Symbiot, Inc.
P.O. Box 9646
Austin, TX 78766-9646

Non-Equilibrium Risk Models in Enterprise Network Security

Paco Nathan, William Hurley

Models derived from biology have proved to be valuable in computer science: neural networks, genetic algorithms, swarms, and to some extent immune systems. Another biological model which is gaining application in computing concerns a category of cybernetics known as autopoietic systems. The application of this theoretical work carries important implications about the role of observers in a control system, adaptation in a running system, and in certain other rigorous notions about cognition. In addition, there is a related body of work in non-equilibrium thermodynamics that provides a mathematical basis for quantitative analysis given those conditions. In this paper we explore the description of the security infrastructure in an enterprise network as an autopoietic system. Building on that, we present indications for a novel form of quantitative analysis for risk aggregation, which we call "non-equilibrium risk models" (NERM). Additional contributors include: Mike Erwin, Jamie Pugh, Frank Milano, Rick Waters, Jim Nasby, Lindsey Simon, Dan Camper.

The Meaning of Risk

Let's start by taking a look at the meaning of *risk*. A good general description[1] of risk is given as a property with two components, *uncertainty* and *exposure*, where risk exists if both are present. The analysis of risk is related the concept of *standard deviation* in statistics. We also need to make a clear distinction between a *measure*, which is a process for assigning a number to some observation, and a *metric*, which is an interpretation of that number. For instance, the method for observing how a 80Gb disk drive has 711Mb space available, that process provides a measure. Determining that the same disk drive is 99.13% full, that interpretation applies a metric. Bringing those definitions together, we can talk about a *risk metric* as a kind of tool used for comparing risk across different situations. Many different kinds of risk metrics exist, for instance *value-at-risk* and *capital-at-risk* are examples of widely used metrics in financial analysis. If you lookup the profile for a stock ticker, its *beta* statistic (or the other "Greeks" similar to it) will be another good example of a risk metric. To recap, we have risk as a property that can be measured in at least a couple of dimensions, and we have risk metrics as a variety of tools for interpreting measurements across different contexts.

The problem of calculating risk metrics may not be simple. In the case of an investment portfolio, it's possible to construct a matrix that describes each of the individual stocks and their trading histories. We apply some linear transforms, manipulate the algebra for a few polynomials, and *voilà!* arrive at, for example, an estimate of how much money stands to be lost in the entire portfolio during the upcoming week of trading. The *value-at-risk* metrics (VaR) work that way. Those calculations tend to use linear methods, requiring particular assumptions. For example, suppose the MSFT stock in your portfolio drops below $3 per share. Does that event prevent you from trading the IBM shares in your portfolio? No. The two issues can be calculated independently. Financial risk metrics[2] make use of methods which are relatively simple to calculate on a computer so long as certain assumptions hold, such as linear independence. The bottom line is that the "capital" in a portfolio is essentially money, and that money could be invested elsewhere.

Stated in the language of systems theory, the elements in a financial portfolio aren't particularly "connected" together. We say that a system is *complex* if some but not all of its elements are connected. In other words, what is the expected number of "hops" between any two elements? A simple system will have a lower number of hops, a complex system will have a higher number. For an illustration, think of the negotiating table for an international treaty where delegates from different countries don't know each other: if the delegates knew each other already, a treaty might be relatively painless to reach, otherwise expect some surprises. The more complex a system is, the more difficult it becomes to model in terms of risk. In the case of financial portfolios, there exist models which aren't relatively as complex as in other fields. Moreover, the risk metrics in most of those models tend to project from "equilibrium" conditions – which is an important point that we'll explore again in just a moment.

What if the capital involved something other than money? Suppose we wanted to calculate risk metrics for the operation of a nuclear power plant – a task which the US Department of Energy spends a lot of resources determining. Building materials for replacing a power plant in the event of some accident might be quantifiable in dollar values, but what about "valuing" the lives of people in a nearby city? Furthermore, the loss of one control system in a nuclear reactor might have cataclysmic effects on the operation of another control system. Unfortunately, the notion of linearly independent equations and so much nifty math developed for financial analysis is no longer applicable. Instead, an analyst must take a careful look at the factors contributing to accidents, and then determine probabilities for how sequences of accidents can lead to risk overall. Here is a point to keep in mind: understanding how sequences of relatively small events may lead to an event of larger consequences[3] is called *risk aggregation.* That represents one of the "holy grail" problems in IT security management.

Risk Models for Enterprise Networks

Let's talk about modeling risk for computer networks. The issues encountered are partly like the financial example and partly like the nuclear power plant. Suppose you have a public-facing network with two web servers, one name server, and one mail server ex-

posed in its DMZ. Suppose the network hosts all the online sales, marketing, and customer support for a catalog business which brings in an average of $25,000 revenue each day. It wouldn't take a math student long to develop a value-at-risk metric to model the risk for operating that network. A good model might require a few more facts, such as relative traffic rates for each server, performance statistics for the upstream providers, and maybe the average value of online purchases, but in any real business those properties are being measured already. You probably also have a spreadsheet with asset values and depreciation schedules for each of those servers (if not, your organization faces an enormous aggregate risk called the IRS, but that's an entirely different problem) and you could add the operating expenses for your IT shop as other line items in that spreadsheet. Those details can be used to develop a capital-at-risk metric for your network. That's one perspective of an enterprise network which fits the assumptions of financial risk modeling.

In practice, computer networks are much more complex than investment portfolios. Suppose your network supports a bank and then one day an attacker breaches security to install some kind of "backdoor" malware. The exposure component of risk in that case is not merely a matter of network service availability, instead there is potential for extortion, loss of customer data, illegal transfer of accounts, etc. Those aspects begin to resemble the risk analysis at a nuclear power plant more so than a stock portfolio. So here is another major point to keep in mind: for networks, we may apply value-at-risk as a kind of preliminary analysis for risk metrics, but we need to consider much more complex models overall for analyzing risk aggregation.

Fortunately, there are some widely available – though perhaps not yet widely practiced – means for complex measures of risk on a computer network. One thing in which networks excel is the production of log files: *iptables*, *snort*, *nagios*, *spamassassin*, *clamav*, to name a few common sources. Commercial and open source solutions[4] exist for using those logs to measure risk. For example, it is possible to model a network (using *auto-discovery* methods) and then correlate events from the log streams (using a SIMS device) to find sequences of events which lead to relative catastrophes. Some anomaly detectors (NBAD) perform a similar kind of analysis to detect zero-day worms: measure the patterns of behavior on a network, and draw comparisons with models for "normal" behavior. We can take that kind of analysis a step further and begin to compare the likelihood for sequences of events on different networks. That's a step toward developing risk measures that could support complex models for risk aggregation.

Security Infrastructures as Complex Systems

Before we can describe risk metrics that can be applied across multiple enterprise networks, we need to look at alternatives for complex models and some of the applicable math for calculating metrics. Any good solution in math starts with a well-defined problem statement… Consider the security infrastructure for an enterprise network, which we'll label ***S***. Let's focus on security for public-facing assets, that is, equipment which touches the DMZ part of the enterprise network. We can say that ***S*** exists as a complex

system within an environment ***E***, which is the TCP/IP space of the public networks. We can also say that ***S*** is distinguished[5] from ***E*** by a topological boundary ***B***, which consists of a set of exposed, filtered services provided on a known set of IP addresses and domain names. Those services are exposed via network traffic exchanged in ***E***.

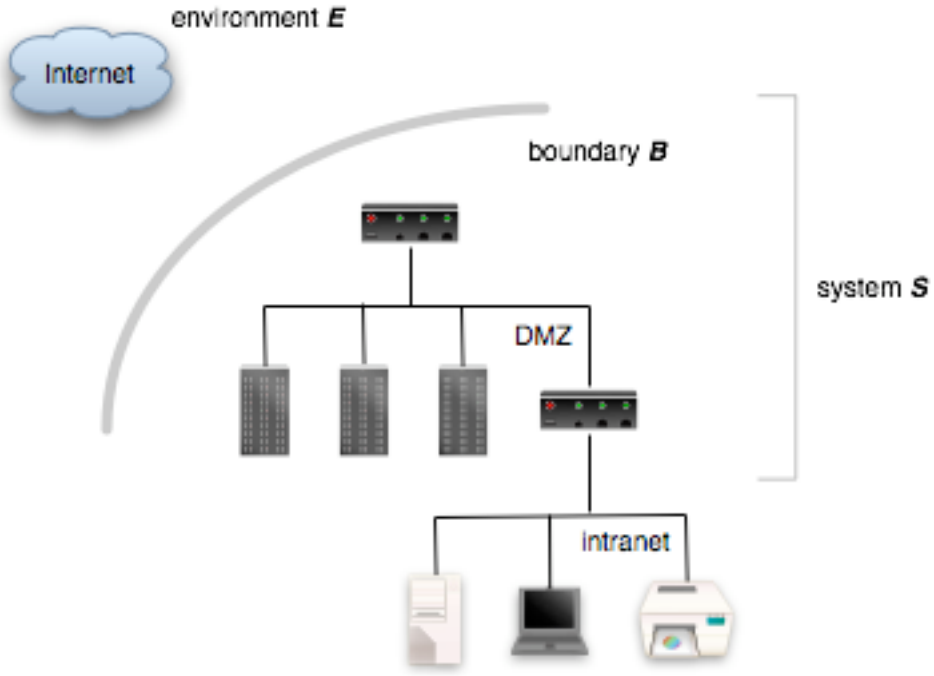

Figure 1 – DMZ for an enterprise network

You might imagine boundary ***B*** as if it were the membrane in an osmosis water filter. On one hand, it may be deformed to some extent, pressured by forces on either side of the membrane, but still function. On the other hand, ultimately it has its breaking points if pushed far enough. Important aspects of those breaking points in the sense of IT security include authentication, quality of service, reliability, data integrity, etc. We'll examine some of those.

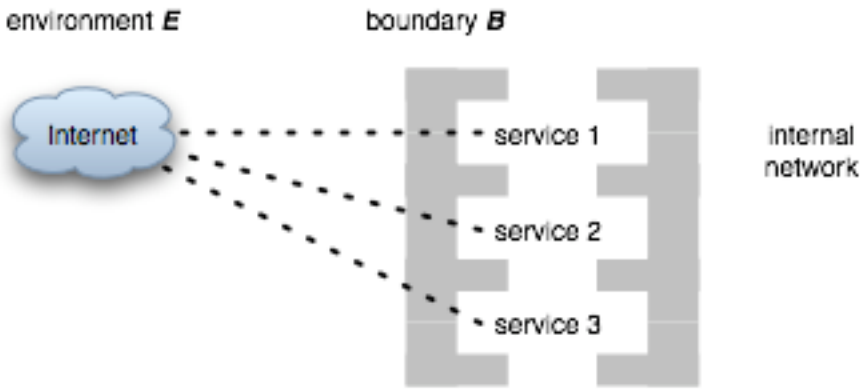

Figure 2 – Logical boundary diagram

We can say that ***S*** incorporates a *community of observers*[6]. In that sense, we can go a step further and say that there are internal regulating processes in ***S*** which produce and protect ***B***.

Taking a broad view of how decisions get made in an enterprise IT operation, the elements of ***S*** which are responsible for regulating and producing ***B*** generally include:

- Network equipment: routers, switches, concentrators
- Hosts and their applications, file systems, and databases
- Security devices: firewalls, proxies, VPN
- System operators
- Executive management decision-makers
- Emergency response team members
- Financial and regulatory auditors

Some elements are hardware appliances and software applications. Those mechanisms can be regulated by configuration rule sets and automated control systems. Other elements are human observers, but they are still considered elements of the overall system. Those individuals will tend to have varying levels of authority for making decisions, and they will contribute in varying degrees to the regulation of ***S***.

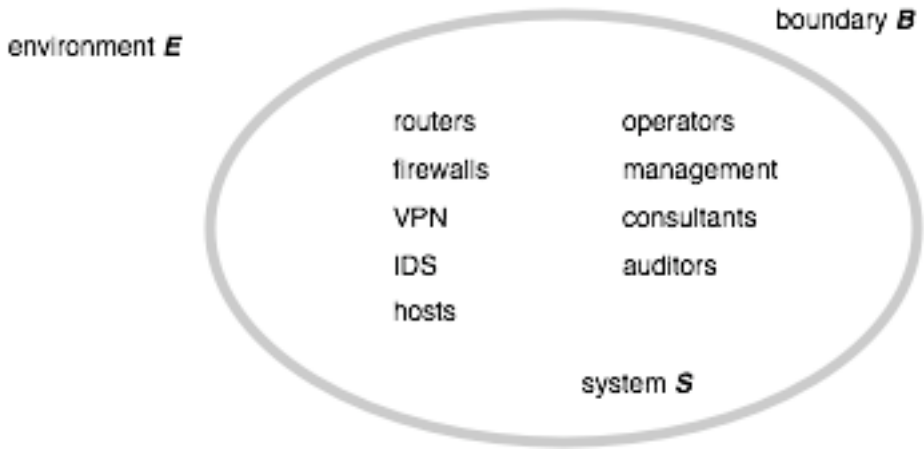

Figure 3 – System, environment, boundary, elements

It's also important to note that the system ***S*** is considered *open* in the sense that it interacts with its environment ***E*** in particular ways. The public services may be filtered, but by definition they are exposed to external influences. Even so, the self-regulation of boundary ***B*** provides some degree of *operational closure* – in other words, it maintains effective network security. That's another key point to keep in mind: effective network security may be modeled as the operational closure in an open, complex system; aggregate risk related to network security exists if that operational closure fails.

Lessons from Autopoiesis

There exists an interesting body of research[7] which seeks to describe the nature of complex systems, pioneered by Humberto Maturana and Francisco Varela. Maturana was one of the early researchers in the field of neural networks – at least, among actual biologists working in that field. Maturana and Varela are noted for developing rigorous definitions in systems theory (*neé* "cybernetics") to describe models of cognition in biology. Their work on *autopoiesis* describes the notion of autonomous or autopoietic systems, as classes of organization.

One interesting aspect of autopoiesis is that Maturana and Varela emphasize the importance of incorporating *observers* as a necessary part of a system – not as external sources of interference, which had been a troubling issue with earlier systems theory. In their formulation, analysis can be extended to make assertions about language, cognition, adaptation, etc. "Anything said is said by an observer"[8], and in the case of security infrastructure, element in ***S*** can be considered an observer. Certainly the human components are, as well as any network device or application that generates an event stream, which is to say almost all of them.

Here we reach a point on which risk modeling for enterprise network security finds some leverage. By definition, an autonomous system has internal processes "related to each other in a network with recursive dependencies", which can be recognized as a "unity".

In the field of enterprise IT, we find related terms[9] used to describe a set of Internet gateways operated by a particular business entity, such that networks have *autonomous system numbers* (ASN) assigned as identifiers. You may have seen ASN listed while running a *traceroute* or *lft* command. In some ways the routing protocols[10] such as BGP-4 put the theory of autonomous systems into widespread practice.

We can establish autonomic qualities for parts of our risk modeling. However, can we argue the point about autopoietic qualities as well? To quote [Whitaker 1995]:

> *The difference between autonomy and autopoiesis is that autopoietic systems must produce their own components in addition to conserving their organization.*

Referring back to our earlier qualitative description of the problem, we have been examining risk models in the context of self-regulation and self-production of boundary ***B*** by the security infrastructure management system ***S***. Plus, we define ***S*** as an open system which, through effective network security procedures, maintains operational closure. In other words, yes.

We'll draw more from systems theory and build this argument further. For now, those properties establish a reasonably good case for modeling enterprise network security infrastructure as autopoietic systems. Leveraging on the work by Maturana, et al., the theoretical work suggests properties, behaviors, and relationships which can be are expected to be observed in autopoietic systems. Those descriptions may be applied to analyze, predict, and reengineer aspects of enterprise network security, particularly the automated control systems. An important corollary is that some degree of adaptation and cognition may be observed and modeled as emergent properties. Overall, that qualitative analysis provides key insights for modeling risk aggregation and developing computable risk metrics.

Dynamics, Adaptation, Catastrophe

We have a description that accounts for the structure of system ***S***. In terms of its dynamics, several feedback loops emerge from interactions between its elements. Consider how some security components (such as IDS, IPS, NBAD) act as internal observers. They monitor network traffic to detect anomalous activity. They generate security events. System administrators also act as observers, monitoring reports of the security event logs or receiving notifications via email, pager, IM, wallboards, etc. In either case, *false positives* among the events[11] make interpretations difficult. Thinking about risk metrics, we can say that security events help provide measures and false positives increase uncertainty.

Administrators apply their expertise with a particular network to discern the more significant events as evidence of potential intrusions. Generally speaking, incidents get reported to managers, who in turn determine policies on behalf of the enterprise and

authorize *countermeasures*[12] to be used in response. Examples of countermeasures include:

- Modifying the configuration of routers, firewalls, etc.
- Tuning the rules used by IDS
- Canceling the customer accounts for potential attackers
- Initiating investigations against identified attackers
- Contacting the upstream providers of sources for recurring attack profiles
- Training system operators to perceive social engineering or hostile situations

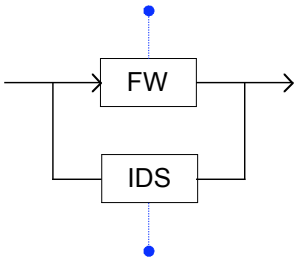

Figure 4 – Security components and structural coupling

Each of those examples represents a dynamic response by ***B*** – some deformation of the boundary. We can say that internal feedback mechanisms exist, for example tracing the data flow in *Figure 5* from firewall to network traffic to IDS to system operator to management to system operator and back to configuration of the firewall. Since the human observers are incorporated with the system, we can certainly say that training for system operators is a deformation of ***B***. In the case of serious incidents, an emergency response team may become involved to provide expert forensic analysis, legal counsel, law enforcement, etc. Periodically, auditors may review the accounting for customer transactions, IT budget, incident reporting and response, etc. Those countermeasures may produce even more substantial deformations.

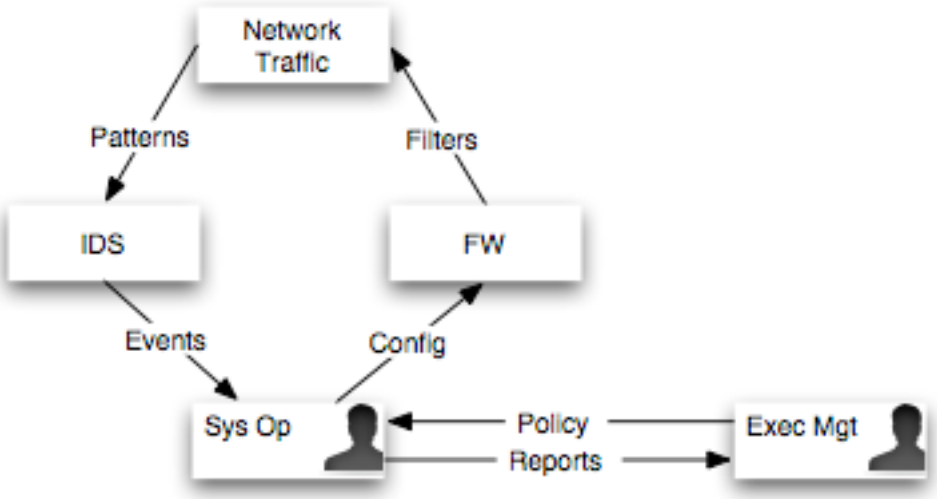

Figure 5 – Internal feedback loops

We can already draw a few conclusions from our qualitative description for a risk model. For instance, we can say that system ***S*** adapts: instances of *structural determination* emerge which alter its dynamics. In fact, multiple instances are occurring at the same time. Whenever system operators apply their local expertise to discern the most significant security events as incidents, they impose a filtering process to disregard less significant events. In that sense, system ***S*** applies *selection* to accept some traffic while rejecting other requests. To the extent that system operators articulate criteria for filtering events – or program an automated component within the security infrastructure to perform that filtering – we can demonstrate adaptation in ***S***.

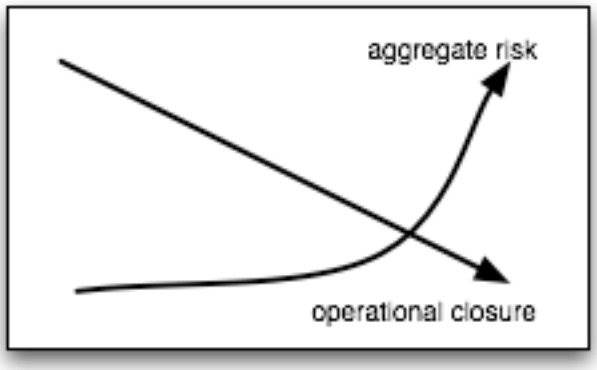

Figure 6 – Catastrophe: closure versus risk

As executive decisions modify network policies and authorize countermeasures, we can say that system ***S*** applies other forms of selection. The available countermeasures may have different relative costs and consequences. New kinds of risk – for example, canceling a valid customer account by mistake – may be introduced, and their risk-benefit trade offs must be considered. However, the end result is that boundary ***B*** becomes

regulated through internal structural determination of system ***S***, so that it responds more effectively to threats in environment ***E***. In that sense, the operational closure of ***S*** may be considered as a *cost optimization* problem.

We can state a general description of risk in enterprise network security as the likelihood that a particular state of ***S*** leads to a catastrophe[13] scenario for the network and organization which it supports. Examples[14] of network catastrophe scenarios include:

- Compromise and illegal access obtained by a remote attacker
- Unintended loss of confidential data
- Covert "rootkit" or backdoor installed
- Misuse as a "spam" relay for unsolicited commercial email
- Appropriation of assets as a "zombie" for attacking elsewhere
- Distributed denial of service
- Infection and subsequent redistribution of malware (worms, viruses)
- Attacks which disable the network's security systems
- Web server defacement

Each of those catastrophe scenarios presents a different sense of deforming ***B***, that is to say some degradation of the operational closure of ***S***. Those catastrophes do not imply the same level of impact to the enterprise network. For instance, a "denial of service" attack could deform ***B*** much less than say having a "rootkit" installed, depending on the context. Stated in terms of risk metrics, the exposure component of risk has a wide range. That aspect adds complexity to the problem of measuring risk.

At this point we can state a systems-theoretic definition[15] for security infrastructure management. Adaptation is put into context as a process of ongoing evolution of threats encountered in ***E***, which drive internal processes of selection in ***S*** to seek better means for producing ***B***. Fitness in that evolutionary context can be described as a condition for maintaining operational closure. We introduced the notion earlier that operational closure equates to effective network security; its degradation implies catastrophe. Now we need to understand more clearly how degradation may be observed and measured.

Equilibrium and Non-Equilibrium

Let's assert that conventional wisdom within the IT security industry tends to frame risk analysis with a bias toward *equilibrium*. Examples include an emphasis in corporate IT management on *best practices* or the ongoing debate[16] over the existence of *return on investment* (ROI). The analytic basis for those kinds of approaches assumes that something is working well and will stay working, ergo the equilibrium point in a model for a system. Likewise, metrics based on value-at-risk or capital-at-risk assume linear extrapolations from some known "good" operational state. That seems unfortunate considering how – as the infamous myth goes – the predecessor of our Internet was designed to withstand a proverbial nuclear war and still maintain *some* routers in operation. In another sense, one of the most essential innovations of internetworking has been to take *non-equilibrium*

conditions (also called *far-from-equilibrium*) into consideration. We're tempted to assert that people who engineer routing protocols might have a bit more insight into the mathematics and nature of complex systems than, say, most people who write articles about IT management; if you don't accept that, try reading some of the literature about QoS algorithms and packet shaping. We make an assertion about non-equilibrium conditions because we're painting a picture here of ***S*** as an autopoietic system, and there happen to be some good, closely related math which may be applied for non-equilibrium conditions. We'll return to that point later.

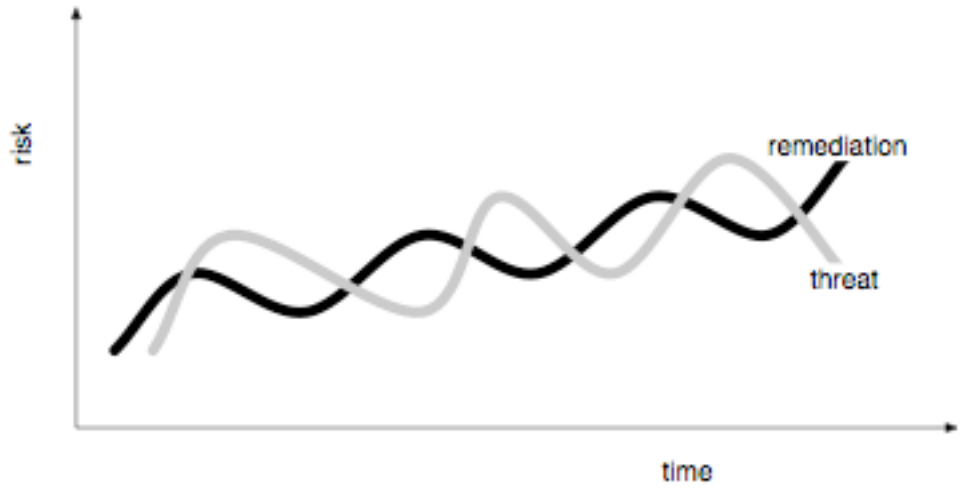

Figure 7 – Co-evolution of threat versus remediation

Looking across the environment ***E*** as a whole, we tend to observe an oscillating escalation[17] of the capabilities of threats versus the effectiveness of remediations. For example, a new worm comes out, and it hits thousands of servers on the public networks using a proverbial *zero-day* exploit. Some enterprise networks may be sufficiently immune but others are not, so the worm propagates and starts to chew up enough bandwidth to capture a lot of attention through the IT industry. Headlines erupt, blogs fill with verbalized annoyance, email lists choke with half-baked ideas, and eventually someone smart analyzes the worm well enough to characterize its behavior, write new IDS signatures that identify it in the wild, and suggest system patches to prevent the thing from propagating so wildly ever again. IT managers wring their hands, people learn from the experience, authors and lawyers gain a wealth of new material, and everybody moves on. That process continues in a kind of cycle. The behaviors of worms evolve, as do the IT practices for contending against them.

Intuitively, the oscillation appears driven by a co-evolution of (1) the complexity and sophistication of attacks in ***E*** and (2) the complexity and sophistication of countermeasures in reaction to those attacks. Taking a global perspective of the aggregate risk involved, we can visualize the pattern of *Figure 7* as a cycle propagating the relation shown in *Figure 6*. In other words, there are "pulses" where the primacy of aggregate risk and operational closure "trade places". To borrow from von Clausewitz, the history of this kind

of warfare may be viewed as an evolving tension between the relative merits of artillery and armor.

Our intent in NERM is to model the "pulses" which punctuate equilibrium, where the level of granularity is an enterprise network. We can apply available analytic approaches to generalize those models across many different enterprise networks, and develop means for calculating non-equilibrium risk metrics. Aside from invasive external practices, such as penetration testing, one area of analysis which appears lacking in IT security would be to have tools that play "Devil's Advocate" to evaluate catastrophe scenarios. In other words, tools that would identify non-equilibrium states in ***S***, anticipating the most likely and effective strategies for attacks which lead into catastrophic states. In military operations this is sometimes called *situational degradation analysis* (SDA) which amounts to planning attacks against one's own strategies and infrastructure. That recalls how risk metrics are applied in the nuclear power plant example.

Metastability

Autopoietic properties[18] of *self-production* and *self-regulation* have been discussed with respect to boundary ***B***, as well as *structural determination*. Other properties include *self-configuration*, which is found in the context of the implementation of countermeasures, and *self-maintenance*, which is the essence of enterprise security itself. Another important property of autopoietic systems concerns *self-reference*, and it can be argued that in the process of implementing *autodiscovery* – or in the course of a financial or regulatory audit – the system ***S*** derives meaning for the significance of its behavior, with respect to itself. Translated from postmodern-speak, those qualities provide a reasonably good argument for self-reference.

The definition of autopoiesis also helps establish a basis for *contextualization*. Again quoting Whitaker:

> *By linking linguistic interaction with structural coupling, the context for signification (determination of meaning) is unified with the context of the interaction. This unification 'grounds' context in the individual's experience, rather than leaving it as a receding horizon of meta-symbolic determinants. This in turn unifies the two senses of 'context'—determinant of linguistic 'meaning' and relevant situational background. This is the strength of autopoietic theory in addressing 'contextualization' in enterprise studies.*

The linkage in our model is that adaptation in ***S*** is contextualized by the evolving threat in ***E***, which is interpreted by the observers' experience. From that we can draw a definition for measuring a sense of *significance* in security events, such that interpretations may be shared among different systems in ***E***. That creates a basis for collaborative risk metrics, which we define as shared interpretations of risk measures based on experience. Keep in mind the point about sharing experiences to develop metrics; some authors[19] have used the term *semantic transports* to describe a related concept. That will come in handy later for developing a quantitative model.

Autopoietic systems theory overlaps to some extent with the definitions for *complex adaptive systems*[20] (CAS). The latter are relatively less rigorous, but worth quoting at this point. [Stacey 1996] presents a set of propositions[21] for analyzing organizational dynamics based on a CAS approach:

> *All organisations are webs of non-linear feedback loops connected to other people and organisations by webs of non-linear feedback loops.*
>
> *Such non-linear feedback systems are capable of operating in states of stable and unstable equilibrium, or in the borders between these states, that is far from equilibrium, in bounded instability at the edge of chaos.*
>
> *All organisations are paradoxes. They are pulled towards stability by the forces of integration, maintenance controls, human desires for security and certainty and adaptation to the environment on the one hand. They are also pulled towards the opposite extreme of unstable equilibrium by the forces of division and decentralization, human desires for excitement and innovation and isolation from the environment.*
>
> *If the organisation gives in to the pull of stability it fails because it becomes ossified and cannot change easily. If it gives in to the pull to instability it disintegrates. Success lies in sustaining an organisation at the border between stability and instability. This is a state of chaos, a difficult to maintain dissipative structure.*

A generalization can be made about how complex systems tend to demonstrate a property called *metastability*. That is to say some tend to operate "best" at non-equilibrium. To illustrate this point, think of a spinning top: so long as it spins well it stays upright, but as it slows it will generally wobble and fall. In the context of our earlier definitions, we can say that metastability is a necessary condition for operational closure, and that it is inversely related to risk.

Biological Models of Aggression

Having described a qualitative model, we can begin to outline means for quantitative analysis. First let's consider some biological models of aggression[22] and then draw comparisons to similar phenomena associated with computer network attacks. [Riznichenko 1999] presents an interesting summary[23] of biophysics models based on non-linear systems of equations for *predator-prey* (Lotka-Volterra) and *propagation of species* (Fisher-Kolmogorov-Petrovskii). Those models assume "the propagation of a species in an active, i.e., rich of energy (food) medium" of unlimited space, the equations are encountered in a more general form as *reaction-diffusion* models in chemistry. *Equation 1* and *Figure 8* both describe a behavior where species x, initially concentrated to the left of domain $r > 0$, propagates into the empty territory according to function $f(x)$, based on diffusion rate D.

$$f(x) = x(1-x)$$

$$\frac{\partial x}{\partial t} = f(x) + D\frac{\partial^2 x}{\partial r^2}$$

Equation 1 – Propagation of species *x*

That math describes biological models, but how can it apply for network security? Let's suppose packets could be exchanged freely from one router to another with no autonomous systems in their path, no filtering, no intrusion detection, and essentially no accountable parties. In that kind of network environment, we would expect to find widespread attacks without remediation. That would engender a situation, as many writers have commented, to the effect that "there is no law on the Internet" – invoking the political context of a libertarian free-for-all, with analogies made to the "Wild West" of the nineteenth century American frontier. Attackers would simply move from one domain of concentration into some other relatively empty territory.

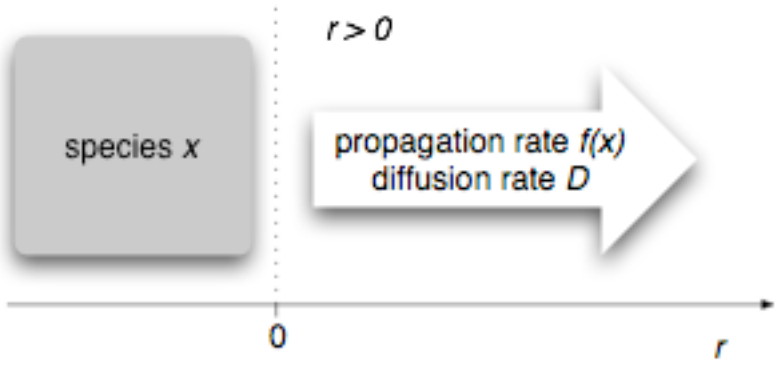

Figure 8 – Species x, domain r > 0, at t = 0

In that case we might expect to find that diffusion models correlate with observed behavior. In fact, it is interesting to compare those predictions against the described phenomena of oscillating escalation (see *Figure 7*) of threat versus remediation over time. The oscillation provides an initial clue that there might not be much of a free-for-all in progress.

Riznichenko, however, goes on to describe that within a limited space, similar underlying processes and conditions result in the emergence of *dissipative structures*[24]. Consider the quote:

> *In linear systems, the diffusion is a process that leads to the equalization of concentrations over the whole reaction volume. However, in the case of nonlinear interaction between the variables* x *and* y*, the instability of homogeneous stationary states can arise, and complex spatio-time regimes form like the autowaves or dissipative structures.*

[Turing 1952] presented the classic work[25] in this area, in terms of chemical *morphogenesis* and the foundations for reaction-diffusion models (see *Equation 2*). Subsequent work[26] by [Thom 1975] explored morphogenesis in the context of *structural stability*, presenting models for "catastrophe theory", extended into many interesting applications. Thom provided remarkable explanations, but he did not develop readily computable models. We reference Thom in the context of developing analysis for catastrophe scenarios in enterprise network security, with an eye toward applying[27] some of his work to risk modeling. That body of work was later superceded[28] by Nobel laureate Ilya Prigogine, who pioneered in the concept of *dissipative structures*. As mentioned, that approach provides a non-linear framework for quantitative modeling. Dissipative structures have been applied for enterprise modeling in other contexts – though to our knowledge not in the context of security infrastructure management. [McCarthy 2000] reports[29] on the analysis of manufacturing organizations in terms of complex systems. That work describes business modeling approaches which seek to create non-equilibrium conditions to affect radical means for adaptation in an organization.

$$\frac{\partial x}{\partial t} = P(x,y) + D_x \frac{\partial^2 x}{\partial r^2}$$
$$\frac{\partial y}{\partial t} = Q(x,y) + D_y \frac{\partial^2 y}{\partial r^2}$$

Equation 2 – An example reaction-diffusion model

Realistically, the "Wild West" scenario for an internetworked environment ***E*** does not hold true in the case of risk aggregation on enterprise networks. Boundaries for gateways are defined by autonomous systems, traffic passing through those boundaries is invariably filtered, and practically all of the defined routes are owned and operated by some enterprise – whether it be a corporation, an educational institution, a governmental agency, a military unit, etc. More to the point, attackers do not quite move into other networks and establish legal dominion – nor do enterprise networks remediate by acquiring the attackers' computers (not typically) – as an equilibrium chemical process for "equalization of concentrations" might imply. In fact, the process of conducting a network-based attack is a rather difficult proposition altogether, on the order of difficulty of building a model ship inside of a bottle.

Stated in another sense, the metastability of a security infrastructure allows for optimal response capability. At equilibrium with its environment, the operational closure required for effective network security of boundary ***B*** would cease to be self-produced. Other indicators include the fact that significant operational costs get spent on security infrastructure – in the financial sense of capital asset depreciation, IT expenditures, corporate overhead, etc., including the time and attention of the human observers. Note that "spending" recalls the description of an "energy rich medium". Moreover, the feedback loops in ***S*** establish non-linear interactions among its observer elements. In short, the complexity of ***S*** allows the enterprise network to dissipate risk over time, and so we look

to dissipative structures to guide the development of non-equilibrium risk models for enterprise network security.

Reaction-Diffusion

The classical notion of morphogenesis uses reaction-diffusion equations to describe the spatial concentrations of two *morphogens* which both diffuse and react with each other. One of these (which we label *a*) is an *activator*. The other (which we label *b*) is called an *inhibitor*. Generally speaking, the activator emerges in barren regions (which Turing called *autocatalysis*), and is consumed to produce the inhibitor. The inhibitor decays naturally, but at an even higher rate where its concentrations are large. Both chemicals diffuse, but the activator normally diffuses at a faster rate than the inhibitor. Recalling the general form of *Equation 1* and *Equation 2*, the function in the first term specifies the reactions between activation and inhibition, which are typically processes of production and decay. The second term is a Laplacian[30] which represents the gradient for diffusion. The variables in those equations are effectively parametric substitutions for ratios of the form *a*/*b* representing the relative concentrations of activator and inhibitor.

Our goal is to develop a quantitative analysis for risk aggregation. In the preceding definitions, we have described properties such as metastability and situational degradation which represent system qualities – by analogy, like the dynamics of a spinning top. We present a working hypothesis: if network-based attacks are considered an emergent property of public networks, then the aggregate situational degradation of networks may be represented as an activator concentration while the aggregate metastability of networks may be represented as an inhibitor concentration. We consider the ratio of metastability to situational degradation to be an estimator for operational closure, such that the concentration ratio *a*/*b* becomes an estimator for aggregate risk.

In our process, we consider the risk measures of security events in a time window, sampling the estimated risk of the interaction between an attacker element a_i and a defender element b_i. Those risk measures allow for a kind of point sampling for mapping the gradient of a field. Using numerical methods based on the system of equations for reaction-diffusion, we can determine an aggregate risk metric. The set $\{b_i\}$ of defender elements belonging to a particular boundary $\boldsymbol{B}$ defines a bundle. Similarly, we can group specific sets $\{a_i\}$ of attacker elements based on upstream analysis and consider them topologically as a bundle. Using those partitions we can resample to determine aggregate risk metrics per attacker or defender.

An Example Case Study

Having worked through the problem definitions, we present an example[31] as a case study and elaborate additional terms and relationships. Suppose that a firm called *Example Networks, Ltd.,* publishes a commercial web site on the Internet for the public sale of widgets. The set of exposed services for their network includes:

- HTTPS for web page content, web services, order forms, etc.
- DNS for announcing the *example.com* domain
- SMTP and instant messaging for customer support

Example Networks places all of their hosts – multiple web servers, load-balanced by a round-robin DNS server, plus a mail server – on a managed switch behind a router and a firewall. The latter three provide some of the filtering for boundary ***B***. Other security devices in the DMZ include an IDS and an email gateway for spam filtering and anti-virus support. Another gateway routes traffic to an internal network which includes database servers, customer authentication, accounting packages, printers, and laptops used by the Example Networks customer support and IT staff – none of which are exposed on the public networks.

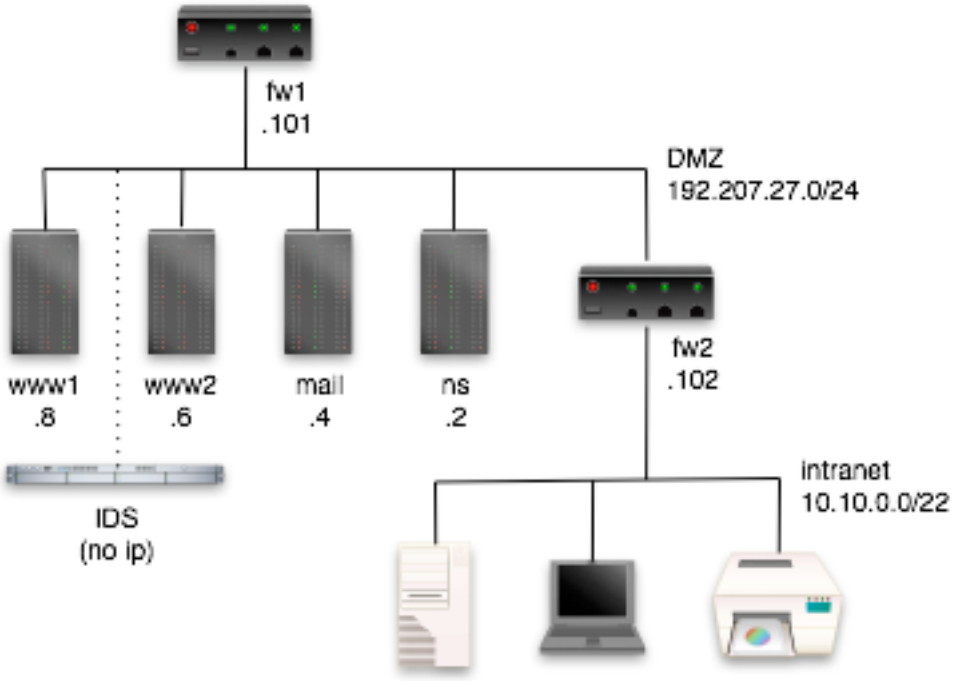

Figure 9 – DMZ network topology for *example.com*

To illustrate a process of catastrophe, one scenario to consider could be when an attacker connects to the host 192.207.27.4 and transacts an SMTP request with the mail server which contains a *buffer overflow* exploit. The intent is to "smash the stack" on the mail server, such that it begins to execute code – probably a call to a shell command interpreter[32] – embedded in the SMTP request. The results are non-deterministic and depend on several factors including the version of the application software running the mail server, the state of its memory stack while the request is parsed, the host's processor architecture, etc. Generally speaking, an attempted exploit needs to be repeated before it succeeds, which increases the opportunities for detection.

When an attempt succeeds, the next step in our hypothetical attack scenario would likely be to send additional exploit code to the obtained shell command interpreter which escalates user privileges, for instance to become the root user. Results for that step are also non-deterministic, depending on the underlying operating system, the shell interpreters it has available, user permissions for the mail server, etc. It may be possible to state statistical descriptions for the likelihood of an exploit succeeding after *N* attempts, the likelihood of an IDS recognizing any of those attempts and notifying a system operator, the likelihood of the mail server application and operating system having vulnerability associated with that exploit, etc. In the case of Example Networks, we can assess risk in terms of measuring factors such as threat (exploit success, detection failure) and vulnerability (applicable to the asset).

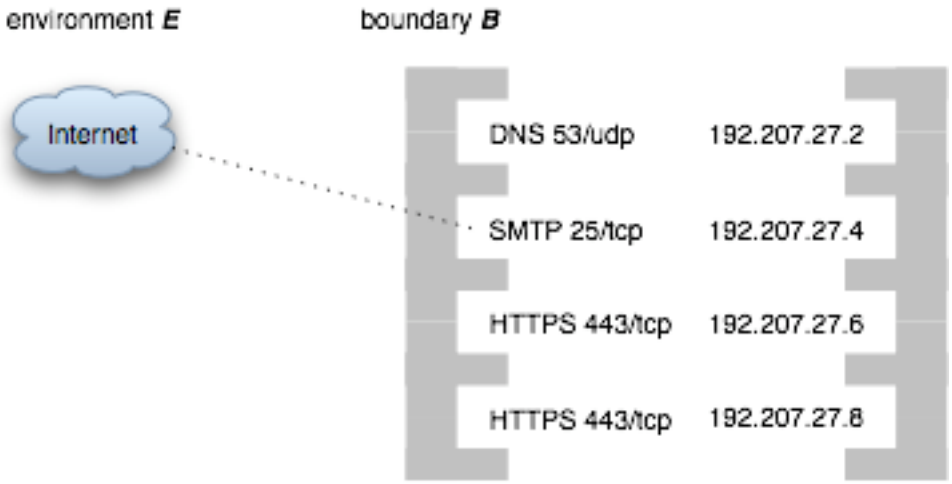

Figure 10 – Logical boundary diagram for *example.com*

In practice, several problems emerge in the assessment of risk for system ***S***. Among the most critical is *data quality*[33]. The streams of security events being aggregated from security components within ***S*** imply several data quality issues. Those streams tend to experience high data rates, such that crucial events may be lost due to transient network errors or client/server synchronization issues. Also, security components in a network infrastructure generally do not all come from the same vendor. For instance, let's say that Example Networks uses Cisco routers, open source *iptables* firewalls, and an open source *Snort* IDS. Using multiple vendors implies potentially different, evolving schema for security events, each of which may change during hardware and software upgrades. In the case of the components mentioned above, each uses a different logging mechanism. That heterogeneity implies consider potential for data loss.

Moreover, there is almost always some likelihood of false positives and false negatives within those event streams. Even if the security event data were all correct, security components themselves introduce latencies which may render events less relevant. For instance, if the IDS for Example Networks takes 10 seconds to identify an incoming worm attack, and the worm tends to infect hosts successfully within 5 seconds, the identi-

fication will arrive for correlation far too late to prevent a security breach. To add to the complexity of the data quality problem, an attack may attempt to use packet fragmentation techniques which disrupt the security event streams or disable the security devices directly.

Another hard problem, related to data quality, is the *autodiscovery* of network assets and services. The point is to construct a network model by using active and passive scanning techniques, and to annotate the model using feedback from the system operators. In our application the autodiscovery of a network model produces an XML document which is created and updated by a Java web application that sequences a set of agents. We can describe that document in more formal terms as a matrix which describes the structure of boundary $\boldsymbol{B}$. In practice the network model has a tree structure since boundary $\boldsymbol{B}$ is a bundle of elements $\{b_i\}$, each of which is a server which in turn represents a bundle of services.

$$\mathbf{B} = \begin{bmatrix} b_1 & b_2 & \dots & b_n \end{bmatrix}$$

Equation 3 – Boundary elements (assets) as a vector

We can also monitor network traffic in the DMZ to construct a triangle matrix of traffic rates transacted among the servers in the model and transacted with the public networks.

$$\text{depend}(\mathbf{B}) = \begin{bmatrix} r_{1x} & r_{21} & \dots & r_{n1} \\ r_{12} & r_{2x} & \dots & r_{n2} \\ \dots & \dots & \dots & \dots \\ r_{1n} & r_{2n} & \dots & r_{nx} \end{bmatrix}$$

where :

$r_{ix} \equiv b_i$ traffic rate externally

$r_{ij} \equiv b_i$ traffic rate from b_j

Equation 4 – Modeling traffic rate dependencies

We transform between that XML document and a spreadsheet[34] to allow the system operators to import and export a view of the network model which describes valuation. The valuation model includes terms for capital asset expenditures and depreciation schedules, IT operating expenses as overhead, revenue tied to the assets, etc.

$$\text{valuation}(b_i, t) = \text{capex}(b_i, t) + \text{opex}(\mathbf{B}, t) / n + \text{revenue}(\mathbf{B}, t) / n$$

Equation 5 – Modeling asset valuation

From these matrices and relations, we can develop estimators for an *impact* model derived from the asset valuations.

$$\text{impact}(\mathbf{B}, t) = \frac{\text{valuation}(\mathbf{B}, t) \cdot \text{depend}(\mathbf{B})}{\left|\text{depend}(\mathbf{B})\right|}$$

Equation 6 – Modeling impact

Another product of autodiscovery is to develop a matrix of vulnerability estimators, such that for a given security event e_{ij} we can estimate the likelihood of the intrusion being applicable for the asset b_i at which it is targeted. The interdependencies of aggregate vulnerability in boundary ***B*** may also be estimated using the traffic rate dependencies.

$$\text{vuln}(b_i \times a_j \rightarrow e_{i,j}) \in [0,1)$$

Equation 7 – Modeling vulnerability

In the case study, Example Networks has only about a half-dozen assets inside its DMZ, and exposes only a few services – which is not a difficult set to discover. In a larger enterprise network, the problem of keeping track of the assets used to produce boundary ***B*** becomes much more complex. Equipment failures, replacements, network outages, software upgrades, configuration changes, and new features development all potentially modify ***B***. Even so, a careful accounting is a prerequisite for risk measurements, particularly for developing vulnerability estimators.

Some types of network security attacks can modify ***B*** structurally, such as denial of service or backdoors. In contrast, if we were talking about risk management in finance, an analogy to the relatively abrupt deformations encountered in network security might be that the securities in an investment portfolio had changed suddenly. A reasonable approach to resolving that problem is to repeat autodiscovery periodically. That is no simple matter, since the operation of different scanning tools can degrade service availability if not managed carefully.

A Pipeline for Staging Analysis

We base our analysis on a generalized SIMS architecture[35] for system ***S***, which is a template used by IT security appliance vendors. Taking the problems listed above into consideration, there are several stages of analyses indicated for building intelligence into the security infrastructure management system ***S***. The data flow between those stages is organized as a kind of pipeline, where most of the analysis is performed locally on the appliance, but there are non-local feedback flows at the later stages.

The first stage of analysis is during the collection of events from the security devices themselves, such as the stream of alerts generated by an IDS. Unfortunately, that stream may be overwhelming for a system operator, and subject to a terribly high rate of false positives.

A second stage of analysis correlates events from security devices and reduces the false positives. Using our case study of Example Networks, system ***S*** implemented as a security appliance could readily discover particular attributes for each of the four servers: IP address, MAC address, host platform (operating system, architecture), uptime, services in operation, applications and their versions, etc. It could also discover from traffic monitoring that the web servers and mail server each depend on some traffic exchanged with the name server, but that the mail server receives relatively few incoming requests compared with the web servers.

From that point, system ***S*** could correlate incoming events to mitigate false positives. For instance, alerts specific to a Microsoft Windows IIS web server could be disregarded if those alerts were associated with traffic destined for a Linux-based mail server.

Table 1 – Stages of analysis

First stage analysis	Streams of events get aggregated from the security devices	Local
Second stage analysis	Security events are correlated to a network model with false-positives reduced	Local
Third stage analysis	Estimators are applied to produce risk metrics for filtering and reporting	Local, consumes non-local feedback
Fourth stage analysis	Non-equilibrium models assess probabilities leading to catastrophe scenarios	Non-local aggregate produces feedback

A third stage of analysis extends the second stage filtering to measure risk and to apply risk metrics. As a security alert arrives, suppose that we have an estimator for the potential threat which it signifies. Assume that we have available patterns of security events which have been modeled for their likelihood of leading to catastrophe scenarios. If an arriving security event fit stochastically within a developing pattern, we can assign an estimator for threat based on that likelihood. Otherwise we can simply use an estimator based on the category of security event.

$$\text{threat}(b_i \times a_j \rightarrow e_{i,j}) \in [0,1)$$

Equation 8 – Modeling threat

We have already determined estimators for the vulnerability of a particular server to different kinds of threats. We also have a model for the impact of the degradation of a particular server. Working from the perspective of an actuarial formula used for insurance risk assessment, and taking into consideration a stream of events within a time window, we can generalize this third stage of analysis as the expected value given in *Equation 9*.

$$\text{E(risk)} = \int \left[\text{threat}(t) \cdot \text{vuln}(t) \cdot \text{impact}(t) \right] dt$$

Equation 9 – Expected value of risk

For example, say that an asset such as the 192.207.27.4 mail server for Example Networks has a valuation – after factoring in asset depreciation and overhead expenses, but no specific revenue attached – of $5000. It encounters a threat of a *sendmail* buffer overflow exploit with a 0.073 probability of significance, which correlates to an estimated 0.0034 vulnerability of effectiveness on a Linux server… that leads to a simple calculation for the expected value of risk:

Table 2 – Third stage analysis for expected value of risk

CVE Identifier	*Threat Estimator*	*Vulnerability Estimator*	*Impact Estimator*	*Expected Value of Risk*
CVE-1999-0047	0.073	0.0034	$5000.00	$1.24

In relation to other security events arriving at that same time, system ***S*** may rank the significance of this event and apply its metrics to evaluate the expected risk of that event leading to a catastrophe scenario. That approach obtains vast improvement over the "off-the-shelf" security devices for managing complex information flows in a security infrastructure. In effect we evaluate the security events $\{e_{i,j}\}$arriving within a time window t as the interaction at boundary ***B*** between some bundle of attackers $\{a_j\}$ and some bundle of defender assets $\{b_i\}$ with estimators for the measures of the expected value of risk associated with that matrix of security incidents.

$$\begin{aligned} b_i \times a_j &\rightarrow e_{i,j} \\ a_j &\in \mathbf{A} \\ b_i &\in \mathbf{B} \\ t &\in [t_0, t_1] \\ \text{E(risk}(\mathbf{A},\mathbf{B},t)) &= \int \left[\text{threat}(\mathbf{A},\mathbf{B},t) \cdot \text{vuln}(\mathbf{B},t) \cdot \text{impact}(\mathbf{B},t) \right] dt \end{aligned}$$

Equation 10 – Risk measure

Even so, this third stage of analysis does not even begin to tackle the problem of modeling risk aggregation. During the course of a day there might be hundreds of attacks detected similar to the buffer overflow exploit described above, none of which actually compromise the network. We cannot merely take a sum of their metrics as an aggregate metric for risk.

We present a fourth stage of analysis to address the hard problem of risk aggregation. Using a modest amount of data mining in the event stream at the third stage, plus a

mechanism for applying *collaborative filters*[36] to event correlation, our research has developed methods for abstracting patterns of events that lead to aggregate risks. Furthermore, we have developed methods for generalizing interpretations of those patterns across multiple enterprise networks participating in a collaborative framework. Those methods provide a process of distributed query, aggregate analysis, and non-linear feedback to the networks participating within that framework. At that point – if we have the math right – our NERM analysis begins to apply a quantitative approach using the reaction-diffusion models cited above.

Summary

We consider the analysis of risk aggregation as a hard problem in security infrastructure management. Given the context of multiple, heterogeneous streams of security events in a complex system of feedback and control, the analysis and interpretation of risk present an unusual challenge. The practice of applying risk metrics helps identify parallels in other fields which have already evolved good methodologies – financial exchanges, critical infrastructure, and routing protocols, to name a few. If costs and risks can be quantified for a particular network, how may those values be compared with risks on other networks? Or on the same network but during a different time period? Those questions articulate the problem of risk metrics, which is to say how may risk factors be measured and interpreted[37] consistently?

In response to those issues, we establish the case for modeling enterprise network security infrastructure as an autopoietic system. Within that class of system, a community of observers is considered an integral part: "Everything said is said by an observer." We can generalize the existing security infrastructure into a larger "unity" which incorporates the people involved with security. In addition, the expected properties of an autopoietic system may be applied to analyze, predict, and reengineer aspects of enterprise network security. Relative degrees of adaptation and cognition are emergent conditions among those properties, and that analysis helps establish a rigorous basis for augmenting the intelligence of security infrastructure. In fact, we present a formal definition for describing how a security infrastructure management system adapts: instances of *structural determination* emerge which alter its internal dynamics. Adaptation is placed into context as the evolution of threats encountered in ***E***, which drive internal processes of selection in ***S*** to seek better means for producing ***B***. Risk metrics can therefore inform the processes of selection and provide a basis for collaboration among distinct systems which are simultaneously adapting to their shared experiences of evolving threats in the public networks. We consider that latter point particularly significant, as evidence of *structural coupling*[38].

Leading into a quantitative analysis, we describe risk in enterprise network security as the likelihood that a particular state of the security infrastructure leads to some catastrophe scenario for the network and the organization which it supports. Risk aggregation is an understanding of how sequences of relatively small events may lead to larger consequences. It is even feasible to compare the likelihood for sequences of events on different networks. As a result, the practice of effective network security may be modeled statisti-

cally as operational closure in an open, complex system; aggregate risk related to network security exists if that operational closure fails.

Notice an important nuance in our approach: the analysis transitions through an arc (1) from contextualized risk, (2) to measurements and estimators used for quantifying factors of risk, and then (3) to metrics used to interpret the significance of security events. In terms of mathematics, we have discussed applying collaborative filters, data mining, statistical estimators, systems of non-linear partial differential equations, and some linear methods as well. The problem space for analyzing catastrophes is large enough and sufficiently complex such that methods other than direct observation and data mining may be required. We anticipate that evolutionary software techniques might be readily applied to the problem of synthesizing how sequences of smaller events lead to catastrophe scenarios.

In terms of complexity, we examine a security infrastructure as having the property of *metastability*: the system is theoretically predicted to operate best at non-equilibrium. We say that metastability is a necessary condition for operational closure. It is inversely related to risk, and allows for optimal response capability. In contrast, at equilibrium with the environment, the operational closure required for effective network security of boundary ***B*** would cease to be self-produced.

In summary, it is the inherent complexity of a security infrastructure – machines and people interacting together – that allows an enterprise network to dissipate risk over time. Consequently, we examine the theory of autopoietic systems and dissipative structures to guide the development of *non-equilibrium risk models* (NERM) for the quantitative analysis of enterprise network security.

Example Network Model

```
<?xml version="1.0" encoding="UTF-8"?>
<NETWORK cidr="192.207.27.0/24" topology="dmz">
  <HOST mac_addr="0060970F86CA" ip_addr="192.207.27.2">
    <PLATFORM osfamily="FreeBSD" osgen="5.X" accuracy="100"
description="FreeBSD 5.2-CURRENT (Jan 2004) on X86" />
    <UPTIME tick="1085501527000" lastboot="Tue May 25 11:12:12 2004" />
    <SERVICE protocol="udp" port="53" name="domain" product="ISC Bind"
version="9.2.2" />
  </HOST>
  <HOST mac_addr="00104B72666A" ip_addr="192.207.27.4">
    <PLATFORM osfamily="Linux" osgen="2.4.X" accuracy="100" description="Linux
Kernel 2.4.0 - 2.5.20" />
    <UPTIME tick="1091470688000" lastboot="Mon Aug  2 13:18:30 2004" />
    <SERVICE protocol="tcp" port="25" name="smtp" product="Sendmail"
version="8.12.10/8.12.9" />
  </HOST>
  <HOST mac_addr="005004997AD4" ip_addr="192.207.27.6">
    <PLATFORM osfamily="Linux" osgen="2.4.X" accuracy="100" description="Linux
Kernel 2.4.0 - 2.5.20" />
    <UPTIME tick="1090510729000" lastboot="Thu Jul 22 10:39:14 2004" />
    <SERVICE protocol="tcp" port="443" name="http" product="Apache httpd"
version="1.3.31" extrainfo="Ben-SSL/1.53 (Debian GNU/Linux)" />
  </HOST>
  <HOST mac_addr="005004982B53" ip_addr="192.207.27.8">
    <PLATFORM osfamily="Linux" osgen="2.4.X" accuracy="100" description="Linux
Kernel 2.4.0 - 2.5.20" />
    <UPTIME tick="1097615315000" lastboot="Tue Oct 12 16:09:02 2004" />
    <SERVICE protocol="tcp" port="443" name="http" product="Apache httpd"
version="1.3.31" extrainfo="Ben-SSL/1.53 (Debian GNU/Linux)" />
  </HOST>
  <HOST mac_addr="009027C5B833" ip_addr="192.207.27.101">
    <PLATFORM osfamily="Linux" osgen="2.4.X" accuracy="100" description="Linux
2.4.18 - 2.6.4 (x86)" />
    <FIREWALL type="iptables" enabled="true" />
  </HOST>
  <HOST mac_addr="00C0F017FB81" ip_addr="192.207.27.102">
    <PLATFORM osfamily="Linux" osgen="2.4.X" accuracy="100" description="Linux
2.4.18 - 2.6.4 (x86)" />
    <FIREWALL type="iptables" enabled="true" />
    <SERVICE protocol="tcp" port="22" name="ssh" product="OpenSSH"
version="3.7.1p2" extrainfo="protocol 2.0" />
    <SERVICE protocol="tcp" port="2601" name="quagga" product="Quagga routing
software" version="0.96.4" extrainfo="Derivative of GNU Zebra" />
  </HOST>
  <HOST mac_addr="000F348126B2" ip_addr="192.207.27.103">
    <PLATFORM osfamily="embedded" accuracy="100" description="Cisco Catalyst
switch" />
    <SERVICE protocol="tcp" port="22" name="ssh" product="Cisco SSH"
version="1.25" extrainfo="protocol 1.5" />
  </HOST>
</NETWORK>
```

Bibliography

[AlephOne 1996] "Aleph One" (pseudonym): "Smashing The Stack For Fun And Profit", *Phrack*, Vol. 7, Issue 49, 08 Nov 1996.
http://www.phrack.org/show.php?p=49&a=14

[Bonabeau 2004] Bonabeau, E.: "Evolving The Bad Guy", *Emerging Technology Conference*, O'Reilly, 10 Feb 2004.
http://conferences.oreillynet.com/cs/et2004/view/e_sess/4847

[Chislenko 1997] Chislenko, A.: *Automated Collaborative Filtering and Semantic Transports*, 15 Oct 1997, (v2).
http://www.ethologic.com/sasha/articles/ACF.html

[Cohen 2004] Cohen, F.: *Risk Aggregation: The Unintended Consequence*, Burton Group, 27 Apr 2004, (v1).

[Gordon 2004] Gordon, L.A, Richardson, R.: "InfoSec Economics", *Security Pipeline*, 15 Apr 2004.
http://nwc.securitypipeline.com/howto/18901529

[Guldimann 1996] Guldimann, T., et al.: *RiskMetrics – Technical Document*, Morgan Guaranty Trust Company of New York, Dec 1996 (Fourth Edition).
http://www.riskmetrics.com/rmcovv.html

[Holland 1996] Holland, J.: *Hidden Order: How Adaptation Builds Complexity*, Perseus, 1996.
http://www.santafe.edu/projects/echo/echo.html

[Holton 1996] Holton, G.: *Contingency Analysis*, 1996.
http://www.riskglossary.com/

[Luhmann 1995] Luhmann, N.: *Social Systems*, Stanford University Press, 1995.

[McCarthy 2000] McCarthy, I.P., Rakotobe-Joel, T., Frizelle, G.: "Complex systems theory: implications and promises for manufacturing organizations", *Int. J. Manufacturing Technology and Management*, Vol. 2, Nos. 1–7, pp. 559–579, 2000.
http://www.ossu.co.uk/Papers/IJTM-Complex%20Systems%202000.pdf

[Maturana 1980] Maturana, H.R., Varela, F.J.: *Autopoiesis and Cognition: The Realization of the Living*, Boston Studies in the Philosophy of Science [Cohen, R.S., Wartofsky, M.W. (eds.)], Vol. 42, Reidel, 1980.
http://web.matriztica.org/555/propertyvalue-6121.html

[Nathan 2003] Nathan, P.X., et al.: *A Trajectory for the Evolution of SIMS Architecture*, Symbiot, 19 Dec 2003 (v3).
http://www.symbiot.com/media/SIMS_Evolution.pdf

[Nathan 2004a] Nathan, P.X., Erwin, M.W., et al.: *On the Rules of Engagement for Information Warfare*, Symbiot, 04 Mar 2004 (v1).
http://www.symbiot.com/media/iwROE.pdf

[Nathan 2004b] Nathan, P.X., Hurley, W., et al.: *Apache Project Proposal: OpenSIMS – A Proposed Framework for Collaborative Risk Metrics*, Symbiot, 28 Jul 2004.
http://opensims.org/rm_proposal.pdf

[Nathan 2004c] Nathan, P.X.: "What 'Countermeasures' Really Means", *O'Reilly Network*, O'Reilly, 03 Aug 2004.
http://www.onlamp.com/pub/a/security/2004/08/03/symbiot.html

[Olson 2002] Olson, J.E.: *Data Quality: The Accuracy Dimension*, Morgan Kaufmann, Dec 2002.
http://www.evokesoft.com/events.htm

[Oram 2004] Oram, A.: "Symbiot on the Rules of Engagement", *ONLamp*, O'Reilly, 10 Mar 2004.
http://www.onlamp.com/pub/a/security/2004/03/10/symbiot.html

[Ozier 2003] Ozier, W.: "Risk Metrics Needed for IT Security", *ITAudit*, Vol. 6, 1 Apr 2003.
http://www.theiia.org/itaudit/index.cfm?fuseaction=forum&fid=5396

[Prigogine 1967] Prigogine, I.: "Dissipative structures in chemical systems", *Fast Reactions and Primary Processes in Chemical Kinetics* [Claesson, S. (ed.)], Interscience, 1967.

[Rekhter 1995] Rekhter, Y., Li, T.: *RFC 1771: A Border Gateway Protocol 4 (BGP-4)*, March 1995.
http://www.faqs.org/rfcs/rfc1771.html

[Riznichenko 1999] Riznichenko, G.: "Mathematical Models in Biophysics", *Computational and Theoretical Biophysics*, Biophysics Textbook Online [Beard, D.A., ed.], 1999.
http://www.biophysics.org/btol/compute.html

[Rosen 1982] Rosen, E.C.: *RFC 827: Exterior Gateway Protocol (EGP)*, Bolt Beranek and Newman Inc., Oct 1982.
http://www.ietf.org/rfc/rfc827.txt

[Scott 2002] Scott, S. J.: *Threat Management Systems: The State of Intrusion Detection*, 09 Aug 2002.
http://www.snort.org/docs/threatmanagement.pdf

[Stacey 1996] Stacey, R.: *Strategic Management and Organization Dynamics*, Pitman, 1996 (2nd ed).

[Teubner 1997] Teubner, G.: "Can Social Systems be Viewed as Autopoietic?", London School of Economics, Complexity Study Group, 18 Jun 1997.
http://bprc.warwick.ac.uk/lsesg3.html

[Thom 1975] Thom, R.: *Structural Stability and Morphogenesis*, W. A. Benjamin, 1975.

[Turing 1952] Turing, A.M.: "The Chemical Basis of Morphogenesis", *Philosophical Transactions of the Royal Society B*, 237, 37-72, 1952.

[Weisstein 1999] Weisstein, E.W.: *MathWorld* – A Wolfram Web Resource, 1999.
http://mathworld.wolfram.com/

[Whitaker 1995] Whitaker, R.A.: *Self-Organization, Autopoiesis, and Enterprise*, Association of Computing Machinery SIGOIS, 1995.
http://www.acm.org/sigois/auto/Main.html

[Winograd 1986] Winograd, T., Flores, F.: *Understanding Computers and Cognition*, Addison-Wesley, 1986.
http://hci.stanford.edu/~winograd/

Endnotes

[1] See [Holton 1996] for related descriptions, specifically starting at: http://www.riskglossary.com/articles/risk.htm

[2] See [Guldimann 1996] for applications of risk metrics in the financial sector.

[3] See [Cohen 2004] for criticism of risk aggregation analysis in the network security industry.

[4] See [Nathan 2004c] and also the OpenSIMS project site at: http://opensims.org/

[5] [Nathan 2003].

[6] See [Nathan 2004c] for a discussion of the human actors in this system.

[7] See [Maturana 1980] for the primary source. While its original focus was an rigorous examination of the biological basis for language and cognition, that body of work has subsequently been applied to social systems by the more controversial [Luhmann 1995] and [Teubner 1997]. Excellent commentary and additional readings can be found at [Whitaker 1995], who applied that work to enterprise networks. See [Winograd 1986] for another application to computer networks (early "groupware") as well as background on the philosophical context. It is interesting to note that Maturana, as a graduate student, worked on one of the original "neural networks" research teams with McCulloch and Pitts.

[8] [Maturana 1980], p. 8.

[9] See [Rosen 1982] for the RFC that describes autonomous systems as a future expectation for the evolution of the Internet. That work has since evolved into the *border gateway protocol* (BGP) – see [Rekhter 1995] for the most recent specification.

[10] See [Rekhter 1995] for discussion of BGP-4 protocol and performance metrics.

[11] See [Gordon 2004] for a quote from the CSO at Oracle, regarding their experience with an IPS at a data center that was generating 60-70% false positives.

[12] See [Oram 2004] and subsequent discussion in [Nathan 2004c].

[13] In general, see [Thom 1975] for the canonical treatment of *catastrophe theory* and related issues of structural stability. The related mathematical modeling has been superceded by the work of [Prigogine 1967].

[14] Developed in private discussions with Jamie Pugh during 2002-2004.

[15] See [Whitaker 1995] for a study of Maturana's views on *languaging* and *contextualization*, with respect to developing an autopoietic definition for *signification*.

[16] [Gordon 2004] and [Ozier 2003] both make compelling economic arguments against using ROI for security. In contrast, the analysis of *return on security investment* (ROSI) has been a central tenet of some security experts such as @Stake, Inc.

[17] Developed in private conversations with Mike Erwin during 2002-2004, and presented in [Nathan 2003]. This phenomena fits well with the analysis of complex systems found in [Bonabeau 2004], in the sense of threat co-evolving with response.

[18] See [Whitaker 1995] for a succinct definition of these terms.

[19] See [Chislenko 1997] for the primary source, and [Nathan 2004b] for an overview of the *Apache OpenSIMS* project which applies these methods. A related end result was obtained by [Guldimann 1996].

[20] Perhaps too many authors write about "complex adaptive systems" to be able to cite a definitive source. Plus, most of those descriptions descend into the morass of *Wired*-esque pop-culture hagiographies for "chaos". In contrast, perhaps one of the best presentations comes through [Holland 1996] who did the pioneering work on genetic algorithms.

[21] [Luhmann 1995] builds a much more substantive case for complex systems emerging from group dynamics, as does [Teubner 1997] for applications of autopoiesis in law. This analysis comes from [Stacey 1996], quoted in *Effecting Change in Higher Education*,
http://www.effectingchange.luton.ac.uk/approaches_to_change/index.php?content=complexitytheory

[22] See [Luhmann 1995], pp. 357-404. The author explores a sociological framework for analyzing conflict in the context of autopoietic systems, building comparisons to immune systems, and in some sense leading to a notion of risk aggregation.

[23] See [Riznichenko 1999], pp. 29-34, leading into [Turing 1952], and also discussed in [Weisstein 1999] http://mathworld.wolfram.com/Lotka-VolterraEquations.html

[24] See [Prigogine 1967] for the primary source on *dissipative structures*. Note that some Russian authors use an alternative description as *autowaves*.

[25] [Turing 1952].

[26] [Thom 1975].

[27] In particular, note the discussions in [Thom 1975], pp. 297-ff, regarding the predation loop and capture morphology.

[28] [Prigogine 1967].

[29] See [McCarthy 2000], pp. 562, 573-ff.

[30] See [Weisstein 1999] http://mathworld.wolfram.com/Laplacian.html and http://mathworld.wolfram.com/Gradient.html

[31] Kudos to Outer.net for allowing the use of their IP address space in this example.

[32] See [AlephOne 1996] for the essential tutorial on constructing buffer overflow exploits, including source code. This document provides the basis for exploits found in the wild, some of which almost use the code verbatim.

[33] See [Olson 2002] which develops a compelling case for data quality issues and methodologies. Those key points are overlooked far too frequently in IT infrastructure, in general, and rarely considered at all with respect to information security, in particular.

[34] Developed in private conversations with Frank Milano during 2003-2004.

[35] The whitepaper by [Scott 2002] describes an idealized architecture common to several vendors, which arguably may be considered an application of the *OODA Loop* decision framework by COL John R. Boyd, which predates it considerably. http://www.mindsim.com/MindSim/Corporate/OODA.html

[36] See [Chislenko 1997] for an overview of the subject.

[37] To quote [Ozier 2003]: "The lack of formalized qualitative and quantitative risk metrics impairs the ability of risk managers and security professionals to effectively and consistently measure risk and points to the absence of a sound framework against which to record quantitative threat-experience data. Establishing a risk-management framework and risk metrics would greatly improve risk management by giving organizations a basis for risk analysis and assessment that would enable them to make business decisions about managing security risks."

[38] See [Whitaker 1995] for a discussion of systems-theoretic nuances that distinguish between *structural determination* and *structural coupling*. In the primary source, these definitions derived from descriptions of neuron physiology. NB: In the earlier [Nathan 2003] paper, we did not distinguish those terms specifically – a situation hopefully clarified in this paper. Original descriptions of autopoiesis by similarly did not distinguish, as in the case of definitions given by [Maturana 1980], p. 136.