Emergent Properties & Security: The Complexity of Security as a Science

Nathaniel Husted
School of Informatics and Computing
Indiana University

Steven Myers
School of Informatics and Computing
Indiana University

ABSTRACT

The notion of emergent properties is becoming common place in the physical and social sciences, with applications in physics, chemistry, biology, medicine, economics, and sociology. Unfortunately, little attention has been given to the discussion of emergence in the realm of computer security, from either the attack or defense perspectives, despite there being examples of such attacks and defenses. We review the concept of emergence, discuss it in the context of computer security, argue that understanding such concepts is essential for securing our current and future systems, give examples of current attacks and defenses that make use of such concepts, and discuss the tools currently available to understand this field. We conclude by arguing that more focus needs to be given to the emergent perspective in information security, especially as we move forward to the Internet of Things and a world full of cyber-physical systems, as we believe many future attacks will make use of such ideas and defenses will require such insights.

1. INTRODUCTION

Emergence is all around us. When a large number of agents act, even seemingly independently, it is possible that their collective behavior results in behavior or a property that is not obvious from the analysis of any particular agent [47]. Such properties or behaviors are known as emergent. For example consider an individual that is not informed about the nature of a modern botnet nor their applications to DDoS (Distributed Denial of Service) attacks. This individual, upon inspection of an individual bot, would probably not consider the bot's ability to direct some limited amount of traffic at a target site particularly odd or menacing. Had the source-code for such a bot been analyzed before there had ever been an actual DDoS attack, the malicious threat might have been laughed off. The potential for the attack's denial of service only emerges when a large number of bots are acting in unison. The DDoS attack can be seen as a *very simple* form of emergence.

NSPW'14, September 15–18, 2014, Victoria, BC, Canada.

http://dx.doi.org/10.1145/2683467.2683468.

Emergent properties are now well studied in a large number of scientific and social scientific fields, spanning from from biology to sociology. Examples here include ant colonies that are seen as having properties that are independent of individual ants to human culture [31]. While there have been some calls for its study by academics in the popular press [27], the academic study of emergence in information security has mostly been restricted to isolated cases, rarely with a view on reflecting on the broader picture of the possibilities of emergent vulnerabilities, attacks, defenses and diagnostics.

Recently there has been much interest in the science of security [55]. The goal of the science of security is to develop information security to the point where it should be possible to design information systems that are secure in understandable, definable ways, against pre-specified adversaries with clear means and abilities. That is, given a relatively well defined adversary, we should be able to design information systems that have clearly defined security properties with respect to the adversary. In order for this scientific program to become a reality, it is necessary to understand the reasonable ways that an adversary or a system can behave. Traditionally, some of the hardest behaviors to understand and model are those composed of a large number of independent systems or agents interacting through a complex environment. In a world where computation is cheap, and everything is networked (such as the world of ubiquitous computing and the Internet of Things – IoT), it is reasonable to assume the adversary will have at its disposal large numbers of agents: in fact, modern botnets are early incarnations of such adversaries. That these agents can and will communicate, and interact with their environment is a given. Thus we need to study emergent attacks, and understand their capabilities both offensively and defensively, as part of the development of such a science.

Unfortunately, the study of emergent properties in security has been more of a byproduct of ad hoc approaches studying specific attacks (e.g., worms and virus propagation [10, 11], or DDoS attacks [78]) , or in attempts to make specific defensive systems such (e.g., Forrest's work on Intrusion Detection Networks or Social/crowdsourced detection of phishing emails) [28, 79, 57]. Emergence has only received a brief nod in early classifications of computer faults [4], or the previously mentioned popular press article [27]. The research itself is not framed under the focus of emergent phenomenon and emergence is not a central topic of these works. Historically this approach has worked well enough. Up until the modern mobile revolution combined with the Internet

of Things, there were few opportunities for novel emergent behavior to manifest. There are a few exceptions, for example, the work of Pastor-Satorras and Vespignani [58] showed why there was no herd immunity from computer viruses on the Internet, unlike the situation with human populations and traditional vaccines.

Individuals in first world societies now have mobile devices in every pocket. We expect to shortly see network connected smart homes, cars that communicate with other cars, and many other devices connected to the Internet of Things. Thus there are significantly more interactions occurring between many computationally independent agents and devices. These interactions allow for the creation of emergent phenomenon, be they new attacks, vulnerabilities, or defenses. Having diagnostic tools to understand these phenomena will be critical.

1.1 Related Work

Emergent properties are widely discussed in many scientific fields' literature. There are now a number of textbooks dealing with emergence under the auspices of Complex Systems [49, 19, 39], and for specific fields (e.g., Social Science, [24, 26], Artificial Life [8], Urban planning [7]). A number of papers in the field of complex systems and physics have also attempted to flesh out the concepts of emergence [20, 62]. In Computer Science, emergent properties are looked at in the study of networks [6]. In Informatics many social networks are studied [80, 2]. Computer scientists have also used other emergent biological systems as inspiration for computing algorithms. This includes genetic algorithms [35], ant colonies [23], and the human brain [40].

The field of computer security has looked at tracking networks [44], Distributed Denial of Service (DDoS) attacks [56], and mobile-to-mobile worms and viruses [44, 10, 82], all of which tend to exhibit elements of emergence. In fact, the word "chaotic" (a common characteristic of emergent systems) is used to describe DDoS traffic patterns, though little work has been done to prove this is an appropriate descriptor [13]. Work has also been done attempting to use the human immune system, itself an emergent system, as the inspiration for antivirus detection [29, 85]. There have been attempts to use epidemiological techniques to model the spread of different forms of malware, and the models of natural biological viral spreads are known to have complex behavior [36]. Many early worms modeled have been simple enough that these dynamics have yet to be seen [86], due to internet network topology and spreading strategy. Local networking and more complex propagation methods are leading to more complex behavior. Wireless worm analysis including 802.11 devices [45, 42], bluetooth devices [10], and human mobility have shown more interesting behaviors.

Systems assurance, especially for critical systems, are often considered part of modern information security. There has been some work using concepts of complex systems in the design and/or analysis of critical systems systems in the assurance community [48]. The focus is on using network analytics to show how critical systems' infrastructures deteriorate over time. The system is modeled as a network, and deterioration based on "node" removal is measured, using techniques based on flow deterioration on complex networks [3]. The work helps determine which component failures are the most dangerous. Critical Systems engineers have also considered safety and security to be emergent properties but their work does not focus on how to measure the risk or threat caused by emergent security phenomenon in any specific system [52].

1.2 Roadmap

We begin with some background on the concept of emergence in Sec. 2. We follow, in Sec. 3, with a definition of the emergent domain, some example problems whose central theme is emergence, and discussion of how the emergent domain manifests within the field of information security. In Sec. 4), a limited set of methodologies and tools that have historically been used to handle problems associated with emergence are presented. Finally, in Sec. 5 we offer some thoughts integrating these methodologies with modern risk/threat analysis and how we will need to mitigate emergent phenomenon.

2. EMERGENCE

Intuitively, emergence is a phenomenon where a system's behavior as a whole is *novel* compared to the behavior of its parts. The canonical example is that of human consciousness being hard to understand or predict by studying individual neurons.

Emergent properties are studied in many branches of science from natural to social [25, 73, 34]. Thus, it is somewhat surprising that a solid and well accepted quantitative definition has yet to been found. Rather, "I know it when I see it", as U.S. Supreme Court Justice Potter Stewart made famous [70], seems to be the prevailing definitional norm. Therefore, while we believe that such definitions would be immensely useful, we are not aware of any that are satisfactory, and our own attempts at formal definitions have been similarly flawed. In fact while there are large groups that agree emergence exists, there is still significant quarreling about what constitutes emergence, or the differing degrees to which emergence is achieved or measured. Our goal is not to settle this quarrel but to find a definition that works well in aiding security practitioners and researchers in understanding how emergence manifests in the security phenomenon they face.

That being said work by Bar-Yam [5], Wolfram [81], and Fromm [31], have created encompassing taxonomies or spectrums of emergent behavior. Heavily mathematical takes are presented by Cucker and Smale[17] and Crutchfield[16]. We present ideas strongly based on Fromm's spectrum of emergence, where Fromm's spectrum is closely based on Bar-Yam's hierarchy for the rest of this work. Fromm provides an easily accessible description of differing levels of emergence without becoming overly formal or technical, making it accessible to those new to the topic. Yet he still makes the semantic differences between types of emergence clear. As we stated earlier, there are no robust, well accepted definitions of emergence yet, so any choice is problematic from some perspective. In particular, we believe that for studying problems we will introduce, the more mathematical definitions will likely provide more technical benefit.

Fromm presents four key distinct points on the spectrum of emergence, with some subdivisions between some of these distinct points. Where a system lies on the spectrum is largely a function of the amount of feedback it gets from itself and other systems or the environment. As one progresses across the spectrum one sees increased difficulty and

expense in predicting the macro behavior of the system from the micro properties of the agents.

2.1 Types of Emergence

2.1.1 Type I: Purposeful Interaction

Type I is the simplest form of emergence. It is characterized by simple, intentional, and designed interactions between components of a system. Type I is split into subcategories of direct and indirect emergence.[1] Direct emergence is when there is a single system, composed of many distinct and differentiable parts, where the function and properties of the system are directly designed. For example a typical program, machine or appliance. Direct, Type I emergence is at the beginning of the spectrum, and is where traditional security mechanisms fall.

For our purposes, emergence becomes interesting as an unstudied area of security with indirect emergence, and higher levels on the spectrum. Indirect emergence is a set of (statistical) properties that emerge from a collection of near identical agents. The bandwidth consumed by a botnet DDoS attack is an example of indirect emergence. Similarly, herd immunity of non-immunized individuals (or systems) in an immunized population is another example.

2.1.2 Type II: Single Source Feedback

Consider a Multi-Agent Systems (MAS) where each agent receives feedback, through independent interactions, from other agents in the same system. Type II typifies the basic notion of emergence where many similar entities interact to create some seemingly higher level organization. These systems are characterized by only having positive *or* negative feedback. A prototypical example is a school of fish: Each fish makes its own decisions, but this behavior can eventually lead to the formation of a school of fish. The school is a system that seems to have its own properties that are hard to determine from direct observation of a fish's individual behavior. The forms of organization that this type of emergence take are far less predictable than that of Type I emergence. In a Type II system the interaction is either *direct*, between the agents, or *indirect*, somehow communicated through the environment. The schooling of fish is an example of direct interaction, while ants use pheromone paths that are deposited and sensed on the ground. Pheromones are an example of indirect communication.

2.1.3 Type III: Multi Source Feedback

Type III expands upon the MAS of Type II emergence to include not only feedback between agents, but also multiple types of feedback caused by interactions on different time scales. Interactions on different time scales allows us to have systems whose emergent behavior contains ebbs and flows as well as chaotic phenomena [81]. Consider financial markets: positive pricing feedback due to herd behavior can cause an asset class (e.g., stocks, houses or tulip bulbs) to reach very high valuations that has no relation to the class' intrinsic value. This is colloquially referred to as a bubble. Eventually the bubble bursts when some other external negative feedback prevails over the positive feedback. The bursting of these bubbles is extremely difficult (or intractable) to predict [65]. Eventually the asset class' prices stabilize again, and will rise when the positive feedback overcomes the negative feedback.

Evolutionary systems can also be considered Type III. In these cases small perturbations, even if marginally negative to a species' survival, can eventually lead to large changes in the system. For example if there is a change in the environment which makes negative changes suddenly beneficial or if the small changes finally interact to create a large benefit under the current environment's fitness function. The best example of such phenomenon is the "cat and mouse" cycle of antivirus protection or spam detection. A number of spam detection techniques are implemented until one is found to work well. This technique then spurs innovation on behalf of the spam senders who have a sudden "catastrophe" effect their revenue generating mechanisms. Type III systems will have both positive *and* negative feedback (differentiating themselves from Type II which have one or the other) and possibly evolutionary components as well.

2.1.4 Type IV: Strong Emergence

Type IV emergence is also referred to as strong emergence. Strong emergence involves feedback between systems of systems which in and of themselves contain many different types of feedback and interactions. Strong emergence cannot be tractably predicted. Frequently, it is argued that prediction would require a model as complex as the original phenomenon [30]. Examples include life emerging from base chemical reactions, consciousness emerging from the brain, or societal culture emerging from groupings of people. In general, Type IV emergence contains some of the largest open questions in science. We note that Type IV emergence has effects on security, as we can view differing societies' cultural approaches and uses of security infrastructure and attacks and defenses, as an example of such an effect. In particular, we note the different behaviors of eastern European attackers, east Asian attackers, and hacktivists in groups like lulzsec. Similarly, our owns consciousnesses and their abilities to reason about attacking and defending systems is an application. However, the direct use in the development of an attack or defense seems unlikely, exactly because we do not understand this form of emergence. Therefore, for the time being Type IV's use in information security seems minimal, and we do no dwell on it further.

3. EMERGENT DOMAIN IN SECURITY

We now discuss how emergence is playing a role in information security. We show that not only are there past examples of emergent attacks, vulnerabilities, and defenses, but that we should expect to find new ones. We argue that when performing risk analysis, threat modeling, planning, engineering, post attack analysis, and auditing one needs to consider emergent behaviors. Further, new tools and techniques need to be used and developed to reason about these emergent properties. In particular, in all domains of science, predicting *specific* novel emergent behavior is not something that has been very successful. However, a number of disciplines including complex systems [58] and computational social science [24] have had success predicting or describing general aspects of emergence. Some of the methodologies and tools from other domains can be borrowed and possibly modified for the specific needs of the security community.

[1] Fromm actually uses the notion of intentional and unintentional emergence, but we feel direct and indirect are better corresponding titles.

Our first goal is to describe differences in the approach between the domains of decomposable and emergent security. Next, we want common diagnostic properties of emergent systems so we can efficiently recognize when time should be spent considering such attacks and defenses.

3.1 Emergent and Decomposable Domains

We split security issues into two domains: decomposable and emergent. Decomposable problems are many of the problems that we have traditionally faced in computer security, such as buffer overflows, access control structures, cryptography, etc. They are problems or systems in which we can seemingly decompose into a number of distinct smaller elements. Those elements can be analyzed independently such that we can combine our analyses and say something concrete about the whole. Information security currently has a rich and mature toolset to measure and mitigate risks associated with decomposable security problems. In contrast, emergent problems are problems that do not allow us to decompose the system into its individual elements. If we decompose an emergent problem, we will find our individual analysis of the components allow us to predict little, if anything, about the larger problem. Thus, if we cannot decompose such a system and study its individual elements, we must study the system as a whole. Unsurprisingly, the two domains call for alternate techniques.

To better understand the emergent and decomposable regimes, we provide a brief example. Let us assume a new piece of malware is discovered. It is possible to look at the code and understand its payload, the vulnerabilities it abuses to infect, etc. However, determining how fast or easy it will spread across a given network as well as through new types of networks is implausible. Consider that this new malware spreads over vehicular networks from vehicle to vehicle over a short-range radio. Even given the amount of time needed to infect, the percentage of vehicles that were susceptible, and a number of other factors, it would be incredibly difficult in most cases to establish whether such a virus would spread widely or be mostly self-contained.

3.2 Diagnosing the emergent domain

When considering a methodology to handle a security problem we must understand if that problem is in the emergent domain or the decomposable domain. In order to diagnose when we are in the emergent domain, we need a list of diagnostic criteria to compare our problem against. These diagnostic criteria should have their basis in the types of emergence presented in Sec. 2.1. The more criteria that are met, the more likely the system should undergo an analysis for emergent attacks, vulnerabilities, and defenses. The default assumption in our field, and thus our diagnosis is that the problem has no emergent properties, and thus is in the traditional decomposable domain. In Fig. 1, we provide a diagnostic flow-chart to use in determining the type of emergence the system may demonstrate.

1. Are there a number of similar agents in the problem space?
2. Do the agents interact or coordinate?
3. Do agents have independent decision making processes?
4. Is the decision making process based on local or environmental information?
5. Is the system-level behavior unpredictable?
6. Is there feedback within the system or between systems (or systems of systems)?
7. Can the system's outcomes be described by a non-linear effect?

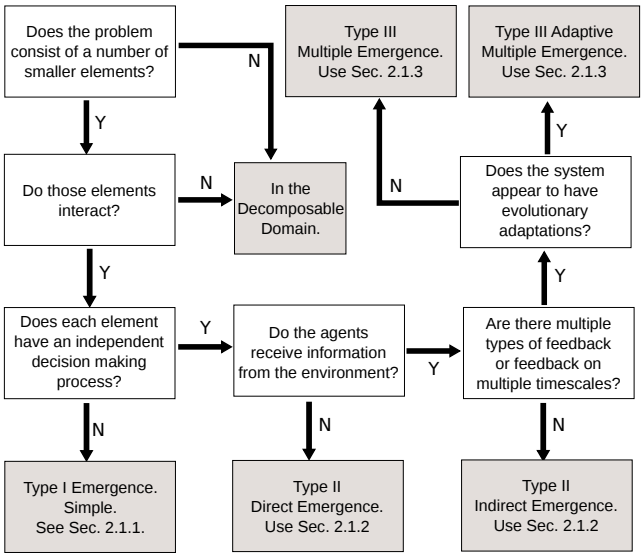

Figure 1: Decision tree to diagnose the highest level of emergence expected from system.

3.3 Emergent Attacks, Vulnerabilities, and Defenses

Emergent properties can result, unsurprisingly, in attacks, vulnerabilities, defenses, and value creation. We describe and distinguish these in the following sections then provide more specific case studies later.

3.3.1 Attacks

Emergent attacks are attacks where the attack itself emerges from a system. They require a large number of participants, such that any individual participating would not have much of an effect, but where large groups of individuals create a disproportionately large effect. Examples of an attack that fits this criteria is a DDoS attack. Any particular bot or agent participating in the DDoS is not generally harmful, but their combined traffic produce devastating results. Similarly, tracking networks [44] are an emergent attack, as a small number of trackers are unlikely to provide much in the form of identifying behavior, but a large number of trackers can result in surprisingly complete geolocation of unsuspecting targets. Pump and dump stock schemes are examples of financial emergent attacks [18]. Biological emergent attacks are represented by certain autoimmune diseases [38].

3.3.2 Defenses

Emergent defenses are a strategy requiring a number of agents working in a self-directed effort to stave off attacks against an adversary. A great biological example of an emergent defense is herd immunity. Herd immunity occurs when a certain portion of the population receives a vaccination from a communicable disease. What is interesting, however,

is that due to the manner in which computer networks have been connected[2], herd immunity does not apply. This result was found not by computer scientists but by physicists and epidemiologists [58].

The ability for the Tor [22] onion routing service to provide anonymity, however, is a great digital example of an emergent defense. If only a small number of users are using Tor, then little in the way of anonymity guarantees are achieved, but if there are large numbers of users, then high levels of anonymity are produced.

3.3.3 Vulnerabilities & Risks

Emergent vulnerabilities and risks are ones where a small number of agents possessing certain properties result in a system with either no vulnerability or a marginal risk, but large numbers of agents present a system with disproportionately sized vulnerabilities or very large risks. Many problems with human memorizable passwords might be viewed as emergent vulnerabilities. Humans asked to memorize one or two passwords might be able to follow complex rules needed to develop strong passwords. However, when asked to develop passwords for many sites, people are unable to consistently develop strong passwords. At some number of passwords m, n where $n > m$ there is a phase change where m passwords are conveniently remembered but n passwords are difficult to remember. Because of this, as users approach their mental limits they typically start to re-use passwords on different sites. As passwords get re-used, a new attack emerges, namely attackers ability to infiltrate a site with weak security. The more individuals that have this problem, the more reused passwords, the greater the incentive for attackers to infiltrate weakly secured sites with the sole purpose of using the results on strongly secured. Alternately, consider systems such as GM's OnStar which can disable vehicle engines to deter theft and prevent high-speed chases. If a hacker were to take control of this system, and disable an individual's vehicle, there would be no emergent risk. If the system permits a hacker to disable an unlimited number of vehicles in a short time period and in a localized area, then an attacker would have the ability to shut down the transportation infrastructure of major US cities. This shutdown is done by simply directing all GM vehicles in the city and its major traffic arteries to stop. There are conceivably enough GM vehicles on the road to halt all, or the majority, of traffic. However, there is a specific catalyst point for this effect to occur. For example, a similar ability to shut down all Ferrari's in a given metropolis is unlikely to affect their transportation infrastructure (even in Monaco). The open problem in the case of this emergent vulnerability is how the number of cars and societal factors affect the risk curve.

3.3.4 Emergent Assets

Emergent assets are assets whose value emerges based on the number of agents that individually value the asset. They are assets at the whim of the network effect [75]. The value that emerges is greater than the sum of the individual values the entities hold. Here we do not mean the general economic maxim that the more demand for an object there is, the more value it has. Most objects have intrinsic value, and their costs rise and fall based on demand. However, something like social networking is valuable only if enough other people participate (cf., Facebook vs. Google+). Similarly, Bitcoins (and paper currency) are inherently uninteresting save for the view that everyone else considers them of value. For instance, producing Bitcoins with an alternate origin block has no inherent value, as the resulting 'Bitcoins' will not be considered valuable by the Bitcoin community, even though the same mathematical process was used to generate the coins. The emergent question that a researcher is concerned with when considering emergent assets is at what point that asset goes from being valueless to having substantial value.

[2]The Internet at a TCP/IP level is fully-connected in that any computer on the Internet can effectively communicate with any other computer directly assuming no security barrier is in place.

3.4 Engineering Emergence and Intent

We note the fact that we are discussing systems that might be specifically engineered to have an emergent property does not mean that the systems are somehow not (or less) emergent, due to intent of the builder. A system, once constructed, can be evaluated for emergence under the many definitions that exist. What is clear is that emergent attacks and defenses become hard to break or counter due to the decentralized nature of emergent systems, and thus there is great incentive for emergent attackers and defenders to make progress in engineering these systems. We note that engineering such systems is still an open question in complexity science [32].

In practice one might argue there has been a repeated history of evolution of different types of security towards engineering more emergent like properties, due to attacks on previous systems. In practice, single centralized control systems are subject to take-down, and thus there is typically a move to have a system achieve its desired outcome without centralized control. Examples include the move from centralized music sharing with Napster, to the completely peer-to-peer structure of bit-torrenting; the use of centralized command and control servers for botnets, to peer-to-peer command structures, to social command structures; the construction of consensus, byzantine agreement, and self-stabilizing algorithms for use in distributed computing. Even Netflix is currently looking at delivering content in a peer-to-peer fashion to overcome its peering "disagreements" with major American ISPs.

On the defensive side, one of the great lessons of emergent systems seems to be to avoid monocultures, and here it is interesting because we have an economic system which, at least in the short term, seems to prefer monocultures (be they organic, software or hardware). Nonetheless, we see the beginnings of engineered heterogeneity in our information systems, designed specifically to combat different attacks. For example, address space layout randomization provides the beginning of heterogeneity to installed operating systems of the same version.

Distributed Computing.

Examples of engineered emergent properties abound in distributed computing. For example, many algorithms for leader election, byzantine agreement or consensus (c.f. [33, 60, 50, 37] for the beginnings of long lines of research on these problems) might very well be considered emergent systems. Many of these protocols exhibit a large number of similar devices executing identical protocols, and a given property emerges (a clearly defined leader, or agreement on a value) as the protocol runs. Yet understanding this prop-

erty by only inspecting one computing device without having a full understanding of its environment, the role other computing devices play in the protocol, and the goal of the protocol would be difficult. Further, most of the computing devices in these protocols provide feedback to other computing devices. Thus we might argue that many of these protocols are in fact Type II systems on our spectrum. Further extending this, Diijkstra [21] introduced the notion of self-stabilized distributed systems. These are like traditional distributed systems but they are supposed to be resilient (in time) to glitches in their environment which alter their internal states: if any (or all) of the computational devices in the system have their internal states randomly modified, they are guaranteed to converge to legitimate configuration for the protocol. Thus we might view random or malicious modifications of a computing device's internal state as input from the environment, and the self-stabilizing algorithm will react to such feedback as well. Further, the reaction to such feedback is often on a different time scale than the distributed algorithm's fundamental emergent property. Therefore, we might see this as being very early attempts at engineering Type III systems.

Of course, the above systems are not specifically related to security, but they show the potential for engineering systems (for both good and bad) that take advantage of emergent phenomena.

Though we have mentioned successes in creating distributed computing systems that have emergent properties, these are more the exception than the rule, and there still considerable difficulty in engineering emergent systems. Difficulty lies in the fact we tend to engineer with a purpose or goal in mind. For example we design gasoline powered car engines to move a car in the most efficient way possible using petrol. As such we may have very complicated systems or systems of Type I complexity due to sheer number of elements in the system. Designing higher level emergent systems where we have a specific, non nebulous, purpose in mind is far more difficult. However, if we choose to design based on certain general properties such as "resilience" or "adaptability" then we are more successfully able to engineer systems exhibiting at least a single emergent property. Again, an example of this would be self-stabilizing distributed systems. The drawback, however, is that we are limited to engineering for the emergent properties we mostly understand, e.g., resiliency. In many situations these engineered resilient systems take the form of distributed computing networks or distributed autonomous agents. In other situations, we tend to engineer systems that mimic natural systems such as ant colonies. In these cases we tend to see behavior that we like (e.g., an immune system's ability to detect the good from the bad) and we attempt to mimic it. In these cases we create bio-inspired computing systems [19] such as artificial immune systems [29]. Eventually, as we better understand higher emergent properties and how they come to be, it may be possible to better engineer emergent systems specifically intended for an emergent behavior. A related difficulty is that we have limited tools for understanding when multiple systems are likely to interact in unexpected ways, and create emergent properties. Typically such interactions are only found through trial-and-error, and post system-failure analyses.

3.5 Micro Case Studies

To better understand the manifestation of emergence, whether it be Type I, II, or III, in the information security context, we provide a number of micro case studies. Each case study describes a relevant information security attack, system, or concept, and provides a description of how its characteristics match with those of Type I, II, or III emergence. The case studies highlight the need for researchers to use the methodologies and tools discussed in Sec. 4 to study emergent problems in security.

3.5.1 DDoS Attacks

A DDoS attack consists of a large number of computers operating in unison in order to flood target servers with networking traffic. DDoS attacks can be sophisticated in that the little traffic sent from each individual bot can take advantage of weaknesses in Internet protocols in order to eventually amplify the amount of traffic received by the target, but the key idea is still that a large number of bots generate traffic directed at a target to overpower their bandwidth. Typically, if we were to look at any individual computer in a DDoS attack, we would not think its ability to send out network traffic is particularly dangerous. However, if one looks at a DDoS attack as a whole, we see that the large amount of traffic and the potential amplification techniques the attack uses can devastate sites with even the largest available bandwidths.

The bots in a DDoS attack can coordinate in several different ways. A first method has the bots all receive a central signal from a command and control (C&C) server telling the bots who to attack. A second method is to distribute the C&C infrastructure in a Peer-to-Peer (P2P) fashion. A third method is that the bots are controlled not technically, but socially: this was seen in 2010 when Anonymous started their Low Orbital Ion Canon (LOIC) attacks against the major credit card companies [61]. In the attacks sponsored by Anonymous, not only were traditional botnets used, but members of the public could volunteer to use their computer as part of the attack by using the LOIC tool to join the DDoS. In this campaign, the attack also had to propagate through a social network just like any other Internet meme or fad. This adds an extra level of complication towards determining the actual strength of the final attack. Each coordination method we observe suggests a higher form of an emergent attack. The central C&C system, would correspond to a Type I indirect system, and a P2P control system would correspond to a Type II. Finally, the use of a social meme to get people involved in the attack would be a Type III. Importantly, the level of difficulty in stopping such attacks seems to scale with emergence level; the strategy for defense changes as well. In Type I schemes, we can take down the central C&C infrastructure. Taking down P2P C&C has proven more difficult, and typically we have resorted to going after the specific individuals in control. Taking down a social movement, such as Anonymous members using LOIC, is unclear. Knowing the level of emergence a security phenomenon displays is important in designing an appropriate mitigation as well as predicting the potential effectiveness of the designed mitigation.

3.5.2 Opportunistic Tracking Networks

Opportunistic tracking networks are networks of sensors that are willing to track an object or agent, but only if the

opportunity arises. An example of such a network is a WiFi based tracking network as discussed by the authors in [44]. Consider a scenario restricted to individuals with mobile phones that can geo-locate themselves and have typical WiFi functionality. We consider that if a large group of those individuals, whether it be through intentional application of software to their phone, or the subversion of their phone by a botnet, promiscuously sniffs WiFi traffic (including protocol management traffic) in attempt to locate the unique identifier of other individuals. Call this group the *trackers*. The trackers, upon sniffing traffic of a non-tracker of interest, geolocates themselves and broadcasts that information to the central storage system with an associated time stamp. If enough trackers opportunistically sense *others* and geolocate themselves and populations are dense enough, then using simple triangulation and correlation of data it is reasonable to expect high fidelity geolocation data of an individual to emerge, unbeknownst to them.

The emergence of the tracking capability contained in such an attack stems from multiple types of feedback caused from the combination of human mobility, city design, population density, and tracking network size. Human mobility is an emergent process in and of itself that can be very similar to flocking. There is feedback in the system leading to individuals clumping together when faster individuals cannot pass slower individuals on a sidewalk. A similar process creates phantom traffic jams, slow traffic that has seemingly no apparent cause [47]. The feedback in human mobility then creates feedback in the tracking network as it increases population density thus the number of individuals detected at any give time. Office organization also will create feedback in the tracking network due to the density of individuals in an office. The density of offices is juxtaposed by individuals commuting where density will be sparser as the population is on the move. The city's urban planning has feedback in the tracking network depending on building placement, size, and location. Denser placement of buildings creates the potential for more detections. A sparser placement of buildings (i.e., wider streets and sidewalks) will create a smaller potential for detections. The effectiveness of the wireless tracking network is a non-linear function dependent upon the size of the network and the socio-geographic properties of where the network is realized [43].

Tracking networks can involve many types of feedback. If we were to assume very simple geographical situations (say a large field or a stadium) such a network might fall under Type II emergence. However, if we assume a metropolitan area with much richer interaction dynamics (see [44]), then we have a system of Type III emergence. Again, note that the effectiveness of a tracking network is essentially impossible to intuit from understanding how the specific triangulation and wireless sniffing work on an individual agent.

3.5.3 Digital Worms and Malware

Digital worms are pieces of malware that spread between devices with minimal or no user interaction. The spread, when user interaction is involved, is unintentional. Through time there have been drastic changes in implementations of contagion vectors: originally, sharing floppy disks in infected machines led to transmission; then through direct Internet network transmission; later the use of email or social networks; to complex multi-phase dynamics such as the Asprox botnet [77]; to local transmission through short-range radio such as Bluetooth and WiFi.

Whether or not a given piece of malware becomes local, regional, or a global threat is a question of emergence. That is if malware does not effectively spread, it is considered non-emergent. If it stays latent in an environment with occasional flare-ups, or spreads throughout the environment then this is emergent behavior. For example, malware that is endemic in a population, i.e., constantly rising and falling but never dying out, is a Type III emergent phenomenon. Malware that becomes pandemic (e.g., Code Red, Melissa) is a Type II emergent phenomenon.

The general emergence of malware strains is probably the emergence related question that security has been studying the longest, although not in a broader emergence context. Our expectations regarding behavior are very much a function of the dynamics that govern the network, and the dynamics that govern the infection mechanism. We will discuss malware epidemiology further in Section 4.1.

3.5.4 Tor and Anonymity

Tor is a well know and widely used anonymity system. The system works by routing all communications through three random encrypted hops on a network. The goal of the system is to anonymize communication, so that it is difficult to determine who is talking to who, and not just keep communications private (as is provided by traditional cryptography). However, here it can be claimed that the anonymity is itself an emergent property. If Tor had only a small number of users then it becomes possible to identify users, and it is only when enough people use the system that appropriate anonymity is achieved. This is because the number of paths one might follow in the Tor network increases super linearly with the number of users. Beyond anonymity, Tor's growth might also be seen as an emergent property due to the network effect as mentioned in Sec. 3.3.4. Specifically, if there are few users of Tor, then in it is less useful for the next individual to commence using it, as little anonymity is offered by the network. However, there is a larger incentive to join when there are a large number of users, as the anonymity offered increases substantially. Anonymity is Type II emergent while the spread of Tor usage would be Type III emergent. Note questions about how much anonymity is achieved by use of the network, and how to encourage participation are important security engineering questions if one's goal is to build an effective anonymity system.

4. METHODOLOGIES & TOOLS OF THE TRADE

Security practitioners should first be focused on *building security in*. That is, we should be focused on first ensuring systems are developed securely to the degree possible by considering abuse cases, performing threat analysis, and determining appropriate security requirements. This work should be followed by risk analysis, secure development and testing to ensure development matches engineered specifications. However, despite our best intentions, we inevitably miss abuse cases, systems may not be implemented correctly, and systems can be abused or used in unintended ways. At this point risk analysis must be performed to determine the threat of vulnerabilities being abused or attacks made in real or potential scenarios.

Unfortunately, there has been little research in predicting abuse cases or performing risk analyses for emergent attacks, vulnerabilities and assets. Only recently there has been some thought towards solving current cyber security issues with tools from Complexity Science [27]. However, if the science of security project is to be successful then such understanding is essential. This is especially true as we see the deployment of more distributed peer-to-peer systems with many emergent properties, and further we evolve towards the Internet of things, ubiquitous computing and cyber-physical systems, where we will have large numbers of small computational devices with both sensors for achieving feedback and network access: a perfect environment for the development of emergent systems. In the case of cyber-physical systems, we have the added concern that these devices will have the ability to have kinetic effects in the real world. A strong set of methodologies dealing with measuring or describing non-linear security risks, long-tailed security events, and socially driven security phenomenon is required.

There is *not yet a* laundry list of traditional attacks or vulnerabilities in the emergent domain, so engineers will need to be creative in thinking of possibilities. The taxonomy discussed in this paper will hopefully spur advances into a development that maps formal taxnomies of traditional vulnerabilities and attacks [41] to vulnerabilities and attacks that exist in the emergent domain. For example, in Sec. 3.5.2 when discussing the possibility of using a botnet of mobile phones to track individuals, it is not immediately clear if tracking targets in a dense city is even feasible without unreasonably high penetrations of sensing bots. Similarly, the effectiveness of the spread of a virus through a vehicular network is rather unintuitive, and depends on traffic density, mobility patterns, susceptibility rates, etc.

Fortunately, we have the beginnings of tools and methodologies for studying potential attacks and vulnerabilities in the emergent domain [27]. Understanding these methodologies will provide a foundation for a discussion of how we can use new analysis techniques to help measure and understand the risk from emergent vulnerabilities and attacks. These techniques can be used prior to system creation to measure potential risks of the system design and to measure their effective mitigation. Techniques can also be used after system creation to study potential unexpected risks created from unintended uses of technology (an emergent process in and of itself). We discuss some of these, briefly, next.

4.1 Epidemiological Models

Epidemiology is the study of communicable diseases [63]. Historically, epidemiology has allowed medical practitioners to quantify how likely a disease is to continue spreading, predict whether it will die out with a few infected patients, become a localized contagion, an epidemic, or a pandemic. Unsurprisingly, epidemiology is now used to study the transmission of malware, and Internet memes as well.

Historically, several epidemiological models such as the SI, SIS, SIR, and SEIR models [9], where the letters in the model names correspond to different states an agent can be in at any time: (S)usceptible, (I)infected, (E)xposed and (R)emoved. In all these models there is a defined Markov process and associated differential equations which dictate the distribution with which agents transition from one state to the other, where the specific transition distributions are based on observed characteristics of the disease. All agents are then prescribed to one of a given number of states, and in many cases a set of closed form equations can be solved to determine what the expected stable ratio of individuals in the population are. Historically, when these models were developed, and because it permitted for closed form equations, homogeneous local mixing of agents was assumed. That is, the population is modeled assuming that people can only infect others that they are close to, and that people tend to move around rather uniformly. Even with simplifying assumptions the models were successful in predicting vaccination rates for herd immunity, if the virus would become endemic, or if the virus would become an epidemic.

In the digital setting with an underlying network of the Internet, the tools' biological success in prediction did not follow. The lack of success was found to be because of the full connectedness of the Internet TCP/IP graph [58]. Pastor-Satorras and Vespignani showed that any system where there was a fully connected communication network would result in a epidemiological constant of 0, which implies that a full on epidemic would arrive, and that regardless of the amount of patching or vaccination applied, there would be no heard immunity, and malware would not die-out. Importantly, these researchers were not in the traditional security field, but in the field of physics and epidemiology, where emergent tools and techniques were better understood. So here we see probably the first important security principle to result from studying emergent behavior models.

These results of Pastor-Satorra and Vespignani might make it appear that epidemiologic models have no use in modern information security, but, with the advent of the Internet of Things, we are likely to see the return of localized communications through localized short-range networks, intentional or unintentional mesh networks, and other such innovations. Epidemiology is useful with localized communication environments as it can be used to determine if new threats will propagate. An added complication is traditional homogeneous mixing models are frequently not sufficient in these environments, leading to the necessity of combining epidemiology with agent based modeling.

The use of epidemiology as a methodology can extend beyond the obvious cases of malware spread. The fundamental problem the methodology studies is the spread of some "artifact" through a society. Thus, security researchers could study the risk of malware technology propagating through the Internet black market just as social scientists use epidemiology to study the spread of memes [53].

4.2 Agent Based Models

Agent based modeling is a method for modeling in a bottom up fashion the individual elements of a system and their interactions. The models consist of a large number of agents interacting with each other and their environment. Essentially, massive computational power is used to model the behavior of agents over time, simulating their interaction between each other and their environment in the aspects that are of interest to the simulation in question. Simplifying assumptions are generally used to reduce the computational power needed. Agent based modeling is particularly good at making statements about the general behavior of a system given its environment, but not necessarily making predictions about a specific agent's behavior. Such global system statements can be done using models *vastly less complex* than the system's being studied [24]. Many disciplines

outside of security now rely on this approach for both prescriptive and descriptive models. For example, it is used extensively in fields as diverse as epidemiology [26] and urban planning [7]. In security, agent based modeling can be used to simulate an attack or defense, and determine its effectiveness.

More specifically, agent based models consist of a number of fundamental elements with the most important being the *Agent*. Each agent consists of a set of *Rules* and a *State*. An agent's state has descriptive information regarding the agent pertaining to a specific point in time. The agent's rules are a set of transition functions $\tau(X, S_t) \rightarrow S_{t+1}$. Given some input X and the agent's state S_t at time t, the set of functions will return the agent's new state S_{t+1} for time $t + 1$. The input X is a combination of inputs provided by the *environment* the agents are in, as well as potential inputs from other agents. The environment can store output from agents as well as provide agents input through their means of sensing the environment. The units of time t are model dependent and can vary from microseconds to centuries.

The development of an agent based model for a specific modeling task is done in multiple steps. First we must decompose the system we are attempting to model and identify the agents, the interconnections between agents, the *behaviors of the agents important to the problem at hand*, the environment the agents exist in, and how the agents interact with their environment in ways that affect the problem at hand. A transition function then formalizes the behavior of the agents in the model, specifically how they interact with other agents, and the environment. A simulation can now be run where in every step all agents and the environment are updated according to the agents' transition functions.

One of the earliest examples of an agent based model, Conway's [15] game of life, a simple cellular automata. Each agent is represented by a single square on the "game board". An agent's state would refer to their (x,y) position in the game board and whether or not they are "alive" (colored) or "dead" (not colored). In the game of life, whether an agent lives or dies in a given time period is dependent on the number of living agents in neighboring cells. The game of life represents an exceedingly simple behavior by the individual agents, but even it is known to produce exotic emergent effects [81].

Writing a custom simulation for the agent based model can provide benefits, especially when creating a complex model as it gives the engineer the utmost flexibility. There are however generic utilities such as NetLogo [74] and Repast [14] which, while slightly constraining, make the development of such models much more efficient. Finally, experiments can be run on the model and results can be analyzed. Besides a thorough analysis of the results from the model, it is to attempt to validate the results against real world data whenever possible. Validating model results versus real world data helps prove either the descriptiveness or prescriptiveness of the model, and can help in improving the modeling of similar problems over time. *The development of agent based models for patterns that are likely to be frequently used in security, such as human mobility in different environments, that are then validated by data need to be seen as a valid research direction.* The development of such useful models that are properly validated will be key in predicting future emergent attacks. Nonetheless, it is worth noting that Nikolic et al. [76] argue that even validation by experts (stakeholders in the problem) is a predominant method of testing the validity of agent based models.

For readers looking for a more detailed, procedural view, of constructing agent based models, we point them towards [76]. However, we describe the general use of agent based models in providing security outcomes in tracking networks and mobile malware spread.

Tracking Networks.

In our work on Tracking networks [44], when one considers the possible tracking attack of a large number of bots in an urban environment, the natural question arises as to whether or not the attack is plausible? As the attack is emergent, quick thought shows it is difficult to create a simple analytical model and any real world observations could be prohibitively expensive (or illegal). Agent based modeling via the UDelModels simulator from the University of Delaware [1] provides a convenient foundation. This model is used to give high-fidelity simulations of humans mobility patterns, matching first and second order real-world statistics, in urban environments and was developed for urban planning. On top of this simulator, we were able to add our tracking network behavior via a secondary agent based model to show show plausibility of the attack; and thus the realistic need to consider defenses.

Mobile Malware & Epidemiology .

There has been much discussion, and many papers published that strongly suggested we would see a large amount of malware spread via a Bluetooth or WiFi vector [71, 46, 10]. However, except for a few famous cases, such as the 2005 world track and field championships in Helsinki [46] where malware did indeed spread via Bluetooth, there have been few cases in the wild. By using epidemiological models on top of the UdelModels simulator [1], previously mentioned, we showed in [45] that such contagions were unlikely in certain classes of modern mobile-to-mobile malware.

4.2.1 Multi-layer Models

As seen with both tracking networks and mobile malware, multiple layers play an important role in modeling emergent phenomenon, especially those of Type III or higher. The reason they are important is the many different types of feedback are sometimes existent on different functional layers of the system. For example, in the case of the tracking network we have feedback existing both at a human mobility level and within the model of the tracking network itself. Each layer (the human mobility layer and the tracking network layer) will show different elements of emergence. Using a multi-layer approach allows us an understandable way to build, generatively, models themselves, from the ground up.

4.3 Systems Theoretic Based Design

The Systems Theoretic Design approach focuses on stringently designing a system from beginning to end, taking the whole system and its social elements into consideration. The goal of this approach is to limit the level of emergence in the system by taking control of any and all interactions both social and technical. Pinpointing the optimal areas to control is done via Systems Theory [12]. The primary control taken in this approach is to limit and remove behavior causing emergence. This step usually manifests as removing feedback loops. Feedback loops are one of the causes of

Type I and Type II emergence in Fromm's spectrum [31]. Systems Theoretic Design can be used prior to the creation of a system or as a triage mechanism after an emergence vulnerability has been discovered in a system. Let us consider a simple example of a peer-to-peer worm. Systems Theoretic Design would have us map out the process flow of the worm, the network it infects, and the human parties that interact with both. The process flow would allow us to pinpoint feedback loops in the spread of a worm giving us information on where to target defenses. The drawback to current Systems Theoretic Design approaches is they tend to provide more qualitative solutions regarding mitigations but can predict little of the quantitative effectiveness of such mitigations. We discuss specific Systems Theoretic Design processes for the rest of this section.

Two means for performing Systems Theoretic Design are the System-Theoretic Accident Model and Process (STAMP) and the Systems-Theoretic Process Analysis (STPA) [52]. STAMP is a method for analyzing accidents in a top-down fashion allowing for consideration of the whole socio-technical system. The STAMP philosophy argues that accidents are caused by failed controls letting unintended feedback into a system. This feedback causes accidents to emerge. The STPA model is similar to STAMP but operates during the design of a system instead of after an accident occurs. In both cases the philosophy is to maximize the ability to control the system as much as possible in order to limit emergence causing feedback. While originally these systems were designed for accidents, they have since been adapted in the form of STAMP-Sec and STPA-Sec to look specifically at security systems [51, 83, 84].

Using STAMP-Sec to qualitatively analyse socio-technical systems after a security incident (cf., the Air Transportation System before and after September 11th [51]) is a straight forward process. Laracy [51] outlines five overall steps that must be performed: 1) Identify system level vulnerabilities and model the system architecture; 2) Enumerate the system's security constraints that were violated by the vulnerability; 3) Define the set of static controls that were in place; 4) Identify what controls defined may lead the system to an insecure state; 5) Using "System Dynamics" [69] determine ways the security constraints and control structures could be violated through emergent phenomenon. STAMP-Sec, however, does not provide prescriptive solutions to mitigate problems in the system, but a descriptive means of explaining how the security, at a systems level, went wrong. In other words STAMP-Sec points out the feedback mechanisms that were not adequately controlled for that allowed the emergent behavior to occur.

The STPA-Sec process is used to analyze a system's architecture before an incident occurs in order to maximize the security of the system by minimizing its threat surface (i.e., minimizing its vulnerabilities). STPA-Sec shares the system level focus of STAMP-Sec. STPA-Sec's system level focus manifests in a process that is concerned with security vulnerabilities and less about attacker tactics. Young et al. [84] consider this a "strategic" approach, not a "tactical" approach, and one better suited for handling problems with security related to emergence. STPA-Sec can be used before or after a system has been built but is preferably used before. STPA-Sec has four major steps: 1) Eliminate any clear vulnerabilities from the conceptual design by designing system definition constraints; 2) If vulnerabilities cannot be eliminated, identify ways to control for them at a system level; 3) Develop a system control structure taking into account the need for enforcing the mitigating controls defined in step 2; 4) Refine the constraints by iterating over the process. The goal of this process is to *control for feedback* that causes a system to enter a vulnerable state. STPA-Sec attempts to mitigate emergence security vulnerabilities by controlling the modeled emergence that causes them to arise. This process is in contrast to the earlier discussed methodologies (Sec. 4.1 and Sec. 4.2) which model the emergent security phenomenon but provide no specific mitigation strategy.

STAMP-Sec and STPA-Sec work best in systems where the architects have full control over every portion of the system, including the social elements, as there will be no other adequate way to control all channels where feedback could manifest. Thus, STPA-Sec and STAMP-Sec are particularly suited for military and governmental projects that are extremely process centric compared to most consumer targeted software. For example, a custom developed missile defense system will have a rigid set of training on its use compared to an Android smartphone which has little to no training. It is less advantageous over a consumer device environment where there is no single entity in control. The goal of STAMP-Sec and STPA-Sec are to reduce the level of emergence displayed by the system (i.e., reduce a Type II system to a Type I system) by controlling for feedback in order to remove the possibility for unexpected outcomes. Given STAMP and STPA's focus on system dynamics and second order cybernetics, their success with Type III systems and above is questionable given the previous failures of second order cybernetics in this area [54].

5. THE FUTURE OF EMERGENCE IN INFORMATION SECURITY

Emergence will play a great role in the future of information security. The rise of mobile devices, ubiquitous computing, and the Internet of Things make the reality of emergence inevitable. As security researchers and practitioners, it is our job to not only determine when our problems have emergent components, but to be able to adequately model the security outcomes emergence brings about. Embracing emergence as an aspect of security provides us the benefit of being able to properly handle problems of emergence. If we ignore the existence of emergence, or disregard it as a fad, we will only curse ourselves to using tools inadequate for the job at hand. We will be measuring risk and developing system mitigations based on assumptions that do not hold true in the systems we are striving to protect. Understanding emergence provides us the framework for developing a toolset to deal with emergent problems we will continue to face into the future. In this work we've outlined a spectrum of emergent behavior and provided an associated diagnostic criteria to allow for researchers to diagnose when their problems exist in the emergent domain. A set of use cases were also given to help connect the concepts of emergence with the discipline of computer security. While it is important to recognize and model the new emergent security phenomenon, we must also consider how to analyze and mitigate threats while considering emergent security phenomenon. The methodologies introduced help provide a foundation in this direction, but

it is useful to consider the bigger picture of risk, emergence, and mitigation.

5.1 Risk and Emergence

Given the importance of economics and threat assessment in modern information security, it is important to know how emergent security phenomenon will affect our ability to assess the security risks of our systems. This is a difficult question to answer as emergent attacks and vulnerabilities exhibit behaviors higher on the emergent spectrum. For example, certain security phenomenon have power-law distributions in their components. P2P botnet structure obeys a power-law [78] and spam transmission based on users' address books also shows certain power-law components [57]. The power-laws are created by the feedback mechanisms inherent in the emergent phenomenon. Many emergent phenomenon occur on power-law based distributions. This is useful for predicting the kinds and distributions of attacks, but not necessarily the specific strength of any upcoming attack. Power-law distributions make it difficult, if not impossible, to predict the risk of any specific event. In other disciplines these have been known as "Black Swans" [72] and "Dragon Kings"[68].

Black Swans are simply events with an extremely small probability of occurring but with an extremely large effect when they do. In particular these Black Swans are events taken from a power-law law distribution of event sizes. They are "events that started small and did not stop growing..." [68]. These events occur, generally, because they involve some form of positive or negative feedback. In fact, many systems that have at least Type II emergence will have some black swan element. In information security a black swan event is an attack which might be extremely rare (and difficult to realize), but if it occurs, the outcome could be catastrophic. For example, a buffer overflow attack against OpenOffice may not effect many users but an attack again OpenSSL can quickly effect the whole Internet.

Dragon Kings are events that are outliers in a power-law distribution of events. They are "extreme events that are statistically and mechanistically different form the rest of their smaller siblings." [67]. A canonical example is the city of Paris, France in a graph of French cities rank-ordered by population. This leads to a fundamental epistemic difference between Dragon Kings and Black Swans in that Dragon Kings are potentially predictable while Black Swans are by definition not predictable [59].

Even though effect sizes are unpredictable (in the case of Black Swans), or potentially unpredictable, it does not mean we should completely ignore mitigation or attempts to develop methods that aide in certain elements of predictability. For example, the methods discussed in Sec. 4.2 allow us to develop systems with some bounded risk. The methods also allow us to analyze the overall effects caused by the emergent phenomenon to made educated decisions on what risks we do want to mitigate to some extent.

5.2 Emergence, Formal Methods, and Understanding Security

The acceptance of emergent phenomenon may fundamentally affect many current views on security, and some may find this concerning. Generally, systems are considered secure if all their components are considered secure. Emergent security phenomenon raise issues in that it is not only the individual elements themselves that need to be of concern but it is also the interactions between those elements, and the interactions of those elements with their greater environment. This creates two issues: 1) It is possible that we can verify the properties of all components but we cannot verify the security of all interactions between all possible components or environments; 2) It is possible that we may need to accept systems as secure if they predominantly exhibit secure behavior but their behavior is not "good" 100% of the time.

Verifying complex interactions between components has recently been a concern within the formal methods community. As systems becoming increasingly complex there has been a greater chance of "unintended consequences" of formally verified equipment. Rushby coined these unintended consequences "emergent misbehavior" [64]. Rushby suggests eliminating and controlling for unanticipated behaviors and interactions. These two principles could be realized by a stringent use of the STAMP and STPA procedures discussed in Sec. 4.3. However, as mentioned previously, eliminating and controlling interactions for emergence can be particularly challenging if higher types (Sec. 2.1 of emergence are involved.

Let us consider cases of high enough complexity (Type III or above) where we cannot adequately reduce the complexity to such an extent tools like formal methods would be feasible. Such cases, as discussed earlier in Sec. 3.5, would arise in situations of consumer technologies that have proliferated to the point of ubiquity (e.g., mobile phones). Even assuming we can guarantee the components, in isolation, behave "securely", we can do little to make claims about all interactions of those components due to the difficulty in predicting all ways in which individuals will use any given product. This creates a situation where "security" as a concept must move from am all-or-nothing paradigm to a paradigm of gradual risk.

A paradigm of gradual risk shifts the focus of the discipline as well. No longer are we specifically studying and modeling engineered artifacts displaying vulnerabilities but we must now study the social aspects surrounding those artifacts. For example, let us take the case discussed in Sec. 3.3.3 regarding the shutdown of vehicles. The main point of analysis is not studying how some digital artifact (i.e., the car's computer) can be exploited but the societal threat of such an exploit. The tool set of the security practitioner expands from engineering to leveraging computational power to model social organizations. In this case the social organization is the traffic patterns of major and minor US cities. To be clear this is not a disregard for the old paradigm but a warning that those entering the new paradigm must expand their skill sets into areas not traditionally viewed as "computer science" or "computer security" to compensate for the new threats.

5.3 Future Directions

Mitigating and analyzing threats has always been a central tenant of both the research and practice of information security and this does not change with emergent security phenomenon. Currently, threat analysis is done, at best, by comparing one or more attack frameworks to the system in question in order to find vulnerabilities [66]. Emergent attacks and vulnerabilities do not change this process but require the security architect to think more broadly about

the threats posed to the system. Mitigating emergent vulnerabilities and attacks may be beyond the traditional approach of engineering some technical solution. The social aspects of emergence mean that purely technical solutions will no longer be feasible in some cases. Embracing and using the tools and methodologies discussed in Sec. 4 is a start to potentially determining the effectiveness of non-technical solutions. Additionally, we need to use tools from economics, sociology, law, criminal justice, and political science which have been used to deal with societal emergence problems. Examples here include policy frameworks, and game theory. We must now look more towards policy and regulatory solutions.

One manner of mitigating emergent vulnerabilities involves leveraging the social element of a socio-technical system in order to exact control over the whole system. The STAMP and STPA frameworks (Sec. 4.3) are based on this philosophy. Their aim is to control all aspects, both social and technological, to such an extent that opportunities for emergence are removed from the system (and thus controlling for security). As mentioned previously, while this level of control may not work with consumer products, the frameworks provide some inspiration for the role policy makers and regulators will play with emergence. Such inspiration should lead us to consider modeling these social elements, for example, when attempting to simulate mitigations for an emergent vulnerability using an agent based model. In a general sense, what we will need is greater elasticity and faster action on behalf of policy makers and regulatory bodies based on the results of these simulations in order to adequately mitigate the risk posed by emergent security phenomenon.

We have seen that there are already many emergent phenomena in security, and it is inevitable that more will become apparent in the next few years. The tools discussed in Sec. 4 will be a start for practitioners looking at studying this behavior in an orderly and methodological manner. It is important we have a methodological basis for studying the inevitable rise of emergent attacks which exhibit some form of self-organization and artificial intelligence as a way of avoiding detection. While malware already has polymorphic qualities it is also possible we will eventually see malware take cues from its environment and use this information in an autopoietic existence creating vast difficulties for defenders. The next generation of threats require the next generation of methodologies and those methodologies must seamlessly feedback with the current. Security as a discipline is, after all, an emergent system.

6. ACKNOWLEDGMENTS

This work is supported by Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory (AFRL) under contract FA8750-11-2-0211. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US government. This material is based upon work supported by the National Science Foundation under Grant No. 1111149.

7. REFERENCES

[1] UDel Models. *http://www.udelmodels.eecis.udel.edu/*, July 2010.

[2] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

[3] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Error and attack tolerance of complex networks. *nature*, 406(6794):378–382, 2000.

[4] Taimur Aslam, Ivan Krsul, and Eugene H Spafford. Use of a taxonomy of security faults. 1996.

[5] Yaneer Bar-Yam. A mathematical theory of strong emergence using multiscale variety. *Complexity*, 9(6):15–24, 2004.

[6] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[7] Michael Batty. *Cities and complexity: understanding cities with cellular automata, agent-based models, and fractals*. The MIT press, 2007.

[8] Valentino Braitenberg. *Vehicles: Experiments in synthetic psychology*. MIT press, 1986.

[9] F. Brauer. Compartmental models in epidemiology. *Mathematical epidemiology*, pages 19–79, 2008.

[10] L. Carettoni, C. Merloni, and S. Zanero. Studying bluetooth malware propagation: The bluebag project. *IEEE Security and Privacy*, 5(2):17–25, 2007.

[11] K. Channakeshava, D. Chafekar, K. Bisset, VS Kumar, and M. Marathe. EpiNet: a simulation framework to study the spread of malware in wireless networks. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, pages 1–10. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.

[12] Peter Checkland. Systems thinking, systems practice: includes a 30-year retrospective. 1999.

[13] Ashley Chonka, Jaipal Singh, and Wanlei Zhou. Chaos theory based detection against network mimicking ddos attacks. *Communications Letters, IEEE*, 13(9):717–719, 2009.

[14] Nick Collier. Repast: An extensible framework for agent simulation. *The University of Chicago's Social Science Research*, 36:2003, 2003.

[15] John Conway. The game of life. *Scientific American*, 223(4):4, 1970.

[16] James P Crutchfield. The calculi of emergence: computation, dynamics and induction. *Physica D: Nonlinear Phenomena*, 75(1):11–54, 1994.

[17] Felipe Cucker and Steve Smale. On the mathematics of emergence. *Japanese Journal of Mathematics*, 2(1):197–227, 2007.

[18] George Cybenko, Annarita Giani, and Paul Thompson. Cognitive hacking: A battle for the mind. *Computer*, 35(8):50–56, 2002.

[19] Leandro Nunes De Castro. *Fundamentals of natural computing: basic concepts, algorithms, and applications*. CRC Press, 2006.

[20] T. De Wolf and T. Holvoet. Emergence versus self-organisation: Different concepts but promising when combined. *Engineering self-organising systems*, pages 77–91, 2005.

[21] Edsger W. Dijkstra. Self-stabilizing systems in spite of distributed control. *Commun. ACM*, 17(11):643–644, 1974.

[22] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Technical report, DTIC Document, 2004.

[23] Marco Dorigo and Mauro Birattari. Ant colony optimization. In *Encyclopedia of Machine Learning*, pages 36–39. Springer, 2010.

[24] Joshua M Epstein. *Growing artificial societies: social science from the bottom up*. Brookings Institution Press, 1996.

[25] Joshua M Epstein. Agent-based computational models and generative social science. *Generative Social Science: Studies in Agent-Based Computational Modeling*, pages 4–46, 1999.

[26] Joshua M Epstein. *Generative social science: Studies in agent-based computational modeling*. Princeton University Press, 2006.

[27] S. Forrest, S. Hofmeyr, and B. Edwards. The complex science of cyber defense. *HBR Blog Network*, Jun 2013.

[28] S. Forrest, S.A. Hofmeyr, A. Somayaji, and T.A. Longstaff. A sense of self for unix processes. *sp*, page 0120, 1996.

[29] Stephanie Forrest and Steven Hofmeyr. Engineering an immune system. *GRAFT-GEORGETOWN-*, 4:369–369, 2001.

[30] Jochen Fromm. Ten questions about emergence. *arXiv preprint nlin/0509049*, 2005.

[31] Jochen Fromm. Types and forms of emergence. *arXiv preprint nlin/0506028*, 2005.

[32] Jochen Fromm. On engineering and emergence. *arXiv preprint nlin/0601002*, 2006.

[33] R. G. Gallager, P. A. Humblet, and P. M. Spira. A distributed algorithm for minimum-weight spanning trees. *ACM Trans. Program. Lang. Syst.*, 5(1):66–77, January 1983.

[34] Nigel Gilbert. *Agent-based models*. Number 7. Sage Publications, Incorporated, 2007.

[35] David E Goldberg. Genetic algorithms in search, optimization and machine learning. 1989.

[36] BT Grenfell, A Kleczkowski, SP Ellner, and BM Bolker. Measles as a case study in nonlinear forecasting and chaos. *Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences*, 348(1688):515–530, 1994.

[37] Maurice Herlihy and Nir Shavit. The topological structure of asynchronous computability. *J. ACM*, 46(6):858–923, November 1999.

[38] Wen-Hsien Ho. Takagi-sugeno fuzzy model of nonlinear hiv dynamics: Chebyshev-series approach integrated with genetic algorithm. *International Journal of Innovative Computing Information and Control*, 8(2):1439–1451, 2012.

[39] C. Hooker. Introduction to philosophy of complex systems: A. *Philosophy of Complex Systems*, 10:3, 2011.

[40] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[41] John D Howard and Thomas A Longstaff. A common language for computer security incidents. *Sandia National Laboratories*, 1998.

[42] Hao Hu, Steven Myers, Vittoria Colizza, and Alessandro Vespignani. Wifi networks and malware epidemiology. *Proceedings of the National Academy of Sciences*, 106(5):1318–1323, 2009.

[43] N. Husted. *ANALYSIS TECHNIQUES FOR EXPLORING EMERGENT VULNERABILITIES AND ATTACKS ON MOBILE DEVICES*.

[44] Nathaniel Husted and Steven Myers. Mobile location tracking in metro areas: malnets and others. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 85–96. ACM, 2010.

[45] Nathaniel Husted and Steven Myers. Why mobile-to-mobile wireless malware won't cause a storm. In *Proceedings of the 4th USENIX conference on Large-scale exploits and emergent threats*, pages 7–7. USENIX Association, 2011.

[46] M. Hypponen. Malware goes mobile. *Scientific American*, 295(5):70–77, 2006.

[47] N.F. Johnson. *Simply Complexity: A clear guide to complexity theory*. Oneworld, 2010.

[48] Henrik Jönsson. *Risk and vulnerability analysis of complex systems: A basis for proactive emergency management*. Fire Safety Engineering and Systems Safety, 2007.

[49] G.J. Klir. *Facets of systems science*, volume 15. Springer, 2001.

[50] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4(3):382–401, July 1982.

[51] Joseph Raymond Laracy. *A systems-theoretic security model for large scale, complex systems applied to the US air transportation system*. PhD thesis, Massachusetts Institute of Technology, 2007.

[52] Nancy Leveson. *Engineering a safer world: Systems thinking applied to safety*. Mit Press, 2011.

[53] Aaron Lynch. *Thought contagion: How belief spreads through society*. Basic Books, 2008.

[54] Eden Medina. *Cybernetic Revolutionaries*. MIT Press, 2011.

[55] Robert Meushaw and Carl Landwehr. Nsa initiatives in cybersecurity science|. *http://www.nsa.gov/research/tnw/tnw194/articles/pdfs/TNW194_article4.pdf*.

[56] Jelena Mirkovic and Peter Reiher. A taxonomy of ddos attack and ddos defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2):39–53, 2004.

[57] P Oscar and VP Roychowdbury. Leveraging social networks to fight spam. *IEEE Computer*, 38(4):61–68, 2005.

[58] Romualdo Pastor-Satorras and Alessandro Vespignani. Epidemic spreading in scale-free networks. *Physical review letters*, 86(14):3200, 2001.

[59] Elisabeth Paté-Cornell. On "black swans" and "perfect storms": Risk analysis and management when statistics are not enough. *Risk analysis*, 32(11):1823–1833, 2012.

[60] M. Pease, R. Shostak, and L. Lamport. Reaching agreement in the presence of faults. *J. ACM*, 27(2):228–234, April 1980.

[61] Aiko Pras, Anna Sperotto, Giovane Moura, Idilio Drago, Rafael Barbosa, Ramin Sadre, Ricardo Schmidt, and Rick Hofstede. Attacks by "anonymous" wikileaks proponents not anonymous. 2010.

[62] Mikhail Prokopenko, Fabio Boschetti, and Alex J Ryan. An information-theoretic primer on complexity, self-organization, and emergence. *Complexity*, 15(1):11–28, 2009.

[63] Kenneth J Rothman, Sander Greenland, and Timothy L Lash. *Modern epidemiology*. Lippincott Williams & Wilkins, 2008.

[64] John Rushby. On emergent misbehavior. *http://www.csl.sri.com/users/rushby/slides/emergentm12.pdf*, 2012.

[65] Robert J Shiller. *Irrational exuberance*. Random House LLC, 2005.

[66] Adam Shostack. *Threat Modeling: Designing for Security*. John Wiley & Sons, 2014.

[67] Didier Sornette. Dragon-kings, black swans and the prediction of crises. *arXiv preprint arXiv:0907.4290*, 2009.

[68] Didier Sornette and Guy Ouillon. Dragon-kings: Mechanisms, statistical methods and empirical evidence. *The European Physical Journal Special Topics*, 205(1):1–26, 2012.

[69] John D Sterman. System dynamics modeling: Tools for learning in a complex world. *California management review*, 43(4), 2001.

[70] P Stewart. Jacobellis v ohio. *US Rep*, 378:184, 1964.

[71] J. Su, K.K.W. Chan, A.G. Miklas, K. Po, A. Akhavan, S. Saroiu, E. de Lara, and A. Goel. A preliminary investigation of worm infections in a bluetooth environment. In *Proceedings of the 4th ACM workshop on Recurring malcode*, page 16. ACM, 2006.

[72] Nassim Nicholas Taleb. Black swans and the domains of statistics. *The American Statistician*, 61(3):198–200, 2007.

[73] Nassim Nicholas Taleb. *The black swan: The impact of the highly improbable*. Random House Trade Paperbacks, 2010.

[74] Seth Tisue and Uri Wilensky. Netlogo: A simple environment for modeling complexity. In *International Conference on Complex Systems*, pages 16–21, 2004.

[75] Brian Uzzi. The sources and consequences of embeddedness for the economic performance of organizations: The network effect. *American sociological review*, pages 674–698, 1996.

[76] Koen H van Dam, Igor Nikolic, and Zofia Lukszo. *Agent-based modelling of socio-technical systems*, volume 9. Springer, 2012.

[77] Nart Villeneuve, Jessa dela Torre, and David Sancho. Asprox reborn. *http://www.trendmicro.com/media/wp/asprox-reborn-whitepaper-en.pdf*.

[78] Qian Wang, Zesheng Chen, Chao Chen, and Niki Pissinou. On the robustness of the botnet topology formed by worm infection, 2010.

[79] C. Warrender, S. Forrest, and B. Pearlmutter. Detecting intrusions using system calls: Alternative data models. *sp*, page 0133, 1999.

[80] Duncan J Watts and Steven H Strogatz. Collective dynamics of small-world networks. *nature*, 393(6684):440–442, 1998.

[81] Stephen Wolfram. Statistical mechanics of cellular automata. *Reviews of modern physics*, 55(3):601, 1983.

[82] Guanhua Yan, Hector D Flores, Leticia Cuellar, Nicolas Hengartner, Stephan Eidenbenz, and Vincent Vu. Bluetooth worm propagation: mobility pattern matters! In *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, pages 32–44. ACM, 2007.

[83] William Young and Nancy Leveson. Systems thinking for safety and security. In *Proceedings of the 29th Annual Computer Security Applications Conference*, pages 1–8. ACM, 2013.

[84] William Young and Nancy G Leveson. An integrated approach to safety and security based on systems theory. *Communications of the ACM*, 57(2):31–35, 2014.

[85] Raed Abu Zitar and Adel Hamdan. Genetic optimized artificial immune system in spam detection: a review and a model. *Artificial Intelligence Review*, 40(3):305–377, 2013.

[86] C.C. Zou, W. Gong, and D. Towsley. Code red worm propagation modeling and analysis. In *Proceedings of the 9th ACM conference on Computer and communications security*, pages 138–147. ACM, 2002.