Laprie, B. Randell, “Fundamental Concepts of Dependability”, Technical report, UCLA CSD Report no. 010028, 2001.

[7] SNORT, http://www.snort.org/, Acc. 19th November 2002.

[8] The Swarm Development Group (SDG), http://www.swarm.org/, Acc. 19th November 2002.