New Challenges in Security

Dan Dongseong Kim
Associate Professor (=full professor in the US),
School of Information Technology and Electrical Engineering,
The University of Queensland, Brisbane, Australia
About myself

• Associate Professor (=full professor in US) at UQ (QS World ranking #48 in 2019), Jan 2019 - present

• Lecturer/Senior Lecturer (Assistant/Associate Prof.) at UC, Aug 2011 – Jan 2019

• Postdoc at Duke University, USA from June 2008- July 2011

• Visiting scholar, University of Maryland, College Park, USA in 2007

• PhD from KAU in South Korea in 2008
The University of Queensland (UQ)

One of Australia’s leading teaching and research institutions

6,600 FTE

- 1,134 Teaching & Research
- 1,455 Research Focused
- 150 Teaching Focused
- 27 Other Academic
- 3,841 Professional

52,000+

- 13,000+ international
- 3,000+ HDR
- 247,000+ alumni

- A global top 50 university in many international rankings
- Annual budget of nearly $1.8 billion
- Outstanding success in the Australian Research Council’s assessment of research performance, the ERA
- Australia’s strongest performer in national teaching awards
- Australia’s strongest performer in research commercialisation
Outline

• Introduction
  ▪ Security objectives
  ▪ Needs for cyber security modeling and analysis
  ▪ STAM terms

• Cyber Security Assessment via 3M

• Two highlights
  ▪ Effectiveness Evaluation of Moving Target Defenses (MTD)
  ▪ Security Analysis of (Software Defined) Internet of Things (IoT)

• Conclusions

• Q&A
Security objectives: CIA triads

Alice in South Island  Internet  Bob in North Island

- The data has not been viewed by a 3rd party
- The data has not been modified in transit
- The data must be available when it is needed

Confidentiality  Integrity  Availability
Needs for cyber security modeling and analysis

• Understand
  ▪ What the systems (e.g., assets) to protect are
  ▪ What the attack goals are
  ▪ What attacks are likely to occur
    • e.g., Probability of attack success

• Plan where/how to best spend a security budget to reduce security risks

• **System**
  - Asset (System Resource): Hardware, Software, Data, communication facilities and networks
  - Vulnerability: A flaw or weakness that could be exploited to violate the system’s security policy

• **Threat**
  - is a possible danger that might exploit a vulnerability

• **Attack**
  - Carried out threat
  - **Attack Surfaces** (Network, Software, Human): consists of the reachable and exploitable vulnerabilities in a system

• **Mitigation (via prevention, detection, tolerance)**
  - The elimination or reduction of the frequency, magnitude, or severity of exposure to risks, or minimization of the potential impact of a threat or warning
STAM – terms: examples

• **System**: Windows OS

• **Threat**: an exploit to a windows OS vulnerability

• **Attack**: an exploit is executed for the Windows vulnerability

• **Mitigation**: a security patch
• Introduction
  ▪ Security objectives
  ▪ Needs for cyber security modeling and analysis
  ▪ STAM terms

• Cyber Security Assessment via 3M
  • Two highlights
    ▪ Effectiveness Evaluation of Moving Target Defenses (MTD)
    ▪ Security Analysis of (Software Defined) Internet of Things (IoT)

• Conclusions

• Q&A
How secure is my network?

How to assess cybersecurity?

NIDS: network intrusion detection system
To assess cyber security, one requires 3\textbf{M}:

1. **Security Measures**
   - To \textit{collected} required information.
   - Vulnerabilities, reachability, \textit{etc.}

2. **Security Metrics**
   - To \textit{represent} the analysis of security
   - Attack-defense scenarios, prob. of attack success

   - To \textit{capture} security using simulation, analytic models, or hybrid models.
Security Measures

• Vulnerabilities and their scores
  ▪ Common Vulnerability and Exposures (CVE)
  ▪ Common Vulnerability Scoring System (CVSS) Base Score (BS): e.g., 9 out of 10.

• Reachability
  ▪ Nmap (network mapping)
  ▪ Network Configurations (e.g., access control by firewalls)

• Mitigations
  ▪ Detection (Intrusion Detection, Vulnerability Identification, …)
  ▪ Countermeasure (Patch, firewall rules changes, …)
Security Measurement: relevant related research

• High Interaction Honeypot [Alata et al. EDCC2006] by LAAS-CNRS in France
• Data Driven Security Analysis [Sharma et al. DSN2011] by UIUC in the US
• Measuring Drive-by Download Defense in Depth [Boggs et al. RAID2014] by Columbia Univ. in the US
• Data-driven Understanding of Telephony Threats [Gupta et al. NDSS2015] by Georgia Tech in the US
• Qualitative Analysis (Metrics)
  ▪ Mincuts (Attack countermeasure scenarios)
  ▪ Importance Measures
  ▪ ...

• Quantitative Analysis (Metrics)
  ▪ Probability of Attacks
  ▪ Adversary’s viewpoint
    • Cost of Attack
    • Return on Attack (ROA)
  ▪ Defender’s Viewpoint
    • Risk = Prob.*Impact
    • Security Investment Cost
    • Return on Investment (ROI)
    • ...

Security Metrics (cont.): a classification

Security Metrics

Host based

Without Probability
- Attack cost
- Impact analysis
- Mean time to compromise
- Mean time to recovery
- Mean time to failure
- Mean time to breach
- ...

With Probability
- Probability of attack success
- Probability of detection
- Probability of success of a mitigation
- ...

Network based

Path based
- Shortest path
- Number of path
- Mean of path
- ...

Non-path based
- Critical Vulnerability Set
- Network Compromise percentage
- ..

- SE Yusuf, JB Hong, M Ge, DS Kim, Composite metrics for network security analysis, Software Networking, 2018
• Page rank based metrics [Mehta et al. RAID 2006]
• Attack Graph based Security metrics [Wang et al. DAS2008][Idika and Bhargave, TDSC2012]
• System level vulnerability metrics [Holm et al. TDSC2012]
• Network Centrality Measures [Hong and Kim, DSN2013, DSN 2014]
Graphical Security Models: a lifecycle

Pre-processing
- Reachability
- Vulnerability
- Other if necc.

Construction (Generation)
- Build/Update Security Model

Representation
- Security Model
- Visualization/Storge

Evaluation
- Security metrics
- Security Analysis
  - Prob. of attack success
  - Return on attacks
  - Risk = prob*I
  - ...
- Updated information

Modification
- Applying security best practices
- Change(s) in the network
- Network

Update
Graphical Security Models: a classification

- **Tree based**
  - Attack Trees
  - Defense Trees
  - ... (to be continued)

- **Graph based**
  - Attack Graphs
  - ... (to be continued)

- **Hybrid**
  - Hierarchical Attack Representation models (HARMs)**


* A. Roy, Dong Seong Kim, Kishor S. Trivedi: Scalable optimal countermeasure selection using implicit enumeration on attack countermeasure trees. in Proc. DSN 2012

Graphical Security Models: relevant related research

• Tree based:
  ▪ Attack Trees [Moore et al. CMU TR2001][Mauw et al. ICISC2004][Ray et al. ESORICS2005]
  ▪ Defense Trees [Bistarelli et al. ARES2006]
  ▪ Protection Trees [Edeg et al. HICSS2007]
  ▪ Attack-Defense Trees [Barbara et al. GameSec 2010]
  ▪ Attack Response Trees [Zonouz et al. DSN 2009]
  ▪ Attack Countermeasure Trees [Roy et al. SCN 2012]

• Graph based:
  ▪ Intrusion Graphs [Foo et al. DSN2005]

• Hybrid:
  ▪ HARMs [Hong and Kim, SECAU2012][Hong and Kim, SEC2013][Hong and Kim, IEEE TDSC2016]

JB Hong, DS Kim, CJ Chung, D Huang, A survey on the usability and practical applications of graphical security models, Computer Science Review 26, 1-16, 2017
• To assess cyber security, one requires 3M:
  1. Security Measures
  2. Security Metrics
  3. (Graphical) Security Models
A Lifecycle of Security Modeling and Analysis using 3M

- Vulnerabilities Database (NVD, CVE, SecurityFocus, etc)
- Connectivity (Topology)
- Attacks (threats)
- Detection/Mitigation
- ...

A system or network

Measurement (Vul. Scanners, system admin, …)

A Graphical Security Model (e.g. Attack Graphs)
A Lifecycle of Security modeling and analysis (cont.)

A Graphical Security Model (e.g. Attack Graphs)

A1

A vul. of H1

Optimal Defense?

A vul. of H2

A vul. of H3

1

2

3

T1

Other metrics…
Impact probability
Attack cost
A Lifecycle of Security modeling and analysis (cont.)

- Vulnerabilities Database (NVD, CVE, SecurityFocus, etc)
- Connectivity (Topology)
- Attacks (threats)
- Detection/Mitigation
- …

• Cloud computing,
• Software Defined Networking (SDN),
• Cyber Physical Systems including SCADA,
  • Internet of Things (IoT),
  • Smart Grid,…
My Research Activities

**Software Tools:**
* Safelite/Safeview
  - Seongmo An (KAU, Korea)
  - Kesang Dorjee (UQ)

**Cloud computing**
- Khaled Khan (Qatar U.)
- Armstrong B. (Qatar U.)
- Noora Fetais (Qatar U.)
- Jin Hong (UWA, AUS)

**Internet of Things (IoT)**
- Mengmeng Ge (Deakin, AU)
- Bilal Ishfaq (UC, NZ)
- Walt Lin (UQ)
- Weilun Liu (UQ)
- Jin-Hee Cho (VT, US)
- Hyounghick Kim (SKKU, Korea)
- Jiwon Yoon (Korea U., Korea)
- Ian Welch (VUW, NZ)

**Al/Intrusion Detection/Response**
- Ke He (UQ)
- Diksa Goel (UQ)
- Abdule Rehman (UC)
- Chun Yong Moon (UQ)
- Marcello Cinque (U Naples II, Italy)

**Moving Target Defenses (MTD)**
- Jin Hong (UWA, AU)
- Hooman A. & Julian (Massy U., NZ)
- Hyuk Lim (GIST, Korea)’s team
- Jin-Hee Cho (VT, US) + US ARL researchers
- Dilli Sharma (UC, NZ)
- Minjune Kim (UQ, AUS)
- Thanh Tung Vo (UQ, AUS)
- Julio Mendoca (UFP, Brazil)

**Software Defined Networking (SDN)**
- Taehoon Eom (KAU, Korea)
- Hyuk Lim (GIST, Korea)’s team
- Dijiang Huang (ASU, US)

**Cyber-Attack & Defense Simulation**
- ADD, Korea
- Huy Kang Kim (Korea U.)’s team
- Chun Yong Moon (UQ)
- Simon Yusuf (UQ)

**Smart (Power) Grid**
- Kieran Morris (UC, NZ) – Reliability/Resilience
  - With ECE/WRC at UC
Cyber Security Modeling: my selected contributions

Pre-processing
Reachability
Other if necc.

Construction (Generation)
Vulnerability
Build/Update Security Model
IM-based Security Model Construction (SecureComm 2013)

Representation
Scalability and Adaptability Analysis
(SEC AU 2012, IFIP SEC 2013, Elsevier JNCA 2016)

Logic Reduction Techniques
(IEEE TrustCom 2013)

Evaluation
Security metrics wrt changes
(Trustcom2017, ComNSec2018, ComNet 2018)

Security Model
IM-based Security Analysis
(DSNW 2013), PMA approach (UI C2014), TrustCom18c, QRS18)

Security metrics
Security economic metrics (AISC 2018)

Modification
Cloud, IoT, SDN
(Applying security best practices)

Network
Change(s) in the network

Update
Updated information
Incorporate MTD techniques into the HARM

Incorporate new attacks paths
(IEEE DSNW 2017a)

Incorporate availability
(IEEE DSNW 2017b)

IFIP SEC 2013: “Performance analysis of scalable attack representation models”
IEEE TrustCom 2013: “Scalable Attack Representation Model Using Logic Reduction Techniques”
IEEE DSNW 2013: “Scalable Security Analysis in Hierarchical Attack Representation Model using Centrality Measures”
SecureComm 2013: “Scalable Security Model Generation and Analysis using k-importance Measure”
IEEE DSNW 2014: “What Vulnerability Do We Need To Patch First?”
Elsevier JNCA 2016: “Towards scalable security analysis using multi-layered security models”
IEEE DSNW2017b: “Evaluating Security and Availability of Multiple Redundancy Designs”

IEEE TrustCom18b: “FRVM: Flexible Random Virtual IP Multiplexing in Software-Defined Networks”
MTD2018@ACM CCS18: “Comprehensive Security Assessment of Combined MTD Techniques for the Cloud”
PRDC2018: “Evaluating the Security of IoT Networks with Mobile Devices”
Elsevier ComNSec “Dynamic Security Metrics for Measuring the Effectiveness of Moving Target Defense Techniques”
Elsevier FGCS 2018 “Proactive defense mechanisms for the software-defined Internet of Things with non-patchable vulnerabilities”
• **Advanced Security Technologies for the Internet of Things**, funded by the Ministry of Business, Innovation and Employment (MBIE) of New Zealand (NZ) (2018-2011) at University of Canterbury (UC): 450K NZD (=400K USD), PI

• **Development and Evaluation of Software Defined Networking based Moving Target Defenses**, funded by the USA RDECOM International Technology Center - Pacific (ITC-PAC) and the US Army Research Lab; 70K USD, PI (2018-2019)

• **Security Risk Assessment for Cloud** by Qatar National Priorities Research Program (NPRP) (2016-2019) at UC: 212,701 USD, NZ PI

• **Cloud Security** by NATO Science for Peace and Security (SPS) programme at UC (2012-2015)
  - [http://cloudsecurity.ece.duke.edu/](http://cloudsecurity.ece.duke.edu/)

• Automated Security modeling tools development at UC (2011-present)

• **MiMANSaS: Metrics, Models and Analysis of Network Security and Survivability** at Duke University, USA NSF (2008-2010)
• Hierarchical Attack Representation Models (HARMS)
• Introduction
  ▪ Security objectives
  ▪ Needs for cyber security modeling and analysis
  ▪ STAM terms

• Cyber Security Assessment via 3M
  • Two highlights
    ▪ Effectiveness Evaluation of Moving Target Defenses (MTD)
    ▪ Security Analysis of (Software Defined) Internet of Things (IoT)

• Conclusions

• Q&A
• **Moving Target Defense (MTD)** techniques (continuously) change the attack surface to thwart attacks.

• Changes in the networked system (attack surface) by deploying MTD techniques are captured via the graphical security model (**HARM**).

Related research projects

- **Development and Evaluation of Service-Aware Intelligent Moving Target Defense for Tactical Networks**, funded by the USA RDECOM International Technology Center - Pacific (ITC-PAC) and the US Army Research Lab; 70K USD, PI (2019-2020)

- **Development and Evaluation of Software Defined Networking based Moving Target Defenses**, funded by the USA RDECOM International Technology Center - Pacific (ITC-PAC) and the US Army Research Lab; 70K USD, PI (2018-2019)

- **Advanced Security Technologies for the Internet of Things**, funded by the Ministry of Business, Innovation and Employment (MBIE) of New Zealand (NZ) (2018-2011) at University of Canterbury (UC): 450K NZD (=400K USD), PI
Three dimensions of MTD

1. What to move? (Moving element(s))
2. How to move? (Techniques)
3. When to move? (Adaptation strategy)

Layer
- App (a)
- OS - Host (o)
- VM - instance (i)
- VMM (v)
- Hardware (h)

Diversity (D) | Redundancy (R) | Shuffle (S)

Time | Event | Hybrid
**MTD: 1. What to move? – Moving element(s)**

<table>
<thead>
<tr>
<th><strong>Strategy Layer</strong></th>
<th><strong>MTD</strong></th>
<th><strong>Diversity (D)</strong></th>
<th><strong>Redundancy (R)</strong></th>
<th><strong>Shuffle (S)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>App (a)</strong></td>
<td>D&lt;sub&gt;a&lt;/sub&gt;</td>
<td>Web, Apache (1), IIS (2), GWS (3), nginx (4)</td>
<td>Web servers replica (1)</td>
<td>TCP/UDP Port Number (1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>App, .NET framework (5), Java (6), PHP (7)</td>
<td>Application replica (2)</td>
<td>S&lt;sub&gt;a&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Database, SQL (8), MySQL (9), Oracle (10)</td>
<td>Database backup and replica (3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other, Mail server (11), Proxy servers (13)</td>
<td>Other service replica (4)</td>
<td></td>
</tr>
<tr>
<td><strong>OS - Host (o)</strong></td>
<td>D&lt;sub&gt;o&lt;/sub&gt;</td>
<td>Windows, Server 2000 (1), 9.x (2), Vista (3), 8 (4), 10 (5)</td>
<td>Host OS and VM instances Replica (1)</td>
<td>S&lt;sub&gt;o&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Linux, RedHat (6), Debian (7), Caldera (8)</td>
<td></td>
<td>IP Address (1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Solaris (9)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other, Unix (10), HP-UX (11)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VM – instance (i)</strong></td>
<td>D&lt;sub&gt;i&lt;/sub&gt;</td>
<td>Same as OS</td>
<td></td>
<td>S&lt;sub&gt;i&lt;/sub&gt;</td>
</tr>
<tr>
<td><strong>VMM (v)</strong></td>
<td>D&lt;sub&gt;v&lt;/sub&gt;</td>
<td>Xen (10)</td>
<td>Hypervisor’s Replica (1)</td>
<td>S&lt;sub&gt;v&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vmware (11)</td>
<td></td>
<td>Failover (1), Switchover (2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ESXi (12)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other, KVM (13), VBox (14), IBM-vSphere (15)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Hardware (h)</strong></td>
<td>D&lt;sub&gt;h&lt;/sub&gt;</td>
<td>Intel (16)</td>
<td>Hardware Backup and Replica (1)</td>
<td>S&lt;sub&gt;h&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HP (17)</td>
<td></td>
<td>Hardware Replacement (1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sun Solaris (18)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other, ARM (19), Atmega (20)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
MTD: 2. How to move? (Techniques)

- Jin Bum Hong, S. Yoon, Hyuk Lim, Dong Seong Kim: Optimal Network Reconfiguration for Software Defined Networks Using Shuffle-Based Online MTD. SRDS 2017: 234-243
- Hooman Alavizadeh, Jin B. Hong, Julian Jang-Jaccard, Dong Seong Kim: Comprehensive Security Assessment of Combined MTD Techniques for the Cloud. MTD@CCS 2018: 11-20
MTD: 3. When to move?

Time-based MTD

Event-based MTD

System starts

Time

An Intrusion Detection Alert
MTD: relationships

Security Analysis

Qualitative Analysis

Probabilistic Analysis

Mincuts

Structural Importance

Prob. of attacks

Cost

Impact

Risk

ROI & ROA

Birnbaum Importance

Graphical Security Modeling

MTD Techniques

Metrics

Security

Economical

Performance

Combinations

I-HARM

AG

AT

Diversity

Redundancy

SHARPE

Analytic Models Evaluation

Cloud Layers for MTD

Hypervisor

VM & Host

Application

Hardware
MTD combinations and metrics

- With VT (Prof JinHee Cho), GIST (Prof Hyuk Lim), the US Army Research Lab Researchers, Massey U. (Prof J. Jang-Jaccard).

### Strategy Example 1 (D-R-S)

\[
\{D_{a1} + D_{a2} + D_i\} + \{R_{a1} + R_{v1}\} + \{S_{a1} + S_{o1}\}
\]

### Strategy Example 2 (R-S)

\[
\{R_{a4} + R_{v1}\} + \{S_{a1} + S_{h1}\}
\]

#### Costs

- **Cost\(_1\)**
  - Opex\(_1\) *(Operational expenses)*
  - Capex\(_1\) *(Capital expenses)*
- **Cost\(_2\)**
  - Opex\(_2\)
  - Capex\(_2\)

---

[Diagram showing MTD combinations and metrics with various metrics and strategies.]
Are MTD techniques useful?

- MTD techniques are effective?
- How we can evaluate the effectiveness?
  - Security enhancement?
  - Other attributes degradation? e.g., Performance and availability
- Details on
MTD implementation in a Cloud

Unitec Cloud.

• Details on
  - Available at: https://arxiv.org/pdf/1904.01758.pdf
Think about what MTD can do for those attack surface?

**ATTACK SURFACE: THE DEVICE**
- Browser
  - Phishing
  - Framing
  - Clickjacking
  - Man-in-the-Middle
  - Buffer Overflow
  - Data Caching
- System
  - No Passcode/Weak Passcode
  - iOS® Jailbreak
  - Android™ Rooting
  - OS Data Caching
  - Passwords and Data Accessible
  - Carrier-Loaded Software
  - No Encryption/Weak Encryption
  - User-Initiated Code
- Phone/SMS
  - Baseband Attacks
  - SMishing

**ATTACK SURFACE: THE NETWORK**
- Wi-Fi (No Encryption/Weak Encryption)
- Rogue Access Point
- Packet Sniffing
- Man-in-the-Middle (MITM)
- Session Hacking
- DNS (Domain Name System) Poisoning
- SSL (Secure Sockets Layer) Strip
- Fake SSL Certificate

**ATTACK SURFACE: THE DATA CENTER**
- Web Server
  - Platform Vulnerabilities
  - Server Misconfiguration
  - Cross-Site Scripting (XSS)
  - Cross-Site Request Forgery (XSRF)
  - Weak Input Validation
  - Brute Force Attacks
- Database
  - SQL Injection
  - Privilege Escalation
  - Data Dumping
  - OS Command Execution
• Introduction
  ▪ Security objectives
  ▪ Needs for cyber security modeling and analysis
  ▪ STAM terms

• Cyber Security Assessment via 3M
  • Two highlights
    ▪ Effectiveness Evaluation of Moving Target Defenses (MTD)
    ▪ Security Analysis of (Software Defined) Internet of Things (IoT)
  • Conclusions
  • Q&A
Security Analysis of (Software Defined) IoT

- IoT security modeling and analysis framework
- Security Analysis of Software Defined IoT
- Network level defenses for IoT
IoT refers to a world where everyday physical objects become locatable, addressable and reachable in the virtual world, and are able to communicate with each other and with other entities (e.g. human beings) [1].

Gartner, Inc. forecasts that 4.9 billion connected things will be in use in 2015, up 30 percent from 2014, and will reach 25 billion by 2020.
Devices
(hardware, operating systems)

IoT architecture design

Communication protocols (WiFi, Bluetooth, ZigBee, etc.)

Services APIs

Service applications

IoT vulnerabilities/attack surfaces
• A study published by HP Fortify on demand:
  - 10 of most popular devices including webcam, home thermostat, home alarm, and garage door opener, etc.
  - An average of 25 vulnerabilities per device.

Introduction (cont.) – Mirai botnet

Source 2: https://commons.wikimedia.org/wiki/File:Mirai-botnet-linked-to-massive-ddos-attacks-on-dyn-dns-gif.gif

By Mengmeng Ge, Jin B. Hong, Walter Guttmann, Dong Seong Kim
## Related Work – IoT security

<table>
<thead>
<tr>
<th>Related work</th>
<th>List of papers</th>
<th>Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Security framework</strong></td>
<td>Radomirovic (2010) [2]</td>
<td>No analytical and/or simulation work</td>
</tr>
<tr>
<td></td>
<td>Yang <em>et al.</em> (2011) [3]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Stepanova <em>et al.</em> (2014) [4]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Atamli <em>et al.</em> (2014) [14]</td>
<td></td>
</tr>
<tr>
<td><strong>Game-based security modeling</strong></td>
<td>Hamdi <em>et al.</em> (2014) [8]</td>
<td>Solutions for certain attacks or scenarios</td>
</tr>
<tr>
<td></td>
<td>Chen <em>et al.</em> (2014) [6]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rontidis <em>et al.</em> (2015) [7]</td>
<td></td>
</tr>
<tr>
<td><strong>Adaptive security models in the ASSET project</strong></td>
<td>Savola <em>et al.</em> (2012) [9]</td>
<td>Solutions for a specific application domain</td>
</tr>
<tr>
<td></td>
<td>Abie <em>et al.</em> (2012) [10]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Torjusen <em>et al.</em> (2014) [12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hamdi <em>et al.</em> (2014) [8]</td>
<td></td>
</tr>
</tbody>
</table>
The goal is to depict potential attack paths in the IoT, evaluate the security level through well-defined security metrics, and assess the effectiveness of different defense strategies.

The Proposed Framework

1. **Phase 1: Data Processing**
   - Input
   - System Information
   - Security Metrics

2. **Phase 2: Security Model Generation**
   - IoT Generator
   - Vulnerabilities and Network Information
   - Security Model Generator

3. **Phase 3: Security Visualization**
   - Security Visualization
   - Security Evaluator
   - SHARPE Tool

4. **Phase 4: Security Analysis**
   - Analysis Results
   - Output
   - Defense Strategies

5. **Phase 5: Model Updates**
   - Security decision maker
   - Input
   - Metric Pool
   - System Information
   - Security Metrics

Security decision maker

Input

System Information

Security Metrics

Metric Pool

Defense Strategies

IoT Generator

Vulnerabilities and Network Information

Security Model Generator

Security Evaluator

SHARPE Tool

Analysis Results

Output

Defense Strategies

Phase 1: Data Processing

Phase 2: Security Model Generation

Phase 3: Security Visualization

Phase 4: Security Analysis

Phase 5: Model Updates
The Proposed Framework (cont.)

Phase 1: Data Processing

Security decision maker

Input
System Information
Security Metrics

IoT Generator

System information
- Subnets
- Nodes
- Vulnerabilities

Security metrics (Static)
- Attack success probability
- Attack cost
- Attack impact
- Mean-time-to-compromise
- Other metrics…
Phase 2: Security Model Generation

- The Model Generator takes the constructed network as inputs and automatically computes all possible attack paths.
- Graphical security model is based on a HARM [16].

Phase 3: Security Visualization

- Attack paths are visualized in the form of an attack graph in the upper layer and middle layer respectively, and a set of attack trees in the lower layer.
Phase 4: Security Analysis

The Security Evaluator can perform one of the two options:

- Carry out the calculation and output results directly
- Generate a textual input file and export the file into the analytic modeling and evaluation tool, named SHARPE* [15]

*SHARPE (Symbolic Hierarchical Automated Reliability and Performance Evaluator)
https://sharpe.pratt.duke.edu/
Phase 5: Model Updates

The deployment of the defense strategy changes:

- Either the vulnerability information; or
- The topology information

Both of them should be updated and taken as the input to the IoT Generator.
An attacker’s path to a target IoT device in a smart home scenario.
1. Construct a malicious media file by using vulnerabilities in supported media formats
2. Upload the file on the Internet
3. the victim downloads the malicious file and starts to play back the video file [17]

1. Write a malware which exploits three bugs to get the root permission
2. change the transmission power of ZigBee chip integrated in the device [18]
- Phase 1: data processing

## Home network information

<table>
<thead>
<tr>
<th>Information</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subnet</td>
<td>WiFi (star), ZigBee (mesh)</td>
</tr>
<tr>
<td>Node</td>
<td>Hub, TV, Tablet, ZigBee routers, ZigBee end devices</td>
</tr>
<tr>
<td>Vulnerabilities</td>
<td>TV: two vulnerabilities in supported media files</td>
</tr>
<tr>
<td></td>
<td>Tablet: three software bugs</td>
</tr>
</tbody>
</table>
• Phase 1: data processing (cont.)

### Security metrics’ values of vulnerabilities

<table>
<thead>
<tr>
<th>Device</th>
<th>Vulnerability</th>
<th>Attack Success Probability</th>
<th>Attack Cost</th>
<th>Attack Impact</th>
<th>Compromise rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>TV</td>
<td>Vulnerability 1</td>
<td>0.3</td>
<td>5.0</td>
<td>6.0</td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>Vulnerability 2</td>
<td>0.3</td>
<td>4.0</td>
<td>6.0</td>
<td>0.4</td>
</tr>
<tr>
<td>Tablet</td>
<td>Vulnerability 1</td>
<td>0.8</td>
<td>4.0</td>
<td>4.0</td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>Vulnerability 2</td>
<td>0.6</td>
<td>6.0</td>
<td>6.0</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Vulnerability 3</td>
<td>0.5</td>
<td>7.0</td>
<td>8.0</td>
<td>0.5</td>
</tr>
</tbody>
</table>

From CVSS BS exploitability/10

From tools that attackers used

From CVSS BS impact

Guessed estimates
• Phase 2 and 3: security model generation and visualization

Phase 4: security analysis

Which device(s) do we need to protect first?

<table>
<thead>
<tr>
<th></th>
<th>Attack Success Probability</th>
<th>Attack Cost</th>
<th>Attack Impact</th>
<th>Mean-Time-To-Compromise</th>
</tr>
</thead>
<tbody>
<tr>
<td>TV</td>
<td>0.51</td>
<td>4.0</td>
<td>6.0</td>
<td>1.43</td>
</tr>
<tr>
<td>Tablet</td>
<td>0.24</td>
<td>17.0</td>
<td>18.0</td>
<td>2.39</td>
</tr>
</tbody>
</table>

- TV is the entry point in the attack path.
- Attacking TV has higher success probability, lower cost and lower mean-time-to-compromise.
• Phase 5: model updates

Which defense strategy is more effective?

<table>
<thead>
<tr>
<th></th>
<th>Attack Success Probability</th>
<th>Attack Cost</th>
<th>Attack Impact</th>
<th>Mean-Time-To-Compromise</th>
</tr>
</thead>
<tbody>
<tr>
<td>No defense</td>
<td>0.12</td>
<td>21.0</td>
<td>24.0</td>
<td>3.81</td>
</tr>
<tr>
<td>Defense (patch) on the</td>
<td>0.07</td>
<td>21.0</td>
<td>24.0</td>
<td>4.89</td>
</tr>
<tr>
<td>Vulnerability 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Defense on the Vulnerability</td>
<td>0.07</td>
<td><strong>22.0</strong></td>
<td>24.0</td>
<td><strong>5.72</strong></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• Defense on vulnerability 2 incurs more attack cost and longer mean-time-to-compromise.
• Attacker models:
  - Multiple attackers - extensible
  - Multiple targets - extensible

• Defense strategies (in the next slides set)
  - IoT devices might have non-patchable vulnerabilities
  - Network-level defense strategies
Security Analysis of (Software Defined) IoT

- IoT security modeling and analysis framework
- Security Analysis of Software Defined IoT
- Network level defenses for IoT
Proactive Defense Mechanisms for the Software-defined Internet of Things with Non-Patchable Vulnerabilities.

*Future Generation Computer Systems*

By Mengmeng Ge, Jin B. Hong, Simon Enoch Yusuf, Dong Seong Kim
IoT vulnerabilities – three types

Vulnerabilities (aka, security weaknesses)

- Known
- Unknown (zero day)
- Forever day
• Problem:
  - Known, zero-day, forever-day vulnerabilities
  - Our focus is **non-patchable** vulnerabilities of IoT devices in an IoT network

• Our approach:
  - Use SDN-based architecture for IoT networks
  - Change the attack surface of the IoT networks to increase the attack efforts and reduce the risk
  - Propose a proactive defense mechanism to reconfigure the network
  - Analyze the security and performance of the network
Proposed Defense Mechanisms for the SD IoT

Defense mechanisms

Proactive
- Topology reconfiguration
- Prioritized patch

Reactive
- Isolate compromised non-patchable nodes
- Patch non-compromised patchable node
An Example Network

Controller software

Base station

Internet

Software-defined sensor node with patchable vulnerabilities

Software-defined sensor node with non-patchable vulnerabilities

Node distance: 30 meters

Communication range: 45 meters
• Reconfiguration goal:
  - Maximize the number of patchable nodes along the path to the base station without changing the hop count

• The reason:
  - Less number of exploitable vulnerabilities for the attacker to reach the base station if some vulnerabilities are patched along the path.
An Example Network (cont.)

After calculation of maximum number of patchable nodes (MNPN) in a path to the BS; Compare the current and new.

Controller software
Base station
Software-defined sensor node with patchable vulnerabilities
Software-defined sensor node with non-patchable vulnerabilities

S1 is the only node to the BS, I don’t reconfigure

S2 has the MNPN to the BS, I don’t reconfigure

No! it increases the hop count!

MNPN =1; No.

MNPN =2 (S5-S2); Yes!

It increases the hop count! Nah…

I don’t want reconfigure because I am directly connected to the BS.

The same with S2; S1 is the only node to the BS, I don’t reconfigure

For S5, S6, Same with S4 😊
Before proactive reconfiguration (initial)

- Software-defined noise sensor node with patchable vulnerabilities

After proactive reconfiguration (reconfigured)

- Software-defined weather and air quality sensor node with non-patchable vulnerabilities
Security Modeling and Analysis (cont.)
• One example network topology (# patchable nodes = 50)
Security Analysis of (Software Defined) IoT

- IoT security modeling and analysis framework
- Security Analysis of Software Defined IoT
- Network level defenses for IoT

Mengmeng Ge$^{1,3}$, Jin-Hee Cho$^2$, Bilal Ishfaq$^1$, and Dong Seong Kim$^{1,4}$

$^1$University of Canterbury, New Zealand  
$^2$Virginia Tech, USA  
$^3$Deakin University, Australia  
$^4$The University of Queensland, Australia
We developed an integrated proactive defense system by proposing an adaptive MTD technique by shuffling a network topology where a network consists of both decoy nodes and real nodes.

- network topology shuffling-based MTD (NTS-MTD) with decoy nodes is to generate a network topology that can maximize disadvantages against the attackers.

We took a metaheuristic approach based on a genetic algorithm (GA) by devising a fitness function that can achieve the objective of our proposed proactive defense mechanism in terms of minimizing defense cost as well as security vulnerability in attack paths.

- We used a software-defined networking (SDN)-based IoT as our network environment.

- We adopted a graphical security model to evaluate the proposed deception and MTD techniques.
Network-level defenses

1. System Information
2. Decoy Deployment
3. Security model Generator takes the shuffled network and generates the HARM to capture attack paths.
4. Shuffling evaluator takes HARM as input with metrics and compute the results.
5. Optimization module applies the multi-objective genetic algorithm to compute the optimal topology for the IoT network.

IoT Generator
- Construct IoT network with node vulnerability information.

Topology Generator
- Randomly generates different topologies with decoy based on shuffling algorithm.

Security model Generator
- Takes the shuffled network and generates the HARM to capture attack paths.

Optimization Module
- Applies the multi-objective genetic algorithm to compute the optimal topology for the IoT network.
Proposed Network-level defenses

Network level Defenses

Reactive
- Topology Reconfiguration
- ...

Proactive
- Deception
- Decoy systems with Moving Target Defense (MTD)
• **Reactive Defense Mechanism:**
  - is triggered when the attack is detected by the Intrusion Detection System (IDS).
  - is based on the topology reconfiguration of IoT network using the optimal reconfiguration algorithm.

• **Example an IoT Network:**

![Diagram of an IoT network with SDN Controller and IDS, base station, and IoT nodes.](image)

- The IoT nodes are non-patchable and two types; easy or hard-to-exploit.
- The reconfiguration algorithm replaces the easy-to-exploit IoT nodes with the hard-to-exploit IoT nodes at the entry points (i.e., S2, S3, S7, S8, S13, S12) of the attackers.
- By doing so, the less chances for the attackers to reach the target (server).
Network-level defenses (cont.)

• After Reconfiguration:

- After reconfiguration of IoT network topology, the easy-to-exploit IoT nodes (S2, S7, S8, S12) are replaced with the hard-to-exploit IoT nodes (S1, S4, S6, S9).
- The new IoT network topology contains the hard-to-exploit IoT nodes at the entry points of the attackers.
Proposed Network-level defences (revisited)

Network level Defenses

Reactive
- Topology Reconfiguration
- ...

Proactive
- Deception
- Decoy systems with Moving Target Defense (MTD)
• Deception technologies:
  ▪ provide an additional defense layer to the traditional security solutions (firewalls, IDS etc.).
  ▪ attract the attacker toward the decoy systems and permits the defender to record the behaviours of the attacker.
  ▪ integrates honeypot technology with visualization and automation techniques.

• Decoy types and authenticity Level:
  ▪ Emulation-based:
    • A variety of fake assets
    • Large scale coverage.
  ▪ Full OS-based:
    • Replication of production devices.

• Decoy deployment location:
  ▪ All the virtual local area networks (VLANs) of an IoT network.
• Deception Technology (cont.)
  ▪ Example IoT Network with Decoy Deployment:

  Decoy deployment as a defense against attackers.

  • In each VLAN, deploy at least one decoy node.
  • Each decoy has one vulnerability to lure the attacker.
  • Once the attacker interact with the decoy, the behaviour of the attacker is captured and he reaches to the decoy server.
  • Each real IoT node has one known un-patched vulnerability which attackers exploit to gain root permission.
Proposed Network-level defences (revisited)

Network level Defenses

Reactive
  - Topology Reconfiguration
  - ...

Proactive
  - Deception
  - Decoy systems with Moving Target Defense (MTD)
• Decoy System with **Moving Target Defense (MTD):**
  - Additionally deploy a network topology shuffling-based MTD technique which is executed adaptively with system security vulnerability.

• Adopt a metaheuristic approach based on a genetic algorithm to achieve the minimum defense cost and security vulnerability in attack paths.

• Develop evaluation metrics to evaluate the effectiveness:
  - Number of attack paths towards decoy targets ($N_{DN}^{AP}$),
  - Mean Time to Security Failure (MTTSF),
  - Defense Cost Per Unit Time
• The two types of ‘when to shuffle a network topology’ strategies:
  - Fixed Shuffling (FS): This shuffling represents a baseline scheme using a fixed time interval to shuffle a given network topology.
  - Adaptive Shuffling (AS): This shuffling is designed to execute the MTD in an adaptive manner based on the system security level detected by the defender

• The two types of ‘how to select a network topology’ strategies are:
  - Random Network Topology (RNT): used as a baseline model that can simply select a network topology based on a simple random selection of a node’s edges to other nodes.
  - GA-based Network Topology (GANT): a network topology to be used for a next round of shuffling is selected based on the network topology that maximizes the objective functions used in the Genetic Algorithm (GA).
• Simulation Steps:
  ▪ Based on Shuffled network topology, mean time to security failure (MTTSF) is calculated.
  ▪ For each shuffled network, HARM is constructed to calculate the potential attack paths.
  ▪ An attacker randomly selects one entry point to compromise the IoT node.
  ▪ The attacker’s intelligence is based on interaction probability with decoy.

• Objective Function Aim:
  ▪ To maximize the $N_{DN}^{AP}$, MTTSF and to minimize the Defense Cost.
    $$\max w_N N_{DN}^{AP} + w_M E(MTTSF) + w_C C_D$$
  ▪ Where $w_N$, $w_M$ and $w_C$ are weights for each metric $w_N + w_M + w_C = 1$.

• Optimal Solution = network topology with maximum objective value.
• In this study, we consider an equal weight for $w_N, w_M$ and $w_C$, respectively, with $1.0/3.0$.

• We assume the following algorithm parameters for the simulations with GANT:
  - population size ($N$) = 100,
  - maximum number of generations ($N_g$) = 100,
  - crossover rate ($r_c$) = 0.8 and
  - mutation rate ($r_m$) = 0.2.

• In RNT, we randomly change the edges between the real IoT nodes and decoy nodes.
  - We also considered the probability that an edge will be shuffled (i.e., add/remove an edge) with $P_r = 0.5$. 
We assume there is an attacker during each simulation run.

The attacker randomly chooses entry points and compromises nodes along the attack paths with the behaviors.

By using the fixed shuffling schemes, the network may be shuffled while a node is under attack.

We assume the attacker is forced to quit the network due to lost connections and needs to find ways to break into the network.

During the subsequent attack after shuffling, the attacker could continue his previous attack action once he encounters the same real node next time (i.e., MTTC for the real node is accumulated throughout the MTTSF).

By using the adaptive shuffling schemes, the network is shuffled due to the system security level detected by the defender.

The attacker is also forced to quit the network after each shuffling due to lost connections and needs to find ways to re-enter the network.

For both schemes, the decoy node is cleared at each shuffling.

Attacker model in simulation
• varying the number of decoys in each VLAN and the level of attackers’ intelligence in detecting decoy nodes (i.e., emulation $P_{d em}^e$ and full OS $P_{d OS}^d$):
  ▪ S(1) one decoy node assigned for each VLAN in the presence of low-intelligent attackers (i.e., $P_{d em}^e = 0.9$ and $P_{d OS}^d = 1.0$);
  ▪ (2) one decoy node assigned for each VLAN in the presence of medium-intelligent attackers (i.e., $P_{d em}^e = 0.5$ and $P_{d OS}^d = 0.9$);
  ▪ (3) two decoy nodes assigned for each VLAN in the presence of low-intelligent attackers (i.e., $P_{d em}^e = 0.9$ and $P_{d OS}^d = 1.0$).
Simulation Results – Scenario 1

- Performance comparison of the four MTD schemes with low-intelligent attackers where one decoy node is with each VLAN.

(a) Average number of attack paths towards decoy targets \( \overline{N_{DT}^{AP}} \)

(b) Average MTTSF \( \overline{MTTSF} \)

(c) Average defense cost per unit time \( \overline{C_D} \)

- Fixed Shuffling (FS)
- Adaptive Shuffling (AS)
- Random Network Topology (RNT)
- GA-based Network Topology (GANT)

Overall, GA-based schemes are capable of preserving higher security level in terms of \( \overline{N_{DN}^{AP}} \) and maintaining lower cost while fixed shuffling schemes incur higher \( \overline{MTTSF} \).
• Performance comparison of the four MTD schemes with medium-intelligent attackers where one decoy is with each VLAN

Simulation Results – Scenario 2

(a) Average number of attack paths towards decoy targets \(N_{DT}^{AP}\)

(b) Average MTTSF \(MTTSF\)

(c) Average defense cost per unit time \(\overline{C_D}\)

• Fixed Shuffling (FS)
• Adaptive Shuffling (AS)
• Random Network Topology (RNT)
• GA-based Network Topology (GANT)

• GA-based schemes are capable of maintaining higher security level in terms of \(N_{DN}^{AP}\) and lower cost while fixed shuffling schemes incur higher \(MTTSF\).
• AS-GANT is resilient under high-intelligent attacks without much reduction of MTTSF when compared with the baseline (scenario I).
• Performance comparison of the four MTD schemes with low-intelligent attackers where two decoy nodes are with each VLAN

(a) Average number of attack paths towards decoy targets ($\overline{N_{DT}^{AP}}$)

(b) Average MTTSF ($\overline{MTTSF}$)

(c) Average defense cost per unit time ($\overline{C_D}$)

Overall, GA-based schemes are capable of preserving higher security level in terms of $N_{DN}^{AP}$. Adaptive shuffling schemes incur lower cost while fixed shuffling schemes incur higher $MTTSF$. We can see there is a balance between MTTSF and defense cost per unit time.

- Fixed Shuffling (FS)
- Adaptive Shuffling (AS)
- Random Network Topology (RNT)
- GA-based Network Topology (GANT)
• (1) Large IoT networks:
  ▪ how to apply our proposed scheme in large-scale IoT networks with a variety of decoy nodes by showing high scalability;

• (2) Sensitivity analysis:
  ▪ carrying out sensitivity analysis by varying other key design parameter values (e.g., weights to consider each system objective, a more number of decoy nodes deployed in the network, a more number of attackers, and/or system security thresholds);

• (3) Other metrics:
  ▪ investigating the effect of deception and/or MTD on service availability, such as delay introduced by the deployment of decoy nodes and/or network topology shuffling; and

• (4) Optimal configurations:
  ▪ identifying an optimal setting of adaptive network topology shuffling algorithms in terms of thresholds in detecting system vulnerabilities.
**Outline**

- **Introduction**
  - Security objectives
  - Needs for cyber security modeling and analysis
  - STAM terms

- **Cyber Security Assessment via 3M**

- **Two highlights**
  - Effectiveness Evaluation of Moving Target Defenses (MTD)
  - Security Analysis of (Software Defined) Internet of Things (IoT)

- **Conclusions**

- **Q&A**
• Graphical Security Models are useful/practical
  ▪ Our formalism named HARMs are scalable and adaptable
  ▪ Our safelite/safeview are being developed for semi-automated cyber security analysis

• Challenges
  ▪ Measurement and Analysis -> Big Data Analytics
  ▪ Automation -> Artificial Intelligence
  ▪ System: various systems (driverless cars, drones, …)
  ▪ Other Defense Requirements
    • e.g., Both Safety and Security
    • e.g., Availability >> Confidentiality
Research vision

Software Defined Cloud Computing

Big Data Analytics

A. I.

Security Models, Measures & Metrics

Automated Security Analysis
New Challenges in Security

Dan Dongseong Kim
Associate Professor,
School of Information Technology and Electrical Engineering,
The University of Queensland, Brisbane, Australia
Thank you

Dr Dan Kim | Associate Professor
School of Information Technology and Electrical Engineering
E-mail: dan.kim@uq.edu.au
Web: https://researchers.uq.edu.au/researcher/23703
My Research Activities

Software Tools: Safelite/Safeview
- Seongmo An (KAU, Korea)
- Kesang Dorjee (UQ)

Al/Intrusion Detection/Response
- Ke He (UQ)
- Diksa Goel (UQ)
- Abdule Rehman (UC)
- Chun Yong Moon (UQ)
- Marcello Cinque (U Naples II, Italy)

Cyber-Attack & Defense Simulation
- ADD, Korea
- Huy Kang Kim (Korea U.)’s team
- Chun Yong Moon (UQ)
- Simon Yusuf (UQ)

Software Defined Networking (SDN)
- Taehoon Eom (KAU, Korea)
- Hyuk Lim (GIST, Korea)’s team
- Dijiang Huang (ASU, US)

Cloud computing
- Khaled Khan (Qatar U.)
- Armstrong B. (Qatar U.)
- Noora Fetais (Qatar U.)
- Jin Hong (UWA, AUS)

Internet of Things (IoT)
- Mengmeng Ge (Deakin, AU)
- Bilal Ishfaq (UC, NZ)
- Walt Lin (UQ)
- Weilun Liu (UQ)
- Jin-Hee Cho (VT, US)
- Hyoungshick Kim (SKKU, Korea)
- Jiwon Yoon (Korea U., Korea)
- Ian Welch (VUW, NZ)

Moving Target Defenses (MTD)
- Jin Hong (UWA, AU)
- Hooman A. & Julian (Massy U., NZ)
- Hyuk Lim (GIST, Korea)’s team
- Jin-Hee Cho (VT, US) + US ARL researchers
- Dilli Sharma (UC, NZ)
- Minjune Kim (UQ, AUS)
- Thanh Tung Vo (UQ, AUS)
- Julio Mendoca (UFP, Brazil)

Graphical Security Measurement, Models, Metrics

Smart (Power) Grid
- Kieran Morris (UC, NZ) – Reliability/Resilience
- With ECE/WRC at UC
• Security for
  ▪ the Cloud, IoT, Edge, and Fog computing
  ▪ Critical Infrastructure protection including CPS, IoT, SCADA, etc.

• Security and Safety for
  ▪ Cyber physical systems including Drones, robots
How to use IDS for security models?

- **NICE+:** Attack Analyzer Workflow

A SDN and GSM (attack graphs)

<table>
<thead>
<tr>
<th>VM ID</th>
<th>CVE ID</th>
<th>CVS</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM1</td>
<td>CVE-2015-0096</td>
<td>9.3</td>
<td>10</td>
</tr>
<tr>
<td>VM2</td>
<td>CVE-2015-0096</td>
<td>9.3</td>
<td>10</td>
</tr>
<tr>
<td>VM3</td>
<td>CVE-2015-0096</td>
<td>9.3</td>
<td>10</td>
</tr>
<tr>
<td>VM4</td>
<td>CVE-2015-2426</td>
<td>9.3</td>
<td>10</td>
</tr>
<tr>
<td>VM5</td>
<td>CVE-2015-1211</td>
<td>7.5</td>
<td>6.4</td>
</tr>
<tr>
<td>VM6</td>
<td>CVE-2015-2426</td>
<td>9.3</td>
<td>10</td>
</tr>
</tbody>
</table>
Vulnerabilities

• Attackers penetrate through networked systems by exploiting vulnerabilities.

• **Three** different types of vulnerabilities:
  1. known vulnerabilities (e.g., WannaCry used it), forever-day vulnerabilities
  2. **zero-day vulnerabilities**
  3. vulnerabilities arising from emerging threats (i.e., new vulnerabilities to the network)
     - e.g., Bring-Your-Own-Device (BYOD) policies, USB file sharing.
Security Assessment using only known vulnerabilities

Collect Security Data → Generate a Security Model → Specify "Unknown Vulnerabilities"

Unspecified vulnerability? yes & no → Security Analysis → Mitigation Strategies

To deal with unknown/new vulnerabilities

Specify "Unknown Vulnerabilities" → Scenario Selection and Security Analysis

Security Decision maker

Countermeasure Selection → Network Change