Verification of Attack Resistance Techniques for Proactive Resiliency
# Cyber Resiliency - Definition

<table>
<thead>
<tr>
<th><strong>Definition</strong></th>
<th><strong>Reference</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>“The ability of the system to <strong>withstand</strong> a major <strong>disruption</strong> within <strong>acceptable degradation</strong> parameters and to <strong>recover</strong> within an <strong>acceptable time</strong> and composite <strong>costs and risks</strong>”.</td>
<td>Haimes, Yacov Y. On the definition of resilience in systems. Risk Analysis, 2009</td>
</tr>
<tr>
<td>“The ability to provide and <strong>maintain</strong> an <strong>acceptable level of service</strong> in the face of <strong>faults</strong> and <strong>challenges</strong> to normal operation.”</td>
<td>Goldman, Harriet. Building secure, resilient architectures for cyber mission assurance.</td>
</tr>
<tr>
<td>“The ability of the system to <strong>recover</strong> from a <strong>cyber intrusion</strong> and to assume close to <strong>normal operations</strong> within an acceptable time and at an acceptable total encompassing cost.”</td>
<td>Chittister, Clyde G and Haimes, Yacov Y. The role of modeling in the resilience of cyberinfrastructure systems and preparedness for cyber intrusions. Journal of Homeland Security and Emergency Management.</td>
</tr>
<tr>
<td>“A resilient system <strong>maintains</strong> state awareness and an <strong>acceptable level of operational normalcy</strong> in response to <strong>disturbances</strong> including <strong>threats</strong> of an unexpected and <strong>malicious nature</strong>.”</td>
<td>Melin, A.M et al. A mathematical framework for the analysis of cyber-resilient control systems. Resilient Control Systems (ISRCS), 2013.</td>
</tr>
</tbody>
</table>

## Cyber Resiliency

The ability of the system to **minimize** the potential **attack damage** in order to maintain the system's state within an **acceptable operational level**.
Resiliency Techniques - Classification

- Proactive
  - Static for Resistance
    - Isolation
    - Diversity
    - Redundancy
  - Dynamic for Deterrence
    - Agility
  - Reactive for Remediation
    - CoA Policies
Cyber Resiliency - Importance

- “A Manifesto for Cyber Resilience” by Semantic.
  - 84% of initial compromises take hours or less.
  - 66% of breaches are not discovered for months.
  - 22% of breaches take months or longer to contain.

  - In 60% of cases, attackers are able to compromise an organization within minutes.
  - 99.9% of the exploited vulnerabilities were compromised more than a year after the CVE was published.
  - 60% of incidents were attributed to errors made by system administrators.
Motivation

- Static resiliency mechanisms.
  - Isolation, diversity, and redundancy.
- Due to systems complexity and human errors it is hard to ensure correct implementation of these mechanisms.
- How to guarantee that they assure system resiliency (resiliency metrics).
- Current configuration techniques does not verify resiliency.
Problem Statement

- **Given**
  - A network configuration.
  - Missions’ requirements.
  - A formal specification for static proactive resiliency techniques.
  - A set of resiliency properties/metrics based on attack attributes.

- **Verify two properties**
  - The network configurations satisfy the static proactive resiliency specification and the mission’s requirements.
  - The network configuration of the proactive resiliency can satisfy the specified metrics.
Challenges

- The specification of resiliency properties and metrics.
  - Provide quantitative measures.
  - Consider attacks with multiple stepping stones.

- Scalability
  - Large and complex networks.
  - Modeling the entire data plan configuration.
  - Large number of attack paths due to stepping stones.
Related Work

● Network Verification Tools.
  ◦ E.g., Veriflow-SIGCOMM’12, NetPlumber-NSDI’13, Anteater-SIGCOMM’11, ConfigChecker-ICNP’09.
  ◦ Verify end-to-end reachability and security.
  ◦ No QoS verification support.
  ◦ Do not focus on resiliency: no indirect reachability, no diversity and isolation properties.

● Network Resiliency Verification.
  ◦ Verifies system traces based and not the real configuration.
  ◦ Verification is limited to one form of resiliency properties.
  ◦ Only consider system failures.
Approach - Overview

Mission Requirements
- End-to-end Reachability
- Resistance Requirements

Network Configuration

Model Builder

SMT-based Model Checker
- Constraints Satisfaction Problem
- Z3 SMT Solver

Resiliency Metrics and Properties

Verification Report

Requirements are not satisfied
(Counterexample)
Mission Requirements (End-to-end Reachability)

**CanReach**
- src/port  # source address(s)
- dst/port  # destination address(s)
- qosExp)   # constraint on the quality of service

**QoS constraints**

Parameter $\rho ::= BW \mid DRate \mid Delay \mid \ldots$

Operator $\sqcd ::= > \mid < \mid \geq \mid \leq \mid ==$

Predicate $\Phi ::= C \mid \max(\rho) \mid \min(\rho) \mid \sum(\rho) \mid \avg(\rho)$

Constraint $\Psi ::= \Phi \sqcd \Phi \mid \Psi \land \Psi \mid \Psi \lor \Psi$

Example.

$\text{CanReach}(\text{clientX/any, serverY/8080, min(DRate) > 512})$
Proactive Static Resiliency Specification (Resistance requirements)

<table>
<thead>
<tr>
<th>Isolation</th>
<th>Diversity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic should traverse a specific sequence of countermeasures before (indirectly) reaching the destination.</td>
<td>The attacker should encounter network components of different attributes in any indirect path to the victim.</td>
</tr>
</tbody>
</table>

resistSpec(src/port, # source address(s)
dst/port, # destination address(s)
a | b% | always, # Min resistance in attack path
BoolExp(isolationCM) | # diversity/isolation requirement
BoolExp(diversity CM))

IsolationCM: Deny | HInspect, DInspect | ESP | AH | …
DiversityCM: os^n | app^n | vul^n | … e.g., (os^{-1} or app^{0})

a: resistance must hold in at least a steps in the attack path
b%: resistance must hold in at least b% steps of the attack path
always: resistance pattern must be maintained in each attack path
# Resistance Specification - Examples

<table>
<thead>
<tr>
<th>Isolation</th>
<th>Diversity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosts in Students labs should not reach students records servers unless the traffic is inspected or tunneled through ESP at least once.</td>
<td>Servers in DMZ should not indirectly reach students record servers unless they encounter different operating systems at least two times.</td>
</tr>
<tr>
<td>\texttt{resistSpec( studentsLabs, Records, 1, (inspect</td>
<td></td>
</tr>
</tbody>
</table>
Bounded Model Checking

Network Configuration
User-defined Functions
Requirements (Reachability + Resistance)
Incremented Sequentially

Transition System
\[ T(s_i, s_j) \]

Property
Temporal logic formula \( f \)

Bound Integer \( k \)

Unroll the transition relation \( k \) times

\[ \bigwedge_{i=0}^{k-1} T(s_i, s_{i+1}) \land f \]

Solve by SMT solvers (Z3)

\{Requirements are Satisfied, Counterexample\}

Encodes packets transformations
\[ T \subseteq loc_i \times pkt_i \times loc_j \times pkt_j \]
Modelling Stepping Stone Attacks

- We extend our model such that packets are forwarded to next potential destinations.
  - Implement a stack to prevent loops.
  - Keep track of the victims in an attack path.
Modeling Middle-boxes

- SDN
  - Support forwarding, filtering, set-header actions.
  - Does not support inspection, encryption, tunneling, … etc.

- This can be resolved by integrating middle-boxes.
  - Not standard SDN configuration.
  - Introduces new actions.
  - SDN forward traffic to appropriate middle-boxes.
Resiliency Properties/Metrics - Verifying Resiliency against DDoS Attacks

- Evaluates the effectiveness of the network to resist specific attacks, given mission and resiliency specs.

\[
\text{CanResist}([\text{MissionReqIDs}], \text{[ResSpecIDs]}, \text{src}/rate; \text{dst}/port \mid \# \text{ddos bots/rates and victims} \\
(x \mid y\%)) \# \text{number or intensity of core flooded links}
\]

Examples.
- \text{canResist}([1-10,15], \text{Isolation5}, \text{labs/10M;Web/80})
- \text{canResist}([1-10,15], \text{Isolation5}, 10\%/10M;\text{Web/80})
  \# \text{sources are } 10\% \text{ of the hosts uniformly distributed}
- \text{canResist}([1-10,15], \text{Isolation5}, 90\%)
  \# \text{sources are do not care}

**Core links** is user-defined as number of hops from any destination in the networks
Resiliency Properties/Metrics - Verifying Resiliency against Worms/APT Attacks

- Evaluates the effectiveness of the network to resist specific attacks, given mission and resiliency specs.

\[
\text{CanResist}(\ \text{MissionReqIDs},
\text{ResSpecIDs},
\text{src/}[\text{vulID}]; \text{dst/port} \mid \# \text{initial compromised hosts} \&
\# \text{potential victims}
\times \mid \gamma \%)
\# \text{threshold for infected victims}
\]

Examples.
- \text{canResist}(([1-10, 15], \text{Isolation5}, \text{labs/any, */any, 75%})
- \text{canResist}(([1-10, 15], \text{Isolation5}, 10%/\text{any, */any, 75%})
- \text{canResist}(([1-10, 15], \text{Isolation5}, \text{lab/any, DB/8080, 50%})
Conclusions

- **Verification of Attack Resistance Techniques for Proactive Resiliency.**
  - We provide a framework to verify the correct deployment of proactive resiliency techniques and verify their integrity and effectiveness with respect to given metrics.
  - We will integrate various middle-boxes in a unified model, complete the resiliency metrics evaluation, and enhance the scalability.

- **Extraction and Verification of Agility Parameters for Dynamic Proactive Resiliency.**
  - We proposed an approach to automatically extract the agility parameters and demonstrated that by examples.
  - We will complete the implementation to extract agility parameters and validate them.

- **Verification of Reactive Resiliency Policies.**
  - We proposed a language for reactive policies and identified a set of conflicts.
  - We will investigate different models to verify policies integrity and effectiveness as well as resolving conflicts.