ADVERSARIAL AND UNCERTAIN REASONING FOR ADAPTIVE CYBER DEFENSE: BUILDING THE SCIENTIFIC FOUNDATION

Sushil Jajodia
George Mason University

Reading

- Sushil Jajodia, George Cybenko, Peng Liu, Cliff Wang, Michael Wellman, eds., Adversarial and Uncertain Reasoning for Adaptive Cyber-Defense, Springer Lecture Notes in Computer Science (State-of-the-Art Survey Series), Vol. 11830, 2019. DOI: 10.1007/978-3-030-30719-6

Outline

- Motivation
  - Current cyber defense landscape & open questions
- Pro-active Defense via Adaptation
  - Adaptation techniques
  - Research challenges
- Our Research

Motivation

Today's Cyber Defenses are Static

- Today's approach to cyber defense is *governed by slow and deliberative processes* such as
  - Security patch deployment, testing, episodic penetration exercises, and human-in-the-loop monitoring of security events
- Adversaries can greatly benefit from this situation
  - They can *continuously and systematically probe targeted networks* with the confidence that those networks will change *slowly if at all*
  - They have the time to engineer reliable exploits and pre-plan their attacks
- Additionally, once an attack succeeds, adversaries persist for long times inside compromised networks and hosts
  - Hosts, networks, software, and services do not reconfigure, adapt, or regenerate except in deterministic ways to support maintenance and uptime requirements

Pro-active Defense via Adaptation

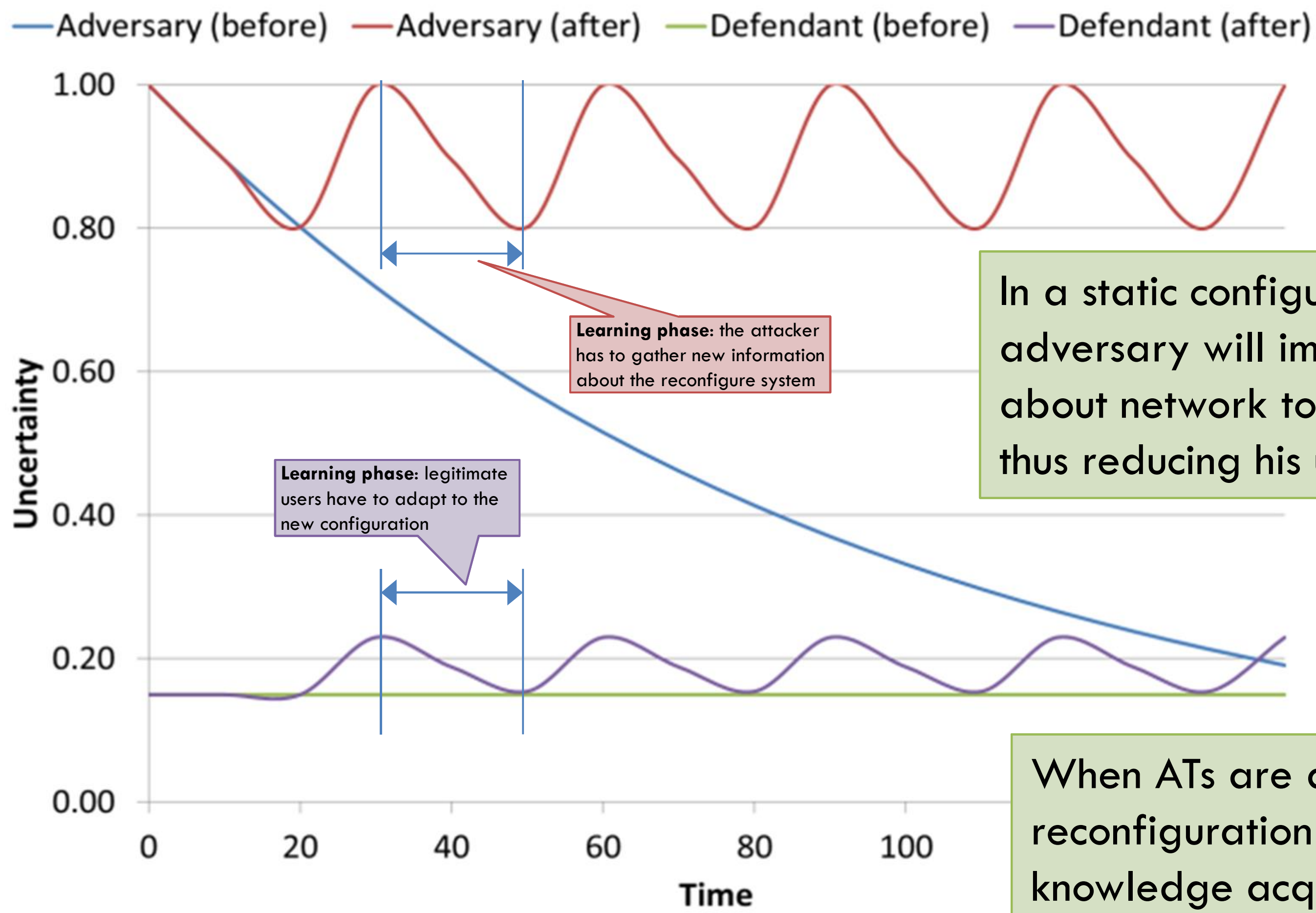
Pro-Active Defense via Adaptation

- To overcome today's limitations, we need to move from reactive defense to proactive defense
  - We propose to use adaptation as the guiding principle enabling proactive defense
    - The ultimate goal is to *adapt* systems to an evolving threat landscape, which includes both known and new threats
    - Systems must be able to change and adapt before *such threats materialize*
      - Adaptation will provide an advantage for the defender

Adaptation Techniques

- Adaptation Techniques (AT) consist of engineering systems that have **homogeneous functionalities** but **randomized manifestations**
  - These techniques *make networked information systems less homogeneous and less predictable*
  - ***Examples:*** Moving Target Defenses (MTD), artificial diversity, and bio-inspired defenses
- **Homogeneous functionality** allows authorized use of networks and services in predictable, standardized ways
- **Randomized manifestations** make it difficult for attackers to engineer exploits remotely, or reuse the same exploit for successful attacks against a multiplicity of hosts

Adversary and Defender Uncertainty



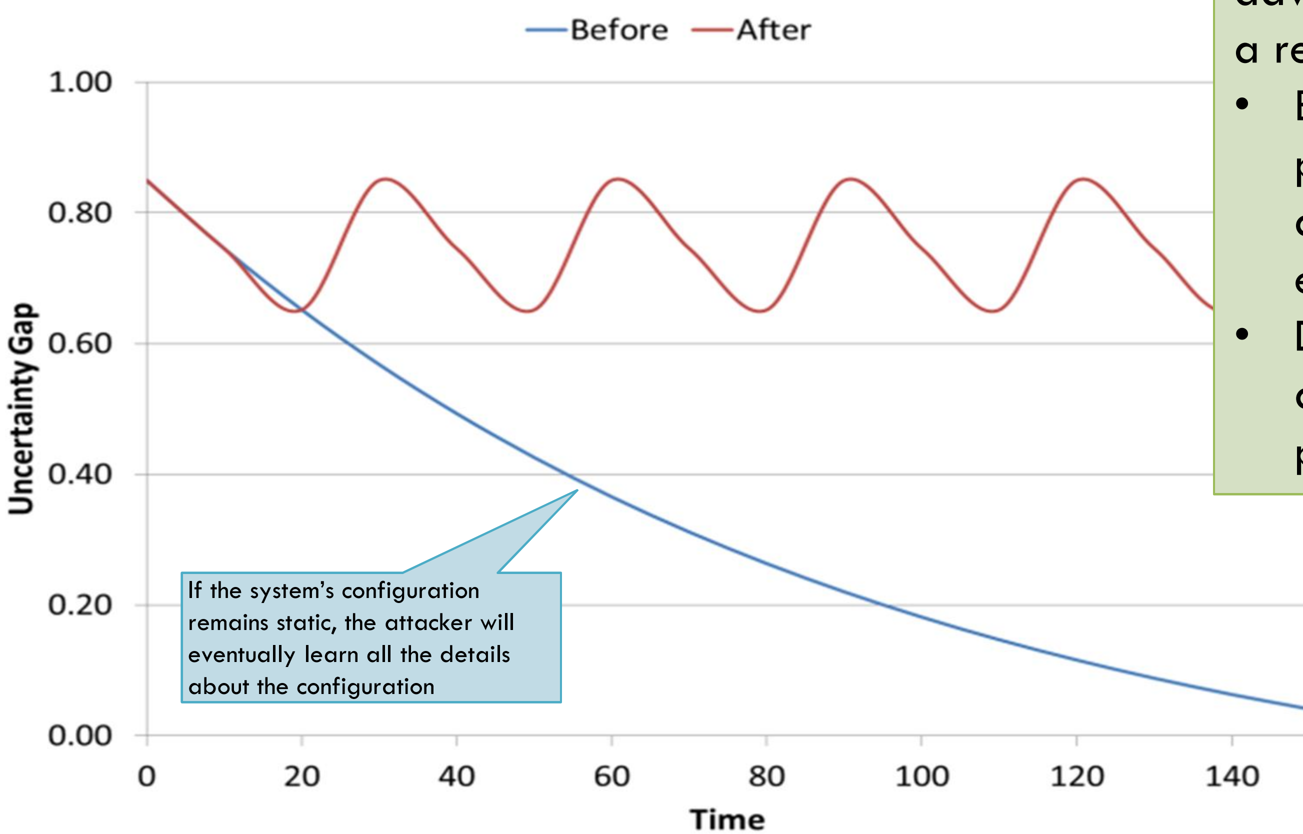
In a static configuration, over time, the adversary will improve his knowledge about network topology and configuration, thus reducing his uncertainty

When ATs are deployed, each system reconfiguration will invalidate previous knowledge acquired by adversaries, thus restoring their uncertainty to higher levels

Uncertainty Gap

ATs enable us to maintain the information gap between adversaries and defenders at a relatively constant level

- Before deploying the proposed mechanisms, the defender's advantage is eroded over time
- Dynamically changing the attack surface ensures a persistent advantage



AT Benefits and AT Classes

- **Benefits of Adaptation Techniques**
  - Increase **complexity, cost, and uncertainty** for attackers
  - Limit exposure of vulnerabilities and opportunities for attack
  - Increase system resiliency against known and unknown threats
  - Offer probabilistic protection despite exposed vulnerabilities, as long as the vulnerabilities are not predictable by the adversary at the time of attack
- **Classes of Adaptation Techniques**
  - Software-based
  - Network-based

Prior MTD Research

ESC-EN-HA-TR-2012-109

Technical Report
1166

Survey of Cyber Moving Targets

H. Okhravi
M.A. Rabe
T.J. Mayberry
W.G. Leonard
T.R. Hobson
D. Bigelow
W.W. Streilein

At least 39 documented in this 2013 MIT Lincoln Labs Report

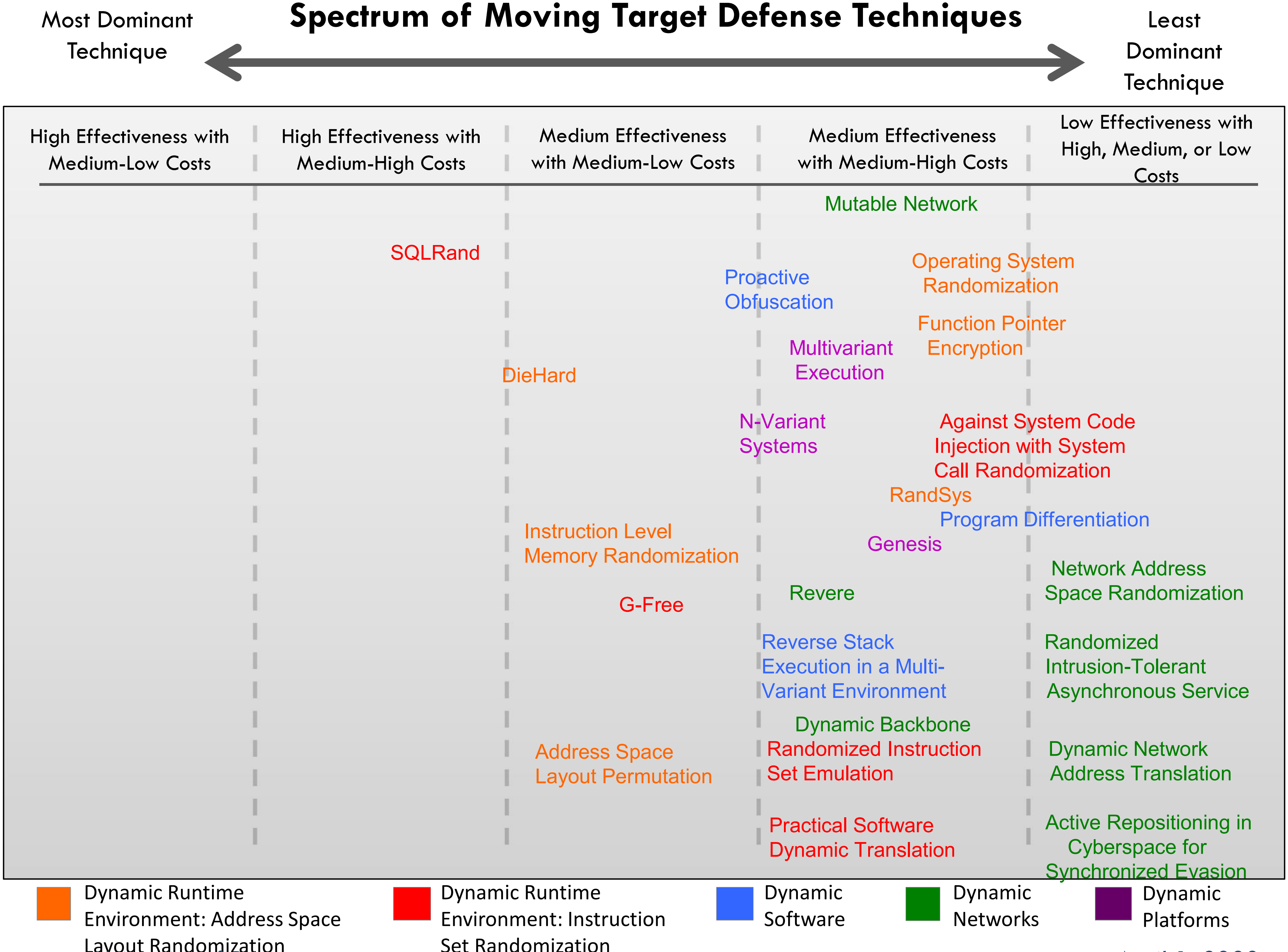
Software-Based Adaptation

- **Address Space Layout Randomization (ASLR)**
  - Randomizes the locations of objects in memory, so that attacks depending on knowledge of the address of specific objects will fail
- **Instruction Set Randomization (ISR)**
  - A technique for preventing code injection attacks by randomly altering the instructions used by a host machine or application
- **Compiler-based Software Diversity**
  - When translating high-level source code to low-level machine code, the compiler diversifies the machine code on different targets, so that vulnerability exploits working on one target *may not work on other targets*

Network-Based Adaptation

- Several Network–based adaptation approaches are being investigated at Mason
  - ID randomization
  - Generation of arbitrary external attack surfaces
  - VM-based dynamic virtualized network
  - Phantom servers to mitigate insider and external attacks
  - Proxy moving and shuffling to detect insider attacks
- Overall, these techniques aim at *giving the attacker a view of the target system that is significantly different from what the system actually is*

Spectrum of Moving Target Defense Techniques



Source: Kate Farris, George Cybenko

Limitations of Current Approaches

- The *contexts in which ATs are useful and their added cost* (in terms of performance and maintainability) to the defenders can vary significantly
  - Most ATs aim at preventing a specific type of attack
- The focus of existing approaches is on *developing new techniques*, not on understanding overall operational costs, when they are most useful, and what their possible interrelationships might be
- While each AT might have some engineering rigor, the *overall discipline is largely ad hoc* when it comes to understanding the totality of AT methods and their optimized application
- AT approaches assume *stochastic, but non-adversarial, environments*

Our Goals

- Building the Scientific Foundation of a new discipline: Adaptive Cyber Defense
  - Incorporating Adversarial and Uncertain Reasoning through the integration of
    - Game theory
    - Control theory
    - Graph and probability theory
    - Stochastic optimization
  - This foundational work is driving the definition of new classes of ACD techniques that
    - present adversaries with *changing attack surfaces* and system configurations
    - force them to *continually re-assess and re-plan their cyber operations*
- Developing model-based algorithms for optimally controlling ACD techniques in specific adversarial environments
- *Understanding* the relative cost and effectiveness of alternative ACD techniques in a variety of operational contexts
- *Raising the capabilities of ACD to meet the challenge of APTs*

Areas of Focus

- New ACD Techniques
- Attack Surface Manipulation
- Quantification of MTD Techniques
- Adversarial Modeling
- Control Theory
- Game Theory
- Defenses against APTs
- Include human component

Contributions

- **Adaptive Cyber Defenses against DDoS Attacks**
  - M. Wright, S. Venkatesan, M. Albanese, and M.P. Wellman, "Moving Target Defense against DDoS Attacks: An Empirical Game-Theoretic Analysis," in Proceedings of the 3rd ACM Workshop on Moving Target Defense (MTD 2016), pages 93-104, Vienna, Austria, October 24-28, 2016.
  - Sridhar Venkatesan, Massimiliano Albanese, Kareem Amin, Sushil Jajodia, Mason Wright, "A moving target defense approach to mitigate DDoS attacks against proxy-based architectures," IEEE Conf. on Communications and Network Security (CNS 2016), Philadelphia, PA, October 17-19, 2016 (Acceptance ratio 38/131).
- **Adversarial Modeling, Cyber Deception, and Game Theory**
  - Sushil Jajodia, Noseong Park, Edoardo Serra, V. S. Subrahmanian, "SHARE: A Stackelberg honey-based adversarial reasoning engine," *ACM Trans. on Internet Technology,* To appear.
  - Sushil Jajodia, Noseong Park, Fabio Pierazzi, Andrea Pugliese, Edoardo Serra, Gerardo I. Simari, V. S. Subrahmanian, "A probabilistic logic of cyber deception," *IEEE Trans. on Information Forensics and Security,* Vol. 12, No. 11, November 2017, pages 2532-2544. DOI: 10.1109/TIFS.2017.2710945
  - Edoardo Serra, Sushil Jajodia, Andrea Pugliese, Antonino Rullo, V. S. Subrahmanian, "Pareto-optimal adversarial defense of enterprise systems," *ACM Trans. on Information and System Security,* Vol. 17, No. 3, Article 11, March 2015, 39 pages. DOI: 10.1145/2699907

Contributions

- **Adaptive Cyber Defenses for Botnet Detection and Mitigation**
  - S. Venkatesan, M. Albanese, C.-Y. J. Chiang, A. Sapello, and R. Chadha, “DeBot: A Novel Mechanism to Detect Exfiltration by Architectural Stealthy Botnets” submitted to IEEE Trans. on Information Forensics and Security [major revision], 2017.
  - Sridhar Venkatesan, Massimiliano Albanese, George Cybenko, Sushil Jajodia, "A moving target defense approach to disrupting stealthy botnets," *Proc. 3rd ACM Workshop on Moving Target Defense (MTD 2016)*, Vienna, Austria, October 24, 2016 (Acceptance ratio 9/26 for regular papers).
  - Sridhar Venkatesan, Massimiliano Albanese, Sushil Jajodia, "Disrupting stealthy botnets through strategic placement of detectors," *IEEE Conf. on Communications and Network Security (CNS 2015)*, Florence, Italy, September 28-30, 2015 (Acceptance ratio 48/171). **Best Paper Runner-Up Award**
  - Sridhar Venkatesan, Massimiliano Albanese, Ankit Shah, Rajesh Ganesan, Sushil Jajodia, "Detecting stealthy botnets in a resource-constrained environment using reinforcement learning," *Proc. 4th ACM Workshop on Moving Target Defense (MTD 2017)*, Dallas, TX, October 30, 2017.

Contributions

- **Adaptive Techniques to Manipulating a System's Attack Surface**
  - Fabio De Gaspari, Sushil Jajodia, Luigi V. Mancini, Agostino Panico, "AHEAD: A new architecture for active defense," *Proc. SafeConfig 2016*, Vienna, Austria, October 24, 2016.
  - Paulo Shakarian, Nimish Kulkarni, Massimiliano Albanese, Sushil Jajodia, "Keeping intruders at bay: A graph-theoretic approach to reducing the probability of successful network intrusions," *Springer Series on Communications in Computer and Information Science, Vol. 554,* 2015, pages 191-211.
  - Massimiliano Albanese, Ermanno Battista, Sushil Jajodia, "A deception based approach for defeating OS and service fingerprinting," *IEEE Conf. on Communications and Network Security (CNS 2015),* Florence, Italy*,* September 28-30, 2015 (Acceptance ratio 48/171).
  - Massimiliano Albanese, Ermanno Battista, Sushil Jajodia, Valentina Casola, "Manipulating the attacker's view of a system's attack surface," *IEEE Conf. on Communications and Network Security (CNS 2014),* San Francisco, CA, October 29-31, 2014, pages 472-480 (Acceptance ratio 38/130).
  - Kun Sun, Sushil Jajodia, "Protecting enterprise networks through attack surface expansion (short paper)," *Proc. ACM SafeConfig 2014: Cyber Security Analytics and Automation,* Scottsdale, AZ, November 3, 2014, pages 29-32. DOI: 10.1145/2665936.2665939
  - Paulo Shakarian, Damon Paulo, Massimiliano Albanese, Sushil Jajodia, "Keeping intruders at large: A graph-theoretic approach to reducing the probability of successful network intrusions," *Proceedings of the 11th International Conference on Security and Cryptography (SECRYPT 2014)*, Vienna, Austria, August 28-30, 2014.

Contributions

- **Security through Diversity to Mitigate Zero-day Attacks**
  - Mengyuan Zhang, Lingyu Wang, Sushil Jajodia, Anoop Singhal, Massimiliano Albanese, "Network diversity: A security metric for evaluating the resilience of networks against zero-day attacks," *IEEE Trans. on Information Forensics and Security,* Vol. 11, No. 5, May 2016, pages 1071-1086. DOI: 10.1109/TIFS.2016.2516916
  - Daniel Borbor, Lingyu Wang, Sushil Jajodia, Anoop Singhal, "Securing networks against unpatchable and unknown vulnerabilities using heterogeneous hardening options," *Proc. 31st IFIP WG 11.3 Conference on Data and Applications Security and Privacy (DBSEC 2017), Springer Lecture Notes in Computer Science, Vol. 10359,* Giovanni Livraga and Sencun Zhu, eds. Philadelphia, July 19-21, 2017, pages 509-528. DOI: [10.1007/978-3-319-61176-1 28](10.1007/978-3-319-61176-1_28)
  - Nawaf Alhebaishi, Lingyu Wang, Sushil Jajodia, Anoop Singhal, "Threat modeling for cloud data center infrastructures," *Proc. 9th International Symposium on Foundations & Practice of Security (FPS 2016), Springer Lecture Notes in Computer Science, Vol. 101128,* Frederic Cuppens, Lingyu Wang, Nora Cuppens-Boulahia, Nadia Tawbi, Joaquin Garcia-Alfaro, eds., Quebec, Canada, October 24-26, 2016, pages 302-319. DOI: 10.1007/978-3-319-51966-1 20
  - Daniel Borbor, Lingyu Wang, Sushil Jajodia, Anoop Singhal, "Diversifying network services under cost constraints for better resilience against unknown attacks," *Proc. 30th IFIP WG 11.3 Conference on Data and Applications Security and Privacy (DBSEC 2016), Springer Lecture Notes in Computer Science, Vol. 9766,* S. Ranise and V. Swarup, eds, Trento, Italy, July 18-21, 2016, pages 295-312. DOI: 10.1007/978-3-319-41483-6 21
  - Lingyu Wang, Mengyuan Zhang, Sushil Jajodia, Anoop Singhal, Massimiliano Albanese, "Modeling network diversity for evaluating the robustness of networks against zero-day attacks," *Proc. 18th European Symp. on Research in Computer Security (ESORICS), Part II, Springer Lecture Notes in Computer Science, Vol. 8713*, Miroslaw Kutylowski, Jaideep Vaidya, eds., Wroclaw, Poland, September 7-11, 2014, pages 494-511 (Acceptance ratio 58/234).

Contributions

- **A Framework for Moving Target Defense Quantification**
  - W. Connell, D. Menasce, and M. Albanese, “Performance Modeling of Moving Target Defenses with Reconfiguration Limits,” submitted to IEEE Trans. on Information Forensics and Security [major revision], 2017.
  - W. Connell, D. Menasce, and M. Albanese, “Performance Modeling of Moving Target Defenses,” to appear in Proceedings of the 4th ACM Workshop on Moving Target Defense (MTD 2017), Dallas, Texas, USA, October 30, 2017.
  - W. Connell, M. Albanese, and S. Venkatesan, “A Framework for Moving Target Defense Quantification,” in Proceedings of the 32nd International Conference on ICT Systems Security and Privacy Protection (IFIP SEC 2017), pages 124-138, Rome, Italy, May 29-31, 2017.
  - L.H. Pham, M. Albanese, and S. Venkatesan, “A Quantitative Risk Assessment Framework for Adaptive Intrusion Detection in the Cloud,” in Proceedings of the 2nd IEEE Workshop on Security and Privacy in the Cloud (SPC 2016), Philadelphia, Pennsylvania, USA, October 19, 2016.
  - Massimiliano Albanese, Sushil Jajodia, "A graphical model to assess the impact of multi-step attacks," *Journal of Defense Modeling and Simulation,* 2017.

Contributions

- **Optimal Management of Cyber Security Operations Centers**
  - Ankit Shah, Arunesh Sinha, Rajesh Ganesan, Sushil Jajodia, Hasan Cam, "Two can play that game: An adversarial evaluation of a cyber-alert inspection system," *ACM Trans. on Intelligent Systems and Technology (TIST)*, 2020, To appear.
  - Ankit Shah, Rajesh Ganesan, Sushil Jajodia, Hasan Cam, "An outsourcing model for alert analysis in a cybersecurity operations center," *ACM Trans. on the Web (TWEB)*, Vol. 14, No. 1, January 2020. DOI: 10.1145/3372498
  - Ankit Shah, Rajesh Ganesan, Sushil Jajodia, Pierangela Samarati, Hasan Cam, "Adaptive alert management for balancing optimal performance among distributed CSOCs using reinforcement learning," *IEEE Trans. on Parallel and Distributed Systems (TPDS),* Vol. 31, No. 1, January 2020, Pages 16-33. First Online: 15 July 2019. DOI: 10.1109/TPDS.2019.2927977
  - Ankit Shah, Rajesh Ganesan, Sushil Jajodia, Hasan Cam, "A two-step approach to optimal selection of alerts for investigation in a CSOC," *IEEE Trans. on Information Forensics and Security (TIFS),* Vol. 14, No. 7, July 2019, pages 1857-1870.

Contributions

- **Optimal Management of Cyber Security Operations Centers**
  - Ankit Shah, Rajesh Ganesan, Sushil Jajodia, Hasan Cam, "Dynamic optimization of the level of operational effectiveness of a CSOC under adverse conditions," *ACM Trans. on Intelligent Systems and Technology,* To appear
  - Ankit Shah, Rajesh Ganesan, Sushil Jajodia, Hasan Cam, "A methodology to measure and monitor level of operational effectiveness in a CSOC," *Springer International Journal of Information Security,* 2017. DOI*:* 10.1007/s10207-017-0365-1
  - Rajesh Ganesan, Sushil Jajodia, Hasan Cam, "Optimal scheduling of cybersecurity analysts for minimizing risk," *ACM Trans. on Intelligent Systems and Technology,* Vol. 8, No. 4, 2017, pages 52:1-52:32. DOI: 10.1145/2914795 Designated by ACM as a paper with practical content
  - Rajesh Ganesan, Sushil Jajodia, Ankit Shah, Hasan Cam, "Dynamic scheduling of cybersecurity analysts for minimizing risk using reinforcement learning," *ACM Trans. on Intelligent Systems and Technology,* Vol. 8, No. 1, 2016. DOI: 10.1145/2882969 Appendix
  - Sushil Jajodia, Noseong Park, Edoardo Serra, V. S. Subrahmanian, "Using temporal probabilistic logic for optimal monitoring of security events with limited resources," *Journal of Computer Security,* Vol. 24, No. 6, 2016, pages 735-791. DOI: 10.3233/JCS-160555

Contributions

- **Fake Document Generation**
  - Prakruthi Karuna, Hemant Purohit, Sushil Jajodia, Rajesh Ganesan, Ozlem Uzuner, "Fake document generation for cyber deception by manipulating text comprehensibility," IEEE Systems Journal, 2020, To appear.
  - Tanmoy Chakraborty, Sushil Jajodia, Jonathan Katz, Antonio Picariello, Giancarlo Sperli, V. S. Subrahmanian, "FORGE: A fake online repository generation engine for cyber deception," IEEE Trans. on Dependable and Secure Computing (TDSC), To appear. First Online: 11 February 2019. DOI: 10.1109/TDSC.2019.2898661
  - Prakruthi Karuna, Hemant Purohit, Rajesh Ganesan, Sushil Jajodia, "Generating hard to comprehend fake documents for defensive cyber deception," IEEE Intelligent Systems, Vol. 33, No. 5, September/October 2018, pages 16-25. DOI: 10.1109/MIS.2018.2877277

Hybrid Adversarial Defense: Merging Traditional Security Defenses and Honeypots

University of Rome

April 1, 2020

Manipulating the Attack Surface

- The common wisdom suggests that *the size of the attack surface is directly related to the security of the enterprise network*
- Researchers have typically focused on methods to reduce the attack surface
  - A smaller attack surface would offer fewer attack vectors to attackers
- Instead, we propose approaches to shift, enlarge, or otherwise manipulate the attack surface observed by attackers

How to Manipulate the Attack Surface

- We may present adversaries with a *varying attack surface*
  - This will create the illusion that the system is changing over time
- We may present adversaries with a *larger* (external) *attack surface* than the actual (internal) attack surface
  - This will create the illusion that the system is more complex than what it actually is
- We may present adversaries with a *realistic but deceiving view* of the (external) attack surface
- All approaches will ultimately have the effect of *increasing the uncertainty for the adversaries*

Our Approach

- Presented different approaches for *manipulating a system's attack surface to increase complexity for the attacker*
  - Virtualizing the Attack Surface
    - Goal: Manipulating how the system responds to probes from potential attackers
      - Attackers will plan attack based on deceiving information
  - Adding Distraction Clusters
    - Goal: Controlling the probability that an intruder may reach a certain goal within a specified amount of time
      - Defenders are buying time
  - Leveraging Network Diversity
    - Goal: Modeling network diversity to evaluate the robustness against known and unknown attacks
      - Proposed three different metrics

Contributions

- Placement of honeypots
- Game-theoretic methods to reason about the adversary and merge traditional security defenses and honeypots
- A novel honeypot architecture

Publications

- Paulo Shakarian, Damon Paulo, Massimiliano Albanese, Sushil Jajodia, "Keeping intruders at large: A graph-theoretic approach to reducing the probability of successful network intrusions," *Proceedings of the 11th International Conference on Security and Cryptography (SECRYPT 2014)*, Vienna, Austria, August 28-30, 2014.
- Paulo Shakarian, Nimish Kulkarni, Massimiliano Albanese, Sushil Jajodia, "Keeping intruders at bay: A graph-theoretic approach to reducing the probability of successful network intrusions," *Springer Series on Communications in Computer and Information Science, Vol. 554,* 2015.
- Sushil Jajodia, Noseong Park, Edoardo Serra, V. S. Subrahmanian, "SHARE: A Stackelberg honey-based adversarial reasoning engine," *ACM Trans. on Internet Technology,* 2018.
- Tanmoy Chakraborty, Sushil Jajodia, Noseong Park, Andrea Pugliese, Edoardo Serra, V. S. Subrahmanian, "Hybrid adversarial defense: Merging traditional security defenses and honeypots," Jour. of Computer Security.
- Fabio De Gaspari, Sushil Jajodia, Luigi V. Mancini, Agostino Panico, "AHEAD: A new architecture for active defense," *Proc. SafeConfig 2016*, Vienna, Austria, October 24, 2016.

A Graph-Theoretic Approach to Increase Complexity for the Attacker and Delay Intrusions: Motivation

- We aim at delaying intrusions
  - Attempting to step all intrusions is unrealistic
- We want to control the *probability that an intruder may reach a certain goal within a specified amount of time*
- *Ultimately, we would like to keep such probability below a given threshold*

Overview of our Approach

- Our method relies on analyzing a graphical representation of the computer network's logical layout and an associated probabilistic model of the adversary's behavior
- We then artificially modify this representation by adding "distraction clusters" at key points of the network in order to increase complexity for the intruders and delay the intrusion

Intruder Penetration Network

- An adversary has a particular target (e.g., an intellectual property repository)
- The target can be reached by sequentially gaining privileges on multiple system resources
- We calculate the probability of reaching the target in a certain amount of time
  - π A function, that given two system-level pairs (s_1, l_1), (s_2, l_2), returns the probability of an intruder gaining access level l_2 on s_2 given that he has access level l_1 on s_1
  - f A function, that, given two system-level pairs (s_1, l_1), (s_2, l_2), returns the a positive value that shows the fitness (attractiveness) of gaining access level l_2 on s_2 for an intruder with access level l_1 on s_1

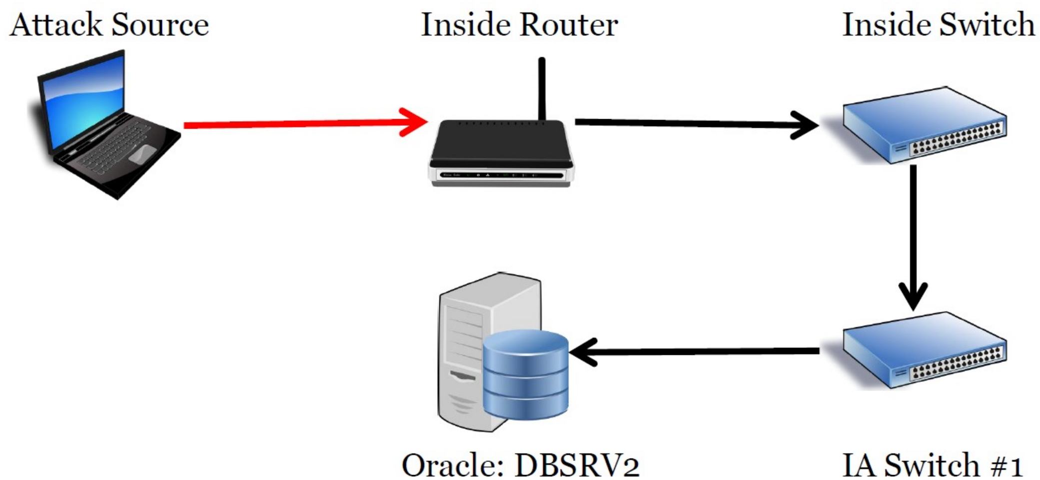
Distraction Cluster

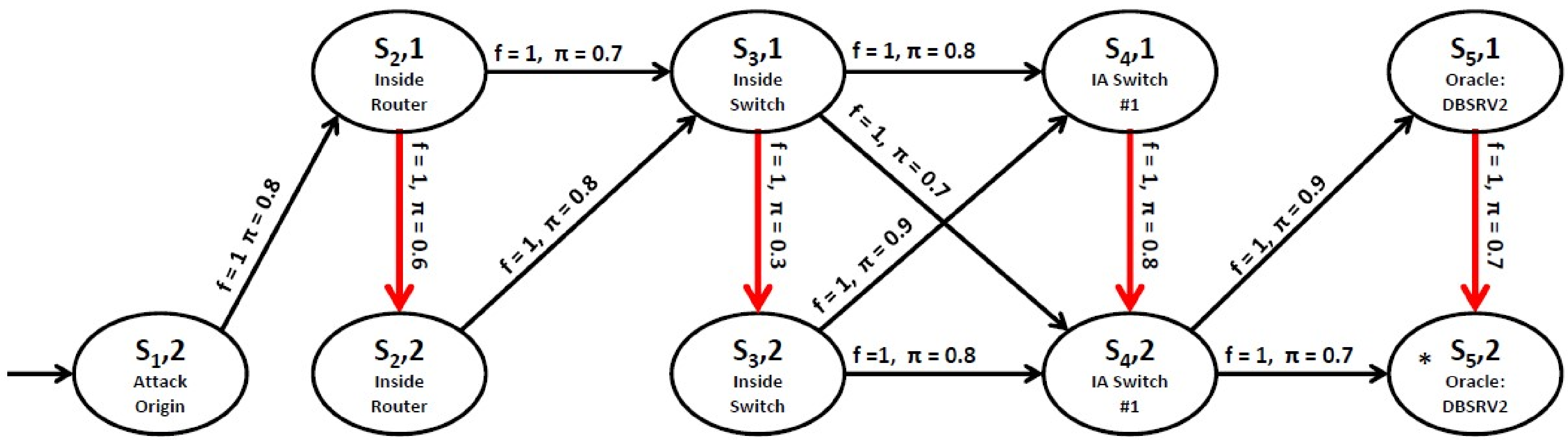
- We then modify our graphical representation by adding "distraction clusters" – collections of interconnected virtual machines – at key points of the network
- We assume a set CFG of virtual machine configurations
- We also assume the existence of arrangements of these virtual machines into network clusters (CL)
- Each cluster in CL has a *lead* and a *last* system which connect to the larger IPN
- In this work we are primarily concerned with where the lead system attaches
- We assume that a cluster can be arbitrarily large to delay the attacker according our specification

Example

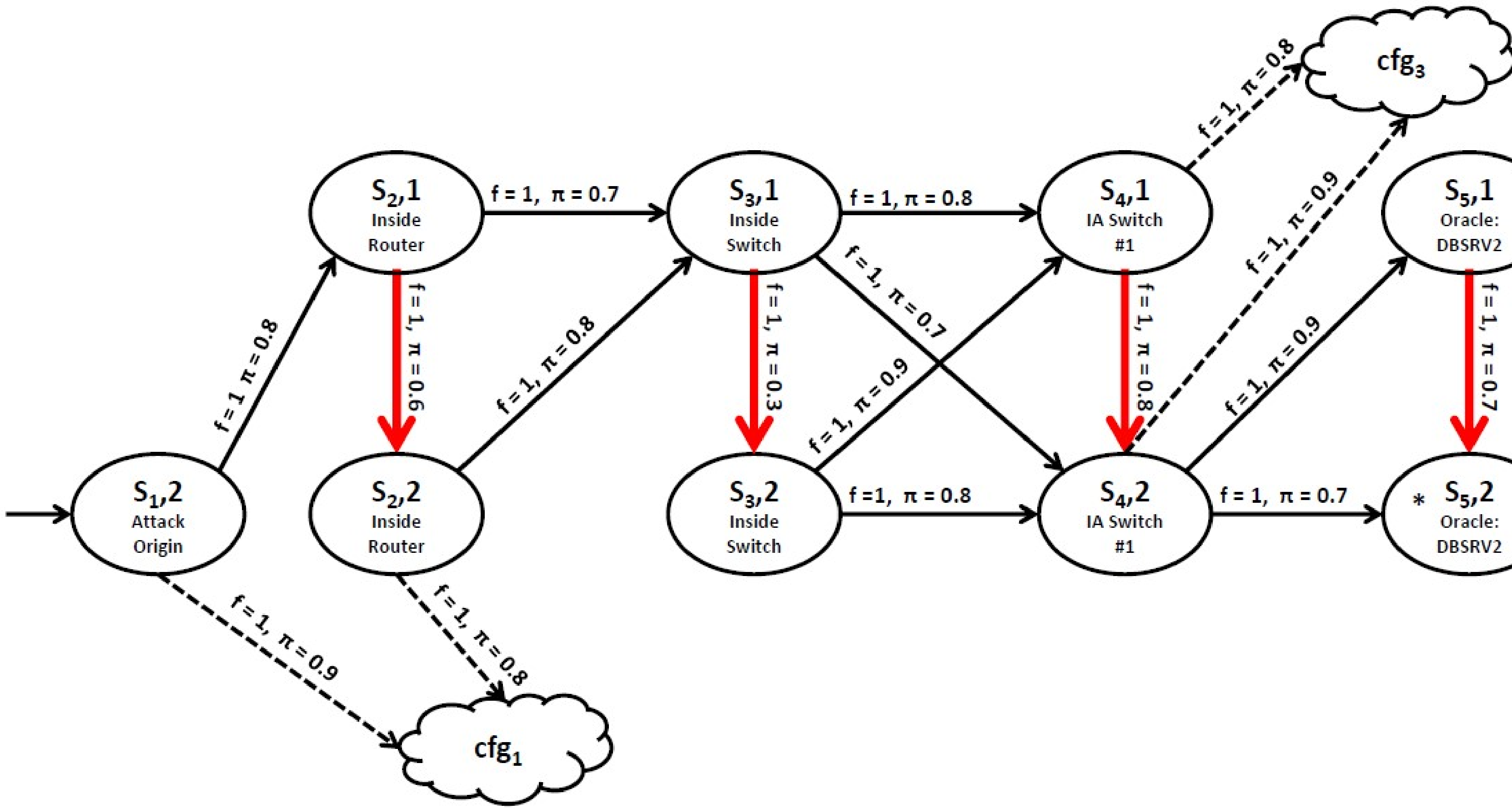
In this network a user can have one of two levels of access on each system

- Guest privileges ($l = 1$)
- Root privileges ($l = 2$)





Example



S1,2
Attack
Origin
S2,1
Inside
Router
S2,2
Inside
Router
S3,1
Inside
Switch
S3,2
Inside
Switch
S4,1
IA Switch
#1
S4,2
IA Switch
#1
S5,1
Oracle:
DBSRV2
S5,2
Oracle:
DBSRV2
cfg1
cfg3
f = 1 π = 0.8
f = 1, π = 0.7
f = 1, π = 0.6
f = 1, π = 0.8
f = 1, π = 0.8
f = 1, π = 0.3
f = 1, π = 0.7
f = 1, π = 0.9
f =1, π = 0.8
f = 1, π = 0.8
f = 1, π = 0.8
f = 1, π = 0.9
f = 1, π = 0.9
f = 1, π = 0.7
f = 1, π = 0.7
f = 1, π = 0.9
f = 1, π = 0.8

Results

- The Cluster Addition Problem is NP-hard
- We provide an approximation algorithm that possesses several useful properties
- We have a prototypal implementation and experimental results

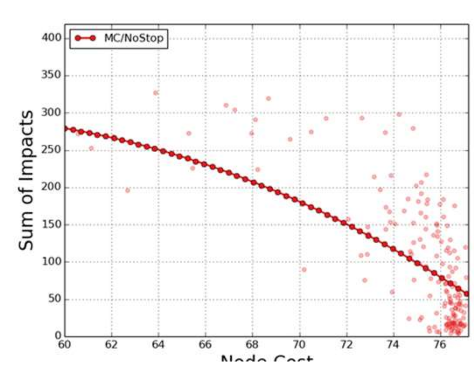
SHARE: A Stackelberg Honey-based Adversarial Reasoning Engine

- Off-line: Game-theoretic framework to optimally locate honeypots in a network, taking vulnerability dependency graph into account
- Online: What to do when an attacker is detected?

Attacker Actions
Scan nodes
Exploit vulnerability

Defender Actions
Place honey nodes/tokens
Patch vulnerability
Deactivate software

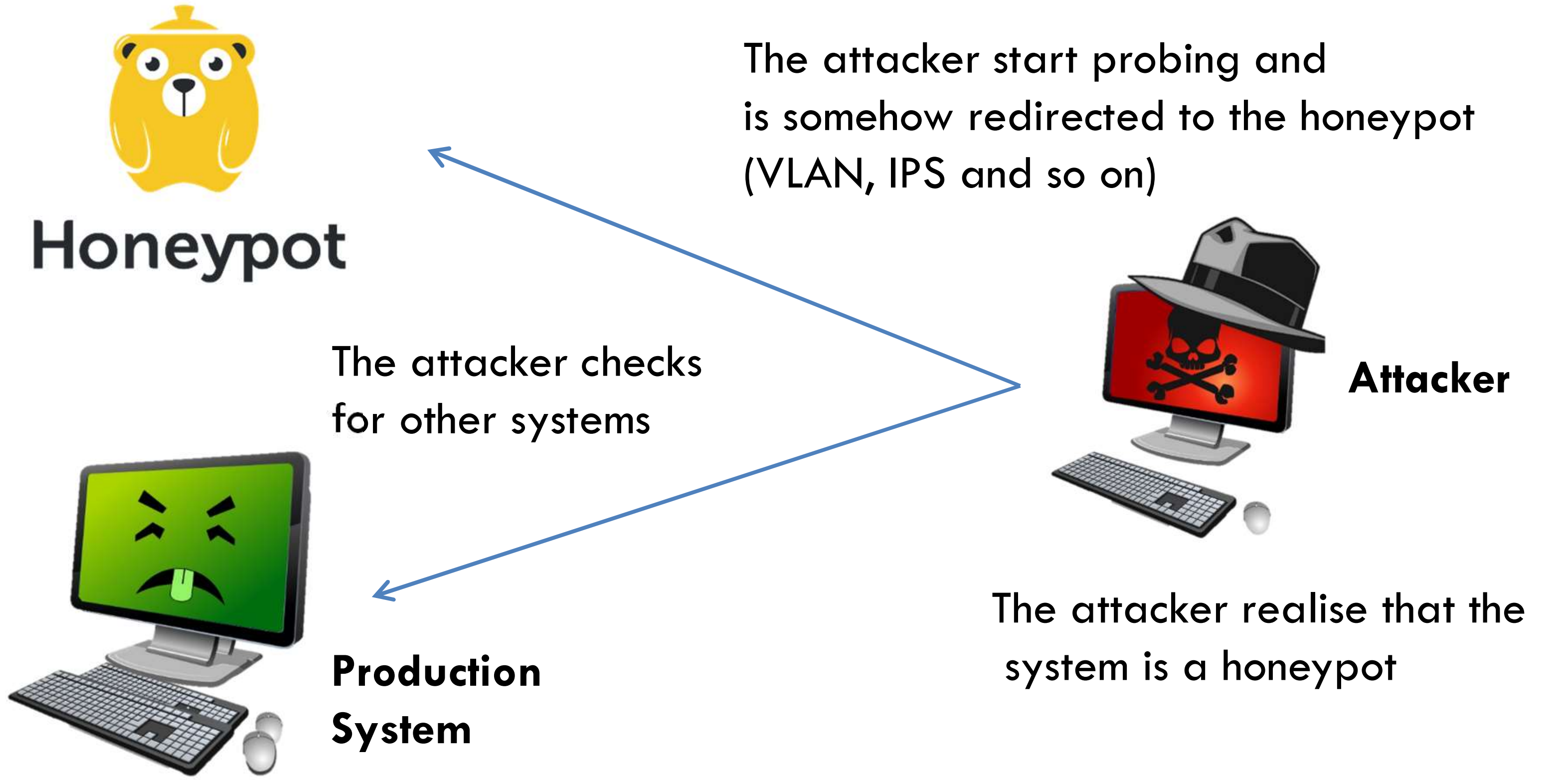
Developed
Attacker model to maximize expected damage
Pareto optimal defender model to:
minimize maximal exp attacker damage
minimize max. attacker success prob.
Showed how attacker can launch new attacks after detection via reinforcement learning
Showed that stopping attacker immediately upon detection is not the best strategy.



A Novel Honeypot Architecture

Classical Approach

Logging the activities



A Different Approach

Logging the activities

The attacker sees directly
the Production System

Option 2. The attacker keep
interacting with the system

The attacker interact
with the system

Production System

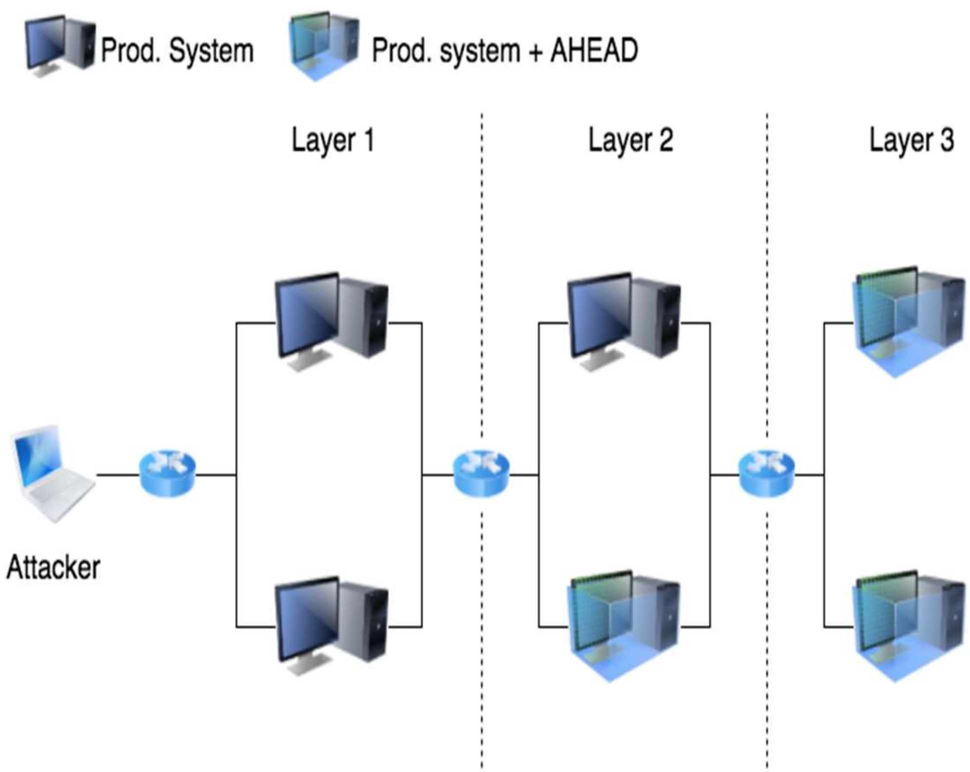
Client/ Server/ Honeypot/
Network Component

Attacker

Option 1
He thinks that the
system is a Honeypot,
look for other systems

Joint work with Prof Luigi Mancini, U of Rome

Evaluation of our Approach



31 last year MSc students

3-layer experiment:

- L1 - No AHEAD deployed
- L2 - AHEAD on one machine
- L3 - AHEAD on both machines

Goal: root privilege in L3 machine

L3 machines and L1 machines had same vulnerable service

Results

Layer	Machine	Success %	Time to Success	Traffic (GB)	Avg. Individual Traffic
L1		90.32%	1h 9m 36s	21.23	0.68
	Prod. System 1	5.34%		7.4305	0.24
	Prod. System 2	84.98%		13.7995	0.44
L2		61%	14h 37m 26s	78.88	2.82
	Prod. System 3	61%	14h 37m 26s	52.0608	1.86
	Prod. System + AHEAD	0%	∞	26.82	0.96
L3		6%	48h 25m 42s	54.89	2.89
	Prod. System1 + AHEAD	0%	∞	23.6027	1.24
	Prod. System2 + AHEAD	6%	48h 25m 42s	31.29	1.65

Optimal Scheduling of Cyber Analysts for Minimizing Risk*

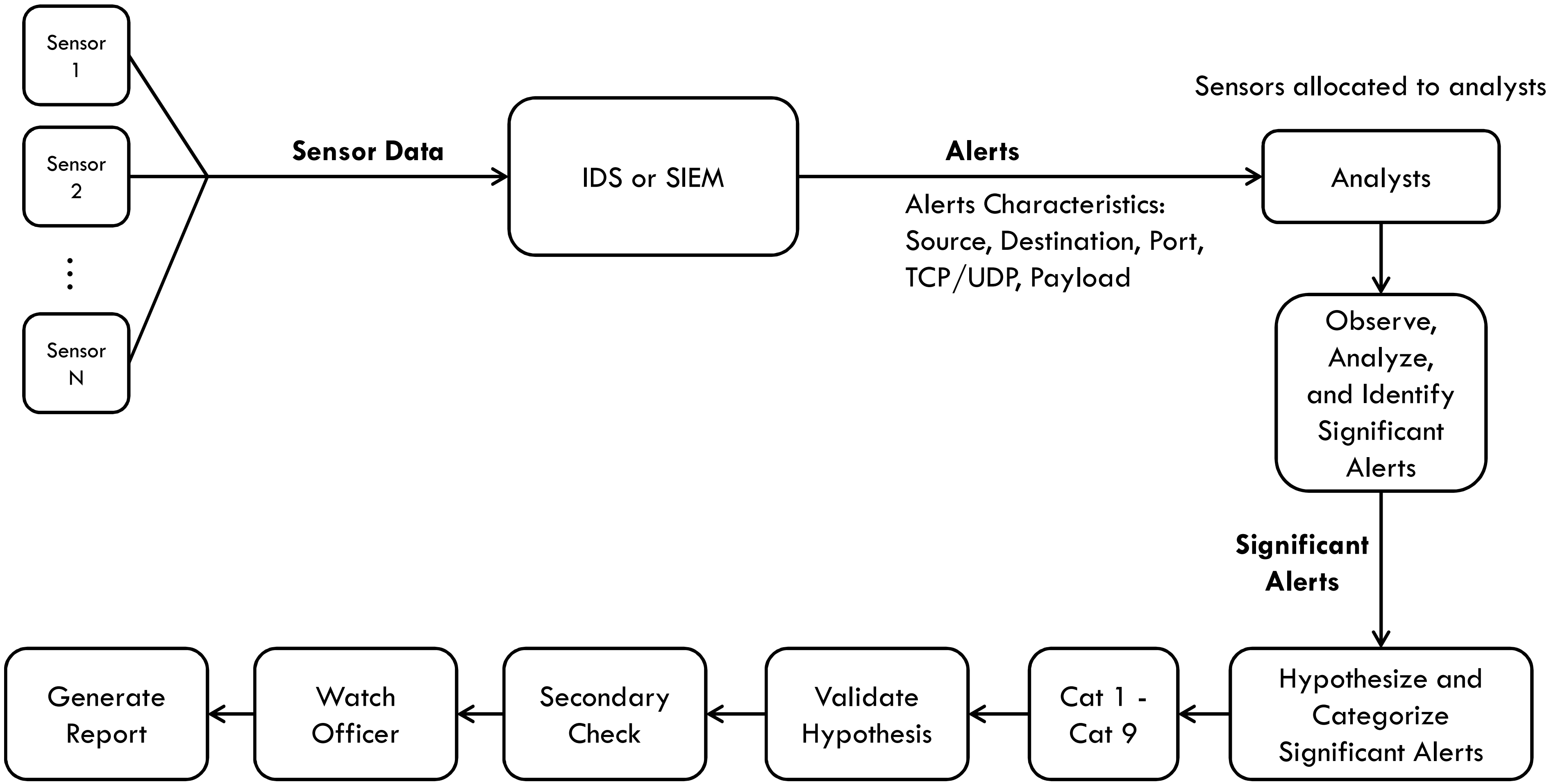
*Joint work with Rajesh Ganesan (GMU), Hasan Cam (ARL), Ankit Shah (GMU)

Statement of Need

- Cybersecurity threats are on the rise
- Demand for Cybersecurity analysts outpaces supply [1] [2]
- Given limited resources (personnel), the analyst workforce must be optimally managed
- Given the current/projected number of alerts it is also necessary to know the optimal workforce size

[1] http://www.rand.org/pubs/research_reports/RR430.html
[2] http://www.rand.org/news/press/2014/06/18.html

Process Flow, Definition of Significant Alerts



Significant Alerts = 1% of all Alerts Generated

Categories 1-9

DON CYBER INCIDENT CATEGORY

Cat 1-9	Description
1	**Root Level Intrusion (Incident):** Unauthorized privileged access (administrative or root access) to a DoD system.
2	**User Level Intrusion (Incident):** Unauthorized non-privileged access (user-level permissions) to a DoD system. Automated tools, targeted exploits, or self-propagating malicious logic may also attain these privileges.
3	**Unsuccessful Activity Attempted (Event):** Attempt to gain unauthorized access to the system, which is defeated by normal defensive mechanisms. Attempt fails to gain access to the system (i.e., attacker attempt valid or potentially valid username and password combinations) and the activity cannot be characterized as exploratory scanning. Can include reporting of quarantined malicious code.
4	**Denial of Service (DOS) (Incident):** Activity that impairs, impedes, or halts normal functionality of a system or network.
5	**Non-Compliance Activity (Event):** This category is used for activity that, due to DoD actions (either configuration or usage) makes DoD systems potentially vulnerable (e.g., missing security patches, connections across security domains, installation of vulnerable applications, etc.). In all cases, this category is not used if an actual compromise has occurred. Information that fits this category is the result of non-compliant or improper configuration changes or handling by authorized users.
6	**Reconnaissance (Event):** An activity (scan/probe) that seeks to identify a computer, an open port, an open service, or any combination for later exploit. This activity does not directly result in a compromise.
7	**Malicious Logic (Incident):** Installation of malicious software (e.g., trojan, backdoor, virus, or worm).
8	**Investigating (Event):** Events that are potentially malicious or anomalous activity deemed suspicious and warrants, or is undergoing, further review. No event will be closed out as a Category 8. Category 8 will be re-categorized to appropriate Category 1-7 or 9 prior to closure.
9	**Explained Anomaly (Event):** Events that are initially suspected as being malicious but after investigation are determined not to fit the criteria for any of the other categories (e.g., system malfunction or false positive).

Source: Dept of Navy, Cybersecurity Handbook, page 20

Statement of Need

- Cybersecurity threats are on the rise
- Demand for Cybersecurity analysts outpaces supply [1] [2]
- Given limited resources (personnel), the analyst workforce must be optimally managed for minimizing today's risk
- Given the current/projected number of alerts it is also necessary to know the optimal workforce size to keep risk under a certain threshold

[1] http://www.rand.org/pubs/research_reports/RR430.html
[2] http://www.rand.org/news/press/2014/06/18.html

Definition of Risk

- Alert Coverage is defined as the % of the significant alerts (1% of the total alerts) that are thoroughly investigated in a work-shift by analysts and the remainder (forms the Risk) is not properly analyzed or unanalyzed because of
  - Sub-optimal shift scheduling
  - Not enough personnel in the organization
  - Lack of time (excessive analyst workload)
  - Not having the right mix of expertise in the shift in which the alert occurs
- Risk % = 100 – Alert Coverage %

Note: From this slide onward, the term alert refers to significant alerts only

Requirements

- The cybersecurity analyst scheduling system
  - Shall ensure that an optimal number of staff is available to meet the demand to analyze alerts
  - Shall ensure that a right mix of analysts are staffed at any given point in time
  - Shall ensure that risks due to threats are maintained below a pre-determined threshold
  - Shall ensure that weekday, weekend, and holiday schedules are drawn such that it conforms to the working hours/leave policy

Problem Description

Risk is proportional to Analyst Characteristics

1. Alert generation rate
2. the number of analysts,
3. their expertise mix,
4. analyst's shift and days-off scheduling,
5. their sensor assignment,
6. Category of alert – analyst workload – time to analyze (input)

Two types of problems to solve:

Simulation: Given all of the above, what level of risk is the organization operating at?
Optimization: Given an upper bound on risk, what are the optimal settings for 1-5?

Minimizing risk vs. Setting an upper bound on risk

- Direct minimizing risk can be achieved by adjusting
  - the number of analysts,
  - their expertise mix,
  - analyst's shift and days-off scheduling,
  - their sensor assignment,
  - Category of alert – analyst workload – time to analyze (input)
- However, which factor(s) to adjust is hard to determine (requires several simulations)
- Running optimization with risk in the objective function is computationally not viable because the solution space is extremely large for these NP Hard problems.
- Instead, we set up an upper bound on risk and determine the optimal settings of the above factors via optimization using metaheuristics.
- Obtain a set of feasible solutions and pick the best (lowest number of analysts, among them the lowest risk).
- A 0% upper bound can also be set, which constitutes the lowest risk attainable.
- The optimization model provides the flexibility to set any upper bound on risk.

Algorithm Contributions

Optimization Algorithm

- Mixed Integer Programming solved using Genetic Algorithm
- Outputs
  - the number of analysts,
  - their expertise mix,
  - their sensor-to-analyst assignment

Scheduling Algorithm

- Integer programming and a heuristic approach
- Output
  - Analyst shift and days-off scheduling

Simulation Algorithm

- Validates optimization
- A tool can be used as a stand-alone algorithm to measure the current risk performance of the organization for a given set of inputs

Research Objective for Optimization

- Objective: Minimize number of personnel and minimize risk
- Subject to following constraints
  - Maintain risk below the upper bound
  - Ensure ≥ 95% analyst utilization
  - Meet the mix (senior, intermediate, junior) specification 20-40% L3, 40-50% L2, and 30-40% L1
  - Number of sensors per analyst constraint
- Outputs
  - Sensor to analyst allocation
  - Total number of analysts and their mix

Sensor

Analyst

	n=1	n=2	n=3
i=1	1	1	0
i=2	0	0	1
i=3	1	0	0

Alert Characteristics

- Sensors generate about 15000 alerts per day
  - All alerts are screened by auto-analysis methods and those that are significant by analysts
  - 1% ~ 150/day ~ avg. 6-7 alerts per hr per sensor are important/have different patterns and requires further investigation by analysts ("significant alerts")
  - Generate alert rate/hr using an arrival probability distribution Poisson (6.5) or Uniform (1,13)
  - Average alert generation rate per hr per sensor can be varied (future work), but for the current model it was kept fixed and equal for all sensors

Analyst Characteristics

- Based on training and experience there are 3 levels of analysts – senior L3, intermediate L2, junior L1
- Over a time interval of one hour,
  - a L1 analyst can handle 5 attacks with simplest actions like blocking an IP address, (Avg 12 min per alert)
  - a L2 analyst can handle 7 or 8 attacks with more complicated actions like blocking a server from an external network (Avg 8 min per alert)
  - a L3 analyst can handle 8+ attacks with the most sophisticated actions (Avg 5 min per alert)
- Alert investigation time could follow a probability distribution – Poisson, normal, triangle, beta

Number of Sensors to Analyst

Constraint - 1

- L3 senior – 4-5 sensors are allocated
- L2 intermediate – 2-3 sensors are allocated
- L1 junior – 1-2 sensors are allocated
- Some overlapping is permissible

- Note: The sensor-to-analyst mapping is an output of optimization

System Requirement Parameters

<u>Constraints – 2 to 4</u>

- Upper bound on Risk - Ex: 10% of the significant alerts are not properly analyzed/unanalyzed
- Analyst Utilization
  - Ensure >95% analyst utilization
- Analyst mix in the organization
  - 20-40% L3, 40-50% L2, and 30-40% L1

Inputs

Inputs that were varied for sensitivity analysis

- Number of sensors -10, 25, 50, 75, 100
- Risk % - 5%, 25%, 45%

Inputs that were maintained fixed for the above studies

- Average alert generation rate using
  - Uniform (0,13) distribution, Mean = 6.5/hr , 6.5*24= 156/day
- Analyst characteristics
  - Average alert investigation rate (time to investigate)
- Number of sensors allocated to analysts

- Optimization was solved using Genetic Algorithm heuristics

Research Findings: Optimization without specifying expertise mix

- Multiple sensors to analyst
- All senior L3 analysts were chosen to minimize personnel
  - No L2 and L1 analysts were selected by optimization
- >95% utilization of analyst time
- At 100% alert coverage (0% Risk), analyst/sensor ratio = 0.7
- At 75% alert coverage (25% Risk), analyst/sensor ratio = 0.5

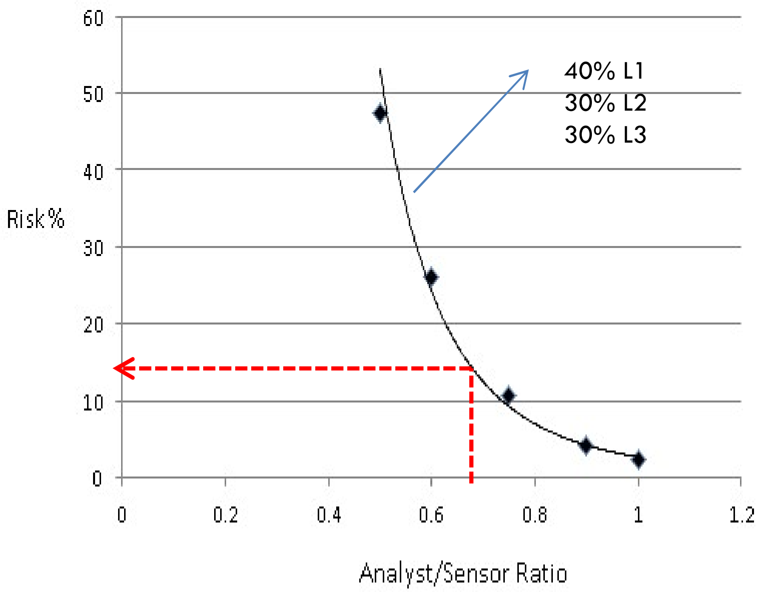
Risk in %	0-5%	25-30%	40-45%
All L3 analysts	7	5	3
Utilization of analyst	>95%	>95%	>95%
Number of sensors	10	10	10
Alert generation rate	U(0,13)	U(0,13)	U(0,13)
Number of sensors per L3 analyst	4-5	4-5	4-5

Research Findings: Optimization specifying expertise mix proportions

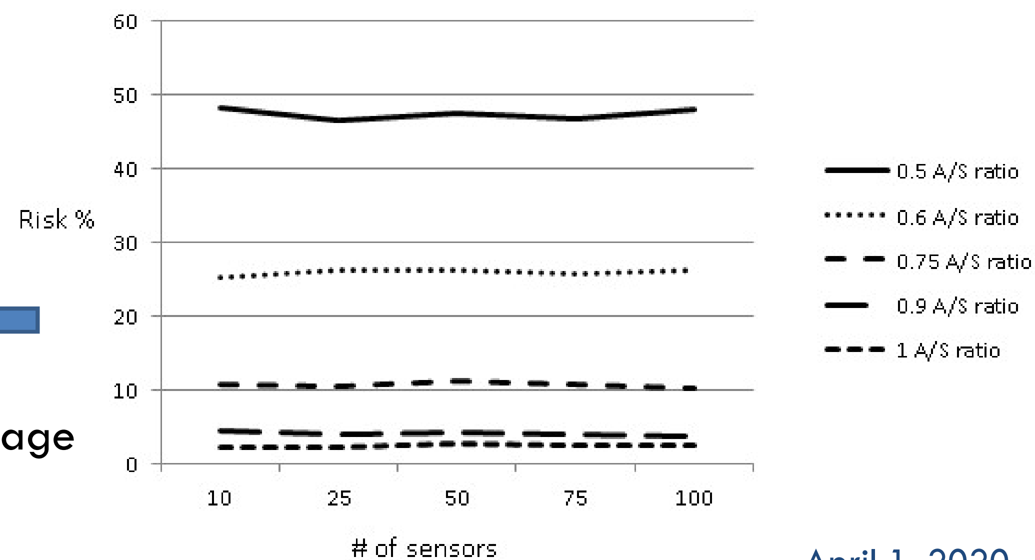
- Multiple sensors to analyst
  - Another input – Proportion of L3, L2, and L1 personnel 20-40% L3, 40-50% L2, and 30-40% L1
- >95% utilization of analyst time
- At 100% alert coverage, analyst/sensor ratio = 0.8
- At 75% alert coverage, analyst/sensor ratio = 0.6
- At 55% alert coverage, analyst/sensor ratio = 0.5

Risk in %	0-5%	25-30%	40-45%
Analysts experience mix	2-L1, 3-L2, 3-L3	1-L1, 3-L2, 2-L3	1-L1, 2-L2, 2-L3
Total number of analysts	8	6	5
Utilization of analyst	>95%	>95%	>95%
Number of sensors	10	10	10
Alert generation rate	U(0,13)	U(0,13)	U(0,13)
Number of sensors per L1 analyst	1-2	1-2	1-2
Number of sensors per L2 analyst	2-3	2-3	2-3
Number of sensors per L3 analyst	4-5	4-5	4-5

Main Results



- Risk% varies non-linearly with analyst/sensor (A/S) ratio
- Plot is useful for hiring decisions
- Assumption: All sensors have the same average alert generation rate, and it remains fixed



For a given analyst/sensor ratio risk is independent of the # of sensors, when the average alert arrival and average service rates remain the same

Sample days off Scheduling

- An analyst works 12*6 + 1*8 = 80 hrs in 2 weeks (7 out of every 14 days from Sun to Sat)
- Gets every other weekend off
- Works no more than 5 consecutive days in a 14 day period

Output of the days-off scheduling algorithm or 10 analysts

Day →	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
1	X	X	X	X			X			X	X				X	X	X	X			X	
2	X	X		X	X	X					X	X			X	X		X	X	X		
3	X	X			X	X					X	X	X		X	X			X	X		
4	X	X				X	X			X			X	X	X	X				X	X	
5	X	X	X				X			X		X		X	X	X	X				X	
6			X	X	X			X	X				X	X			X	X	X			X
7				X	X			X	X	X	X			X				X	X			X
8			X		X	X		X	X		X	X					X		X	X		X
9				X		X	X	X	X			X	X					X		X	X	X
10			X				X	X	X	X			X	X			X				X	X

X – off days

Optimization Recommendations

For an organization that seeks a mix of L3, L2, and L1 analysts

- Use single queue system of alerts in the sensor group
  - When a group of analysts are allocated to a group of sensors by the optimization algorithm, the alerts generated by that group of sensors are arranged in a single queue based on their arrival time-stamp
  - the next available analyst within that group will draw the alerts from the single queue based on a first-in-first-out rule.
- Set proportion of mix L3, L2, L1 level
  - Optimization tends to maximize number of L3 analysts (budget is not considered)
- Do not allocate a sensor only to a junior L1 analyst
  - A junior must be assigned to a sensor that also has a senior L3 person
- All sensors must have at least 1 senior level personnel
- Do not let everyone work on all sensors as an when they become available.
  - The juniors will reduce the overall efficiency of the system.
  - Let optimization decide which junior is paired with a senior and on which sensor.

Questions?

Sushil Jajodia

jajodia@gmu.edu

http://csis.gmu.edu/jajodia