Learning Intrusion Prevention Policies through Optimal Stopping

Kim Hammar (kimham@kth.se) & Rolf Stadler, KTH Royal Institute of Technology, Sweden

CIFAR Deep Learning + Reinforcement Learning (DLRL) Summer School 2021

Overview

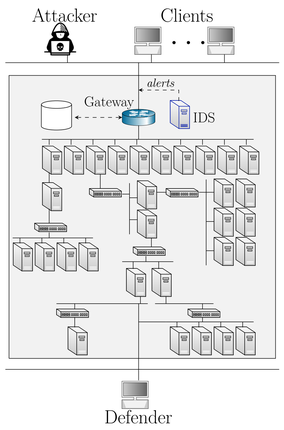
Motivation
Cyber attacks are evolving quickly and getting increasingly automated. As a consequence, a defender must constantly adapt and improve the target system in order to remain effective.

Approach
We formulate the problem of intrusion prevention as an optimal stopping problem and use a reinforcement learning approach to automatically find intrusion prevention policies.

Contributions
First, we formulate the problem of intrusion prevention as a problem of optimal stopping which allows us to derive properties of the optimal defender policy. Second, we present an approach based on reinforcement learning, system emulation, and simulations to approximate the optimal defender policy.

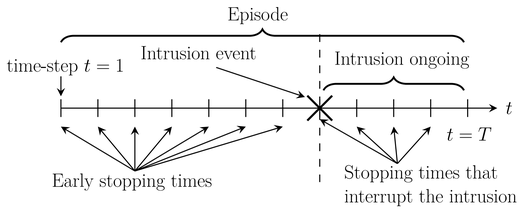
Use Case: Intrusion Prevention

We consider an intrusion prevention use case that involves the IT infrastructure of an organization. The operator of this infrastructure, which we call the defender, takes measures to protect it against an attacker while, at the same time, providing a service to a client population.



POMDP for Intrusion Prevention

We model intrusion prevention as a partially observed optimal stopping problem where the stopping action refers to blocking the gateway.



States $\mathcal{S}$ and Observations $\mathcal{O}$: The state is defined by the number of IDS alerts x_t, y_t , login attempts z_t, the intrusion time-step i_t, and the current time t: $s_t = (x_t, y_t, z_t, i_t, t)$. The defender observes the vector $o_t = (x_t, y_t, z_t, t)$. This implies that $\mathcal{Z}(o_t, s_t, \cdot) = 1$. The terminal state is $\emptyset$.

Actions $\mathcal{A}$: The defender has two actions: "stop" (S) and "continue" (C). The action space is thus $\mathcal{A} = \{S, C\}$.

Transition Probabilities $\mathcal{P}^a_{ss'}$: IDS alerts and login attempts generated during a single time-step are random variables $X \sim f_X$, $Y \sim f_Y$, $Z \sim f_Z$, with joint pmf $f_{X,Y,Z}(\Delta x, \Delta y, \Delta z | i_t, t)$ which is estimated based on empirical data.

Reward Function $\mathcal{R}^a_{ss'}$: The defender receives a positive reward for maintaining service, and incurs a loss for early stopping and being intruded. Further, the defender receives a time-decaying reward for stopping an ongoing intrusion:

$$r\left((x_t, y_t, z_t, i_t, t), S\right) = \frac{R_{st}}{(t - i_t + 1)^{1.05}} \tag{1}$$

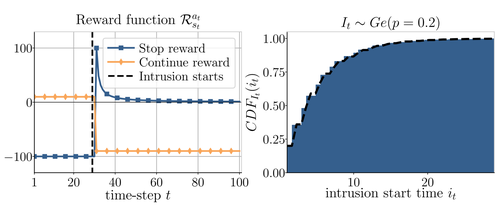


Figure: The reward function and the CDF of the intrusion start.

Threshold Property of the Optimal Policy

$$\pi^*((x_t, y_t, z_t, b_t, t)) = \text{argmax}\left[\underbrace{\beta_t}_{\text{stopping reward}}, \underbrace{\omega_t + \sum_{\Delta x \Delta y \Delta z} \mathbb{P}[\Delta x, \Delta y, \Delta z] V^*((x', y', z', b', t'))}_{\text{continue return } \alpha_t}\right] \tag{2}$$

If $\beta_t = \alpha_t$, both actions of the defender, continuing and stopping, are optimal. If $\beta_t > \alpha_t$, it is optimal for the defender to stop. Therefore, the optimal policy is determined by the scalar sequence of thresholds $(\alpha_t)_{t=1}^{T_\emptyset}$, which are monotonically decreasing in t.

Evaluation Results: Learning Defender Policies

We use the PPO reinforcement learning algorithm to learn a policy $\pi_\theta : \mathcal{O} \mapsto \mathcal{A}$, where π_θ is a feed-forward neural network. The policy is learned through simulation of the POMDP. Specifically, the simulation trajectories are used to estimate the expectation of the policy gradient $\mathbb{E}_{\pi_\theta}[\nabla_\theta \log \pi_\theta(a|o) A^{\pi_\theta}(o, a)]$. The gradient is then used to update the policy with the PPO algorithm.

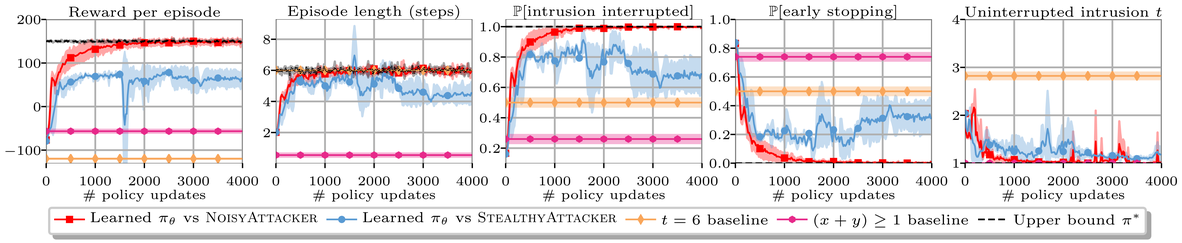


Figure: Learning curves; the curves show the averages and the standard deviations of three training runs with different random seeds..

Threshold Properties of the Learned Policies

The learned policies can be expressed through thresholds, just like the optimal policy. Specifically, the learned policies implement a soft threshold on the number alerts by stopping with high probability if $x_t + y_t > 130$.

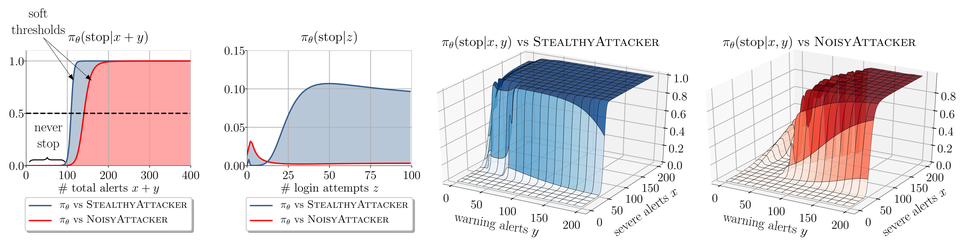


Figure: Probability of the stop action by the learned policies π_θ in function of the number of alerts x, y and login attempts z.

More Information

- **Paper:** https://arxiv.org/pdf/2106.07160.pdf
- **Code:** https://github.com/Limmen/gym-idsgame, https://github.com/Limmen/gym-optimal-intrusion-response