**Introduction**

Traditional security measures use passive defense strategies to address cybersecurity issues. The need for more robust measures are being discovered in the field of **PSYBERSECURITY and COGNITIVE MODELING**. The approaches use novel frameworks that utilize a process of models replicating human behavior to help design security measures to address flaws in securing computer systems and networks.

One method of cognitive modeling is using a Stackelberg Game and Instance-Based Learning. The purpose of these is to maximize security with limited resources and to use past instances to influence future actions. We can construct computer models to help us view virtual possibilities to influence real-world decision-making.

These models use the following foundational principles:
- **Instances**: A result that occurs at a specific time stamp under certain parameters.
- **Activation**: The use of previous instances to influence future decisions in the model. Can be edited via noise or decay.
- **Blended Value**: The probability value when activation, parameters, and instances are all considered.
- **Similarity**: The method of viewing a previous instance and its relation to the current instance. Can be calculated using a linear modeling technique.

**Methodology**

Using a pre-built IBL Model in Google Co-Lab, we were able to change a binary-choice task into a multiple-choice task. We aimed to have six nodes (computers) to attract a hacker with varying rewards, penalties, and monitoring probabilities.

The original settings were:
- **Trials** = 100
- **Participants** = 100
- **Noise** = 25
- **Decay** = 5
- **Default Utility** = 10

Rewards and Penalties came from *Deceptive Signaling: Understanding Human Behavior Against Signaling Algorithms*.

**ORIGINAL RESULTS**:

The model originally showed us that the model could not determine a singular node that was rewarding after all trials. This is due to each node being relatively like one another. However, this still leads the model to lose points and the probabilities of choices to be semi-distributed.

**MANIPULATING THE REWARD STRUCTURE**:

Changing one node to have a reward of 100 results in the model quickly learning what node to attack and reversing the trend of losing. Thus, placing the most secure material on a single computer is a flaw that can be exploited by hackers immediately.

**RESULTS AND DISCUSSION**:

**RESULTS AND DISCUSSION**:

Decay involves how many instances can be recalled before the model makes the next decision. We did notice that not being able to recall previous instances did influence the probabilities. Though it was slight, it still made us discover we could hope hackers are not quick of mind. However, this is a liability in security instances, and we should always assume hackers are on the top of their game when trying to take advantage of a system.

**RESULTS AND DISCUSSION**:

Increasing noise in the model did allow for a significant increase in losses on the part of the hacker. When noise was reduced, the model was allowed to determine the best decisions and reduced losses. This could be beneficial when constructing systems by ensuring there are enough distractors to confuse the attacker. However, security must not be a gamble on the part of the company.

---

**Conclusion**

- The experience of taking a model and manipulating parameters was unique to see that computers can perform tasks in a loop to replicate human decision-making.
- We learned several mathematical equations that demonstrate the concepts we discussed in the introduction.
- Further research will include implementing the calculation of similarity and replicating the usage of a signal as done in the research done before the project.
- Overall, we see the importance and variety of factors that go into designing cybersecurity systems for the future to ensure their safety from hackers.

---

**Acknowledgements**

Thanks Dr. Pahl Aggarwal & Arif Khan for Supervising our Research! This work is supported by National Science Foundation Award # 2206982