SANDIA REPORT

2013-1259
Unlimited Release
Printed February 2013

Dynamic Defense Workshop: From Research to Practice

Sean Crosby, Justin E. Doak , Jason J. Haas, Ryan Helinski, Christopher C. Lamb

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

2013-1259
Unlimited Release
Printed February 2013

Dynamic Defense Workshop: From Research to Practice

Sean Crosby
smcrosb@sandia.gov
Threat Analysis Technologies

Justin E. (J.D.) Doak
jedoak@sandia.gov
Analytics and Cryptography

Jason J. Haas
jjhaas@sandia.gov
Cyber Education and Research

Ryan Helinski
rhelins@sandia.gov
Cyber Education and Research

Christopher C. Lamb
cclamb@sandia.gov
Requirements & Architecture

Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM, 87185

Abstract

On September 5th and 6th, 2012, the *Dynamic Defense Workshop: From Research to Practice* brought together researchers from academia, industry, and Sandia with the goals of increasing collaboration between Sandia National Laboratories and external organizations, defining and understanding dynamic, or moving target, defense concepts and directions, and gaining a greater understanding of the state of the art for dynamic defense. Through the workshop, we broadened and refined our definition and understanding, identified new approaches to inherent challenges, and defined principles of dynamic defense.

Half of the workshop was devoted to presentations of current state-of-the-art work. Presentation topics included areas such as the failure of current defenses, threats, techniques, goals of dynamic defense, theory, foundations of dynamic defense, future directions and open research questions related to dynamic defense. The remainder of the workshop was discussion, which was broken down into sessions on defining challenges, applications to host or mobile environments, applications to enterprise network environments, exploring research and operational taxonomies, and determining how to apply scientific rigor to and investigating the field of dynamic defense.

Acknowledgments

We would like to thank all the input gathered from the workshop attendees. Specifically, we are very greatful for the direction and contributions from Professors Yih-Chun Hu, Karl N. Levitt, Sam Liles, Radha Poovenderan, Neil C. Rowe, and Jared Saia. Further, we would like to thank Sil Han, Ray Parks, Roger Suppona, and Ed Talbot for their insight and direction, and the experience they brought to the workshop. We also want to thank Ann Campbell, Ben Cook, Bill Hart, Han Lin, and David White for their input and enabling the team to host this workshop. Finally, we want to thank Carmen Good and Abel Saenz for their tireless effort in making sure the workshop could take place.

Contents

Acknowledgments 4

1 Introduction 9

2 Taxonomies: Structures for DD Thought 9

2.1 Proactive vs. Reactive 10

2.2 Application Domains 10

2.3 Deterministic vs. Random 11

2.4 Defensive Models 11

2.5 Defensive Goals 12

2.6 Attacker Goals 13

2.7 Efficiency 13

2.8 Future Development 13

3 Challenges and Opportunities 14

3.1 Enterprise Networking DD 15

Challenges 15

Opportunities 16

3.2 Mobile and Host-Based DD 16

Attacks 17

Mitigations 17

4 The Path Forward: Experimentation and Evaluation 18

4.1 Experimentation Techniques 19

Human-In-The-Loop Testing 20

4.2 Metrics 20

4.3 A Competition Scenario 22

5 Conclusions 22

List of Figures

1 A Potential Structural Taxonomy 11

1 Introduction

Cyber defenders are facing an increasingly uphill battle in defending networks, computers, and devices as both the frequency and breadth of attacks in cyber space increase at a dramatic pace. Although data about attacks is often sparse in this space, the recent increase in activity has gained widespread attention. In early 2012, CNET compiled a list of "hacking events" from 2011 through the beginning of 2012, which lists attacks on governments, law enforcement agencies, and major corporations. [1] The problem has become so pandemic that many experts agree we need to adapt and function while attackers are in our networks [2]. Furthermore, the problem is spreading; as mobile devices proliferate, so is malware [3].

Dynamic defense (DD) is a promising mitigation to the problem of increasing attacker activity by increasing attackers' uncertainties. We will discuss this area as *dynamic defense*, though the term *moving target defense* is also used in the literature. We will discuss these terms and their meanings further in Section 2. Attackers benefit from the largely static nature of traditional cyber defenses. For example, the vast majority of users use email; thus, phishing is an effective entry mechanism[1] as it largely bypasses firewalls and network intrusion detection systems. Further, attackers can usually test their phishing emails against such tools. However, dynamic defense provides a means to increase attacker uncertainty by changing the environment through hiding or deception, topics we discuss below. Defenders can use these techniques to accomplish denying an attacker access to critical systems or information, or providing misinformation. However, these benefits of DD also represent new challenges to network defenders in knowing and managing their own networks.

DD is a relatively new area of research in cyber defense and thus there are many unknowns associated with it. In this document, we report on the work accomplished at the *Dynamic Defense Workshop: From Research to Practice*, held at Sandia National Laboratories on September 5-6, 2012. This workshop brought together members of academia, industry, and research labs to present, discuss, and work on problems related to DD. We present the discussion from the workshop in addressing the development of useful taxonomies to think about DD (Section 2), DD challenges (Section 3), how to advance the field through experimentation and evaluation (Section 4), and the conclusions we draw (Section 5).

2 Taxonomies: Structures for DD Thought

During the workshop, it became clear that DD techniques meant different things to different people. Some research groups were focused on large-scale honeypot-centric experiments, while others were engaged in thought experiments and theoretical analysis of possible resource contention schemes. Still, other participants were convinced that DD approaches were not worth the cost imposed on defenders. A common theme began to emerge – with-

[1] An entry mechanism that gives an attacker access to the internals of a network is sometimes referred to as longitudinal movement [4].

out a common structure upon which to discuss various approaches and potential drawbacks, effectively debating DD techniques was an awkward and unnecessarily difficult proposition.

The goals for the developed taxonomy are to drive investment and help organizations and researchers make strategic decisions with respect to future technology development. A well-defined and established taxonomy can help researchers identify potential techniques and avenues of research that have yet to be explored. It can also help organizations recognize areas in which they can invest resources and expect some kind of acceptable return. For example, while researchers may use this kind of taxonomy to help them identify areas in which they can perform new, original research, organizations would use the same taxonomy to identify mature areas of potential investment from an enterprise security perspective. Here, we show the elements of a possible taxonomy with the understanding that additional work is required to appropriately unify these various dimensions to build a coherent taxonomy.

It may be impractical from a cost and latency standpoint for large-scale human involvement in DD techniques, though some human involvement may be possible or even desirable depending on the level of the system where that involvement is to be. Thus, the taxonomy is focused on classification of automated rather than manual solutions. Peisert et al. have begun to investigate how to categorize these factors, including human elements, in a simple layered model that could be incorporated into a future taxonomy [5].

2.1 Proactive vs. Reactive

A proactive dynamic defense technique is executed on some pre-determined schedule, perhaps at random, that is not in response to any outside stimulus. Thus, these techniques do not require any kind of detection component. Reactive techniques, on the other hand, are triggered and take some kind of action based upon detection of activity of interest. A specific example of a proactive approach is changing the IP addresses of machines on a network according to some kind of schedule [6]. An example reactive approach is MOB-C, which requires the detection of attempted lateral movement that then triggers "cocooning" or switching a potential adversary from real services to one or more emulated services [7].

2.2 Application Domains

Figure 1 shows a potential division of defensive techniques by domain, breaking potential areas of DD application into network- and host-based approaches first. The network element is then subdivided into the seven layers of the OSI model. The host element is divided into system and hypervisor sub-elements to reflect typical virtual deployments as well as traditional bare-metal systems. The system elements are further separated by the security rings of applications, device drivers, and kernel elements. We chose these specific breakdowns for the purpose of discussion, though other breakdowns are possible.

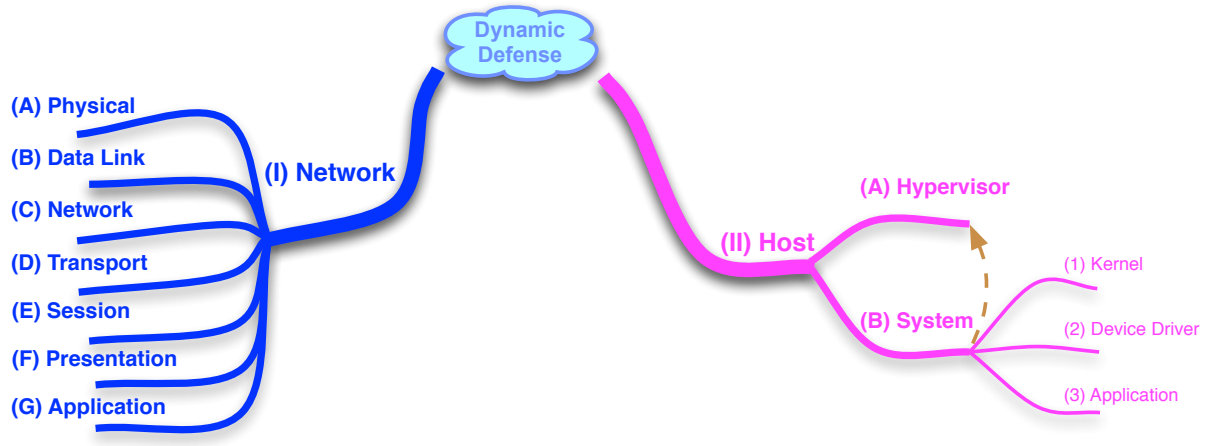

Figure 1. A Potential Structural Taxonomy

2.3 Deterministic vs. Random

Another way to characterize a DD approach is by how deterministic or random it is. Take again the example of changing IP addresses on a network in an attempt to thwart reconnaissance efforts by an adversary. A completely deterministic approach might change the IP addresses every 60 minutes. A completely random approach might make this change in some entirely unpredictable way. Some approaches might fall somewhere in the middle. For example, we might draw from a normal distribution with a mean of 60 minutes and some standard deviation to determine exactly when to change IP addresses. Deterministic approaches might be easier for defenders to manage, but their predictability will likely eventually lead to their subversion by a determined adversary. Randomized changes, on the other hand, may be less predictable and thus harder for an adversary to thwart. However, such changes may make it much more difficult for administrators to manage networks and systems.

2.4 Defensive Models

Discussions at the workshop yielded four potential models for DD, which we list below.

- Control theory and optimization
- Game theory
- Biology-inspired approaches
- Information theory

These (and perhaps other) strategies are not necessarily disjoint and can potentially be used in hybrid model configurations.

The approach of Rowe et al., for example, uses control theoretic constructs to determine an optimal defense for a given network [8]. The primary idea behind this defensive model is that thwarting attacks in progress is infeasible and automated defenses can be too risky, especially as deployed sensors may have high false positive rates. As a result, in order to effectively implement a DD system, we need to determine an optimal balance between possible degradation of operational performance for authorized users and negative effects on attackers.

Work by Gilbert et al. on resource management and scarcity as a possible approach to dynamically defending systems implies that game theory is also a potential model for building DD techniques [9]. By creating situations where rational agents compete for scarce resources, defenders are in fact designing a specific mechanism that may control the behavior of actors in a given scenario. This is the core of most game theoretic approaches, and implies that a similar analysis can be applied to DD.

Previous work using biological models shows potential application to dynamic security systems. Specifically, work drawing parallels between system defense and biological immune systems may be particularly relevant.

Finally, information theory may be a potential model for dynamic system design. An example of such a resulting system might be the use of various entropy measures to help deceive system attackers. Information theory might also provide a theoretical basis for assessing the security of the resulting systems [10].

2.5 Defensive Goals

Defender goals can also differentiate DD techniques. Defenders may place a premium on deception, or perhaps they wish to delay intruders to maximize the probability of detection or allow sufficient time to implement appropriate mitigations. Perhaps the defenders are most interested in deterring adversaries thereby causing them to move on to the next target.

Deception-centric systems need to follow distinct strategies to be effective, as described in Rowe's taxonomy of deception [11]. Furthermore, systems that use deception must mask the fact that they are attempting to deceive intruders to be effective. Deceptive techniques must also constantly change, and must be focused on deceiving attackers rather than authorized users. Delay-focused approaches raise the costs of intrusion to the point where attackers may be challenged to avoid detection and thus face defensive countermeasures. This effectively slows down attacks, increasing the time-to-compromise of a system, forcing the attackers to maintain a larger temporal footprint on the attacked systems, which provides greater opportunity for analysts to find the attacker. Intrusion deterrence may focus on trying to make the initial intrusion as difficult as possible or increasing the potential consequences to

an attacker to the point where the risk-reward equation is disagreeable. The goals of the defender may influence the selection of specific DD techniques.

Usability Since any system being defended exists for the use of some target audience, *usability* of the system after a DD technique is deployed on that system is another concern of defenders. If the system becomes unusable to the target audience because of the DD mechanism, those end users may either stop using the system or try to circumvent the DD mechanism. This could result in additional vulnerabilities, loss of visibility to defenders, or the DD mechanism simply being ineffective.

2.6 Attacker Goals

Often, classes of attackers or threats are defined by the tactics, techniques, and procedures (TTPs) they have in common. Additionally, threats often share common goals or targets. Thus, classifying DD techniques by attacker or threat goals is another possible means of dividing and understanding DD techniques. Advanced persistent threats (APTs) often try to establish a solid presence on a target network and expand from an initially compromised host out into the rest of the network. Hacktivists (e.g., Anonymous, Anti-Sec) often target web servers and email servers, which handle front-door type traffic. DD solutions focused on each of these different threats could look extremely different and have different goals.

2.7 Efficiency

Inherently interesting to defenders is the *efficiency* of potential DD mechanisms. Efficiency is a natural means of comparing two mechanisms and could take many forms including energy costs, computational requirements, complexity, communication latency or bandwidth, and memory usage or storage requirements. These types of comparisons are already well-understood and would provide an immediate means of comparing different mechanisms, as is done in many other fields of computer science and engineering.

2.8 Future Development

Combining these various classifications, which may overlap or be completely disjoint, results in a structure that spans physical DD applications as well as the basic strategies and goals used in the application of DD capabilities. This structure provides a way to categorize various DD tactics, as well as a way to investigate the strengths and weaknesses of different approaches. This overall structure can be applied equally well to cloud systems, bare-metal systems, mobile devices, and indeed all computing platforms. Our intention is that the structure presented here can be used to facilitate communication about various DD tech-

niques and identify productive niches for future work. Further, we anticipate the structure developed here could lead immediately to a more-complete, coherent taxonomy in the future.

3 Challenges and Opportunities

Moving target defense or dynamic defense (DD) is a broad area that can operate at many different levels, including the physical world, the layers of the OSI model, and even at human and social levels [5]. DD can be implemented as a moving target or as a moving defense. An adversary can pre-stage their attack, enabling them to carry out their attack very efficiently. Because of pre-staging, humans can be too slow to stop an attack. Automated defenses can surpass a human in speed and stop an attack while it is still in progress.

New systems can be designed to support DD. However, retrofitting existing, statically-configured systems for DD may not be a viable option. In other words, tight coupling between components in complex systems may prohibit DD add-ons. The implementation of DDs within the layers of the OSI model often faces these challenges. While there are some properties and relationships that can be dynamically configured on a host or network, most properties and relationships are static. Network protocols are dependent on the implementation of the protocols on which they are built. In addition, applications are also dependent on protocols. Hence, dynamic reconfiguration of a protocol's interface would likely break higher-level protocols or applications which depend on that protocol. Both applications and protocols are dependent on certain network properties, such as IP address. Changes in these properties can invalidate the protocol or application state.

In reactive DD, changes to the system are made after the detection of some event. While there is a benefit to having the change take place at the time it is needed, it necessitates a detection mechanism. One of the attractive benefits of DD is protection against unknown threats. One of the benefits of proactive DD is that it may provide some protection against unknown threats. However, because reactive DD only makes changes after the detection of an event, it will generally only protect against known threats. Controlled triggering of the defenses may cause harm to the system, making it more vulnerable. For example, a reaction may be to shut down a particular service, making the system vulnerable to a denial of service (DoS) attack. Hence, dynamics alone may not be sufficient; systems need unpredictability.

Proactive DD also needs some level of unpredictability. Otherwise, the defense mechanism will be obscurity and it will just be a matter of time until the defense is defeated. Therefore, care must be taken to ensure that the amount of time required for an attacker to understand the strategy exceeds the amount of time that strategy is employed.

The implementation of DD may reach deep into the system it is defending. The added features and the changes they make increase the complexity of the system. Increasing the complexity of the system increases the burden of system administration. Administration tools generally look at static processes, and will also have to be modified to adapt to the new defenses and the raised complexity of the system. The added complexity may reduce the

usability of the system and reduce the ability of administrators to understand what is taking place within the system. Defenders need to receive digestible system information in a timely manner so they can respond quickly. Overlaying a control structure may help to maintain dynamic defenses. However, this would add a new channel and further complicate the system. The more complicated a system is, the more potential there is for new vulnerabilities and attack vectors. In other words, the added complexities of dynamic defenses will make the system harder to understand and defend. As the idiom goes, "Better the devil you know than the devil you don't know."

The introduction of DDs into a system should occur during system design. Whether something is fixed or dynamic sometimes depends on the timescale. We need new semantics for designing more dynamic systems.

3.1 Enterprise Networking DD

While DD represents a new, exciting, and in some ways, fundamentally different approach to cyber security, it is also in some sense just another tool for the job. In other words, many of the same criteria that are required of cyber security tools in existing enterprise environments will be applied to DD tools to potentially be deployed in those same environments. However, the space for opportunity is extremely large with some of the resources already in or applied to today's enterprise environments.

Challenges

Current enterprises can be extremely complex systems-of-systems, which mandates a constant effort from skillful analysts, highly familiar with their environments. Thus, keeping DD tools and deployments open, understandable, and transparent to analysts is a key challenge. While tools may provide additional defenses and capabilities, analysts are responsible for making decisions at a higher level about whether to shut down a compromised system, appropriate mitigations to an attacker, threat, or vulnerability, and when or how to escalate an incident. Thus, tools need to be supportive of these decisions; they need to inform the analyst, not cloud the issue or the facts. To be informative, what a tool is doing and the information it is presenting to an analyst needs to be clear and easily or quickly understood, which may require transparency of what the mechanisms a tool uses are and how they work. For example, if system IP addresses on a LAN are cycling following pseudo-random number generators (PRNGs), then an analyst should be able to track any one system symbolically (using its MAC address, for example) rather than tracking it by its current IP address. Thus, what is needed are tools that provide *grey-* or *white-box security* as opposed to *black-box security*.

An additional complication to moving DD ideas and tools to an operational setting is the widespread disconnect between research and operations. Numerous tools and ideas having great merit never leave the research community due to lack of effort in bridging this divide.

How to and what efficient processes can move tools and ideas from the research community to operational enterprise settings is an important area of future work.

Opportunities

The resources in many of today's enterprise networks are extensive. These resources afford new opportunities to incorporate dynamics and parallels from the physical security world not yet or not widely incorporated into cyber security. The advent of new or resurgent technologies such as software-defined networking and cloud computing complement the adoption of these dynamic ideas to the cyber security world.

Flexibility Software-defined networking and cloud computing both allow for increased flexibility in how an enterprise is constructed and where computation is done. Software defined networking allows one to redefine the structure of the network on-the-fly. This dynamism has the possibility to confuse an attacker (slowing them down, or thwarting them) during the reconnaissance phase of an attack. Cloud computing, and specifically virtualization allows for computation to be moved from one cloud or part of a cloud to another. Combining these two technologies is a means for reconfiguring an enterprise. However, the scale required in an enterprise to house these types of facilities may prohibit where tools making use of this type of flexibility can be deployed. The greater resources, independent of scale or reconfigurability, also allows for greater flexibility in the sense of there being more places to hide.

Deception Deception is often used in physical security and has been used in the physical world for millennia. Simple examples include camouflage, decoys, and concealment. While there are existing tools where deception has been employed (e.g., honeypots), there are numerous ways to use deception that have not been explored or deployed [11].

Introspection Virtualization allows for greater or more wide-spread introspection into hosts and their processes. Using introspection, we may be able to gain greater insight into how an attacker operates and the tools he uses.

3.2 Mobile and Host-Based DD

While mobile devices and mobile computing is not a new concept or area of cyber security application, as we have had laptops for almost 40 years, new mobile devices such as tablets and cell phones coupled with the increased computational power both they and the more traditional laptops bring, and the proliferation of such devices has brought a fundamental change to computing. This increased power, increased accessibility, and increased prevalence have resulted in traditional sensitive computing being done on these devices. Examples of

this shift include banking applications and the *bring your own device* (BYOD) approach to business computing.[2]

These shifts introduce new challenges to secure computing. First, these devices exist and operate outside of the traditional enterprise networking environment. Furthermore, these devices may not even belong to the organization owning the enterprise environment in which they periodically exist. Thus, common enterprise tools may not be useful for mitigating attacks. Since these devices often are outside an enterprise environment, they cannot hide from an attacker among other enterprise hosts. Thus, an even greater challenge is *how can we hide changes that are part of a dynamic defense from an attacker already on a system?* Though this may seem to be an insurmountable problem, there have already been successes in this area. One example is address space layout randomization which breaks an attacker's static executable code but allows legitimate software to run. DD solutions that work for mobile devices may also apply to individual hosts in other environments (e.g., enterprise), however, for brevity, here we will only discuss mobile hosts.

Attacks

Mobile devices face some attacks that do not have direct analogs in the enterprise environment. For example, mobile devices have and use wireless interfaces, which can be jammed by an attacker, thus affecting availability. Additionally, the software landscape on enterprise hosts is more static and well-defined than on mobile hosts, which have a plethora of applications at their disposal. There are also a large number of mobile device hardware types and a high churn rate for their operating system software. Applications from so-called app stores may be malicious, and the proliferation of malicious apps has been well documented recently [3].

The goals of these malicious apps are diverse and include data theft, direct profit for the app creator or an associate, and privacy compromise. Since mobile devices are used for processing sensitive information (e.g., banking data, phone books, contact information, or business data in BYOD environments), attackers may be interested in stealing this information. Malicious apps may also be used to call or send text messages to premium numbers, from which the attackers profit. Since mobile devices include such a wide variety of information sources, they are readily used for compromising a person's privacy, which can be accomplished, for example, through reading their location via GPS data, or simply eavesdropping on phone conversations.

Mitigations

When operating on a system or environment shared by an attacker, there are two strategies a defender can employ in trying to defeat the attacker. First, the defender can change faster

[2]See `http://www.govinfosecurity.com/webinars/mobile-learn-from-intels-ciso-on-securing-employee-owned-devices-w-264`.

than the attacker can keep up. This is the same premise on which much of cryptography is based; that is, it will take the attacker much longer to break the cryptography than the useful lifetime of the information being protected, or the lifetime of the parameters or keys used in protecting the information. An existing example that applies to jamming attacks is uncoordinated frequency hopping in the face of a resource constrained attacker (i.e., the attacker cannot simply jam all channels) [12]. This strategy could be applied to host internals by moving software necessary for an attacker's exploit to succeed [8]. The success of this type of defense depends on not underestimating the sophistication of the attackers we face. With regards to cryptographic security, which is almost exclusively computational security, the limitations are generally widely known because the best hardware available for breaking the cryptography is assumed to be commercially available. However, algorithmic weaknesses may not be openly shared, as could the case be for tools used to break the dynamics of DD techniques. A sufficiently sophisticated adversary could use automated tools to detect and respond to these changes to keep up with the dynamics.

An alternative strategy is to create or make use of separation between defenders and attackers. This strategy employs defense-in-depth concepts: by creating more separation, there is more "cyber-ground" to give while slowing or stopping an attack. Again, returning to the jamming attack example, using physical diversity through directional antennas against an attacker who cannot simultaneously be everywhere or transmit at all times allows nodes in a wireless network to successfully route packets in the presence of a wireless jammer [13]. Implementing virtualization of the operating system and using hypervisors is an example of this type of separation at the host level that is emerging for mobile devices.[3] Sandboxing has also been applied at the application layer in mobile devices.[4] This separation can be used as a means for inserting dynamic defenses. Defenders have the ability to change the conditions and environments these boundaries represent, forcing attackers to cross them.

4 The Path Forward: Experimentation and Evaluation

Moving research into operations will require evidence that an approach or tool is effective in some scenario or sense. Data from experimentation can fulfill this requirement. For industry, adoption may require a favorable outcome of a cost-benefit analysis, which may serve as a standard way of evaluation DD strategies. In this section, we present steps and methods which we believe will be useful and potentially necessary before deploying ideas in the research and development phase onto production systems. However, these considerations may not provide a complete path from research to practice.

[3]See `http://www.computerworld.com/s/article/print/9233834/Dual_identity_smartphones_could_bridge_BYOD_private_corporate_divide?taxonomyName=Bring+Your+Own+Device+(BYOD)&taxonomyId=227` (accessed 12/14/12)

[4]For example, see `http://developer.apple.com/library/ios/#documentation/iphone/conceptual/iphoneosprogrammingguide/TheiOSEnvironment/TheiOSEnvironment.html` (accessed 12/14/12).

4.1 Experimentation Techniques

During the workshop, participants suggested numerous approaches to experimentation. Simulation is a standard evaluation technique that allows experimentation and is already widely used. One drawback with using simulation is its lack of real-world attacks. Another problem is that there is no standard simulation environment. Instead, simulation environments and attack models vary as widely as defense techniques do. Wide-spread testing using *honeyfarms*, red teams, and cyber exercises are alternatives that use actual attacks. When choosing an experimental technique, there are a number of experiment outcomes to consider. Experiments performed on the open Internet may or may not attract the type of attackers and the level of activity desired for the experiment. If the experiments fail in this aspect, more control is needed, and methodologies including red teams or cyber exercises may be preferable.

Honeyfarms A *honeypot* is a single system or service which provides information that serves as bait for an attacker, luring them away from real sensitive data or into a trap. A *honeynet* is a network of such false systems or services that serve to mislead, confuse or trap an attacker, preventing him from understanding the real network. Mass deployment of honeypots or honeynets forming a *honeyfarm* would allow experimentation and data gathering on a large scale. One could envision thousands of such machines or networks deployed widely across institutions participating in an evaluation. This technique could be compared to experimental medicine. Like experimental medicine, the honeyfarm approach would require control groups to demonstrate differences between systems deploying a specific DD tool and those not deploying the tool.

Wide-spread deployment and long-running experiments would enable researchers to gather statistically significant amounts of data. Furthermore, it would allow for variance in attacker interest, targeting, and volition (i.e., when they decide to act). Since a honeyfarm is by definition covert (hopefully, at least to potential attackers) and exposed to any attacker (i.e., Internet-accessible), there may be multiple simultaneous attacks on such a test platform, which might confuse the data gathered and its meaning. Deploying a sufficiently large honeyfarm might also have problems with scalability. Recording data sufficient to understand an attack and the relevant performance of the tool or system under test across thousands of machines would require storing very large amounts of data. Searching or processing this data would present a further problem in terms of computation and network communication (if the participating locations are sufficiently geographically diverse).

It is hard to evaluate a defense without operational deployment. Exposing attackers to a honeyfarm and observing their actions is better than testing a particular DD strategy against synthetic attackers, but may not be as valuable as exposing them to real resources (i.e., not synthetic) because of the human factor.

Human-In-The-Loop Testing

The human factor is a strong component of the complete cyber security picture or problem. Furthermore, DD techniques that incorporate elements of deception are fundamentally focused on the human-driven aspect of attacks, as it will be humans that are deceived.

Red Teaming Red teaming a system to find potential vulnerabilities and assess the security of a system is well-established and widely accepted. However, red teaming is usually applied to production or near-production systems. Using red teaming to assess the viability of research ideas early in the development or prototyping phase is another means for gathering experimental data for assessing research ideas, but can be less effective. Assessing prototypes in a test environment that provides a sufficient amount of supporting infrastructure will allow the red team effort to gather data more relevant to operations. For example, with an address-hopping DD on a testbed network, the red team can measure how long it takes them to move from a position outside the network deploying the defense to inside that network, thus measuring the delay induced by the address hopping.

Due to the in-depth nature of red teaming exercises, this line of experimentation can take significant amounts of time and resources to accomplish. Thus, developing quicker turn-around experimental methods will also be important, especially in the early phase of development.

Cyber Exercises Cyber exercises (often *cyber defense exercises*) are designed to improve training through providing operational, near-operational, or time-sensitive settings to test and improve skills and processes. Some of these types of exercises take the form of capture-the-flag contests. These exercises take place over the course of a couple days to a couple weeks depending on the specific exercise. The set up and planning for these exercises can also take a similar amount of time depending on the repetition designed into the exercise. Thus cyber exercises could be another means for experimentally evaluating DD tools, and they would provide a much faster turn-around than red teaming, although the depth of the analysis and testing may be less.

By using an experimental methodology that can be instantiated more quickly, more instantiations of the experiment can be run. By running a sufficient number of these experiments, researchers may be able to provide statistically significant data about their proposals, tools, and prototypes. This data will provide a good foundation for moving tools from research to practice.

4.2 Metrics

Meaningful metrics have long been both a much sought after and elusive goal for cyber security researchers. Indeed, experimentation will be superfluous without meaningful metrics.

Meaningful metrics for cyber security need to be insightful – they must result in actionable information. For metrics to be insightful, they must also be relevant, and the resulting measurements or data must also be relevant. An experiment that attracts only attackers from threat models that are of little or no concern to a deploying institution may not provide relevant data. Thus, metrics must not only be relevant themselves, but the experiments where they are used must be designed in a way to use them in a relevant manner. Furthermore, humans are often unpredictable. Their creativity and ingenuity is highly desirable in certain settings. However, this unpredictability can complicate designing experiments and gathering meaningful data in experimental settings.

Operations Operational metrics revolve around the human analyst defending a network or system. The impact of a tool on the analyst and his work can be measured in terms of both time and workload. For example, in a repeated experiment, the reaction time of an analyst to a common threat and attack could indicate either the complexity of using a tool or the benefit of additional information provided by a tool. Measuring the amount of information, screens, or events an analyst must track could indicate the level of workload introduced or alleviated by a tool. The operational cost *savings* should also be considered. Attacks can be classified into types. Then the frequency of each type of attack, and the time and monetary costs can be considered. A reduction in a specific type of attack equates to an operational cost savings. Therefore, the impact of a tool on the cost of operations could be the decisive factor for determining deployment in some cases.

Overhead The impact of security mechanisms on normal operations can be measured in terms of energy, bandwidth, or computation consumed by the mechanism. In networking, goodput is defined as throughput less overhead. Measuring goodput provides a means for comparing different networking security mechanisms. This can be applied to both computational (cycles per second) and bandwidth (bytes per second) resources. In energy-constrained environments (e.g., sensor networks or data centers), measuring the energy consumption provides another means for comparison.

Engineering Moving a tool from research into production requires engineering and maintenance. The amount of time and money required to make this move or keep a tool running is a measure of the maturity of a tool and its acceptability into practice. For example, if a tool can simply be moved onto a production system, but it requires daily maintenance by a large team of highly skilled engineers or scientists, that tool may be less desirable than one that takes more effort initially to move into production but little or none once in production. Ideally, a tool could be developed once and then deployed anywhere without requiring maintenance.

4.3 A Competition Scenario

During the workshop, a scenario was developed for an experimental evaluation based on a two-round competition, which we describe here. The first round would be focused on selecting DD techniques for further testing, while the second round would focus on the actual testing of the selected techniques. The discussion focused on the second round as the means of applying scientific rigor to the process.

During the second round, the teams submitting the defense techniques would be eligible for the test phase (excluding their own techniques). There should be rewards in the form of cash prizes, hardware, publicity, or internship or employment opportunities. The second phase itself could be held either on a closed network, on a VPN, or on the Internet. This red-team approach would significantly reduce the amount of time required for evaluation and remove the uncertainty that is inherent in some of the online tests described above. The validity of the evaluation rests on the innovation of the red teams and not the application of known attacks alone. For this reason, the red team should be given details about the design and the opportunity to craft new attacks. However, this sort of red-team evaluation still cannot assess all unknown or future threats and techniques. Therefore, further testing would be recommended before deployment.

The competition approach fits into the scientific method in the following ways. The scientific hypothesis is that a given DD technique thwarts attacks and reduces the total cost or amount of human effort spent on defense among both the administrators and users of a system.[5] The scientific experiment is to select a set of techniques that show some merit and challenge a number of red teams to assess the techniques on a live system. The scientific analysis is to compare the red teams' successes, having both pre-existing and innovative attack elements as described above, to the merits of the techniques, making use of control groups where necessary. For example, the number and severity of compromises and should be taken into account. The scientific conclusion should determine if the hypothesis of the reduction in overall defense costs or efforts, or the reduction in successful attacks is true or not.

5 Conclusions

Dynamic Defense is a promising direction for cyber security because it is a move away from traditional static defenses on which adversaries can learn and practice. Fundamentally, DD incorporates changing the environment to increase an adversary's uncertainty, and the variance increases in terms of their expected payoff. As is true of all new areas of research, there are many unanswered questions and unknowns, which we cannot claim to have addressed or solved during this workshop. However, we believe that we have provided directions for beginning to address and answer those questions.

[5]The technique should not simply shift the defense effort from the part of the administrators to the users or result in an undue burden on the users.

The scientific basis for cyber security in general as well as for DD in particular has been questioned, and there are numerous efforts to find or define that basis. While there are few efforts, though informative or promising, theoretical bases for the cyber security of large-scale, real-world systems, experimental validation is a promising parallel path for establishing a scientific basis or method to cyber security. At the core of experimental evaluation is the fact that security is a human problem in addition to a technical problem. Thus, experimental evaluation necessitates human involvement and measurement.

The ultimate goal of DD, as with any area of cyber security, is to have an effect on and to complement operational deployments. To reach this goal, researchers and analysts, those ultimately responsible for the operational security of networks and devices, need to cooperate and collaborate to bridge the gap from research to practice. Ultimately, the decisions of when or whether a tool or system is operationally viable will be in the hands of those in operations. Using experimentation and testing in an operational setting will also be key to helping bridge this gap, especially when incorporating new non-static elements to defenses, as will be the case with DD.

References

[1] E. Mills, "Attacks on sony, others show it's open hacking season," June 2011. (Accessed November 13, 2012).

[2] B. Donohue, "Experts tell senate: Government networks owned, resistence is futile," March 2012. (Accessed November 13, 2012).

[3] McAfee Labs, "McAfee Threats Report: Second Quarter 2012," tech. rep., 2012. (Accessed November 13, 2012).

[4] C. Kreibich, N. Weaver, C. Kanich, W. Cui, and V. Paxson, "Gq: practical containment for measuring modern malware systems," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, IMC '11, (New York, NY, USA), pp. 397–412, ACM, 2011.

[5] S. Peisert, E. Talbot, and M. Bishop, "Turtles all the way down: A clean-slate, ground-up, first-principles approach to secure systems," in *Proceedings of the 2012 New Security Paradigms Workshop (NSPW)*, ACM, September 2012.

[6] M. Dunlop, S. Groat, W. Urbanski, R. Marchany, and J. Tront, "Mt6d: A moving target ipv6 defense," in *MILITARY COMMUNICATIONS CONFERENCE, 2011 - MILCOM 2011*, pp. 1321 –1326, nov. 2011.

[7] J. J. Haas, J. Hamlet, and J. Doak, "Machine-oriented biometrics and cocooning for dynamic network defense," in *Cyber Security and Information Intelligence Research Workshop 8 - CSIIRW 2013*, ACM, January 2013.

[8] J. Rowe, K. N. Levitt, T. Demir, and R. Erbacher, "Artificial diversity as maneuvers in a control theoretic moving target defense," in *National Symposium on Moving Target Research*, June 2012.

[9] S. Gilbert, J. Saia, V. King, and M. Young, "Resource-competitive analysis: a new perspective on attack-resistant distributed computing," in *Proceedings of the 8th International Workshop on Foundations of Mobile Computing*, FOMC '12, (New York, NY, USA), pp. 1:1–1:6, ACM, 2012.

[10] G. Novark and E. D. Berger, "Dieharder: securing the heap," in *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, (New York, NY, USA), pp. 573–584, ACM, 2010.

[11] N. Rowe, "A taxonomy of deception in cyberspace," in *International Conference on Information Warfare and Security*, pp. 173–181, March 2006.

[12] M. Strasser, S. Capkun, C. Popper, and M. Cagalj, "Jamming-resistant key establishment using uncoordinated frequency hopping," in *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pp. 64–78, May 2008.

[13] J. T. Chiang, D. Kim, and Y.-C. Hu, "JIM-Beam: using spatial randomness to build jamming-resilient wireless flooding networks," in *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '12, (New York, NY, USA), pp. 255–256, ACM, 2012.

DISTRIBUTION:

1 MS 0899 Technical Library, 9536 (electronic copy)

Sandia National Laboratories