RISCS

Research Institute in Science of Cyber Security

Imperial College London

Queen Mary University of London

Game Theory Meets Information Security Management

Andrew Fielder
Research Associate
Imperial College London

Overview

- Research Objectives
- Cyber Security Resources
- Model Design
- Game Theoretic Formulation
- Solving The Game
- Experiment Overview
- Results
- Future Work

Research Objectives

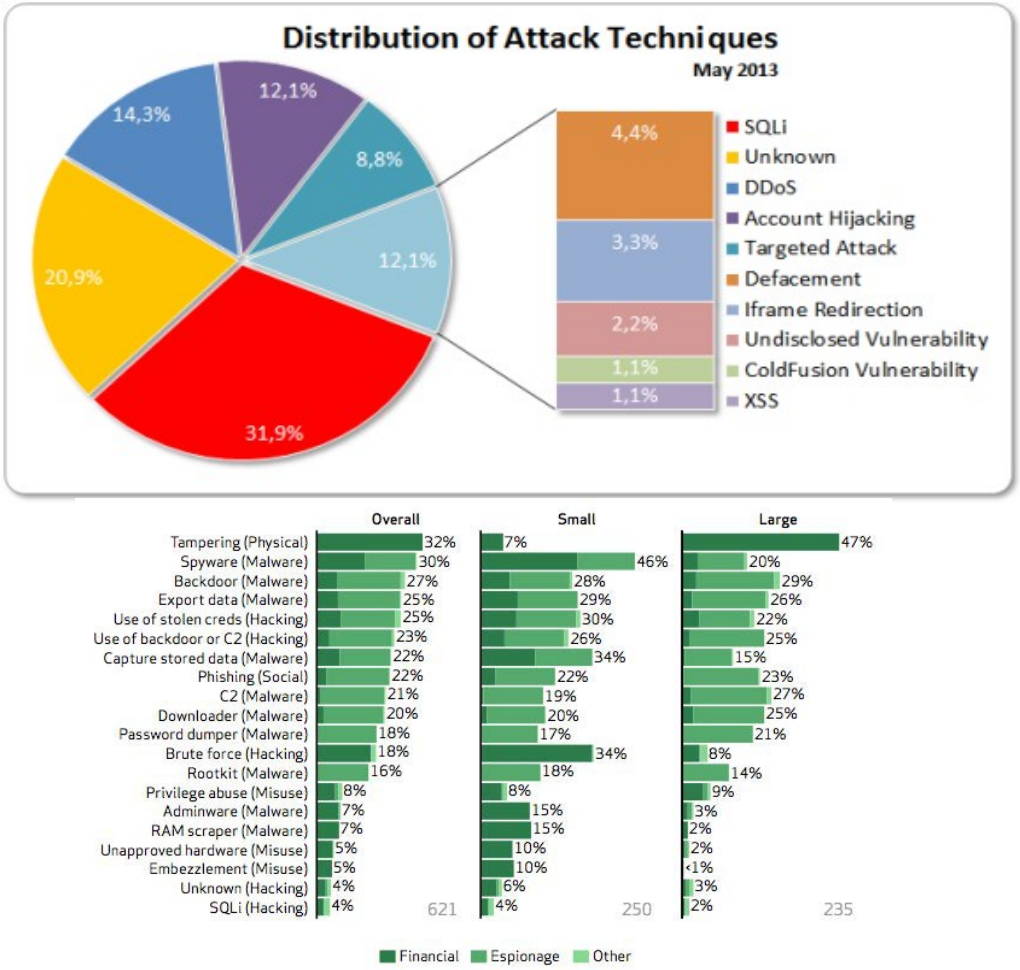
- Questions
  - How do we make better security decisions
  - Development of effective strategies by CISOs
  - Optimal Levels of Funding
- Objectives
  - Game Theory to model Complex Scenarios
  - Abstract These Models
  - Build Proof of Concept tools
- Expected Results
  - New Theory and understanding of games in a cyber security environment
  - Empirical evaluation (against real data where possible)
  - New Policy Advice

Cyber Security Resources

Hackmageddon[1]:
Compiled Statistics from attacks made public.
Current Data Source used in Experiments

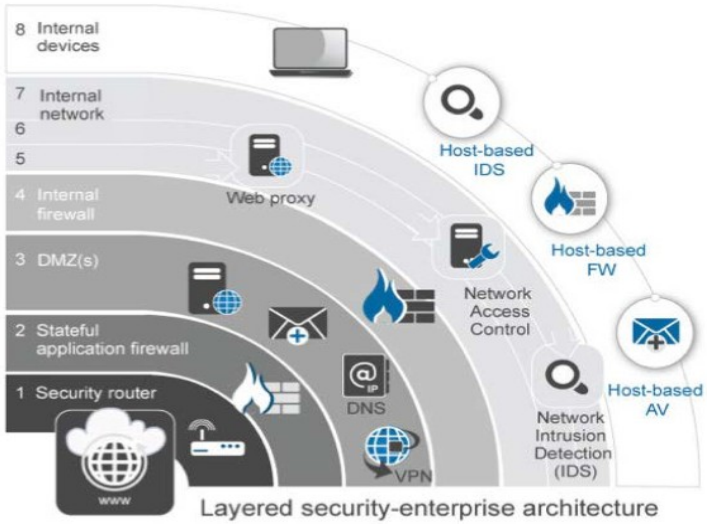
Verizon Data Breach Report[2]:
47,000+ security incidents analysed
621 confirmed data breaches Studied
19 international Contributors

Interviews:
QMUL – Working with an Systems Administrator with SME Experience
Imperial – Attack and Log data available for study



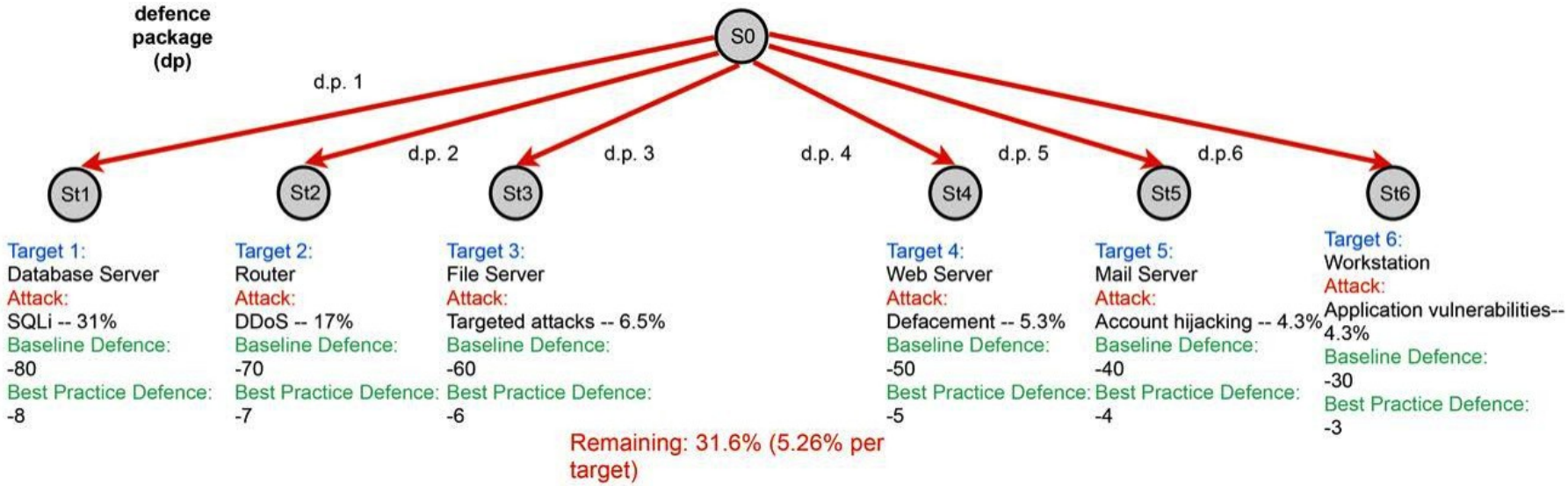
Model Design

- What are our Targets?
  - Data Assets
- How do we consider Attacks?
  - Exploitation of a Vulnerability
  - Unique Path
- How do we consider Defences?
  - All processes for upgrading network defence
- Why do we consider Administrators' Time?



Layered security-enterprise architecture

Model Design



- An example of an attack graph that represents the kind of problem we try to solve
  - Paths
  - Targets
  - Defence Packages

Game Theoretic Formulation

- Players
  - Defender $\mathcal{D}$
  - Attacker $\mathcal{A}$
- Targets
  $T = t_1, t_2, \ldots, t_n$
- Schedules
  $\mathcal{S} \subseteq \{0,1\}^n$
- Utilities of Targets
  - BaseLine $U_{\mathcal{D}}^{bl}(t_i)$
  - Best Practice $U_{\mathcal{D}}^{bp}(t_i)$

Game Theoretic Formulation

- Utilities

$$U_{\mathcal{D}}(\mathbf{D},\mathbf{A}) = \sum_{i=1}^{n} E_{D,C}(t_i) = \sum_{i=1}^{n} a_i(c_i U_{\mathcal{D}}^{bp}(t_i) + (1-c_i)U_{\mathcal{D}}^{bl}(t_i))$$

- Nash Equilibrium

$$\mathcal{D}\ plays\ a\ best\text{-}response\ that\ is\ \ U_{\mathcal{D}}(\mathbf{D},\mathbf{A}) \geq U_{\mathcal{D}}(\mathbf{D}',\mathbf{A}), \forall\ \mathbf{D}'$$
$$\mathcal{A}\ plays\ a\ best\text{-}response\ that\ is\ \ U_{\mathcal{A}}(\mathbf{D},\mathbf{A}) \geq U_{\mathcal{A}}(\mathbf{D},\mathbf{A}'), \forall\ \mathbf{A}'$$

- Game uses a Perfect Affine Transformation
- Games are general-sum, but min-max solution is equal to Nash.

Solving The Game

- A Python Based Min-Max Solver
- We have used a method based on Singular Value Decomposition (SVD) to compute equilibria in large games where a large number of assets of the defending party must be protected against adversaries.
- Our method provides reasonably close solutions to the original game solutions and a significant speed up of the computation.

Experimental Overview

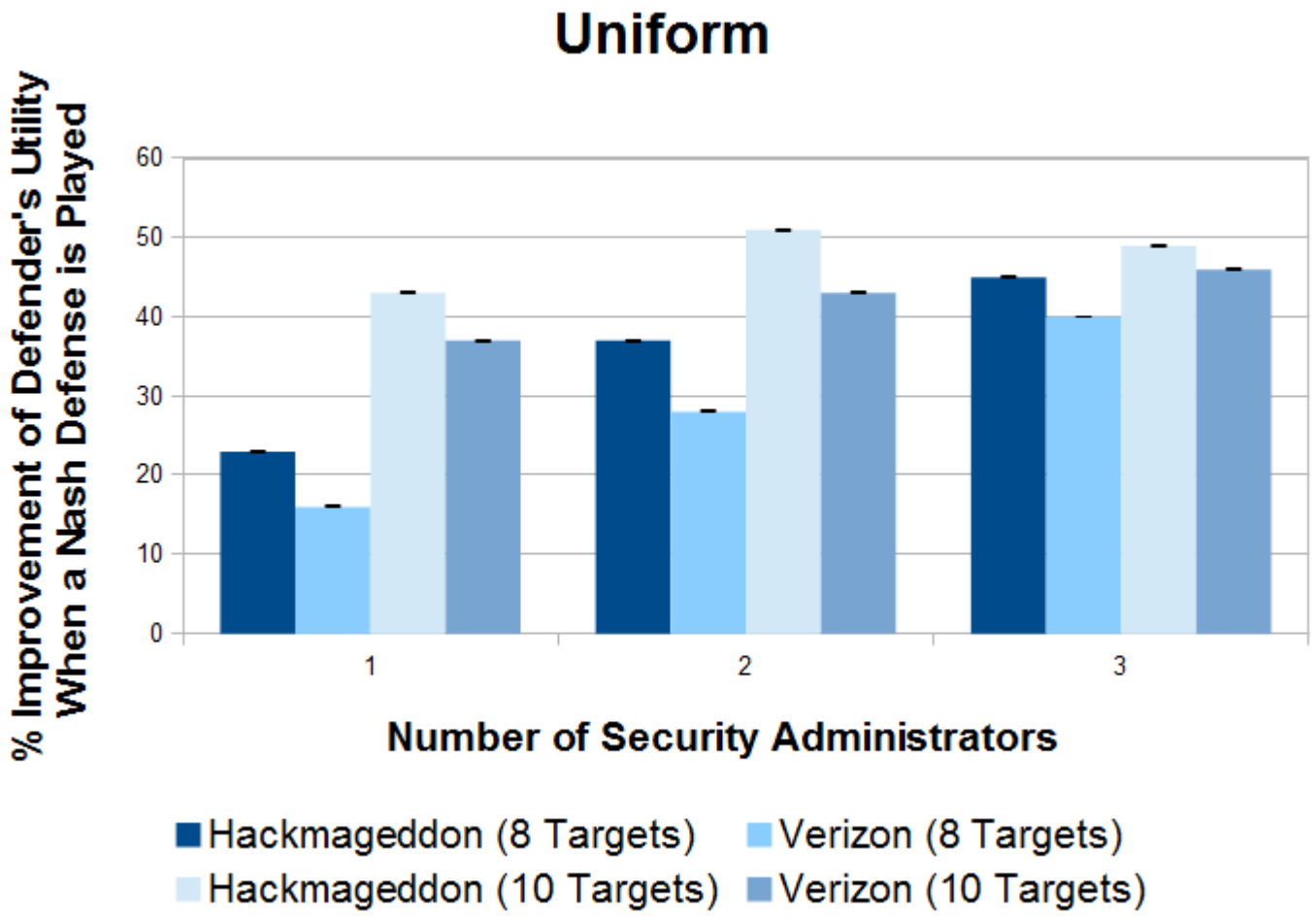
- We compare the outcome of the Min-Max to Two Common Sense Approaches:
  - Uniform – A Naive Approach, where everything is treated equally and the schedule is evenly spread.
  - Weighted – Schedule Time based on the relative value of the target.

- We additionally compare the Min-Max to a Optimisation Based Approach:
  - AC - An optimisation method that aims to reduce the amount of damage expected.

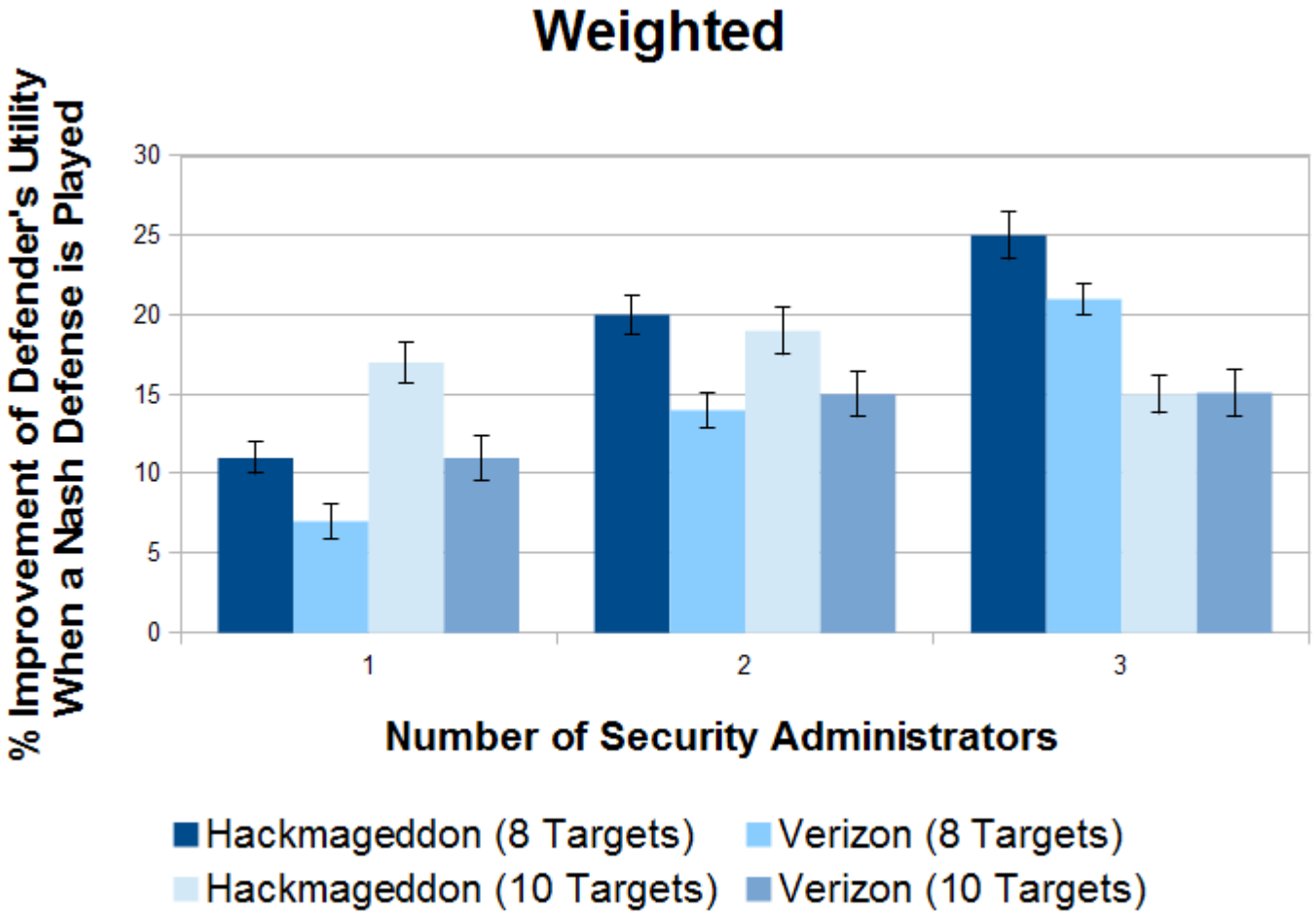
Experimental Overview

- Attack Sets
  - Hackmageddon
  - Verizon Data Breach Report
- Data Loss Costs
  - Ponemon Institute
- Experiment Specific Data
  - Number of Administrators - 1, 2 and 3
  - Number of Targets - 8 and 10
  - Sample Size - 10000 Sample Attacks
- Variance of Asset Utilities

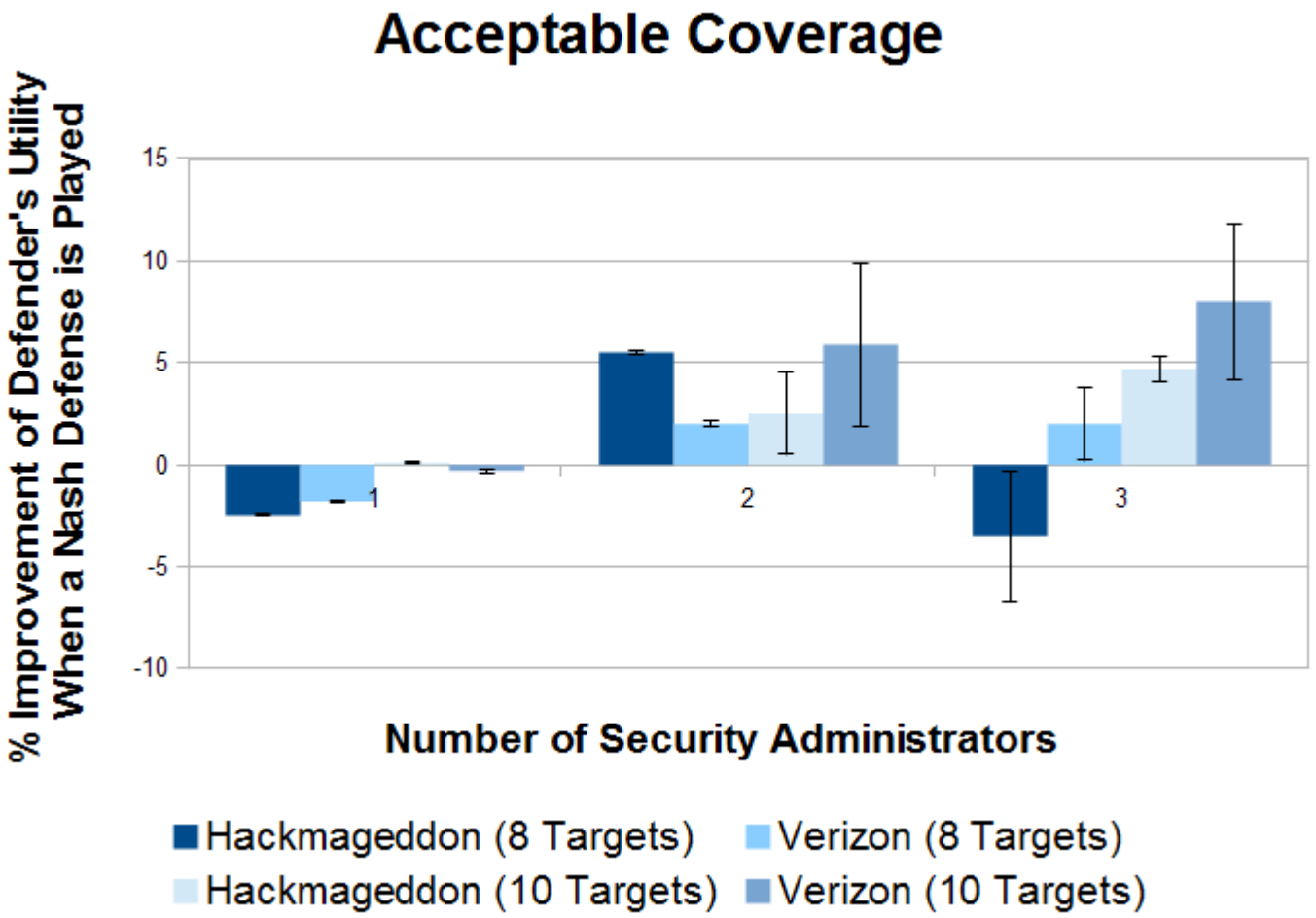
Results


Uniform
% Improvement of Defender's Utility When a Nash Defense is Played
60
50
40
30
20
10
0
1
2
3
Number of Security Administrators
Hackmageddon (8 Targets)
Verizon (8 Targets)
Hackmageddon (10 Targets)
Verizon (10 Targets)

Results


Weighted
% Improvement of Defender's Utility When a Nash Defense is Played
30
25
20
15
10
5
0
1
2
3
Number of Security Administrators
Hackmageddon (8 Targets)
Verizon (8 Targets)
Hackmageddon (10 Targets)
Verizon (10 Targets)

Results


Acceptable Coverage
% Improvement of Defender's Utility When a Nash Defense is Played
15
10
5
0
-5
-10
1
2
3
Number of Security Administrators
Hackmageddon (8 Targets)
Verizon (8 Targets)
Hackmageddon (10 Targets)
Verizon (10 Targets)

Future Work

- Interdependencies
- Multi-Stage Games
- Development of the model beyond Time
- Looking more towards investment
- Improved Sources of Data