**Problem Setting**

IT managers regularly face the challenging problem of deploying mitigations to improve cybersecurity.

**Mitigation selection challenges:**
- Appropriately allocating resources over time to achieve security quickly and efficiently.
- Addressing precedence relations when implementing multiple mitigations.
- Prioritizing important vulnerabilities without sacrificing time-efficient overall coverage.

**Problem**

What is the best way to schedule the implementation of mitigations subject to resource, budget, and precedence constraints, to achieve maximal coverage of vulnerability nodes when covering a node multiple times gives diminishing returns?

**Existing Model Limitations**

- **Mitigation selection** [2]: Provide ways to choose mitigations under this multiple-coverage notion, but cannot model the multiple-coverage objective.
- **Resource Constrained Project Scheduling** [1]: RCPSPs match our problem’s scheduling structure, but cannot model the multiple-coverage objective.

**IP Model**

\[
\begin{align*}
T & \text{ set } \{1, ..., T\} \text{ of time periods} \\
N & \text{ set of nodes} \\
J & \text{ set of jobs/ mitigations} \\
R & \text{ set of resources} \\
P & \text{ set of precedence relations} \\
w_{jn} & \text{ benefit of completing } j \in J \text{ on } n \in N \\
\tau_j & \text{ time required to complete } j \in J \\
c_{jr} & \text{ cost per period of } r \in R \text{ for } j \in J \\
c_j & \text{ cost of } j \in J \text{ for total budget} \\
b_{jt} & \text{ budget for } r \in R \text{ for period } t \in T \\
B & \text{ total budget (not time indexed)} \\
f_{\alpha} & \text{ piecewise-linear concave function for objective} \\
\alpha & \text{ time-weighting parameter} \\
z_{nt} & \text{ amount of coverage for } n \in N \text{ at time } t \in T \\
\tau_{jt} & = 1 \text{ if job } j \in J \text{ finishes at time } t \in T \text{, binary} \\
\end{align*}
\]

**Benefit of Integrated Model**

What if we don’t combine coverage and scheduling?

- **RCPSP**
  - We relax the notion of coverage by removing constraints with \( z \), and using the objective
  \[
  \max \sum_{j \in J} \sum_{t=1}^{T} a_t h_j x_{jt}
  \]
  where \( h_j \) is some estimation of coverage provided by job \( j \), and \( a_t \) is a time-weighting parameter.
  - **Select then Schedule (STS)**
    - Ignore scheduling, choose jobs for best coverage.
    - Then use an RCPSP to schedule these jobs.

**Solving Large Instances**

We propose a rolling horizon heuristic using an interval model derived from [1].

**Interval Model**

- Group time periods into a set of time intervals
- Provides a relaxation of our full model
- Schedules jobs into intervals, which we can use to schedule into time periods

**Rolling Horizon Heuristic (Int-roll)**

- Use interval model to find a solution.
- Fix some jobs at beginning of horizon.
- Repeat, fixing more jobs later into the horizon at each iteration.

**Computational Comparisons**

- **RCPSP** and STS average 7% optimality gaps, showing benefit to an integrated model.
- **Int-roll** finds good solutions faster than other methods given a time limit.

**References**


Ashley Peper: apeper2@wisc.edu

This work was in part funded by the National Science Foundation Award 2000986.