Chapter 1
Cybersecurity Dynamics: A Foundation for the Science of Cybersecurity

Shouhuai Xu

Abstract Cybersecurity Dynamics is new concept that aims to achieve the modeling, analysis, quantification, and management of cybersecurity from a holistic perspective, rather than from a building-blocks perspective. It is centered at modeling and analyzing the attack-defense interactions in cyberspace, which cause a "natural" phenomenon — the evolution of the global cybersecurity state. In this Chapter, we systematically introduce and review the Cybersecurity Dynamics foundation for the Science of Cybersecurity. We review the core concepts, technical approaches, research axes, and results that have been obtained in this endeavor. We outline a research roadmap towards the ultimate research goal, including a systematic set of technical barriers.

1.1 Introduction

The fundamental concepts of confidentiality, integrity, and availability have been at the core of information security research over the past decades. These concepts have led to the development of many building-block techniques, such as cryptographic mechanisms, which can be rigorously analyzed in a sound scientific framework. This motivated us to seek fundamental concepts and frameworks that can guide our investigation of cybersecurity, which has to be understood from a holistic perspective (i.e., by treating a network of interest as a whole, rather than investigating their building-blocks separately).

In the course of our endeavor, the concept of *cybersecurity dynamics* emerges [125]. Intuitively, the concept of cybersecurity dynamics reflects the *evolution* of the global cybersecurity state of a network, where "evolution" is caused by the interactions between the human parties involved — dubbed *attack-defense interactions*. The human parties involved include attackers who wage attacks against a network,

Laboratory for Cybersecurity Dynamics, Department of Computer Science, University of Texas at San Antonio. Email: e-mail: `shxu@cs.utsa.edu`. Web: `www.cs.utsa.edu/~shxu`

defenders who employ defense mechanisms to protect a network in question, and users who may be exploited by the attackers to wage attacks.

The concept of Cybersecurity Dynamics is appealing because of the following. First, the global cybersecurity state of a network reflects the real-time situation, which “naturally” evolves over time because of the attack-defense interactions. Knowing the real-time global cybersecurity state or situation is of high interest to cyber defense decision-makers, who often need to adjust their defense posture (including policies, architectures, and mechanisms) to mitigate or minimize the damage of cyber attacks. Second, the effects of employing new cyber defense postures are reflected by the resulting global cybersecurity state. This means that we can compare the effectiveness of one defense posture against another. Third, looking at the evolution of the global cybersecurity state allows us to build systematic models with *descriptive* power (i.e., characterizing what phenomenon can happen under what circumstances), *prescriptive* power (i.e., guiding the adjustment to defense postures to mitigate or minimize the damage of cyber attacks), and *predictive* power (i.e., forecasting what will happen with or without making adjustments to the defense posture). Four, modeling the evolution of the global cybersecurity state makes security quantification an inherent task, which paves the way for quantitative decision-making in the course of cyber defense operations. In particular, the concept of Cybersecurity Dynamics naturally leads to the notion of *macroscopic cybersecurity*, with models that will use parameters to describe or represent (among other things) attacks and defenses.

Our contributions. The present chapter systematically refines and extends an earlier treatment of the Cybersecurity Dynamics foundation given in [128], while accommodating the many advancements that have been made during the past few years. More specifically, we systematically introduce and review the Cybersecurity Dynamics foundation (or framework), while focusing on three orthogonal, coherent “axes”: (i) the cybersecurity metrics axis aims to develop a systematic set of metrics that can adequately describe cybersecurity; (ii) the cybersecurity first-principle modeling and analysis axis aims to establish cybersecurity laws governing the evolution of the global cybersecurity state; and (iii) the cybersecurity data analytics axis aims to extract model parameters and validate/invalidate models developed in the first-principle modeling and analysis axis. In particular, we discuss the deep connections between these three axes. Despite the many efforts and significant results, there are many outstanding problems that have yet to be tackled. We hope the present chapter will inspire many more studies to address the many open problems.

Chapter outline. The chapter is organized as follows. Section 1.2 presents an overview of the Cybersecurity Dynamics foundation. Section 1.3 reviews the recent advancement in cybersecurity metrics research. Section 1.4 reviews the recent advancement in cybersecurity first-principle modeling and analysis. Section 1.5 reviews the recent advancement in cybersecurity data analytics. Section 1.6 discusses future research directions, including technical barriers that need to be tackled. Section 1.7 reviews related prior studies. Section 1.8 concludes the present chapter.

1.2 Overview of the Cybersecurity Dynamics Foundation

1.2.1 Terminology

By “network” we mean an arbitrary (cyber, cyber-physical, Internet of Things or IoT) network of interest that is enabled or interconnected by the TCP/IP technology, regardless of the underlying communication being wired or wireless. A network can have an arbitrarily large size (e.g., an enterprise network or even the entire cyberspace). By “computer” we mean a computer or device (e.g., smartphones, IoT devices) with a software stack, which typically includes some applications, library functions, and an operating system.

A network is protected by some *defenders*, who may or may not be under the same administrative jurisdiction (e.g., a network of interest consisting of multiple independently managed enterprise networks). Each network has a number of *users*, who are often subject to attacks (e.g., social-engineering attacks). The *attacker* attempts to compromise the computers in a network, by exploiting weaknesses in the network software and hardware as well as weaknesses in the users or defenders (e.g., making them become insider threats).

In the context of the present chapter, the terms *cybersecurity* and *security* are used interchangeably. In order to model cybersecurity from a holistic perspective (in contrast to building-block perspectives), we need to have the notion of *model resolution*, reflecting the level of abstraction. For example, we can treat a computer or software component as an indivisible unit, dubbed “*atoms*” of a model. Throughout the chapter, we will use the term “atom” to indicate the unit from a modeling point of view. Because each “atom” will be represented as a vertex or node in a graph-theoretic model, we also call an “atom” a *node*. When we treat a computer as a unit or “atom”, we are dealing with a coarse-grained model because the internal components of the computer are treated as transparent. As a consequence, compromise of any program in the user space of a computer would force us to treat the entire computer as compromised. When we treat a software component (e.g., software program or even program function) as an “atom”, we are dealing with a fine-grained model because the compromise of one component in a computer (e.g., application) does not necessarily mean the compromise of another component in the same computer (e.g., the operating system).

For each “atom” mentioned above, we can define its *security state*, which can be either *secure* but possibly vulnerable to attacks because it contain some vulnerabilities, or *compromised*. In the real world, the security state of an “atom” is dynamic (i.e., changing over time), rather than static, because it can become compromised (because of some attack actions), then become secure (because of some defense actions), then become compromised, and so on. This naturally leads to the view that the security state evolves. We call the security state of an “atom” a *local* cybersecurity state because it deals with an individual “atom”; we call the security state of an entire network the *global* cybersecurity state, which can be represented as a vector of the local cybersecurity states of the “atoms”.

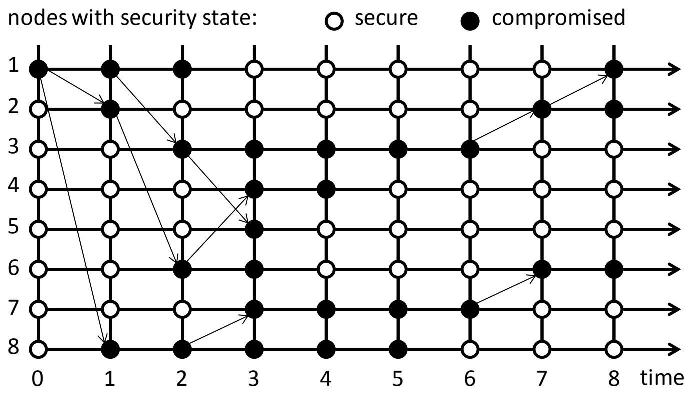

Fig. 1.1 Illustration of the evolution of the global cybersecurity state in a small network of 8 “atoms” at an appropriate model resolution. The “atoms” are represented as *nodes* (e.g., computers, devices, or software components). In the discrete-time model, each “atom” or node has a cybersecurity state at any point in time, either *secure* (represented as an empty circle) or *compromised* (represented as a filled circle) in this example. Each arrow represents a successful attack from a compromised node against a secure node, causing the latter to become compromised. A compromised node may become secure again because of some defense activities. A secure node may be attacked by multiple compromised nodes at the same time.

Figure 1.1 illustrates the evolution of the global cybersecurity state of a network, reflected by the evolution of the local cybersecurity states of individual “atoms” that are represented as “nodes” 1, . . ., 8. In this illustration, a node has two possible states at any point in time, *secure* (empty circle) or *compromised* (filled circle). A secure node may be attacked by one or multiple compromised nodes and then become compromised; a compromised node may become secure again because of some defense activities. An arrow indicates a successful attack.

1.2.2 Research Objectives

The evolution of the global cybersecurity state, as illustrated in Figure 1.1, is a *natural* phenomenon in cyberspace. The core research objectives of Cybersecurity Dynamics are centered at *understanding*, *managing* (or controlling), and *forecasting* the evolution. Understanding the evolution means we want to gain deep insights into the laws that govern the evolution. For this purpose, we need to build *descriptive* models to analyze how the attack-defense interactions govern the evolution of the global cybersecurity state. Managing the evolution means that we want to mitigate or control, if not minimize, the damage so as to benefit the defender. For this purpose, we need to build *prescriptive* models that can guide the orchestration of cyber defense activities in an optimal or cost-effective fashion. Forecasting means that we want to be able to forecast or predict the evolution so as to facilitate adaptive and/or proactive cyber defense. For this purpose, we need to build *predictive* models that can forecast, among other things, the evolution of the global cybersecurity state and the incoming threats against a network of interest.

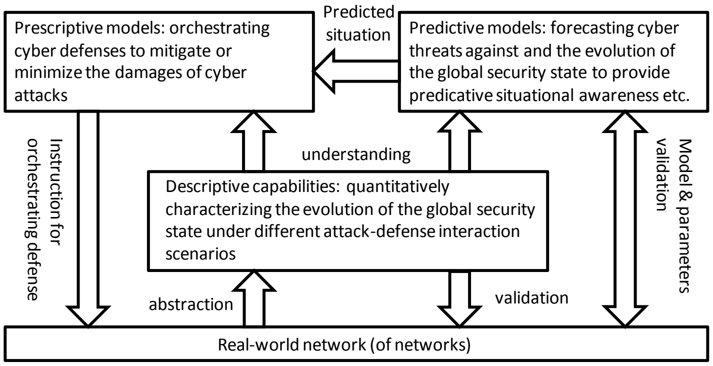
Prescriptive models: orchestrating cyber defenses to mitigate or minimize the damages of cyber attacks
Predicted situation
Predictive models: forecasting cyber threats against and the evolution of the global security state to provide predicative situational awareness etc.
Instruction for orchestrating defense
understanding
Model & parameters validation
Descriptive capabilities: quantitatively characterizing the evolution of the global security state under different attack-defense interaction scenarios
abstraction
validation
Real-world network (of networks)

Fig. 1.2 Three core research objectives of Cybersecurity Dynamics: descriptive capabilities, prescriptive capabilities, and predictive capabilities.

Figure 1.2 highlights the aforementioned three core research objectives and the relationship between them. Descriptive models are abstracted from the real-world networks by faithfully representing the attack-defense interactions. These models will be validated (or invalidated) according to real-world data or experiments. Predictive models are built on top of the description models and are also validated (or invalidated) according to real-world data. Prescriptive models are also built on top of descriptive models, while possibly taking into consideration the situations predicted or forecasted by the predictive models. The prescriptive models will guide the orchestration of cyber defense so as to benefit the defender in a cost-effective, if not optimal, fashion.

1.2.3 Scope

Figure 1.3 highlights the scope of the present chapter, which focuses on discussing three axes of Cybersecurity Dynamics research: (i) Cybersecurity metrics, which are driven by applications (e.g., for orchestrating cyber defenses to mitigate or minimize the damage of cyber attacks) and semantics (e.g., what aspects of cybersecurity would reflect the competence of cyber defense?). (ii) Cybersecurity first-principle modeling and analysis, which are driven by assumptions. First-principle models are useful in the absence of real-world data and can be inspired by the properties exhibited by real-world datasets. (iii) Cybersecurity data analytics, which are driven by real-world data or experiments.

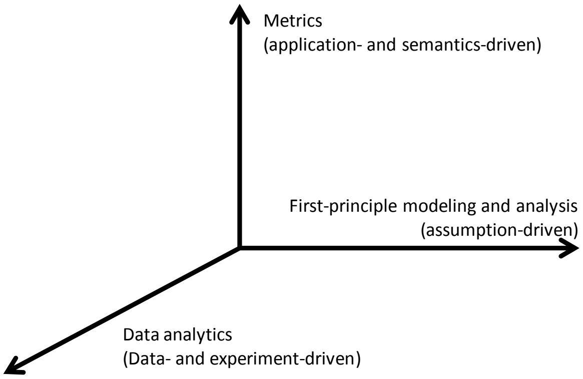
Metrics
(application- and semantics-driven)
First-principle modeling and analysis
(assumption-driven)
Data analytics
(Data- and experiment-driven)

Fig. 1.3 Scope of the present chapter: Three research axes towards achieving the research objectives of Cybersecurity Dynamics.

Figure 1.4 highlights the relationship between the three research axes. The cybersecurity metrics axis aims to rigorously define metrics to measure and quantify cybersecurity from a holistic perspective, and therefore provides conceptual guidance to the other two axes because those quantitative models are often centered at some metrics. Along this axis, significant progress has been made [94, 104, 89, 23, 18, 19, 80, 22].

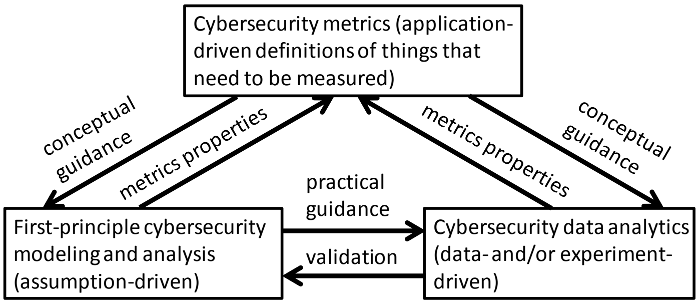
Cybersecurity metrics (application-driven definitions of things that need to be measured)
conceptual guidance
metrics properties
metrics properties
conceptual guidance
practical guidance
validation
First-principle cybersecurity modeling and analysis (assumption-driven)
Cybersecurity data analytics (data- and/or experiment-driven)

Fig. 1.4 Relationship between the three research axes.

The cybersecurity first-principle modeling and analysis axis aims to build, under appropriate assumptions, mathematical models to describe the evolution of the global cybersecurity state caused by cyber attack-defense interactions. By “first-principle” we mean the use of as-simple-as-possible models with as-few-as-possible parameters, while making as-weak-as-possible assumptions; of course, these models must make sense from a cybersecurity perspective and can be validated/invalidated (e.g., through the validation/invalidation of the assumptions they make). This axis aims to establish cybersecurity laws governing the evolution of the global cybersecurity state. For example, these first-principle models aim to derive macroscopic phenomena (or characteristics or properties) from the underlying microscopic attack-defense interactions. This axis supports the cybersecurity metrics axis by providing insights into the properties of metrics (e.g., do they converge or oscillate over time), and provides practical guidance to the cybersecurity data analytics axis (e.g., by showing that some model parameters are necessary and therefore cannot be replaced with any alternatives). Along this axis, significant progress has been made [69, 131, 133, 122, 74, 132, 28, 129, 42, 120, 130, 148, 147].

The cybersecurity data analytics axis aims to use data- and/or experiment-driven studies to obtain model parameters and validate/invalidate first-principle models. This is because first-principle models typically, and legitimately, assume away the obtaining of model parameters. This axis supports the cybersecurity metrics axis by providing insights into the properties of metrics (e.g., some metrics are hard or costly to measure, suggesting the need to define and use alternate metrics), and helps validate first-principle models (e.g., by showing that an assumption underlying a first-principle model is not valid). Along this axis, significant progress has been made [138, 139, 140, 20, 121, 95, 96, 119].

1.3 Cybersecurity Metrics

The most outstanding open problem in cybersecurity research is arguably cybersecurity metrics [88, 94, 104]. Despite its clear importance, the problem is largely open as evidenced by the fact that it has been constantly listed as one of the hard problems [27, 108, 87]. Recently, the problem has received systematic attention [94, 104, 89, 23, 18, 19, 80, 22].

In Cybersecurity Dynamics [128, 94], the following five kinds of cybersecurity metrics have been proposed to systematically describe the evolution of the global cybersecurity state [94]: (i) metrics for describing a network including its configurations; (ii) metrics for describing systems and human vulnerabilities; (iii) metrics for describing defenses employed to protect networks; (iv) metrics for describing cyber attacks (i.e., threat models); and (v) metrics for describing the global cybersecurity state or cybersecurity situational awareness.

Specifically, let $security\_state(t)$ denote the global cybersecurity state at time t, $C(t)$ denote a network of interest at time t (including its hardware and software configurations), $L(t)$ denote the vulnerabilities in the network at time t (including possibly zero-day vulnerabilities, human factors with uncertainty), $D(t)$ denote the defense posture at time t (i.e., the defense that are employed at time t to protect the network), and $A(t)$ denote the attacks that are waged against the network at time t. The framework aims to obtain families of mathematical functions, denoted by $\{f\}$, such that

$$security\_state(t) = f(C(t), L(t), D(t), A(t)). \tag{1.1}$$

Eq. (1.1), once achieved, has many applications. For example, it allows us to compare the global cybersecurity of networks deploying two different configurations, say $C(t)$ vs. $C'(t)$, or two different defense postures, say $D(t)$ vs. $D'(t)$, through the difference between the corresponding evolution of $security\_state(t)$ and $security\_state'(t)$ over time. As we will discuss later, some concrete f's have been investigated in the cybersecurity first-principle modeling and analysis axis and the cybersecurity data analytics axis.

In what follows, we discuss how to obtain mathematical representations of network configurations $C(t)$, vulnerabilities $L(t)$, defense postures $D(t)$, and threats $A(t)$. These representations naturally lead to quantitative metrics.

1.3.1 Representation of Network Configuration and Metrics

Representation. At a high level, configurations can be reflected by an *attack-defense structure*, which can be described as a graph $G(t) = (V(t), E(t))$, where $V(t)$ is the node or vertex set at time t, and $E(t)$ is the edge or arc set at time t. A node $v \in V(t)$ represents an "atom" mentioned above (e.g., a computer or software component). An edge or arc $(u, v) \in E(t)$ means that node u can attack node v, meaning that the communication from node u to node v may not be filtered, for example, by host-based intrusion prevention (when u and v belong to the same computer) or by network-based intrusion prevention (when u and v represent, or belong to, different computers). Moreover, $(u, v) \in E(t)$ means that the compromise of node u can cause the compromise of node v. Note that $E(t)$ does not necessarily represent the physical network topology in general (except perhaps for sensor networks or IoT networks where nodes can only afford to have short-range communications); in general, $(u, v) \in E(t)$ represents a communication link or path in a network. It turns out that filtering unauthorized communication relations $(u, v) \notin E(t)$ is an important defense means (see, for example, [131, 147, 18, 19]).

Recently, researchers have started to investigate how to represent networks at finer granularities [18, 19]. Suppose a network of interest is composed of $n(t)$ computers or devices at time t. In order to obtain the attack-defense structure $G(t) = (V(t), E(t))$, we need to first represent the *software stacks* on each computer or device, meaning that we need to model the applications, operating systems, and possibly library functions. Then, a computer or device, denoted by i, may be represent by a graph $G_i(t) = (V_i(t), E_i(t))$, where $v \in V_i$ represents an "atom" (e.g., application, operating system, or function), and $(u, v) \in E_i(t)$ means either u can call v (i.e., caller-callee dependence relation) or u can communicate with v (i.e., inter-application communication relation). Another edge set $E_0(t)$ may be defined to represent the authorized inter-computer communications within the network at time t. Yet another edge $E_*(t)$ may be defined to represent the authorized inter-network communication relations between the network and the external networks (i.e., internal-external communication relations). Note that $E_i(t)$ reflects a host-based access control policy (if employed), and $E_0(t)$ and $E_*(t)$ reflect network-wide access control policies (if employed). As a result, the attack-defense structure $G(t) = (V(t), E(t))$ may be derived as follows [18, 19]:

$$V(t) = V_1(t) \cup \ldots \cup V_{n(t)}(t) \text{ and } E(t) = E_1(t) \cup \ldots \cup E_{n(t)}(t) \cup E_0(t) \cup E_*(t).$$

Metrics. Having obtained the graph-theoretic representation $G(t) = (V(t), E(t))$, we may define metrics to characterize $G(t)$. For example, we may use nodes' degree distribution to characterize the structure of $G(t)$; we may characterize the evolution of $G(t)$ over time; we may quantify the difference of two defense policies by comparing the attack-defense structures resulting from their respective employments.

1.3.2 Representation of Vulnerabilities and Metrics

Representation. We propose classifying vulnerabilities into three kinds: software, hardware, and human vulnerabilities, which are all used in a broad sense.

- We use the term "software vulnerabilities" to describe the vulnerabilities in the entire software stack, including applications, library functions, and operating systems. Software vulnerabilities are the root cause of many real-world attacks. For example, the problem of *vulnerability detection* is an active research topic (see, for example, [71, 62, 70]).
- We use the term "hardware vulnerabilities" to describe the vulnerabilities in the hardware, architecture, and firmware. The number of hardware vulnerabilities is often much smaller than the number of software vulnerabilities, but the damage caused by a hardware vulnerability is often severe because of the wide use of the hardware. Two recent examples of hardware vulnerabilities are Spectre and Meltdown (see, for example, [113, 63]).
- We use the term "human vulnerabilities" to describe the vulnerabilities of the users and administrators, such as vulnerabilities to social-engineering attacks (e.g., phishing) as well as insider threats and the vulnerabilities caused by the use of weak passwords.

Each vulnerability may be associated with a set of attributes. For example, a software vulnerability may have the following attributes: (i) the privilege that is required in order to exploit the vulnerability (e.g., local access vs. remote access); (ii) what is the chance that there is a zero-day vulnerability in a software component? (iii) what is the security consequence of the exploitation of a vulnerability?

Metrics. Corresponding to these vulnerabilities, metrics need to be defined to quantify them. Two approaches have been proposed in the literature to measure software vulnerabilities, *coarse-grained* vs. *fine-grained*.

- Fine-grained approach: In this approach, vulnerabilities are considered at fine-grained granularities by separating the vulnerabilities of applications, library functions, and operating systems [18, 19].

- Coarse-grained approach: In this approach, vulnerabilities are often discussed at an aggregate level. For example, when treating a computer as an "atom", we consider the overall vulnerability of a computer, which can be aggregated from the vulnerabilities in the applications, library functions, and operating systems. This approach has been used in numerous cybersecurity first-principle models (see [131, 147] and the references therein).

Similarly, hardware vulnerabilities may be characterized by, for example, the chance that a vulnerability can be exploited; human vulnerabilities may be described by the chance that a user or defender is vulnerable to social-engineering attacks.

1.3.3 Representation of Defenses and Metrics

Representation. There are many kinds of defense mechanisms that need to be represented for modeling purposes, such as firewalls, host-based intrusion prevention/detection systems, and network-based intrusion prevention/detection systems. Moreover, access control policies also need to be represented. For example, a tight access control policy would filter or block any unauthorized communication or function call; in contrast, a loose access control policy would not filter or block any unauthorized communication or function call, which can happen when some "atoms" are compromised. For modeling purposes, we classify defenses into preventive, reactive, proactive, adaptive, and active defenses.

- Preventive defenses aim to prevent attacks from succeeding or even reaching the target of interest. Mechanisms such as whitelisting, access control, and firewall are examples of preventive defenses.
- Reactive defenses aim to detect successful attacks and "clean up" their damage. Mechanisms such as anti-malware tools are examples of reactive defenses.
- Adaptive defenses aim to dynamically adjust the defense posture so as to mitigate or disrupt ongoing attacks that have been detected by the defender. Examples include the use of Software-Defined Networking (SDN) technology to change network configurations, or route network traffic through dynamically employed network security tools such as firewalls and intrusion prevention/detection systems. A concrete example for protecting systems with known, but unpatched, vulnerabilities is shown in [17].
- Proactive defenses aim to dynamically adjust the defense posture so as to mitigate or disrupt attacks, whose presence is not necessarily known to the defender. Mechanisms such as Moving Target Defense (MTD) are examples of proactive defenses.
- Active defenses aim to deploy defense mechanisms (or defenseware) to "patrol" networks to detect and clean up compromises. In the context of the present Chapter, active defenses are not meant to be "hacking back" because the defenseware are deployed within the boundary of the defender's network.

Metrics. Metrics need to be defined to measure the defense capabilities of a defender. For a preventive defense mechanism, we need to measure what kinds of cyber attacks that can or cannot be prevented by it. For a reactive defense mechanism, its detection capabilities can be measured by the false-positive rate, false-negative rate, and related metrics; similarly, its "cleaning" capabilities may not be perfect as well (because there is evidence showing that using multiple anti-malware tools together is not adequate to clean up malware infecting a computer [81, 97, 82, 36]). For adaptive defense, its capabilities before and after an adaptation should be different (e.g., in terms of both attack-prevention and attack-detection capabilities). For proactive defense, its capabilities can be measured by the extent to which the compromised nodes can be cleaned by such mechanisms. For active defense, its capabilities can be measured by what kinds of attacks can be detected and cleaned up by such mechanisms.

1.3.4 Representation of Attacks and Metrics

Representation. There are many kinds of cyber attacks, which can be characterized from multiple perspectives. From the perspective of *attack freshness*, which often reflects the *attack evasion capability*, we can classify attacks into the following categories:

- Zero-day attacks: These attacks can be further divided into two sub-categories, depending on the freshness of the vulnerabilities they exploit.
  - Zero-day attacks exploiting zero-day vulnerabilities: These attacks exploit zero-day vulnerabilities which are not known to anyone but the attacker, the exploit writer, or the entity that discovered the vulnerability. These attacks are often difficult to detect, let alone prevent. These attacks can also accommodate the exploitation of newly compromised employees as *insider threats*.
  - Zero-day attacks exploiting known vulnerabilities: These attacks exploit known, but unpatched, vulnerabilities, while possibly able to evade any existing defense systems (e.g., intrusion prevention/detection systems).
- Known attacks: These attacks are recognized by defense systems and therefore can be blocked before they cause any damage or detected after they penetrate into computers or devices.

From the perspective of *attack behaviors*, which often reflect the characteristics of attackers, we can classify attacks into the following categories:

- Machine-waged attacks: These attacks are largely waged by machines and are largely automated.
  - Push-based attacks: These attacks actively seek to compromise other computers or devices [131]. Examples of these attacks are computer malware,

  - which actively search for vulnerable victims. Social engineering attacks also fall into this category.
  - Pull-based attacks: These attacks passively wait to compromise other computers or devices [131]. Examples of these attacks are “drive-by download” by which a malicious web server waits for connections from vulnerable browsers and then compromises the latter [102].

- Human-waged attacks: These attacks are largely waged by human attackers and are largely manual.
  - Advanced Persistent Threats (APTs): These attacks are often waged by patient attackers targeting high-value assets. These attacks are often carefully planned.
  - Insider Threats: These attacks are largely waged by compromised users who are authorized with some privileges. These attackers are often victims of social engineering attacks, but are aware of their own malicious activities (in contrast to other victims of social engineering attacks, such as those who are lured to double-click a malicious email attachment or access a malicious website).

From the perspective of *attack objectives*, we can classify attacks into the following categories:

- Attacks against confidentiality: These attacks attempt to compromise the confidentiality of data, either during transmission, which is possible when the cryptographic protection mechanisms or protocols are flawed, or during storage in computer memory or disks, which is possible by penetrating into the computers [24, 43, 91, 41] or using side-channel attacks [64].
- Attacks against integrity: These attacks attempt to compromise the integrity of data, either during transmission, which is possible when the cryptographic protection mechanisms or protocols are flawed, or during storage in computer memory or disks, which is possible (for example) when the storage provider is malicious (see, e.g., [55, 142, 143, 145, 146, 144]).
- Attacks against availability: These attacks attempt to make services unavailable to their users [50]. These attacks are often waged by many compromised computers or devices, such as botnets [141, 29, 67, 58].

Faithful threat or attack models are important. For example, both random and targeted deletions of nodes from computer networks [6] oversimplifies real-world attacks [109, 114].

Metrics. Many kinds of metrics can be defined to measure attack capabilities, such as (i) the *exploits* that can be used by the attacker; (ii) the *agility* of the attacker, and (iii) the *strategy* that can be used by the attacker.

- Characterizing exploits: An exploit can be described by its attributes, such as: whether it exploits a zero-day vulnerability or an unpatched but known vulnerability.

- Characterizing attack agility: This attribute aims to describe how active and agile the attacker is. For example, one attacker may only reactively update its exploits after the defender updates its defenses. The first study at modeling and quantifying the agility of attackers is reported in [80], which presents a metrics framework for transforming well-defined security metrics (e.g., false-positive rate and false-negative rates) to measure attacker agility.
- Attack strategies: Examples of attack strategies are Lockheed Martin's Cyber Kill Chain [51] and Mandiant's Attack Life Cycle [78]. A general attack strategy may include the following phases: reconnaissance, weaponization, initial compromise, further reconnaissance, privilege escalation, and lateral movement. At each phase, metrics need to be defined to measure the attack capabilities.

1.3.5 Security State Metrics

For any model resolution (e.g., treating a computer/device as an atom vs. treating a software component as an atom), the security state of an "atom" can be in one of multiple states, such as *secure* vs. *compromised*, denoted by

$$security\_state(atom,t) = \begin{cases} 0 \text{ the ``atom'' is in \textit{secure} state at time } t \\ 1 \text{ the ``atom'' is in \textit{compromised} state at time } t \end{cases}$$

Therefore, at any point in time, the *global* cybersecurity state can be defined as

$$global\_security(t) = \frac{\text{the number of ``atoms'' in the \textit{compromised} state at time } t}{\text{the total number of ``atoms'' at time } t},$$

while noting that the total number of "atoms" can dynamically evolve. This is arguably one of the most fundamental metrics and has been the center of numerous cybersecurity first-principle models [128].

1.4 Cybersecurity First-Principle Modeling and Analysis

At a high level, cybersecurity first-principle modeling aims to design and characterize the various kinds of mathematical functions f illustrated in Eq. (1.1). Several kinds of f's have been proposed to describe different kinds of attack-defense interactions and the resulting dynamics [128]: preventive and reactive cyber defense dynamics [69, 131, 133, 122, 74, 120, 147, 18, 19, 38]; adaptive cyber defense dynamics [132, 28]; proactive cyber defense dynamics [42]; and active cyber defense dynamics [130, 148].

1.4.1 Preventive and Reactive Cyber Defense Dynamics

The systematic preventive and reactive cyber defense dynamics model presented in [68] accommodates arbitrary, but time-independent, attack-defense structures $G = (V, E)$, push-based attacks (e.g., malware spreading), and pull-based attacks (e.g., drive-by download). The analytic result presented in [131] gives a sufficient condition (i.e., a specific parameter regime) under which the dynamics converge to a unique equilibrium, namely $\Pr(global\_state(t \to \infty) = 0) = 1$, meaning that all compromises will eventually be cleaned up. However, the properties of the dynamics in parameter regimes other than the specific regime characterized in [131] are not known until [147], which proves that the dynamics are *globally stable* in the *entire* parameter universe (i.e., the dynamics always converges to a unique equilibrium). This result remains true if the model parameters are extended to be node-dependent (i.e., different nodes $v \in V$ exhibit different cybersecurity characteristics, such as different host-based intrusion prevention/detection capabilities), and/or edge-dependent (i.e., different edges $e \in E$ exhibit different cybersecurity characteristics, such as different network-based intrusion prevention/detection capabilities) [147]. Moreover, the convergence speed is proven [147] to be exponential, except for a very special parameter regime (within which the dynamics converge polynomially). Although there is no closed-expression for the unique equilibrium, upper and lower bounds of the equilibrium can be obtained [131, 147]. Another important insight, which shows the value of theoretic studies, is that there is a practical statistical method that can be used to estimate the global cybersecurity state at equilibrium *without* knowing the model parameters, thanks to the global stability of the dynamics [131, 147].

The investigations mentioned above make the *independence* assumption that cyber attacks are waged independently of each other, which may not be the case when attacks are coordinated [127]. This highlights the importance of weakening, if not eliminating, the independence assumption. Initial results have been reported in [122, 28, 120]. An important finding is that assuming away the due dependence can lead to results that are unnecessarily restrictive, if not incorrect. Since the aforementioned dependence can be caused by multiple cyber attackers, preventive and reactive cyber defense dynamics have been extended to investigate the effect of multiple cyber attackers [133], which may even fight against each other. This leads to an interesting insight: the defender can leverage one attacker, say Alice, to “defeat” another attacker, say Bob, when the defender can more effectively defend against Alice than Bob.

In summary, we have a pretty deep understanding of preventive and reactive cyber defense dynamics. For example, the effectiveness of preventive and reactive cyber defenses is limited by a fundamental attack-defense asymmetry: the attack consequence is automatically amplified by a network effect reflected by the largest eigenvalue (in modulus) of the attack-defense structure G; in contrast, the defense effectiveness is not amplified by any network effect. This attack-defense asymmetry highlights the importance of enforcing strict network access control policy (e.g., direct communication between computers is allowed only when missions demand it), which effectively reduce the largest eigenvalue.

1.4.2 Adaptive Cyber Defense Dynamics

Cyber defense is often adaptive because the defender needs to adapt to the evolution of cyber attacks. Adaptive cyber defense dynamics have been investigated in [132, 28] while considering arbitrary attack-defense structure $G = (V, E)$. In [132], both semi-adaptive defenses (i.e., the defender dynamically adjusts the defense, but not necessarily geared towards the evolution of cyber attacks) and fully-adaptive defenses (i.e., the defender dynamically adjusts the defense geared towards the observed evolution of cyber attacks) are investigated. Adaptive control strategies can be used to force the dynamics to follow a trajectory that benefits the defender (e.g., forcing the dynamics to converge to a certain equilibrium). In [28], a new approach is proposed to model adaptive cyber defense dynamics with adaptive cyber attacks. An interesting finding is that the global cybersecurity state is relatively easy to quantify when the defense is either highly effectively or highly ineffective.

In summary, both cyber attacks and defenses are often adaptive, but they are challenging to to model and analyze mathematically. For example, the intuitive concept of *adaptation agility* needs to be systematically investigated, with an initial effort presented in [80].

1.4.3 Proactive Cyber Defense Dynamics

Adaptive defenses may rely on the successful detection of attacks. Proactive defense does not suffer from this restriction because the defender can adjust the defense regardless of whether there are successful attacks or not. Moving-Target Defense (MTD) is a popular example of proactive defense. Many MTD techniques have been proposed (see, e.g., [90] and the numerous references therein) and many aspects of MTD have been investigated (see, e.g., [52, 103, 77, 25, 84]). However, very few studies have aimed at systematically quantifying the effectiveness of MTD. In what follows we outline a systematic use of MTD.

As highlighted in Figure 1.5, MTD can be employed at one or multiple layers of the software stack. Specifically, MTD can be employed at the operating system/hypervisor layer by frequently changing the underlying operating system/hypervisor environment (e.g., using VM migration). Anonymous communication can be leveraged to disrupt the attacker's reconnaissance capabilities by degrading the attacker's capability from waging *targeted and adaptive attacks* to *random attacks* [123, 124]. This means that anonymous communication can be leveraged for MTD to substantially increase the attacker's reconnaissance effort by dynamically adjusting, for example, the underlying anonymous communication infrastructure. At

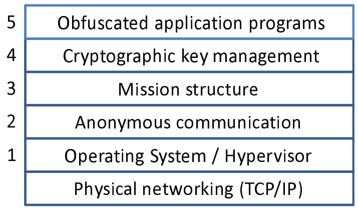
5
Obfuscated application programs
4
Cryptographic key management
3
Mission structure
2
Anonymous communication
1
Operating System / Hypervisor
Physical networking (TCP/IP)

Fig. 1.5 An example architecture showing that MTD can be employed at different layers, individually or collectively.

the *mission structure* layer, "mission structure" may be represented by a subgraph $G_M(t) = (V_M(t), E_M(t))$ of the aforementioned attack-defense structure $G(t) = (V(t), E(t))$ with $V_M(t) \subseteq V(t)$ and $E_M(t) \subseteq E(t)$. In order to prevent the attacker from identifying a target node (e.g., the cyber command-and-control center), the defender can frequently relocate the target node. At the cryptographic key management layer, proactive cryptosystems [46], key-insulated cryptosystems [33, 34, 35], or leak-free cryptosystems [31, 32] can be used to tolerate the compromise of some computers, which hold some short-lived cryptographic key or cryptographic key shares [30, 123, 124]. Moreover, dynamic re-keying (e.g., [126, 149]) can be frequently enforced even in the absence of *detected* compromises because this can make the compromised cryptographic keys useless or can increase the chance that the compromise is detected [134]. At the application layer, the defender can use the following kinds of MTD to slow down the attacker: (i) re-obfuscating the application programs frequently; (ii) dynamically re-shuffle honeypot IP addresses within a production network to capture new attacks [75].

While it is intuitive that MTD can be employed at each of these five layers, the main question is: When should the defender employ MTD and at which layers? Towards answering this question, the first systematic quantification study is presented in [42], which uses cybersecurity dynamics to quantify the effectiveness of MTD. However, the investigation treats MTD as a means, rather than a goal. That is, the effectiveness of MTD is *indirectly*, rather than directly, measured in [42]. In summary, proactive cyber defense is one of the very few approaches that can potentially defend against sophisticated attacks, such as zero-day attacks and Advanced Persistent Threats (APTs). More research needs to be done in order to systematically and directly quantify the effectiveness of proactive defense, including MTD.

1.4.4 Active Cyber Defense Dynamics

In the context of this chapter, active cyber defense means the use of "defenseware" (e.g., white worms or "malware killer" programs) to detect and clean up compromised computers. That is, active cyber defense is not about hacking back because it is employed within the administrative boundary of the network in question. The systematic modeling study of active cyber defense dynamics is initiated in [130], which formulates a mathematical model to quantify the effectiveness of active cyber defense. In active cyber defense dynamics, we need to consider a pair of attack-defense structures, denoted by $G_A(t) = (V_A(t), E_A(t))$ and $G_D(t) = (V_D(t), E_D(t))$. Note that $G_A(t)$ is centered at the attacker's point of view, and $G_D(t)$ is centered at the defender's point of view, while noting that it is possible that $G_A(t) = G_D(t)$. This leads to the identification of the optimal $G_D(t)$ under certain circumstances. In particular, it is shown [130] that active cyber defense can benefit the defender substantially by eliminating the aforementioned asymmetry, which is inherent to preventive and reactive cyber defense dynamics.

In [74], further investigation is conducted to identify optimal strategies for orchestrating active cyber defense against non-strategic or strategic attackers. In order to effectively defend against a non-strategic attacker, two flavors of optimal control strategies are investigated (i.e., *infinite-time horizon* control vs. *fast* control), by showing when the defender should adjust its active defense (including the extreme case of giving up the use of active defense, and instead using other kinds of defenses). In order to effectively defend against a strategic attacker, we identify Nash equilibrium strategies, while considering factors such as whether or not the attacker is willing to expose its advanced or zero-day attacks (exposure implying likelihood that these attacks will soon become useless).

In [148], it is shown for the first time that active cyber defense dynamics can exhibit *bifurcation* and *chaos*. Their cybersecurity implications include (i) it is not feasible or possible to seek to predict active cyber defense dynamics under certain circumstances, such as those reported in [148]; (ii) the defender should seek to manipulate active cyber defense dynamics to avoid such "unmanageable" situations. In summary, the defender can use active cyber defense to offset the asymmetry advantage of the attacker in preventive and reactive cyber defense dynamics. However, active cyber defense is no panacea, and should be used together with other kinds of defenses [74]. Additional research needs to be conducted to deepen our understanding of active cyber defense dynamics.

1.5 Cybersecurity Data Analytics

Like cybersecurity first-principle modeling and analysis, cybersecurity data analytics is also centered at some well-defined cybersecurity metrics. However, cybersecurity data analytics is complementary to the cybersecurity first-principle modeling and analysis because the former is data- and experiment-driven (rather than assumption- or semantics-driven). More specifically, cybersecurity data analytics aim to achieve a range of objectives, including: (i) obtaining model parameters used by cybersecurity first-principle models, (ii) validating or invalidating the assumptions made by cybersecurity first-principle models, and (iii) helping tackle the *transient behavior* of cybersecurity dynamics. The state-of-the-art is that significant progress has been made in the aforementioned objectives (i) and (iii), which are reviewed below, but not in (ii) due to the lack of real-world datasets.

1.5.1 Obtaining Model Parameters

Measuring the attack-defense structure $G(t)$. In cybersecurity first-principle modeling and analysis, obtaining the attack-defense structure $G(t)$ is typically, and legitimately, treated as an orthogonal effort because it copes with a different aspect of the cybersecurity problem. As discussed above, researchers have recently started to investigate how to represent networks and computers at finer-grained granularities [18, 19]. Once a modeling resolution is determined, we need to represent the software stack (including applications and operating systems) of individual computers, represent individual computers as well as the dependence and communication relations within individual computers, represent the communication relations between computers (e.g., which computer or application is authorized to communicate with which other computer or application in a network), and represent the communication relation between a network and its external environment networks.

Measuring susceptibility of software systems. Cybersecurity first-principle models often assume parameters describing the *susceptibility* of an "atom" (e.g., computer or software component). In order to measure this parameter or metric, we need to measure the vulnerability of the "atom". From the perspective of software vulnerability, we need to measure to what extent a software program is vulnerable and susceptible to exploits. For this purpose, we need to understand and characterize the capabilities of *vulnerability detection* capabilities. For example, static analysis of software source code is one approach to detecting vulnerabilities. This approach can be further divided into two methods: *code similarity-based* [62, 71] vs. *pattern-based* [3, 1, 2, 40, 85, 137, 136, 70, 72]. The former method is effective in detecting vulnerabilities caused by certain kinds of code cloning [70]. Pattern-based methods are not limited to detecting clone-caused vulnerabilities. Pattern-based detection methods detect vulnerabilities at a coarse granularity, such as at the level of individual programs [40], individual components [85], individual files [111], or individual functions [135, 136]. More recent studies focus on fine-grained vulnerability detection [62, 71, 70, 72]. Studies in vulnerability detection represent a first step towards quantifying the susceptibility of software systems.

Measuring defense capabilities. The accurate measurement of defense capabilities is one outstanding open problem. For example, most existing measurements often assume the availability of the ground truth in question. In the real world, ground truth is difficult to obtain. Therefore, it is important to investigate to what extent we can get rid of the ground truth, if possible at all. In the context of evaluating the detection capabilities of malware detectors, this problem has been investigated in [57, 36, 16]. In particular, statistical estimators are designed and evaluated in [36]. Moreover, relative accuracy of malware detectors, rather than absolute accuracy, can be estimated under much weaker assumptions [16].

Quantifying attack capabilities. In order to measure the capabilities of pull-based attacks (e.g., drive-by download), it is necessary to measure the extent at which malicious websites can evade detection systems. There have been many proposals for detecting malicious websites (see, e.g., [76, 117]). However, the open problem is that the attacker, who knows the detection model or the dataset from which the model is learned, can manipulate the malicious websites to evade the detection systems in question. The investigation of this problem is initiated in [118], but there are no satisfactory solutions yet. For example, the proactive training approach used in [118] can only make the detection accuracy around 70-80%, which is far from sufficient.

1.5.2 *Tackling the Transient Behavior Barrier*

Towards ultimately tackling the transient behavior barrier, Figure 1.6 highlights the "grey-box" statistical methodology initiated in [138]. The term "grey-box" means that the methodology first aims to characterize the statistical properties exhibited by the data (e.g., long-range dependence, extreme value, dependence, burstiness), and then uses these properties to guide the development of prediction models.

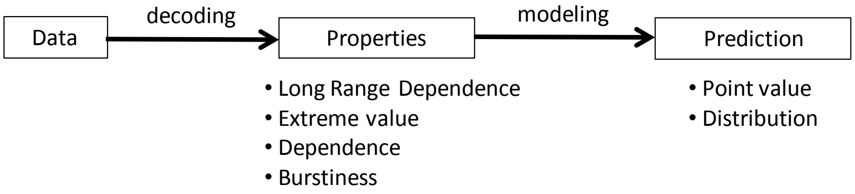

Fig. 1.6 The "grey-box" statistical methodology for cybersecurity data analytics.

Progress in coping with cybersecurity univariate time series. A particular kind of univariate time series, dubbed *stochastic cyber attack processes*, has been substantially investigated [138, 140, 20, 95, 119]. These cyber attack processes describe the number of cyber attacks or incidents against a target of interest (e.g., a network, a computer, or even a particular port). Specifically, leveraging the statistical property known as *long-range dependence*, which is exhibited by the stochastic cyber attack processes corresponding to a dataset collected at a honeypot, the "grey-box" methodology leads to an 80% accuracy in forecasting the number of attacks coming to a network one hour ahead of time. By further extending the model to accommodate the *extreme values* exhibited by the dataset, the one-hour ahead forecasting accuracy is improved to 88% [140]. A preliminary analysis of the spatiotemporal predictability shows that the forecasting upper bound is around 93% [20]. Focusing on the extreme values only, a *marked point process* model is developed to forecast the distribution of extreme values with good accuracy [95]. Another study focuses on the statistical analysis of breach incidents occurring between 2005-2017 [119], which shows that in contrast to previous beliefs, both the inter-arrival times and the breach sizes of hacking breach incidents should be described using stochastic processes, rather than probabilistic distributions, because of the autocorrelations exhibited by the data. These properties can be exploited to build accurate forecasting models [119]. These results evidently show predictability in cyberspace, at least from the perspectives that have been explored.

Progress in coping with cybersecurity multivariate time series. Many cybersecurity datasets can be represented by multivariate time series. The first investigation of this kind is to characterize and forecast the effectiveness of cyber defense early-warnings [121]. The idea of early-warning is to filter the cyber attacks, which are detected at cyber defense instruments (e.g., honeypot [101] or network telescope [12]) or third parties [75], against a network of interest. A unique research challenge, when compared with univariate time series, is to cope with the dependence between the time series, which manifests the dependence barrier [128] from a statistical perspective. For this purpose, the copula technique [54] turns out to be useful. A more general investigation of multivariate time series of cyber risks is conducted in [96]. The idea is to use a Copula-GARCH model to describe the multivariate dependence between stochastic cyber attack processes. In [121, 96], it is shown that assuming away the due dependence between stochastic cyber attack processes (i.e., the time series) can cause a severe underestimation of cybersecurity risks.

Progress in coping with cybersecurity multivariate time series. Many cybersecurity datasets can be represented by graph time series. A concrete example is the reconnaissance behaviors of cyber attackers, which can be represented as a time series of bipartite graphs [139, 39], which reflects one particular kind of the aforementioned attack-defense structure $G(t) = (V(t), E(t))$ over time t. For studying such time series of graphs, a systematic methodology is presented in [39]. At a high level, the methodology is to characterize the evolution (i.e., time series) of the *similarity* between two adjacent graphs $G(t)$ and $G(t+1)$, where the notion of *similarity* can have many different definitions (leading to various kinds of analyses). Using a real-world dataset, it is shown, among other things, that a couple of time resolutions are sufficient to accommodate and describe the *temporal* characteristics of these time series. This finding offers an effective guideline in coping with real-time data streams of this kind in real-world defense operations.

1.6 Future Research Directions

In this section we discuss future research directions with respect to the three axes mentioned above.

1.6.1 Cybersecurity Metrics

Towards ultimately tackling the problem cybersecurity metrics, the following two outstanding issues need to be resolved as soon as possible.

- Identifying a systematic, ideally complete, set of metrics that must be measured: Although many metrics have been proposed in the literature [94], the state-of-the-art is still that we do not know which metrics are essential to define and measure. This is because most existing metrics are introduced simply because they can be measured; in contrast, we need to know what metrics must be measured [98]. A fundamental question is: What kinds of metrics have to be measured in order to quantify cybersecurity? Therefore, we need to know a systematic set of metrics that can adequately describe cybersecurity. Better yet, it is important to know if there is a *complete* set of metrics, where "complete" means that any metric of interest can be derived from this set of metrics. Moreover, it is important to investigate the cost for measuring each of these metrics. This is because if one metric is costly to measure, we may need to seek easy-to-measure, alternate metric(s) to replace the hard-to-measure one as long as the former can answer the same kinds of questions as the latter does.
- Investigating mathematical properties of cybersecurity metrics and the operators that can be applied to them: As mentioned above, cybersecurity can be characterized at multiple model resolutions, reminiscent of the idea of considering security at multiple layers of abstractions [66]. Ideally, cybersecurity metrics at a lower model resolution (i.e., a higher level of abstraction or macroscopic cybersecurity) should be a mathematical function of the cybersecurity metrics defined and measured at some higher model resolutions (i.e., lower levels of abstractions or microscopic cybersecurity). This incurs the issue of aggregating lower levels of cybersecurity metrics into higher levels metrics [94, 99, 98]. For this purpose, we need to investigate the mathematical properties that should be satisfied by cybersecurity metrics, including axiomatic properties.

1.6.2 Cybersecurity First-Principle Modeling and Analysis

The following unique set of technical barriers need to be adequately tackled.

- The *scalability* barrier [128]: A first-principle, native approach to modeling the evolution of global cybersecurity state caused by the attack-defense interactions would be Stochastic Process models, which would incur an exponentially-large state space that is not tractable in general. How can this problem be tackled while preserving information in the model as much as possible? The current approach is to use mean-field style treatment, which essentially reduces the number of dimensions from exponentially-many to a number of dimensions that is proportional to the size of the attack-defense structure $G(t)$.

- The *nonlinearity* barrier [128]: It has been hypothesized that Cybersecurity Dynamics models are often highly nonlinear. The lack of real-world data has hindered the validation or rejection of this hypothesis, while many researchers believe in the nonlinearity. Coping with nonlinearity is a well known hard problem in general.
- The *dependence* barrier [128]: The security states of the "atoms" are not independent of each other because, for example, some software may have the same vulnerabilities. It is an outstanding open problem to cope with the dependence between the security state of the "atoms" (i.e., random variables), for which initial progress has been made as mentioned above [120, 28].
- The *structural dynamics* barrier [128]: The attack-defense structure $G(t)$ itself evolves over time. There have been some studies on accommodating specific kind of evolutions (see, for example, [42]). However, we need to establish a mathematical description of general evolution of $G(t)$.
- The *transient behavior* barrier [128]: Existing first-principle models often analyze the asymptotic behaviors of Cybersecurity Dynamics as time $t \to \infty$ (i.e., for sufficiently large t). For cybersecurity purposes, it is perhaps even more interesting to characterize the evolution of the global cybersecurity state before the dynamics converge to an equilibrium, if it does at all. This manifests the importance of cybersecurity data analytics. Despite the progress reviewed above (e.g., [138, 140, 20, 95, 119]), our understanding of the problem is still at the infant stage.
- The *uncertainty* barrier: Cybersecurity first-principle modeling often assumes the availability of complete information, meaning that the model parameters can be obtained precisely. This represents a first-step in building analytic models for baseline understanding. In practice, model parameters may not be known or may not be precisely measured, which highlights the importance of quantifying the consequences caused by uncertainties in the models and/or parameters.
- The *deception* barrier: In the cybersecurity domain, data or information not only can be missing or noisy, but also can be malicious because the attacker can intentionally inject or manipulate the measurements in question to mislead the defenders. This kind of deceptive data/information needs to be rigorously treated.
- The *human factor* barrier: The degrees that human users or defenders are vulnerable to social-engineering attacks need to be measured and quantified.

1.6.3 Cybersecurity Data Analytics

The following research problems need to be adequately resolved as soon as possible.

- Building a full-fledged statistical methodology to forecast holistic cybersecurity situational awareness, including the the emergence of software zero-day vulnerabilities and attacks exploiting them. Although the studies reviewed above already showed the feasibility of predicting cybersecurity situational awareness from certain specific perspectives [138, 139, 140, 20, 121, 95, 96, 119], these results only represent a first step towards the ultimate goal.

- Tackling the dependence barrier as manifested in cybersecurity data analytics. Cybersecurity data can have extremely high dimensions, while dependence can be inherent to them. Therefore, we need to investigate forecasting models that can adequately accommodate the dependence "encoded" in real-world data. The results mentioned above [121, 96] only address a small tip of the iceberg.

1.7 Related Work

Prior studies related to the Cybersecurity Dynamics foundation. The present chapter systematically refines and extends an earlier treatment of Cybersecurity Dynamics [128], while accommodating the many advancements during the past few years. Although there have been investigations on exploring the various aspects (or characteristics) of the science of cybersecurity [44, 106, 107, 112, 65], to the best of our knowledge, we are the first to systematically map out a concrete framework as reviewed in the present chapter.

Prior studies related to cybersecurity metrics. There are several recent surveys related to cybersecurity metrics [94, 104, 89, 22]. Moreover, the problem has been rejuvenated by new efforts [94, 104, 89, 23, 18, 19, 80, 22]. We treat cybersecurity metrics systematically, as highlighted in Eq. (1.1), for describing the configurations of networks, for describing systems and human vulnerabilities, for describing defense postures, for describing cyber attacks (i.e., threat models), and for describing the global cybersecurity state or cybersecurity situational awareness.

Prior studies related to cybersecurity first-principle modeling and analysis. As discussed in [128], cybersecurity first-principle modeling is inspired by multiple endeavors in several disciplines, including: (i) Biological epidemic models [79, 61, 13, 11, 47]: These models have been adapted to the Internet setting (or cyber epidemic models) since Kephart and White [59, 60]. Later efforts aim to accommodate general network structures, including power-law network structures [92, 83, 92, 93, 86, 14] and *arbitrary* network structures (e.g., [116, 37, 15, 115]). (ii) Interacting particle systems [73]: These models investigate the collective behaviors of interacting components and the phenomena that can emerge from these interactions. (iii) Microfoundation in Economics [49]: This effort aims to make connections between macroeconomic models to the underlying microeconomic models. However, the aforementioned technical barriers distinguish cybersecurity first-principle models from the models in the literature mentioned above. Moreover, it is the Cybersecurity Dynamics foundation that stresses that the attack-defense structure reflects, among other things, the access control policies that are enforced in a network, rather than the physical communication network. Furthermore, the foundation offers a unique set of cyber defense dynamics as reviewed above: preventive and reactive cyber defense dynamics, adaptive cyber defense dynamics, proactive cyber defense dynamics, and active cyber defense dynamics.

It is worth mentioning that cybersecurity first-principle models in the context of Cybersecurity Dynamics are different from the models in the context of Attack Graphs (see, for example, [100, 110, 105, 53, 10, 5, 48, 21]). This is because models in the context of Attack Graphs are *combinatorial* in nature (e.g., computing or enumerating attack paths with respect to a target); in contrast, models in the context of Cybersecurity Dynamics are *stochastic processes* in nature because they explicitly model the evolution of the global cybersecurity state over time t. This explains why these models are, as mentioned above, inspired by Biological Epidemic Models, Interacting Particle Systems, and Microfoundation in Economics, and why we can model various kinds of cyber defense dynamics.

Prior studies related to cybersecurity data analytics. There are numerous data-driven cybersecurity research activities, which however are often geared towards some specific events, attacks, or defenses. For example, honeypot-captured cyber attack data have been used for purposes of visualization [45], clustering attacks [9, 8, 7, 26], and characterizing attack behaviors such as inter-arrival times [4, 56]. In contrast, cybersecurity data analytics in the context of the present chapter is meant to become an inherent pillar of the Cybersecurity Dynamics foundation, by interacting with the other two pillars (i.e., cybersecurity first-principle modeling and analysis and cybersecurity metrics) as shown in Figure 1.4.

1.8 Conclusion

We have systematically reviewed the Cybersecurity Dynamics foundation, with emphasis on the three active research axes or pillars in cybersecurity metrics, cybersecurity first-principle modeling and analysis, and cybersecurity data analytics. We discussed the progress in each of these axes and future research directions. We hope that we have clearly and successfully conveyed the following message: This is an exciting, but challenging, research endeavor that deserves a community effort to explore. We hope the present chapter will inspire many more studies towards achieving the ultimate, full-fledged Cybersecurity Dynamics foundation for the Science of Cybersecurity.

Acknowledgements This work was supported in part by ARO Grant # W911NF-17-1-0566 and ARL Grant # W911NF-17-2-0127. The author would like to thank his mentors for their encouragement, and his collaborators (including his former and current students) for deepening his understanding of the problem and potential solutions. The author would also like to thank Lisa Ho and John Charlton for proofreading the present chapter.

References

1. Rough Audit Tool for Security, 2014. `https://code.google.com/archive/p/rough-auditing-tool-for-security/`.

2. Checkmarx, 2018. `https://www.checkmarx.com/`.
3. Flawfinder, 2018. `http://www.dwheeler.com/flawfinder`.
4. E. Alata, M. Dacier, Y. Deswarte, M. Kaaâniche, K. Kortchinsky, V. Nicomette, V. Pham, and F. Pouget. Collection and analysis of attack data based on honeypots deployed on the internet. In *Proc. Quality of Protection - Security Measurements and Metrics*, pages 79–91, 2006.
5. Massimiliano Albanese, Sushil Jajodia, and Steven Noel. Time-efficient and cost-effective network hardening using attack graphs. In *Proc. IEEE DSN'12*, pages 1–12, 2012.
6. R. Albert, H. Jeong, and A. Barabasi. Error and attack tolerance of complex networks. *Nature*, 406:378–482, 2000.
7. S. Almotairi, A. Clark, M. Dacier, C. Leita, G. Mohay, V. Pham, O. Thonnard, and J. Zimmermann. Extracting inter-arrival time based behaviour from honeypot traffic using cliques. In *5th Australian Digital Forensics Conference*, pages 79–87, 2007.
8. S. Almotairi, A. Clark, G. Mohay, and J. Zimmermann. Characterization of attackers' activities in honeypot traffic using principal component analysis. In *Proc. IFIP International Conference on Network and Parallel Computing*, pages 147–154, 2008.
9. S. Almotairi, A. Clark, G. Mohay, and J. Zimmermann. A technique for detecting new attacks in low-interaction honeypot traffic. In *Proc. International Conference on Internet Monitoring and Protection*, pages 7–13, 2009.
10. Paul Ammann, Duminda Wijesekera, and Saket Kaushik. Scalable, graph-based network vulnerability analysis. In *Proc. ACM CCS'02*, pages 217–224.
11. R. Anderson and R. May. *Infectious Diseases of Humans*. Oxford University Press, 1991.
12. M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson. Internet motion sensor: A distributed blackhole monitoring system. In *Proceedings of The 12th Network and Distributed System Security Symposium (NDSS'05)*, 2005.
13. N. Bailey. *The Mathematical Theory of Infectious Diseases and Its Applications*. 2nd Edition. Griffin, London, 1975.
14. A. Barrat, M. Barthlemy, and A. Vespignani. *Dynamical Processes on Complex Networks*. Cambridge University Press, 2008.
15. D. Chakrabarti, Y. Wang, C. Wang, J. Leskovec, and C. Faloutsos. Epidemic thresholds in real networks. *ACM Trans. Inf. Syst. Secur.*, 10(4):1–26, 2008.
16. John Charlton, Pang Du, Jin-Hee Cho, and Shouhuai Xu. Measuring relative accuracy of malware detectors in the absence of ground truth. manuscript under review, 2018.
17. Haoyu Chen, Deqing Zou, Shouhuai Xu, Hai Jin, Bin Yuan, and Yu Lu. Audy: Towards automated generation and deployment of dynamic network security rules. manuscript under review, 2018.
18. Huashan Chen, Jin-Hee Cho, and Shouhuai Xu. Quantifying the security effectiveness of firewalls and dmzs. In *Proceedings of the 5th Annual Symposium and Bootcamp on Hot Topics in the Science of Security (HoTSoS'2018)*, pages 9:1–9:11, 2018.
19. Huashan Chen, Jin-Hee Cho, and Shouhuai Xu. Quantifying the security effectiveness of network diversity: poster. In *Proceedings of the 5th Annual Symposium and Bootcamp on Hot Topics in the Science of Security (HoTSoS'2018)*, page 24:1, 2018.
20. Yu-Zhong Chen, Zi-Gang Huang, Shouhuai Xu, and Ying-Cheng Lai. Spatiotemporal patterns and predictability of cyberattacks. *PLoS One*, 10(5):e0124472, 05 2015.
21. Yi Cheng, Julia Deng, Jason Li, ScottA. DeLoach, Anoop Singhal, and Xinming Ou. Metrics of security. In *Cyber Defense and Situational Awareness*, volume 62. 2014.
22. J. Cho, S. Xu, P. Hurley, M. Mackay, T. Benjamin, and M. Beaumont. Stram: Measuring the trustworthiness of computer-based systems. manuscript in submission, 2018.
23. Jin-Hee Cho, Packtrick Hurley, and Shouhuai Xu. Metrics and measurement of trustworthy systems. In *IEEE Military Communication Conference (MILCOM 2016)*, 2016.
24. J. Chow, B. Pfaff, T. Garfinkel, K. Christopher, and M. Rosenblum. Understanding data lifetime via whole system simulation. In *Proceedings of Usenix Security Symposium 2004*, 2004.

25. Warren Connell, Daniel A. Menascé, and Massimiliano Albanese. Performance modeling of moving target defenses. In *Proceedings of the 2017 Workshop on Moving Target Defense*, MTD '17, pages 53–63, 2017.
26. G. Conti and K. Abdullah. Passive visual fingerprinting of network attack tools. In *Proc. 2004 ACM workshop on Visualization and data mining for computer security*, pages 45–54, 2004.
27. INFOSEC Research Council. Hard problem list. `http://www.infosec-research.org/docs_public/20051130-IRC-HPL-FINAL.pdf`, 2007.
28. Gaofeng Da, Maochao Xu, and Shouhuai Xu. A new approach to modeling and analyzing security of networked systems. In *Proceedings of the 2014 Symposium and Bootcamp on the Science of Security (HotSoS'14)*, pages 6:1–6:12, 2014.
29. David Dagon, Guofei Gu, Christopher P. Lee, and Wenke Lee. A taxonomy of botnet structures. In *23rd Annual Computer Security Applications Conference (ACSAC'07)*, pages 325–339, 2007.
30. Y. Desmedt and Y. Frankel. Threshold cryptosystems. In *Proc. CRYPTO 89*, pages 307–315, 1989.
31. X. Ding, G. Tsudik, and S. Xu. Leak-free group signatures with immediate revocation. In *24th International Conference on Distributed Computing Systems (ICDCS 2004)*, pages 608–615. IEEE Computer Society, 2004.
32. Xuhua Ding, Gene Tsudik, and Shouhuai Xu. Leak-free mediated group signatures. *Journal of Computer Security*, 17(4):489–514, 2009.
33. Y. Dodis, J. Katz, S. Xu, and M. Yung. Key-insulated public key cryptosystems. In Lars R. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 65–82. Springer, 2002.
34. Y. Dodis, J. Katz, S. Xu, and M. Yung. Strong key-insulated signature schemes. In *Public Key Cryptography (PKC'03)*, pages 130–144, 2003.
35. Yevgeniy Dodis, Weiliang Luo, Shouhuai Xu, and Moti Yung. Key-insulated symmetric key cryptography and mitigating attacks against cryptographic cloud software. In *7th ACM Symposium on Information, Compuer and Communications Security, ASIACCS '12*, pages 57–58, 2012.
36. P. Du, Z. Sun, H. Chen, J. H. Cho, and S. Xu. Statistical estimation of malware detection metrics in the absence of ground truth. *IEEE Transactions on Information Forensics and Security*, pages 1–1, 2018.
37. A. Ganesh, L. Massoulie, and D. Towsley. The effect of network topology on the spread of epidemics. In *Proceedings of IEEE Infocom 2005*, 2005.
38. Richard Garcia-Lebron, David J. Myers, Shouhuai Xu, and Jie Sun. Node diversification in complex networks by decentralized coloring). manuscript under review, 2018.
39. Richard Garcia-Lebron, Kristin Schweitzer, Raymond Bateman, and Shouhuai Xu. A framework for characterizing the evolution of cyber attacker-victim relation graphs. manuscript in submission, 2018.
40. Gustavo Grieco, Guillermo Luis Grinblat, Lucas C. Uzal, Sanjay Rawat, Josselin Feist, and Laurent Mounier. Toward large-scale vulnerability discovery using machine learning. In *Proceedings of the Sixth ACM on Conference on Data and Application Security and Privacy, CODASPY 2016, New Orleans, LA, USA*, pages 85–96, 2016.
41. Le Guan, Jingqiang Lin, Bo Luo, Jiwu Jing, and Jing Wang. Protecting private keys against memory disclosure attacks using hardware transactional memory. In *Proceedings of the 2015 IEEE Symposium on Security and Privacy*, SP '15, pages 3–19, 2015.
42. Yujuan Han, Wnelian Lu, and Shouhuai Xu. Characterizing the power of moving target defense via cyber epidemic dynamics. In *Proc. 2014 Symposium and Bootcamp on the Science of Security (HotSoS'14)*, pages 10:1–10:12, 2014.
43. K. Harrison and S. Xu. Protecting cryptographic keys from memory disclosures. In *Proceedings of the 2007 IEEE/IFIP International Conference on Dependable Systems and Networks (DSN-DCCS'07)*, pages 137–143. IEEE Computer Society, 2007.

44. C. Herley and P. C. v. Oorschot. Sok: Science, security and the elusive goal of security as a scientific pursuit. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 99–120, May 2017.
45. A. Herrero, U. Zurutuza, and E. Corchado. A neural-visualization ids for honeynet data. *Int. J. Neural Syst.*, 22(2), 2012.
46. A. Herzberg, M. Jakobsson, S. Jarecki, H. Krawczyk, and M. Yung. Proactive public key and signature schemes. In *Proceedings of the Fourth Annual Conference on Computer and Communications Security*, pages 100–110. ACM, 1997.
47. H. Hethcote. The mathematics of infectious diseases. *SIAM Rev.*, 42(4):599–653, 2000.
48. John Homer, Su Zhang, Xinming Ou, David Schmidt, Yanhui Du, S. Raj Rajagopalan, and Anoop Singhal. Aggregating vulnerability metrics in enterprise networks using attack graphs. *J. Comput. Secur.*, 21(4):561–597, 2013.
49. K. Hoover. Idealizing reduction: The microfoundations of macroeconomics. *Erkenntnis*, 73:329–347, 2010.
50. A. Hussain, J. Heidemann, and C. Papadopoulos. A framework for classifying denial of service attacks. In *Proceedings of ACM SIGCOMM'03*, pages 99–110, 2003.
51. Eric M. Hutchins, Michael J. Cloppert, and Rohan M. Amin. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. In *2011 International Conference on Information Warfare and Security*.
52. Jafar Haadi Jafarian, Ehab Al-Shaer, and Qi Duan. Openflow random host mutation: Transparent moving target defense using software defined networking. In *Proceedings of the First Workshop on Hot Topics in Software Defined Networks (HotSDN'12)*, pages 127–132, 2012.
53. S. Jha, O. Sheyner, and J. Wing. Two formal analys s of attack graphs. In *Proc. IEEE Workshop on Computer Security Foundations*, pages 49–59, 2002.
54. Harry Joe. *Dependence Modeling with Copulas*. CRC Press, 2014.
55. Ari Juels and Burton S. Kaliski Jr. Pors: proofs of retrievability for large files. In *Proc. ACM Conference on Computer and Communications Security (CCS'07)*, pages 584–597, 2007.
56. M. Kaâniche, Y. Deswarte, E. Alata, M. Dacier, and V. Nicomette. Empirical analysis and statistical modeling of attack processes based on honeypots. *CoRR*, abs/0704.0861, 2007.
57. Alex Kantchelian, Michael Carl Tschantz, Sadia Afroz, Brad Miller, Vaishaal Shankar, Rekha Bachwani, Anthony D Joseph, and J Doug Tygar. Better malware ground truth: Techniques for weighting anti-virus vendor labels. In *Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security*, pages 45–56. ACM, 2015.
58. Erhan J. Kartaltepe, Jose Andre Morales, Shouhuai Xu, and Ravi S. Sandhu. Social network-based botnet command-and-control: Emerging threats and countermeasures. In *ACNS*, pages 511–528, 2010.
59. J. Kephart and S. White. Directed-graph epidemiological models of computer viruses. In *IEEE Symposium on Security and Privacy*, pages 343–361, 1991.
60. J. Kephart and S. White. Measuring and modeling computer virus prevalence. In *IEEE Symposium on Security and Privacy*, pages 2–15, 1993.
61. W. Kermack and A. McKendrick. A contribution to the mathematical theory of epidemics. *Proc. of Roy. Soc. Lond. A*, 115:700–721, 1927.
62. Seulbae Kim, Seunghoon Woo, Heejo Lee, and Hakjoo Oh. VUDDY: A scalable approach for vulnerable code clone discovery. In *2017 IEEE Symposium on Security and Privacy*, pages 595–614, 2017.
63. Paul Kocher, Daniel Genkin, Daniel Gruss, Werner Haas, Mike Hamburg, Moritz Lipp, Stefan Mangard, Thomas Prescher, Michael Schwarz, and Yuval Yarom. Spectre attacks: Exploiting speculative execution. *CoRR*, abs/1801.01203, 2018.
64. Boris Köpf and David Basin. An information-theoretic model for adaptive side-channel attacks. In *Proc. ACM conference on Computer and communications security*, pages 286–296. ACM, 2007.
65. Alexander Kott. Towards fundamental science of cyber security. In Robinson E. Pino, editor, *Network Science and Cybersecurity*, volume 55 of *Advances in Information Security*, pages 1–13. Springer New York, 2014.

66. Butler Lampson. Practical principles for computer security, 2006.
67. Justin Leonard, Shouhuai Xu, and Ravi S. Sandhu. A framework for understanding botnets. In *Proceedings of the The Forth International Conference on Availability, Reliability and Security, ARES 2009*, pages 917–922, 2009.
68. X. Li, P. Parker, and S. Xu. Towards quantifying the (in)security of networked systems. In *21st IEEE International Conference on Advanced Information Networking and Applications (AINA'07)*, pages 420–427, 2007.
69. X. Li, P. Parker, and S. Xu. A stochastic model for quantitative security analysis of networked systems. *IEEE Transactions on Dependable and Secure Computing*, 8(1):28–43, 2011.
70. Z. Li, D. Zou, S. Xu, X. Ou, H. Jin, S. Wang, Z. Deng, and Y. Zhong. Vuldeepecker: A deep learning-based system for vulnerability detection. In *Proceedings of the 25th Annual Network and Distributed System Security Symposium (NDSS'2018)*, 2018.
71. Zhen Li, Deqing Zou, Shouhuai Xu, Hai Jin, Hanchao Qi, and Jie Hu. Vulpecker: An automated vulnerability detection system based on code similarity analysis. In *Proceedings of the 32nd Annual Conference on Computer Security Applications, ACSAC 2016, Los Angeles, CA, USA*, pages 201–213, 2016.
72. Zhen Li, Deqing Zou, Shouhuai Xu, Hai Jin, Yawei Zhu, Zhaoxuan Chen, Sujuan Wang, and Jialai Wang. Sysevr: A framework for using deep learning to detect software vulnerabilities. Manuscript in submission, 2018.
73. T. Liggett. *Interacting Particle Systems*. Springer, 1985.
74. Wenlian Lu, Shouhuai Xu, and Xinlei Yi. Optimizing active cyber defense dynamics. In *Proceedings of the 4th International Conference on Decision and Game Theory for Security (GameSec'13)*, pages 206–225, 2013.
75. Weiliang Luo, Li Xu, Zhenxin Zhan, Qingji Zheng, and Shouhuai Xu. Federated cloud security architecture for secure and agile clouds. In Keesook J. Han, Baek-Young Choi, and Sejun Song, editors, *High Performance Cloud Auditing and Applications*, pages 169–188. Springer New York, 2014.
76. Justin Ma, Lawrence K. Saul, Stefan Savage, and Geoffrey M. Voelker. Learning to detect malicious urls. *ACM TIST*, 2(3):30:1–30:24, 2011.
77. Hoda Maleki, Saeed Valizadeh, William Koch, Azer Bestavros, and Marten van Dijk. Markov modeling of moving target defense games. In *Proceedings of the 2016 ACM Workshop on Moving Target Defense*, MTD '16, pages 81–92, 2016.
78. Mandiant. Apt1 report. `https://www.fireeye.com/content/dam/fireeyewww/services/pdfs/mandiant-apt1-report.pdf`, February 16, 2013 (Accessed July 08, 2016).
79. A. McKendrick. Applications of mathematics to medical problems. *Proc. of Edin. Math. Soceity*, 14:98–130, 1926.
80. Jose Mireles, Eric Ficke, Jin-Hee Cho, Patrick Hurley, and Shouhuai Xu. Metrics towards measuring cyber agility. manuscript in submission, 2018.
81. Aziz Mohaisen and Omar Alrawi. Av-meter: An evaluation of antivirus scans and labels. In *Detection of Intrusions and Malware, and Vulnerability Assessment - 11th International Conference, DIMVA 2014, Proceedings*, pages 112–131, 2014.
82. Jose Morales, Shouhuai Xu, and Ravi Sandhu. Analyzing malware detection efficiency with multiple anti-malware programs. In *Proceedings of 2012 ASE International Conference on Cyber Security (CyberSecurity'12)*, 2012.
83. Y. Moreno, R. Pastor-Satorras, and A. Vespignani. Epidemic outbreaks in complex heterogeneous networks. *European Physical Journal B*, 26:521–529, 2002.
84. Dieudonne Mulamba and Indrajit Ray. Resilient reference monitor for distributed access control via moving target defense. In Giovanni Livraga and Sencun Zhu, editors, *Data and Applications Security and Privacy XXXI*, pages 20–40, 2017.
85. Stephan Neuhaus, Thomas Zimmermann, Christian Holler, and Andreas Zeller. Predicting vulnerable software components. In *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA*, pages 529–540, 2007.
86. M. Newman. The structure and function of complex networks. *SIAM Review*, 45:167, 2003.

87. David Nicol, Bill Sanders, Jonathan Katz, Bill Scherlis, Tudor Dumitra, Laurie Williams, and Munindar P. Singh. The science of security 5 hard problems (august 2015). `http://cps-vo.org/node/21590`.
88. David M. Nicol, William H. Sanders, and Kishor S. Trivedi. Model-based evaluation: From dependability to security. *IEEE Trans. Dependable Sec. Comput.*, 1(1):48–65, 2004.
89. Steven Noel and Sushil Jajodia. *A Suite of Metrics for Network Attack Graph Analytics*, pages 141–176. Springer International Publishing, Cham, 2017.
90. H. Okhravi, M. Rabe, T. Mayberry, W. Leonard, T. Hobson, D. Bigelow, and W. Streilein. Survey of cyber moving targets (mit lincoln lab technical report), 2013.
91. T. Paul Parker and Shouhuai Xu. A method for safekeeping cryptographic keys from memory disclosure attacks. In *First International Conference on Trusted Systems (INTRUST'2009)*, pages 39–59, 2009.
92. R. Pastor-Satorras and A. Vespignani. Epidemic dynamics and endemic states in complex networks. *Physical Review E*, 63:066117, 2001.
93. R. Pastor-Satorras and A. Vespignani. Epidemic dynamics in finite size scale-free networks. *Physical Review E*, 65:035108, 2002.
94. Marcus Pendleton, Richard Garcia-Lebron, Jin-Hee Cho, and Shouhuai Xu. A survey on systems security metrics. *ACM Comput. Surv.*, 49(4):62:1–62:35, December 2016.
95. Chen Peng, Maochao Xu, Shouhuai Xu, and Taizhong Hu. Modeling and predicting extreme cyber attack rates via marked point processes. *Journal of Applied Statistics*, 44(14):2534–2563, 2017.
96. Chen Peng, Maochao Xu, Shouhuai Xu, and Taizhong Hu. Modeling multivariate cybersecurity risks. *Journal of Applied Statistics*, 0(0):1–23, 2018.
97. Roberto Perdisci and ManChon U. Vamo: Towards a fully automated malware clustering validity analysis. In *Proceedings of the 28th Annual Computer Security Applications Conference*, ACSAC '12, pages 329–338, 2012.
98. Shari Lawrence Pfleeger. Useful cybersecurity metrics. *IT Professional*, 11(3):38–45, 2009.
99. S.L. Pfleeger and R.K. Cunningham. Why measuring security is hard. *Security Privacy, IEEE*, 8(4):46–54, July 2010.
100. Cynthia Phillips and Laura Painton Swiler. A graph-based system for network-vulnerability analysis. In *Proc. 1998 Workshop on New Security Paradigms*, NSPW '98, pages 71–79, 1998.
101. N. Provos. A virtual honeypot framework. In *USENIX Security Symposium*, pages 1–14, 2004.
102. N. Provos, D. McNamee, P. Mavrommatis, K. Wang, and N. Modadugu. The ghost in the browser analysis of web-based malware. In *Proceedings of the First Workshop on Hot Topics in Understanding Botnets (HotBots'07)*, 2007.
103. Mohammad Ashiqur Rahman, Ehab Al-Shaer, and Rakesh B. Bobba. Moving target defense for hardening the security of the power system state estimation. In *Proceedings of the First ACM Workshop on Moving Target Defense, MTD '14*, pages 59–68, 2014.
104. A. Ramos, M. Lazar, R. H. Filho, and J. J. P. C. Rodrigues. Model-based quantitative network security metrics: A survey. *IEEE Communications Surveys Tutorials*, 19(4):2704–2734, 2017.
105. Ronald W. Ritchey and Paul Ammann. Using model checking to analyze network vulnerabilities. In *Proc. IEEE Symposium on Security and Privacy*, pages 156–165, 2000.
106. Antonio Roque, Kevin B. Bush, and Christopher Degni. Security is about control: insights from cybernetics. In *Proceedings of the Symposium and Bootcamp on the Science of Security, Pittsburgh, PA, USA, April 19-21, 2016*, pages 17–24, 2016.
107. Fred Schneider. Blueprint for a science of cybersecurity. Technical report, Cornell University, May 2011. Also to appear in The Next Wave.
108. National Science and Technology Council. Trustworthy cyberspace: Strategic plan for the federal cybersecurity research and development program. `https://www.nitrd.gov/SUBCOMMITTEE/csia/Fed_Cybersecurity_RD_Strategic_Plan_2011.pdf`, 2011.

109. Y. Shang, W. Luo, and S. Xu. l-hop percolation on networks with arbitrary degree distributions and its applications. *Phys. Rev. E*, 84:031113, Sep 2011.
110. O. Sheyner, J. Haines, S. Jha, R. Lippmann, and J. Wing. Automated generation and analysis of attack graphs. In *IEEE Symposium on Security and Privacy*, pages 273–284, 2002.
111. Yonghee Shin, Andrew Meneely, Laurie Williams, and Jason A Osborne. Evaluating complexity, code churn, and developer activity metrics as indicators of software vulnerabilities. *IEEE Transactions on Software Engineering*, 37(6):772–787, 2011.
112. Jonathan M. Spring, Tyler Moore, and David J. Pym. Practicing a science of security: A philosophy of science perspective. In *Proceedings of the 2017 New Security Paradigms Workshop, NSPW 2017*, pages 1–18, 2017.
113. Caroline Trippel, Daniel Lustig, and Margaret Martonosi. Meltdownprime and spectreprime: Automatically-synthesized attacks exploiting invalidation-based coherence protocols. *CoRR*, abs/1802.03802, 2018.
114. Adam Tyra, Jingtao Li, Yilun Shang, Shuo Jiang, Yanjun Zhao, and Shouhuai Xu. Robustness of non-interdependent and interdependent networks against dependent and adaptive attacks. *Physica A: Statistical Mechanics and its Applications*, 482:713 – 727, 2017.
115. Piet Van Mieghem, Jasmina Omic, and Robert Kooij. Virus spread in networks. *IEEE/ACM Trans. Netw.*, 17(1):1–14, February 2009.
116. Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos. Epidemic spreading in real networks: An eigenvalue viewpoint. In *Proc. of the 22nd IEEE Symposium on Reliable Distributed Systems (SRDS'03)*, pages 25–34, 2003.
117. Li Xu, Zhenxin Zhan, Shouhuai Xu, and Keying Ye. Cross-layer detection of malicious websites. In *Third ACM Conference on Data and Application Security and Privacy (ACM CODASPY'13)*, pages 141–152, 2013.
118. Li Xu, Zhenxin Zhan, Shouhuai Xu, and Keying Ye. An evasion and counter-evasion study in malicious websites detection. In *IEEE Conference on Communications and Network Security (CNS'14)*, pages 141–152, 2013.
119. M. Xu, K. M. Schweitzer, R. M. Bateman, and S. Xu. Modeling and predicting cyber hacking breaches. *IEEE Transactions on Information Forensics and Security*, 13(11):2856–2871, Nov 2018.
120. Maochao Xu, Gaofeng Da, and Shouhuai Xu. Cyber epidemic models with dependences. *Internet Mathematics*, 11(1):62–92, 2015.
121. Maochao Xu, Lei Hua, and Shouhuai Xu. A vine copula model for predicting the effectiveness of cyber defense early-warning. *Technometrics*, 59(4):508–520, 2017.
122. Maochao Xu and Shouhuai Xu. An extended stochastic model for quantitative security analysis of networked systems. *Internet Mathematics*, 8(3):288–320, 2012.
123. S. Xu, X. Li, and P. Parker. Exploiting social networks for threshold signing: Attack-resilience vs. availability. In *ACM Symposium on Information, Computer and Communications Security (ASIACCS'08)*, pages 325–336, 2008.
124. S. Xu, X. Li, T. Parker, and X. Wang. Exploiting trust-based social networks for distributed protection of sensitive data. *IEEE Transactions on Information Forensics and Security*, 6(1):39–52, 2011.
125. Shouhuai Xu. Cybersecurity dynamics publications. `http://www.cs.utsa.edu/~shxu/socs/`.
126. Shouhuai Xu. On the security of group communication schemes. *Journal of Computer Security*, 15(1):129–169, 2007.
127. Shouhuai Xu. Collaborative attack vs. collaborative defense. In *Collaborative Computing: Networking, Applications and Worksharing, 4th International Conference (CollaborateCom'2008)*, pages 217–228, 2008.
128. Shouhuai Xu. Cybersecurity dynamics. In *Proc. Symposium and Bootcamp on the Science of Security (HotSoS'14)*, pages 14:1–14:2, 2014.
129. Shouhuai Xu. Emergent behavior in cybersecurity. In *Proceedings of the 2014 Symposium and Bootcamp on the Science of Security (HotSoS'14)*, pages 13:1–13:2, 2014.
130. Shouhuai Xu, Wenlian Lu, and Hualun Li. A stochastic model of active cyber defense dynamics. *Internet Mathematics*, 11(1):23–61, 2015.

131. Shouhuai Xu, Wenlian Lu, and Li Xu. Push- and pull-based epidemic spreading in arbitrary networks: Thresholds and deeper insights. *ACM Transactions on Autonomous and Adaptive Systems (ACM TAAS)*, 7(3):32:1–32:26, 2012.
132. Shouhuai Xu, Wenlian Lu, Li Xu, and Zhenxin Zhan. Adaptive epidemic dynamics in networks: Thresholds and control. *ACM Transactions on Autonomous and Adaptive Systems (ACM TAAS)*, 8(4):19, 2014.
133. Shouhuai Xu, Wenlian Lu, and Zhenxin Zhan. A stochastic model of multivirus dynamics. *IEEE Transactions on Dependable and Secure Computing*, 9(1):30–45, 2012.
134. Shouhuai Xu and Moti Yung. Expecting the unexpected: Towards robust credential infrastructure. In *Financial Cryptography and Data Security, 13th International Conference (FC'09)*, pages 201–221, 2009.
135. Fabian Yamaguchi, Felix "FX" Lindner, and Konrad Rieck. Vulnerability extrapolation: Assisted discovery of vulnerabilities using machine learning. In *5th USENIX Workshop on Offensive Technologies, WOOT'11, August 8, 2011, San Francisco, CA, USA, Proceedings*, pages 118–127, 2011.
136. Fabian Yamaguchi, Markus Lottmann, and Konrad Rieck. Generalized vulnerability extrapolation using abstract syntax trees. In *28th Annual Computer Security Applications Conference, ACSAC 2012, Orlando, FL, USA*, pages 359–368, 2012.
137. Fabian Yamaguchi, Christian Wressnegger, Hugo Gascon, and Konrad Rieck. Chucky: Exposing missing checks in source code for vulnerability discovery. In *2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13, Berlin, Germany*, pages 499–510, 2013.
138. Zhenxin Zhan, Maochao Xu, and Shouhuai Xu. Characterizing honeypot-captured cyber attacks: Statistical framework and case study. *IEEE Transactions on Information Forensics and Security*, 8(11):1775–1789, 2013.
139. Zhenxin Zhan, Maochao Xu, and Shouhuai Xu. A characterization of cybersecurity posture from network telescope data. In *Proc. of the 6th International Conference on Trustworthy Systems (InTrust'14)*, pages 105–126, 2014.
140. Zhenxin Zhan, Maochao Xu, and Shouhuai Xu. Predicting cyber attack rates with extreme values. *IEEE Transactions on Information Forensics and Security*, 10(8):1666–1677, 2015.
141. Y. Zhao, Y. Xie, F. Yu, Q. Ke, Y. Yu, Y. Chen, and E. Gillum. Botgraph: large scale spamming botnet detection. In *Proc. NSDI'09*, pages 321–334, 2009.
142. Qingji Zheng and Shouhuai Xu. Fair and dynamic proofs of retrievability. In *First ACM Conference on Data and Application Security and Privacy, (CODASPY'2011)*, pages 237–248, 2011.
143. Qingji Zheng and Shouhuai Xu. Secure and efficient proof of storage with deduplication. In *Second ACM Conference on Data and Application Security and Privacy (CODASPY'2012)*, pages 1–12, 2012.
144. Qingji Zheng and Shouhuai Xu. Verifiable delegated set intersection operations on outsourced encrypted data. In *2015 IEEE International Conference on Cloud Engineering, IC2E 2015*, pages 175–184, 2015.
145. Qingji Zheng, Shouhuai Xu, and Giuseppe Ateniese. Efficient query integrity for outsourced dynamic databases. In *Proceedings of the 2012 ACM Workshop on Cloud computing security, CCSW 2012, Raleigh, NC, USA, October 19, 2012.*, pages 71–82, 2012.
146. Qingji Zheng, Shouhuai Xu, and Giuseppe Ateniese. VABKS: verifiable attribute-based keyword search over outsourced encrypted data. In *Proc. 2014 IEEE Conference on Computer Communications (INFOCOM'2014)*, pages 522–530, 2014.
147. R. Zheng, W. Lu, and S. Xu. Preventive and reactive cyber defense dynamics is globally stable. *IEEE Transactions on Network Science and Engineering*, pages 1–1, 2017.
148. Ren Zheng, Wenlian Lu, and Shouhuai Xu. Active cyber defense dynamics exhibiting rich phenomena. In *Proc. 2015 Symposium and Bootcamp on the Science of Security (HotSoS'15)*, pages 2:1–2:12, 2015.
149. S. Zhu, S. Setia, S. Xu, and S. Jajodia. Gkmpan: An efficient group rekeying scheme for secure multicast in ad-hoc networks. *Journal of Computer Security*, 14(4):301–325, 2006.