Quantitative Underpinnings of Secure, Graceful Degradation

Ryan Wagner, Matt Fredrikson, David Garlan

Carnegie Mellon University

Pittsburgh, PA, USA

How do we reason architecturally to trade off functionality for security in the presence of sophisticated adversaries?

High Level Approach:

Axioms:

- Two subtypes of connectors: unprivileged, privileged, and exploited
    - Privileged connectors are a result of the architectural instance and style (rules)
    - An attacker cannot create new connections—must exploit only existing ones
    - Attack traces must follow privileged and exploited connections
- Defenders have a limited budget of tactics
- Attackers have a limited budget of exploits
- Exploits can be reused at no additional cost to attacker
- Attacker budget (capability) is viewed by the defender as a probability mass function

Algorithm:

For each possible defensive tactic set (i.e., within defender's budget to implement):

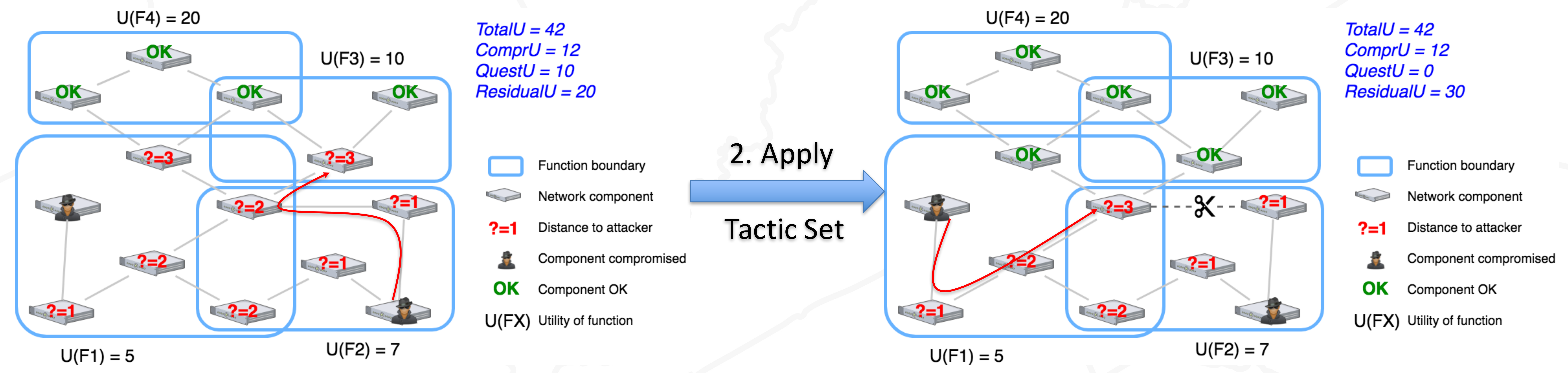
1. Apply the tactics in the set to create an architectural alternative (Datalog)
2. Determine all possible attack traces within the attacker's maximum anticipated capability budget (Datalog)
3. Find worst case attack trace at each possible attacker capability (Python)
4. Based on probability mass function of attacker capability, determine *expected* utility (to defender) of architectural alternative (Python)

Emit best tactic set corresponding to optimal architectural alternative

Detail of Evaluating Attack Traces:

1. Find Worst Case Attack Trace Given Attacker Capability

3. Find *New* Worst Case Attack Trace Given Attacker Capability

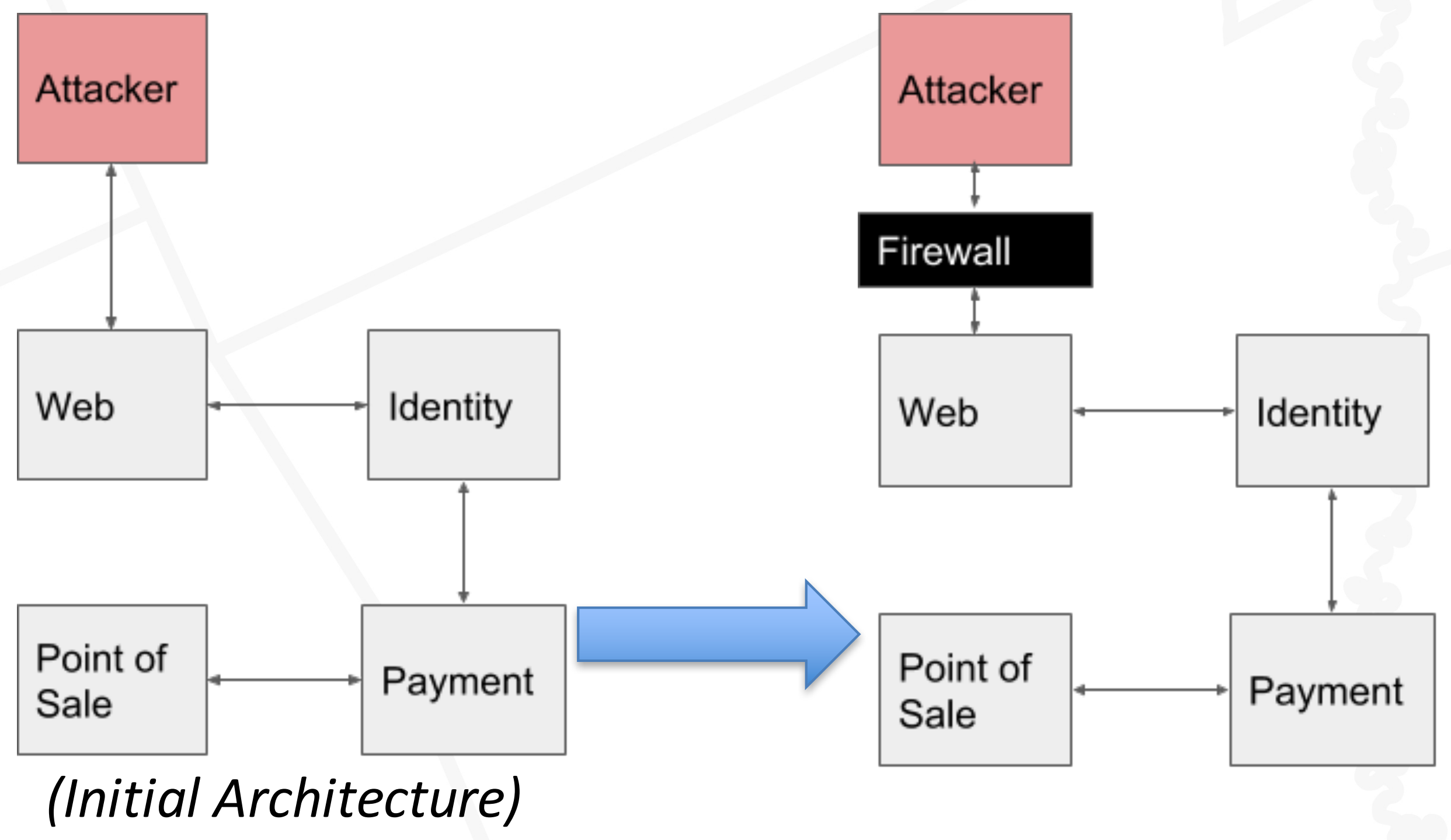


The worst-case attack trace affects three functions. Only one function is operable and secure.

Cutting a connection sacrifices one function to remove worst case attack. Now, two functions are operable and secure.

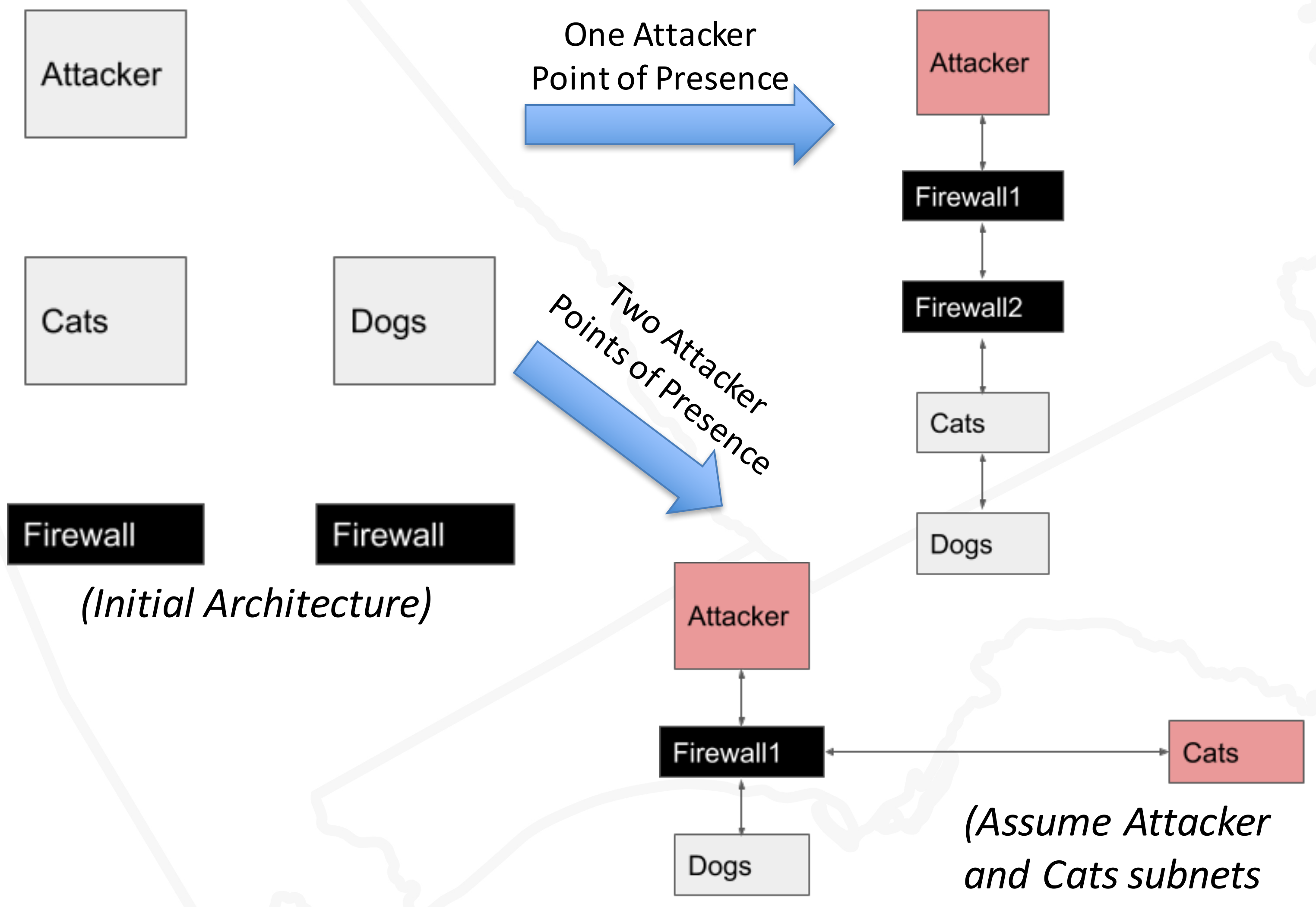
Example Results:

Correct Placement of a Firewall:



(Initial Architecture)

Correct Arrangement of Subnetworks:



(Initial Architecture)

(Assume Attacker and Cats subnets are compromised)

Benefits:

- Generalizable approach that works at multiple levels of abstraction (e.g., host-level, network-level)
- Limited information required for results: no *a priori* knowledge of vulnerabilities needed
- Demonstrates a path forward for adapting architectures in response to sophisticated adversaries

This work is sponsored by the DoD

HOTSOS 2018

isr institute for SOFTWARE RESEARCH

Carnegie Mellon University CyLab Security and Privacy Institute

5TH ANNUAL
HOT TOPICS in the SCIENCE OF SECURITY
APRIL 10-11, 2018 | RALEIGH, NORTH CAROLINA